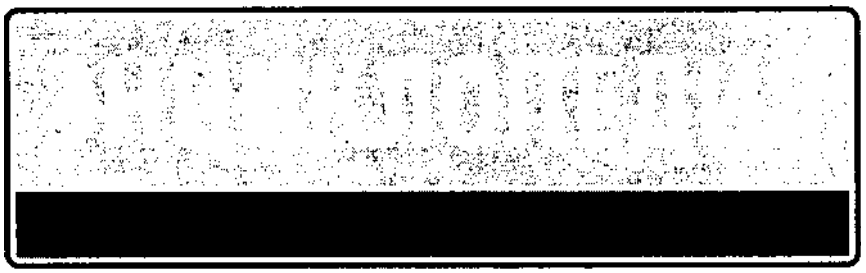
Михаил Гук



Аппаратные средства IBM PC

ПИТЕР

2006

Михаил Гук

Аппаратные средства IBM PC. Энциклопедия

3-е издание

Заведующий редакцией	А. *Кривцов*
Руководитель проекта	Ю. *Суркис*
Литературный редактор	А. *Жданов*
Художник	А. *Татарко*
Иллюстрации	В. *Демидова*
Корректоры	А. *Моносов, Н. Рощина*
Верстка	*Л. Харитонов*

ББК 32.973.23-04я20 УДК 004.3(03)

Гук М. Ю.

Г93 Аппаратные средства IBM PC. Энциклопедия. 3-е изд. — СПб.: Питер, 2006. — 1072 с.: ил. ISBN 5-469-01182-8

На страницах третьего издания вы найдете систематизированное описание «железной» части семейства самых распространенных персональных компьютеров. Книга дает глубокие знания как по отдельным электронным подсистемам (память, процессоры, диски и т. п.), так и по их соединению в единое целое — персональный компьютер со всеми его достоинствами и недостатками.

Описание сигналов в книге проиллюстрировано временными диаграммами и схемами типовых подключений различных устройств. Приводятся сведения по установке и конфигурированию аппаратных средств, не оставляются в стороне и практические вопросы диагностики, а также проблемы электробезопасности.

Для неподготовленных читателей во введении даются основы компьютерной техники и некоторые другие сведения, необходимые для понимания материала, что позволяет рекомендовать книгу самому широкому кругу читателей.

По сравнению с предыдущим изданием, имевшим большой успех у читателей, в книгу добавлен целый ряд новых сведений и отражены все новшества, появившиеся за последнее время: новые процессоры, включая 64-битные расширения; новые типы памяти; новые шины (PCI-x, PCI Express, Hyper Transport); Serial ATA и SCSI; подробное описание FireWire. Кроме того, исправлены обнаруженные ошибки и некоторые структурные огрехи, ликвидированы «белые пятна».

ISBN 5-469-01182-8

ООО «Питер Принт». 194044, Санкт-Петербург, Б Сампсониевский пр , 29а.
Лицензия ИД№ 05784 от 07.09.01.
Налоговая льгота — общероссийский классификатор продукции ОК 005-93, том 2; 953005 — литература учебная.
Подписано в печать 21.12.05 Формат 70X100/16 Усл. п. л. 86,43. Тираж 5000 экз. Заказ №6939.
Отпечатано с готовых диапозитивов в ФГУП «Печатный двор» им. А. М. Горького Федерального агентства по печати и массовым коммуникациям.
197110, Санкт-Петербург, Чкаловский пр., 15.

Краткое содержание

От автора 16
Часть I. Устройство и общая архитектура компьютера 18
Глава 1. Основы компьютерной техники 19
Глава 2. Устройство персонального компьютера 37
Глава 3. Питание компьютеров и периферийных устройств 72
Глава 4. Архитектура IBM PC-совместимого компьютера 96
Глава 5. Организация ввода-вывода и BIOS 147
Часть II. Ядро компьютера — системная плата, процессор и память 183
Глава 6. Системная плата 184
Глава 7. Процессоры 258
Глава 8. Электронная память 334
Часть III. Периферийные устройства 417
Глава 9. Устройства хранения данных 418
Глава 10. Видеосистема 559
Глава 11. Устройства ввода-вывода и их интерфейсы 642
Глава 12. Аудиосистема ПК 687
Глава 13. Коммуникационные устройства 741
Часть IV. Интерфейсы периферийных устройств 766
Глава 14. Шины расширения 767
Глава 15. Параллельный интерфейс — LPT-порт 823
Глава 16. Проводные и беспроводные последовательные интерфейсы 842
Глава 17. Шина USB 872
Глава 18. Шина IEEE 1394 — FireWire 900
Глава 19. Интерфейс IDE — ATA/ATAPI и SATA 927
Глава 20. Интерфейс SCSI 978
Глава 21. Интерфейс Fibre Channel 1024
Список литературы 1033
Алфавитный указатель 1034

Содержание

От автора 16
От издательства 17

Часть I. Устройство и общая архитектура компьютера 18

Глава 1. Основы компьютерной техники 19
1.1. Из чего состоит компьютер? 20
1.2. Биты, байты, слова, параграфы 22
1.3. Ячейки памяти, порты и регистры 25
1.4. Подсистемы памяти и хранения данных 27
Внутренняя и внешняя память 28
Диски и файловые системы 30
1.5. Устройства ввода-вывода и коммуникаций 31
1.6. Адаптеры, контроллеры и иерархия подключений периферийных устройств 32
1.7. Программное обеспечение 33

Глава 2. Устройство персонального компьютера 37
2.1. Настольные компьютеры 38
2.2. Малогабаритные компьютеры 43
2.3. Промышленные и инструментальные компьютеры 46
2.4. Периферийные устройства 49
2.5. Интерфейсы подключения периферийных устройств 56
Виды передаваемой информации 56
Параллельные и последовательные интерфейсы 57
Сигналы и среда передачи 60
Гальваническая развязка устройств 62
Достоверность, надежность передачи и управление потоком 63
Асинхронные, синхронные и изохронные передачи 65
2.6. Карты, сокеты, слоты, джамперы 66
2.7. Кабели и разъемы 69

Глава 3. Питание компьютеров и периферийных устройств 72
3.1. Схемотехника блоков питания 72
3.2. Блок питания PC 75
Блок питания для корпусов AT 78

Блок питания ATX и ATX12V....79
Питание блокнотных ПК....83
3.3. Питание процессоров....83
3.4. Охлаждение компонентов системного блока....85
3.5. Общие вопросы электропитания и заземления....89
3.6. Средства улучшения качества электропитания....92

Глава 4. Архитектура IBM PC-совместимого компьютера....96
4.1. Структурная схема....96
4.2. Распределение пространства памяти....98
Память для режима SMM....102
Верхняя память — UMA....103
Теневая память — Shadow ROM и Shadow RAM....105
Оперативная память для MS-DOS....105
Виртуальная память....108
4.3. Пространство ввода-вывода....109
4.4. Аппаратные прерывания....112
Немаскируемые прерывания....112
Маскируемые прерывания....ИЗ
Традиционный контроллер прерываний — PIC....117
Улучшенный контроллер прерываний — APIC....119
Проблема разделяемых прерываний....122
4.5. Прямой доступ к памяти — DMA....124
4.6. Аксессуары системной платы IBM PC....126
Системный таймер — 8253/8254....127
Канал управления звуком — PC speaker....128
Батарейная память и часы — CMOS Memory, RTC....129
Системная поддержка таймеров — Int 1Ah и Int 15h BIOS....133
4.7. Распределение системных ресурсов....134
4.8. Функционирование компьютера....137
Начальный запуск и самотестирование....137
Загрузка ОС и прикладных программ....138
«Засыпание» и «пробуждение» компьютера....143

Глава 5. Организация ввода-вывода и BIOS....147
5.1. Взаимодействие программ с периферийными устройствами....147
Взаимодействие через пространство памяти....152
Взаимодействие через пространство ввода-вывода....154
Синхронизация программ и устройств....156
Буферизация данных в устройствах....157
5.2. Системный модуль ROM BIOS....160
Тест начального включения — POST....162
Начальная загрузка....167
Сервисы и другие векторы прерываний BIOS....169
32-разрядные вызовы — BIOS32....172
Области данных ROM BIOS — BDA....173
PnP BIOS....173
Флэш-BIOS....173

5.3. Расширения ROM BIOS ... 175
5.4. DMI BIOS ... 177
5.5. Интерфейс ACPI ... 179

Часть II. Ядро компьютера — системная плата, процессор и память ... 183

Глава 6. Системная плата ... 184

6.1. Архитектура системной платы ... 185
Шинно-мостовая архитектура ... 186
Хабовая архитектура ... 187
Архитектура HyperTransport ... 189
Северные мосты и хабы ... 192
Южные мосты и хабы ... 194
Синхронизация и потоки данных ... 196
Чипсеты и платы ... 198
6.2. Установка и конфигурирование компонентов ... 199
Процессоры ... 199
Оперативная память (DRAM) ... 215
Слоты расширения ... 218
Синхронизация и разгон ... 221
BIOS ... 224
Память CMOS — питание и обнуление ... 228
6.3. Конструктивы и установка плат ... 231
6.4. Подключение системной платы ... 234
6.5. «Оживление» системной платы ... 240
6.6. Конфигурирование компьютера — CMOS Setup ... 244
Вход, выход и сохранение параметров Setup ... 245
Общее конфигурирование ... 246
Управление процессором ... 247
Управление памятью. 248
Конфигурирование шин ISA, PCI и порта AGP ... 249
Встроенная периферия ... 251
Управление загрузкой ... 252
Управление энергосбережением и питанием ... 253
Мониторинг состояния ... 255
6.7. Выбор системной платы PC ... 256

Глава 7. Процессоры ... 258

7.1. Исполнение программного кода ... 259
Переключение задач и виртуальные машины ... 261
Защищенный режим и виртуальная память ... 262
Архитектура и микроархитектура процессоров ... 263
7.2. Программная модель современных процессоров x86 ... 267
Режимы работы процессоров ... 268
Архитектурные регистры и типы данных ... 269
Набор инструкций (система команд) ... 274
События — прерывания и исключения ... 275

7.3. Организация памяти ... 278
Эффективный адрес ... 279
Преобразование адресов ... 280
Страничная трансляция адресов и виртуальная память ... 282
Стек ... 285
Кэширование памяти ... 285
Управление кэшированием и обращениями к памяти ... 292
7.4. Особые режимы работы процессора ... 294
Запуск и инициализация процессоров ... 294
Переключение между реальным и защищенным режимами ... 295
Обновление микрокода ... 296
Режим системного управления ... 297
Управление энергопотреблением и производительностью ... 298
7.5. Мультипроцессорные и избыточные системы ... 304
Симметричные мультипроцессорные системы ... 305
Объединение процессоров на локальной шине ... 305
Гиперпотоковые и мультиядерные процессоры ... 307
Мультипроцессорные системы Athlon и Opteron ... 308
7.6. Совместимость и идентификация процессоров ... 310
Совместимость процессоров ... 310
Идентификация процессоров ... 311
Основные характеристики процессоров ... 316
7.7. Процессоры фирмы Intel ... 319
Процессоры P6 ... 320
Процессоры Pentium 4 ... 324
7.8. Процессоры AMD ... 330

Глава 8. Электронная память ... 334

8.1. Структура оперативной памяти ... 335
Быстродействие и производительность памяти ... 336
Достоверность хранения данных ... 338
Кэширование оперативной памяти ... 340
Режим пакетной передачи данных ... 341
8.2. Динамическая память ... 342
Основы работы DRAM ... 343
Регенерация ... 346
Асинхронная память — FPM, EDO и BEDO DRAM ... 348
Синхронная память — SDRAM, DDR и DDR2 SDRAM ... 352
Память Rambus DRAM — RDRAM и XDRAM ... 361
Память с виртуальными каналами — VC DRAM ... 366
Сравнительные характеристики и перспективные типы динамической памяти ... 368
8.3. Применение модулей DRAM в оперативной памяти ... 370
Модули динамической памяти ... 372
Нюансы применения DRAM ... 386
Рекомендации по выбору модулей динамической памяти ... 388
Тестирование оперативной памяти ... 390

8.4. Статическая память ... 392
Разновидности статической памяти ... 393
Применение статической памяти для кэширования ОЗУ ... 396
Напряжение питания SRAM ... 397
8.5. Энергонезависимая память ... 398
Постоянная и полупостоянная память — ROM, PROM, EPROM 401
Флэш-память и EEPROM ... 404
Энергонезависимая память с последовательными интерфейсами 415

Часть III. Периферийные устройства ... 417

Глава 9. Устройства хранения данных ... 418

9.1. Принцип действия и назначение устройств хранения ... 418
9.2. Основные характеристики и конструктивы устройств хранения 420
9.3. Интерфейсы устройств хранения ... 425
9.4. Преодоление физических ограничений — массивы RAID ... 431
9.5. Устройства, системы и сети хранения данных ... 434
9.6. Логическая структура дисков ... 437
Разделы и логические диски ... 437
Логический диск с файловой системой FAT ... 441
9.7. Устройства хранения на магнитных дисках ... 443
Накопители на гибких магнитных дисках ... 447
Накопители на жестких магнитных дисках — винчестеры ... 457
Сменные магнитные диски большой емкости ... 480
Магнитооптические диски ... 483
9.8. Оптические диски — CD, DVD, PD ... 488
Диски CD - CD, CD-R, CD-RW ... 488
Диски DVD ... 496
Устройство приводов CD-ROM, CD-R, CD-RW и DVD ... 502
Файловые системы для CD и DVD ... 506
Запись на оптические диски ... 507
Оптические диски с прямым доступом ... 513
9.9. Ленточные устройства — стримеры ... 514
9.10. Твердотельные устройства хранения ... 516
Флэш-память USB ... 519
CompactFlash ... 520
SmartMedia Card ... 522
MultiMediaCard и Secure Digital ... 525
Miniature Card ... 530
9.11. Системная поддержка внешней памяти ... 531
Традиционный сервис BIOS INT 13h ... 532
Расширенный сервис BIOS INT 13h ... 533
Преодоление барьера 528 Мбайт (ECHS и LBA) ... 534
Сервисы операционной системы ... 538
Системная поддержка CD-ROM ... 539
Загружаемые диски CD-ROM ... 539
9.12. Установка и обслуживание устройств ... 541
Установка новых устройств ... 541

Проблемы использования больших дисков 546
Конфигурирование, форматирование и обслуживание дисков 554
Основные причины отказов дисков 556

Глава 10. Видеосистема 559

10.1. Принципы вывода изображений 560
Графический режим 564
Текстовый режим 567
Обработка видеоизображений 569
Стандарты MPEG 575
10.2. Акселератор — «интеллект» графического адаптера 580
Трехмерная графика 583
Память для графического акселератора 591
10.3. Дисплей 594
Электронно-лучевой дисплей 594
Матричные дисплеи 604
Трехмерный вывод изображения и виртуальная реальность 607
10.4. Интерфейсы мониторов и видеосистем 609
Аналоговые интерфейсы RGB 610
Цифровые интерфейсы P&D, DVI и DFP 613
Телевизионные интерфейсы 617
10.5. Дисплейные адаптеры 619
Компоненты дисплейного адаптера 619
Программные модели стандартных адаптеров 627
Адаптеры с интерфейсами PCI, AGP и PCI-E 628
Мультидисплейные системы 631
10.6. Видеосервис BIOS 633
10.7. Параметры видеосистемы 636

Глава 11. Устройства ввода-вывода и их интерфейсы 642

11.1. Клавиатура 642
Интерфейс клавиатуры 644
Контроллер интерфейса клавиатуры и мыши 8042/8242 646
Скан-коды 648
Системная поддержка и программный интерфейс 649
11.2. Манипуляторы-указатели — мышь, трекбол 652
Последовательные мыши — MS Mouse и PC Mouse 655
Мышь PS/2 656
Беспроводные мыши и клавиатуры 656
11.3. Планшеты 657
11.4. Сканеры 658
11.5. Принтеры и плоттеры 660
Матричные игольчатые принтеры 661
Термопринтеры 664
Струйные принтеры 665
Твердокрасочные и сублимационные принтеры 666
Лазерные и светодиодные принтеры 666
Цветная печать и фотопринтеры 668

Плоттеры 669
Форматы данных 671
Интерфейсы принтеров и плоттеров 674
Системная поддержка принтера 680
11.6. Игровые устройства — джойстик, руль, педали 681
11.7. Коммутаторы устройств ввода-вывода 684

Глава 12. Аудиосистема ПК 687
12.1. Краткий экскурс в прикладную звукотехнику 690
Оцифровка звуковых сигналов 692
Использование ПК для обработки «цифрового» звука 695
Методы компрессии звуковой информации 697
Методы синтеза звуков 699
Стереофоническое и объемное воспроизведение 704
Трехмерный звук 705
Аудиоданные на дисках CD и DVD 707
12.2. Звуковые карты PC 709
Аналоговые звуковые карты 711
Цифровые технологии в звуковых картах 712
Аудиокодек AC'97 715
Многоканальный звук — High Definition Audio 720
Интерфейсы звуковых карт 726
«Исторические» модели звуковых карт 730
12.3. Интерфейс MIDI 735

Глава 13. Коммуникационные устройства 741
13.1. Модемы и факс-модемы 741
Модемы для телефонных линий 744
Технологии xDSL и кабельные модемы 747
Модемы для выделенных линий 749
13.2. Подключение к проводным локальным сетям 750
Организация сетей Ethernet 750
Сетевые адаптеры 753
13.3. Подключение к беспроводным сетям (Wi-Fi) 758
13.4. ПК и Интернет 761
Варианты подключения 761
IP-телефония и передача факсов по IP-сетям 763

Часть IV. Интерфейсы периферийных устройств 766

Глава 14. Шины расширения 767
14.1. Организация шин PCI и PCI-X 768
Взаимодействие устройств 770
Шины, устройства, функции и хост 772
Спецификации PCI и PCI-X 774
14.2. Протокол, команды и транзакции шин PCI и PCI-X 775
Команды шины PCI 779

Особенности PCI-X ... 780
Время выполнения транзакций, таймеры и буферы ... 781
14.3. Прямой доступ к памяти, эмуляция ISA DMA (PC/PCI, DDMA) . . 783
14.4. Пропускная способность шин PCI и PCI-X ... 784
14.5. Прерывания PCI — INTx#, PME#, MSI и SERR# ... 785
Традиционные прерывания PCI — INTx# ... 785
Сигнализация событий управления энергопотреблением — PME# . . . 786
Прерывания сообщениями — MSI ... 786
14.6. Мосты PCI и PCI-X ... 787
Транслирование транзакций и буферизация ... 790
Порядок выполнения операций и синхронизация ... 791
14.7. Конфигурирование и BIOS устройств PCI и PCI-X ... 791
PCI BIOS ... 792
Expansion ROM карт PCI ... 793
14.8. Слоты и карты PCI/PCI-X ... 793
Инициализация и определение режима работы шины PCI-X ... 797
Малогабаритные конструктивы с шиной PCI ... 798
14.9. Порт графического акселератора — AGP ... 798
Протоколы транзакций ... 801
Трансляция адресов — апертура AGP и GART ... 804
Изохронные транзакции в AGP 3.0 ... 805
Конфигурационные регистры AGP ... 806
Слоты и карты AGP ... 806
14.10. PCI Express ... 807
Элементы и топология соединений PCI Express ... 808
Архитектурная модель PCI Express ... 810
Физический уровень и конструктивы PCI Express ... 815
14.11. Шины расширения блокнотных ПК ... 818
Конструктивы Small PCI, Mini PCI и Mini PCI Express ... 818
Карты PCMCIA: интерфейсы PC Card, CardBus и Express Card 820

Глава 15. Параллельный интерфейс — LPT-порт ... 823
15.1. Традиционный LPT-порт ... 824
15.2. Расширения параллельного порта ... 825
15.3. Стандарт IEEE 1284 ... 826
Полубайтный режим ввода ... 827
Байтный режим ввода ... 827
Режим EPP ... 828
Режим ECP ... 830
Согласование режимов IEEE 1284 ... 833
Физический и электрический интерфейсы ... 833
Подключение цепочек устройств и мультиплексоров ... 834
15.4. Системная поддержка LPT-порта ... 836
15.5. Параллельный порт и функции PnP ... 837
15.6. Применение LPT-порта ... 838
15.7. Конфигурирование LPT-порта ... 839
15.8. Неисправности и тестирование параллельного порта ... 839

Глава 16. Проводные и беспроводные последовательные интерфейсы ... 842
16.1. Интерфейс RS-232C — COM-порт ... 842
Протокол RS-232C ... 844
Управление потоком данных ... 848
Микросхемы асинхронных приемопередатчиков ... 849
Системная поддержка COM-портов ... 850
Конфигурирование COM-портов ... 851
Использование COM-портов ... 852
COM-порт и PnP ... 855
Неисправности и тестирование COM-портов ... 856
16.2. Инфракрасный интерфейс IrDA ... 860
16.3. Радиоинтерфейс Bluetooth ... 864
Физические каналы и пикосети ... 864
Синхронизация и установление соединений ... 867
Логический транспорт, пакеты и каналы ... 868
Протоколы Bluetooth ... 870

Глава 17. Шина USB ... 872
17.1. Архитектура USB ... 872
17.2. Топология шины ... 874
17.3. Модель передачи данных ... 876
Запросы, пакеты и транзакции ... 878
Каналы ... 879
17.4. Организация обменов по шине ... 880
Кадры и микрокадры ... 880
Протокол шины USB ... 880
Пропускная способность и совместная работа устройств с разными скоростями ... 882
Синхронизация при изохронной передаче ... 884
17.5. Электрический интерфейс ... 886
Кабели и разъемы ... 886
Сигнальный интерфейс ... 888
Питание от шины ... 890
17.6. Хабы USB ... 891
17.7. Хост-контроллер ... 891
Универсальный хост-контроллер ... 892
Открытый хост-контроллер ... 893
Расширенный хост-контроллер ... 893
17.8. USB без ПК — расширение OTG ... 895
17.9. Автоматическое конфигурирование устройств ... 896
17.10. Проблемы при подключении устройств USB ... 897

Глава 18. Шина IEEE 1394 — FireWire ... 900
18.1. Спецификации ... 900
18.2. Организация, топология и архитектура ... 901
Топология ... 902

Архитектура сети ... 902
Архитектура узла ... 903
Адресное пространство сети и узла ... 905
18.3. Физический интерфейс ... 906
Кабели и коннекторы ... 906
Питание от шины ... 909
18.4. Конфигурирование ... 909
Идентификация дерева ... 910
Самоидентификация узлов ... 911
18.5. Передача данных ... 911
Арбитраж ... 912
Организация потоковых передач и изохронный обмен ... 914
18.6. Управление ... 915
18.7. Применение ... 916
Шина 1394 в компьютерах ... 916
Шина 1394 для устройств хранения данных ... 917
Шина 1394 для передачи и печати изображений ... 917
Шина 1394 для аудио- и видеоустройств ... 917
Защита передаваемой информации ... 919
18.8. Открытый хост-контроллер ... 919
Устройство контроллера OHC ... 921
Взаимодействие хоста и OHC ... 923
18.9. Протокол SBP-2 ... 923
Организация взаимодействия устройств ... 924
Структура целевого устройства ... 925
Запросы ... 925
Агенты целевого устройства ... 925
Потоки ... 926

Глава 19. Интерфейс IDE — ATA/ATAPI и SATA ... 927
19.1. Устройства, адаптеры, контроллеры и интерфейсы IDE ... 929
19.2. Параллельный интерфейс ATA ... 933
Физический интерфейс ... 933
Назначение сигналов ATA ... 937
Подключение и конфигурирование устройств ATA/ATAPI ... 938
Режимы передачи данных для устройств ATA ... 941
19.3. Интерфейс Serial ATA ... 944
Физический интерфейс SATA ... 946
Расширения SATA для систем хранения данных ... 948
19.4. Адаптеры и контроллеры ATA ... 949
Традиционный адаптер шины ATA ... 950
Контроллер PCI IDE Bus Master ... 951
Контроллер SATA Intel 31244 ... 953
Контроллер AHCI ... 955
19.5. Программное взаимодействие с устройствами ATA/ATAPI и SATA . 959
Адресация блоков данных ... 960

Регистры устройств ATA ... 962
Регистры Serial ATA ... 968
19.6. Система команд ATA/ATAPI и SATA ... 969
Команды доступа к данным ATA ... 969
Пакетный интерфейс ATAPI ... 970
Инициализация, идентификация и конфигурирование устройств ... 971
Журналы ошибок и событий ... 972
Мониторинг состояния — S.M.A.R.T. ... 973
Работа со сменными носителями ... 973
Поддержка флэш-памяти и компактных карт ... 974
Управление энергопотреблением и шумом ... 974
Защита данных ... 975
Потоковое расширение команд ... 977

Глава 20. Интерфейс SCSI ... 978
20.1. Спецификации SCSI ... 979
20.2. Архитектурная модель SAM ... 980
Команды, задания и очереди ... 981
Соединения ... 982
Состояния, исключения и асинхронные события ... 982
Типы периферийных устройств ... 983
Система команд SCSI ... 984
Отличия ATAPI от SCSI ... 985
20.3. Хост-адаптер SCSI ... 986
20.4. Параллельные шины SCSI ... 987
Версии параллельной шины ... 987
Протокол параллельной шины ... 990
Процессы ввода-вывода на шине SCSI ... 997
Физический и электрический интерфейсы ... 999
Экспандеры ... 1005
Подключение устройств к шине ... 1007
Конфигурирование устройств ... 1012
20.5. Устройства SCSI с последовательным интерфейсом — SAS ... 1013
Устройства, порты и соединения SAS ... 1014
Топология домена и маршрутизация ... 1016
Архитектурная модель SAS ... 1018
Физический уровень SAS ... 1018
Протокол SSP ... 1020
Протокол SMP ... 1021
Протокол STP ... 1022
Определение структуры домена ... 1023

Глава 21. Интерфейс Fibre Channel ... 1024
21.1. Топология и типы портов ... 1025
21.2. Архитектура стандарта Fibre Channel ... 1027
21.3. Среда и скорости передачи ... 1028

21.4. Адресация и подключение узлов 1029
21.5. Арбитражное кольцо — FC-AL....1030
Арбитраж и открытие соединений 1030
Инициализация кольца 1031
21.6. Протокол FCP — Fibre Channel для SCSI 1031

Список литературы....1033

Алфавитный указатель 1034

От автора

Энциклопедия «Аппаратные средства IBM PC» завоевала популярность с пер- вого издания (1998 г.). В 2001 году вышло второе издание, в котором были исправлены огрехи структурирования и некоторые ошибки, а также освещены появившиеся за три года новинки. Судя по многочисленным отзывам, книга используется и как справочник, и как неофициальный учебник в ряде известных вузов страны, ее читают и за рубежом.

Перед вами третье издание, в котором я постарался «подтянуть» материал к уровню 2005 года, исправить обнаруженные неточности и донести до читателя свое текущее понимание рассматриваемых вопросов. Затянувшаяся пауза между изданиями обусловлена сменой рода деятельности: началом преподавания «компьютерных наук» в Политехническом университете (ленинградском «Политехе») и руководством группой разработчиков периферийных устройств в ЦНИИ РТК. С одной стороны, это отвлекло от писания книг, но с другой - позволило иначе посмотреть на вопросы изложения материала и приобрести дополнительные знания и практический опыт их приложения.

В результате некоторой реструктуризации материала энциклопедия разделилась на четыре части. Первая часть посвящена фон-неймановской *архитектуре компьютеров* и ее самым распространенным представителям — IBM PC-со- вместимым персональным компьютерам. По сравнению с предыдущим изданием здесь больше внимания уделено подключениям периферийных устройств и их взаимодействию с программным обеспечением. Обновлены материалы о конструктивных особенностях современных ПК, в том числе малогабаритных (блокнотных и планшетных), а также системах их питания и охлаждения.

Вторая часть книги описывает *ядро компьютера*: системную плату, процессор и оперативную память. Здесь отражены новинки архитектуры системных плат, связанные с появлением PCI Express и HyperTransport. По сравнению с предыдущим изданием более подробно рассказывается о процессорах, от 32-разрядных 6-7-го поколений (Pentium III—4) до 64-разрядных x86 8-го поколения. Рассматриваются особенности гиперпотоковых и мультиядерных процессоров и мультипроцессорных систем на их основе. Добавлены материалы о современных технологиях в области памяти (DDR2 SDRAM, XDRAM, DDR SRAM, QDR и QDRII SRAM).

Третья часть книги посвящена *периферийным устройствам.* Обновлены материалы об устройствах хранения данных, появились разделы о стримерах, успешно конкурирующих с популярными оптическими дисками; расширены сведения о твердотельной памяти — флэш-картах разных типов. В описании видеосистемы больше внимания уделено матричным дисплеям и их интерфей-

сам. В главе об устройствах ввода-вывода появились разделы о сканерах и планшетах (дигитайзерах). В описании аудиосистемы отражены такие новшества, как метод оцифровки звука в Super Audio CD, спецификации High Definition Audio, автоматическое конфигурирование гнезд. В разделах, относящихся к коммуникационным устройствам, появились материалы о беспроводных сетях.

Четвертая часть книги посвящена *интерфейсам периферийных устройств.* Она начинается с главы о шинах расширения, в которых основное внимание уделено PCI, PCI-X и PCI Express. В описании параллельных и последовательных интерфейсов исправлены некоторые неточности; принципиальных новинок в этой области не появилось. Главы, посвященные шинам USB и FireWire, отражают современное состояние этих интерфейсов (описание FireWire в предыдущем издании энциклопедии явно оставляло желать лучшего). Главы, относящиеся к интерфейсам устройств хранения, тоже значительно переработаны. В них отражена тенденция перехода на последовательные интерфейсы: SATA, SAS, Fi- breChannel. Кроме того, значительное внимание уделяется идеологии взаимодействия с устройствами хранения и раскрывается эволюция интерфейса ATA (постепенное приближение к поддержке многозадачности).

Таковы основные новшества данного издания. Кроме того, проведена работа по устранению неоднозначности и небрежности толкований различных понятий. Для облегчения восприятия в книге используются шрифтовые выделения названий сигналов (Frame#, D+), инструкций, регистров и битов (SYNC). Курсивом выделены *ключевые понятия*, а также названия команд *(READ)*, пакетов (*Data0*), состояний *(Idle)*, Штриховка на рисунках, изображающих назначение регистров и программных структур, означает зарезервированные или не используемые поля.

Я благодарен любознательным и внимательным читателям, присылающим свои замечания, вопросы и отзывы о моих книгах. На все технические вопросы и замечания по книге я готов ответить по электронной почте mgook@stu.neva.ru. Пользуясь случаем, еще раз обращаюсь к читателям: пишите письма! С вашей помощью исправляются многие ошибки.

Как и все предыдущие, эта книга не смогла бы появиться без информационной поддержки коллектива RUSNet (http://www.neva.ru), обеспечивающего доступ к Сети в ЦНИИ РТК — «базовом лагере» автора. Интернет является основным источником информации — на сайтах фирм-производителей аппаратуры и программного обеспечения из моря рекламной информации удается выуживать и ценные сведения.

От издательства

Ваши замечания, предложения и вопросы отправляйте по адресу электронной почты comp@piter.com (издательство «Питер», компьютерная редакция).

Мы будем рады узнать ваше мнение!

Подробную информацию о наших книгах вы найдете на веб-сайте издательства http://www.piter.com.

Часть I

Устройство и общая архитектура компьютера

ГЛАВА 1

Основы компьютерной техники

Компьютер представляет собой устройство, способное исполнять четко определенную последовательность операций, предписанную программой. Понятие «компьютер» является более широким, чем «электронно-вычислительная машина» (ЭВМ), поскольку в последнем явный акцент делается на вычисления. Персональный компьютер (ПК) характерен тем, что им может пользоваться один человек, не прибегая к помощи бригады обслуживающего персонала и не отводя под него специального зала с особым климатом, мощной системой электропитания и прочими атрибутами больших вычислительных машин. Персональный компьютер обычно в значительной степени ориентирован на интерактивное взаимодействие с одним пользователем (в играх иногда и с двумя), причем взаимодействие происходит через множество сред общения — от алфавитно-цифрового и графического диалога посредством дисплея, клавиатуры и мыши до устройств виртуальной реальности, в которых пока не задействованы, наверное, только запахи. Когда используется аббревиатура PC (Personal Computer), подразумевается ПК, совместимый с самым массовым семейством персональных компьютеров фирмы IBM и их клонов. Конечно же, это не единственное в мире семейство — есть множество других достойных компьютерных линий, но данная книга посвящена именно IBM PC-совместимым персональным компьютерам. Чем они отличаются от других, можно узнать в главе 4. PC поддерживает и коллективную работу: возможности многих компьютеров этого семейства позволяют использовать их в качестве серверов в сетях или в локальных многотерминальных системах. Таким образом, можно объяснить словосочетание PC-сервер, которое неявно предполагает повышенную мощность (скорость вычислений, объем оперативной и внешней памяти) и особое конструктивное исполнение (просторный корпус) компьютера. Словосочетание «ПК-сервер» уже выглядит довольно странно, хотя в одноранговых сетях и этому словосочетанию можно найти объяснение — персональный компьютер может предоставлять свои ресурсы (например, дисковое пространство, принтеры или модемы) другим компьютерам, для которых он становится невыделенным сервером. Попутно отметим и термин «рабочая станция» (Workstation, WS), который может иметь два значения. В компьютерной сети рабочей станцией называют компьютер пользователя (как противоположность серверу). Однако

рабочей станцией могут назвать и изолированный компьютер (standalone computer), когда хотят подчеркнуть его особенную мощность (подключение к сети такого компьютера, конечно же, не исключается). В этом случае часто подразумевается архитектура, отличающаяся от IBM PC-совместимой (например, компьютер на RISC-процессоре). Для мощного IBM PC-совместимого компьютера применяют англоязычный термин High End PC, которому короткого русского аналога пока нет.

Персональные компьютеры, совместимые с IBM PC, делятся на несколько поколений (или классов), которые начинались со следующих «исторических» моделей:

- IBM PC первой модели: процессор Intel 8088, адресуемая память — 1 Мбайт, шина расширения — ISA (8 бит), накопители на гибких дисках (НГМД) — до 360 Кбайт;
- IBM PC/XT (extended Technology — расширенная технология) — все то же, но с винчестерами (накопителями на жестких дисках, НЖМД) и возможностью установки математического сопроцессора Intel 8087;
- IBM PC/AT (Advanced Technology — прогрессивная, или «продвинутая», технология): процессор — Intel 80286/80287, адресуемая память — 16 Мбайт, шина - ISA 16 бит, НГМД - 1,2 и 1,44 Мбайт, НЖМД.

В настоящее время класс машин AT развивается в нескольких направлениях: 16-разрядный процессор заменен 32-разрядным (уже класса P6 и выше), а теперь уже и 64-разрядным; память адресуется в пространстве до 4 или 64 Гбайт (и больше); применяются более эффективные шины расширения (PCI, PCI-X и PCI Express) с сохранением (и то уже не всегда) ISA для совместимости со старыми адаптерами; расширяется состав устройств, имеющих системную поддержку. Компьютеры выпускаются в разных исполнениях — от настольных (напольных) до блокнотных, причем их предельные возможности не так уж сильно различаются. Есть и специальные конструкции для встраивания в технологическое и иное оборудование. Самые маленькие, помещающиеся на ладони (palm top), пока что сильно отличаются от своих более крупных сородичей, и в этой книге им внимание практически не уделяется.

1.1. Из чего состоит компьютер?

Любой IBM PC-совместимый компьютер представляет собой реализацию так называемой фон-неймановской архитектуры вычислительных машин. Эта архитектура была представлена Джорджем фон Нейманом (George von Neumann) еще в 1945 году и имеет следующие основные признаки. Машина состоит из *блока управления*, *арифметико-логического устройства* (АЛУ), *памяти* и *устройств ввода-вывода*. В ней реализуется *концепция хранимой программы*: программы и данные хранятся в одной и той же памяти. Выполняемые действия определяются блоком управления и АЛУ, которые вместе являются основой центрального процессора. Центральный процессор выбирает и исполняет команды из памяти последовательно, адрес очередной команды задается «счетчиком адреса» в блоке управления. Этот принцип исполнения называется *после*

довательной передачей управления. Данные, с которыми работает программа, могут включать *переменные* — именованные области памяти, в которых сохраняются значения с целью дальнейшего использования в программе. Фон-неймановская архитектура — не единственный вариант построения ЭВМ, есть и другие, которые не соответствуют указанным принципам (например, потоковые машины). Однако подавляющее большинство современных компьютеров основаны именно на указанных принципах, включая и сложные многопроцессорные комплексы, которые можно рассматривать как объединение фон-неймановских машин. Конечно же, за более чем полувековую историю ЭВМ классическая архитектура прошла длинный путь развития. Тем не менее ПК можно «разложить по полочкам» следующим образом.

Центральный процессор (АЛУ с блоком управления) реализуется микропроцессором семейства x86 — от 8086/88 до новейших процессоров Pentium, Athlon и Opteron (и это не конец истории). При всей внутренней суперскалярности, суперконвейеризированности и спекулятивности (см. главу 7) современного процессора внешне он соблюдает вышеупомянутый принцип последовательной передачи управления. Набор арифметических, логических и прочих инструкций насчитывает несколько сотен, а для потоковой обработки придуман принцип SIMD (Single Instruction Multiple Data — множество комплектов данных, обрабатываемых одной инструкцией), по которому работают расширения MMX, 3DNow!, SSE. Процессор имеет набор регистров, часть которых доступна для хранения операндов, выполнения действий над ними и формирования адреса инструкций и операндов в памяти. Другая часть регистров используется процессором для служебных (системных) целей, доступ к ним может быть ограничен (есть даже программно-невидимые регистры). Все компоненты компьютера представляются для процессора в виде наборов *ячеек памяти* или/и *портов ввода-вывода*, в которые процессор может записывать и/или из которых может считывать содержимое.

Память «расползлась» по многим компонентам. *Оперативная память* (ОЗУ) — самый большой массив ячеек памяти со смежными адресами — реализуется, как правило, на модулях (микросхемах) динамической памяти. Для повышения производительности обмена данными (включая и считывание команд) оперативная память кэшируется сверхоперативной памятью (см. 7.3). Два уровня кэширования территориально располагаются в микропроцессоре. Оперативная память вместе с кэшем всех уровней (в настоящее время — до трех) представляет собой единый массив памяти, непосредственно доступный процессору для записи и чтения данных, а также считывания программного кода. Помимо оперативной память включает также постоянную (ПЗУ), из которой можно только считывать команды и данные, и некоторые виды специальной памяти (например, видеопамять графического адаптера). Вся эта память (вместе с оперативной) располагается в едином пространстве с линейной адресацией. В любом компьютере обязательно есть *энергонезависимая память*, в которой хранится программа начального запуска компьютера и минимально необходимый набор сервисов (ROM BIOS).

Процессор (один или несколько), память и необходимые элементы, связывающие их между собой и с другими устройствами, называют *центральной частью*,

или *ядром*, компьютера (или просто *центром*). То, что в фон-неймановском компьютере называлось устройствами ввода-вывода (УВВ), удобнее называть периферийными устройствами.

Периферийные устройства (ПУ) — это все программно-доступные компоненты компьютера, не попавшие в его центральную часть. Их можно разделить по назначению на несколько классов:

- *Устройства хранения данных* (устройства внешней памяти) — дисковые (магнитные, оптические, магнитооптические), ленточные (стримеры), твердотельные (карты, модули и USB-устройства на флэш-памяти). Эти устройства используются для сохранения информации, находящейся в памяти, на энергонезависимых носителях и загрузки этой информации в оперативную память. В каком виде хранится информация на этих устройствах, нам не так уж важно (главное — правильно считать то, что сохранили).
- *Устройства ввода-вывода* служат для преобразования информации из внутреннего представления компьютера (биты и байты) в форму, понятную ок*ружающим,* и обратно. Под окружающими подразумеваются человек (и другие биологические объекты) и различные технические устройства (компьютер можно приспособить для управления любым оборудованием, были бы датчики и исполнительные устройства). В какую форму эти устройства преобразуют двоичную информацию — определяется их назначением.
- *Коммуникационные устройства* служат для передачи информации между компьютерами и/или их частями. Сюда относят модемы (проводные, радио, оптические, инфракрасные...), адаптеры локальных и глобальных сетей. В данном случае преобразование формы представления информации требуется только для передачи ее на расстояние.

Процессор, память и периферийные устройства взаимодействуют между собой с помощью шин и интерфейсов, аппаратных и программных; стандартизация интерфейсов делает архитектуру компьютеров открытой.

1.2. Биты, байты, слова, параграфы

Компьютер работает в двоичной системе счисления — минимальным информационным элементом является бит, который может принимать значение 0 или 1. Этим значениям соответствуют различимые физические состояния ячейки, чаще всего — уровень напряжения (низкий или высокий). Биты организуются в более крупные образования — ячейки памяти и регистры. Каждая ячейка памяти (и каждый регистр) имеет свой *адрес,* однозначно ее Идентифицирующий в определенной системе координат. Минимальной адресуемой (пересылаемой между компонентами компьютера) единицей информации является *байту* состоящий, как правило, из 8 бит[1]. Два байта со смежными адресами образуют

[1] Существуют процессоры и компьютеры с разрядностью обрабатываемого слова, не кратной 8 (например, 5, 7, 9...). Байты таких процессоров не 8-битные, но в мире PC столкновение с ними маловероятно. Кроме того, в некоторых системах (обычно коммуникационных) совокупность восьми соседних битов данных называют октетом. Название «октет» обычно подразумевает, что эти 8 бит не имеют явного адреса, а характеризуются только своим местоположением в длинной цепочке битов.

слово (word) разрядностью 16 бит, два смежных слова — *двойное слово* (double word) разрядностью 32 бита, два смежных двойных слова — *учетверенное слово* (quad word) разрядностью 64 бита. Байт (8 бит) делится на пару *тетрад* (nibble): старшую тетраду — биты [7:4] и младшую тетраду — биты [3:0].

В двухбайтном слове принят *LH-порядок следования байтов:* адрес слова указывает на младший байт L (Low), а старший байт H (High) размещается по адресу, большему на единицу. В двойном слове порядок аналогичен — адрес указывает на самый младший байт, после которого размещены следующие по старшинству. Этот порядок, называемый форматом *Little Endian* и естественный для процессоров Intel, применяется не во всех микропроцессорных семействах. В формате *Big Endian* адрес указывает на самый старший байт (двойного, учетверенного...) слова, остальные байты размещаются по нарастающим адресам. С несовпадением форматов представления приходится сталкиваться, например, при передаче информации между различными подсистемами (формат Big Endian использу-ется в сетевых протоколах и шине FireWire).

В технической документации, электрических схемах и текстах программ могут применяться разные способы представления чисел:

- Двоичные (binary) числа — каждая цифра отражает значение одного бита (0 или 1), старший бит всегда пишется слева, после числа ставится буква «b». Для удобства восприятия тетрады могут быть разделены пробелами, например, 1010 0101b.
- Шестнадцатеричные (hexadecimal) числа — каждая тетрада представляется одним символом 0...9, A, B, ..., F. Обозначаться такое представление может по-разному, в данной книге используется только символ «h» после последней шестнадцатеричной цифры, например, A5h. В текстах программ это же число может обозначаться и как 0xA5, и как 0A5h, в зависимости от синтаксиса языка программирования. Незначащий ноль (0) добавляется слева от старшей шестнадцатеричной цифры, изображаемой буквой, чтобы различать числа и символические имена.
- Десятичные (decimal) числа — каждый байт (слово, двойное слово) представляется обычным числом, а признак десятичного представления (букву «d») обычно опускают. Байт из предыдущих примеров имеет десятичное значение 165. В отличие от двоичной и шестнадцатеричной форм записи, по десятичной трудно в уме определить значение каждого бита, что иногда приходится делать.
- Восьмеричные (octal) числа — каждая тройка битов (разделение начинается с младшего) записывается в виде цифры из интервала 0-7, в конце ставится признак «o». То же самое число записывается как 245o. Восьмеричная система неудобна тем, что байт не разделить поровну, но зато все цифры — привычные. В «про-интеловских» системах это представление непопулярно (у него «DEC'ов-ское» происхождение).

В табл. 1.1 приведены разные представления одной тетрады (4 бит). Чтобы перевести любое 8-битное число в десятичное, нужно десятичный эквивалент старшей тетрады умножить на 16 и сложить с эквивалентом младшей тетрады.

Для нашего примера A5h = 10 x 16 + 5 = 165. Обратный перевод тоже несложен: десятичное число делится на 16, целая часть даст значение старшей тетрады, остаток — младшей.

Таблица 1.1. Представление двоичных чисел в разных системах счисления

Двоичное (bin)	Шестнадцатеричное (hex)	Десятичное (dec)	Восьмеричное (oct)
0000	0	0	0
0001	1	1	1
0010	2	2	2
0011	3	3	3
0100	4	4	4
0101	5	5	5
0110	6	6	6
0111	7	7	7
1000	8	8	10
1001	9	9	11
1010	A	10	12
1011	B	11	13
1100	C	12	14
1101	D	13	15
1110	E	14	16
1111	F	15	17

В «наследство» от процессоров 8086/88 достался своеобразный способ задания адреса ячейки памяти в виде *указателя* «seg:offset», состоящего из двух слов: *сегмента* (seg — segment) и *смещения* (offset). Такая запись предполагает вычисление полного адреса по формуле *addr* = 16 • *seg* + *offset*. Такое представление 20-битного адреса двумя 16-битными числами в процессорах 8086/88 поддерживается и в реальном режиме всех последующих процессоров x86 (подробнее об адресации памяти см. в 7.3). Здесь сегмент указывает адрес *параграфа* — 16-байтной области памяти. Выравнивание адреса по границе параграфа означает, что он кратен 16 (4 младших бита нулевые). Нетрудно увидеть, что один и тот же адрес можно задавать разными сочетаниями этих двух компонентов. Так, например, адрес начала области данных BIOS (BIOS Data Area) 00400h представляют и как 0000:0400, и как 0040:0000 (шестнадцатеричное представление подразумевается). Возможны и другие варианты, но их не используют. В данной книге в основном будем пользоваться первым способом, причем нулевое значение сегмента будем представлять кратко, то есть 0:0400. К счастью, в 32-разрядном (и 64-разрядном) режиме работы процессоров в современных ОС и приложениях сегментация не применяется, а адрес выражается одним (32- или 64-битным) числом.

Обозначение и порядок битов и байтов шин адреса и данных, принятое в аппаратуре PC, пришло от процессоров Intel 8086/88 (и даже от 8080). Самый младший бит (Least Significant Bit, *LSB*) имеет номер 0, самый старший (Most

Significant Bit, *MSB)* бит байта — 7, слова — 15, двойного слова — 31. На рисунках принято старший бит изображать слева, а младший — справа.

1.3. Ячейки памяти, порты и регистры

Поясним разницу между ячейками памяти, портами и регистрами. *Ячейки памяти* служат лишь для хранения информации — сначала ее записывают в ячейку, а потом могут прочитать, а также записать иную информацию. *Порты ввода-вывода*, как правило, служат для преобразования двоичной информации в какие-либо физические сигналы и обратно. Например, порт данных параллельного интерфейса формирует электрические сигналы на разъеме, к которому обычно подключают принтер. Электрические сигналы, поступающие от принтера, порт состояния того же интерфейса отображает в виде набора битов, который может быть считан процессором. *Регистр* — довольно широкое понятие, которое зачастую используется как синоним порта. Регистры могут служить для управления устройствами (и их контроллерами) и для чтения их состояния. Регистры (как и порты) могут образовывать *каналы*:

- *Каналы ввода-вывода данных.* Пример — регистр данных COM-порта: байты, записываемые друг за другом в этот регистр, в том же порядке будут передаваться по последовательному интерфейсу, то есть поступать в *канал вывода.* Если этот интерфейс подключить к COM-порту другого компьютера и выполнять программные чтения его регистра данных, мы получим байт за байтом переданные данные. Таким образом, здесь регистр играет роль *канала ввода.*
- *Каналы управления.* Если запись в регистр определенных данных (битовых комбинаций) изменяет состояние некоего устройства (сигнал светофора, положение какого-то механизма...), то регистр образует канал управления.
- *Каналы состояния.* Пример — регистр игрового порта (game-порт), к которому подключен джойстик. Чтение регистра дает информацию о состоянии кнопок джойстика (нажаты или нет).

Канал отличается от ячейки памяти рядом свойств. Если в *ячейку памяти* записывать раз за разом информацию, то последующее считывание возвращает результат последней записи, а все предшествующие записи оказываются бесполезными. Если ячейку памяти считывать раз за разом, не выполняя запись в нее, то результат считывания каждый раз будет одним и тем же (при исправной памяти). «Лишнее» чтение ячейки памяти не приведет ни к каким побочным эффектам. На этих свойствах «настоящей» памяти основаны методы ускорения работы с ней: кэширование и спекулятивное чтение. С *регистрами, образующими каналы*, такие вольности недопустимы. Здесь все обращения приводят к каким-либо изменениям. Кэширование и спекулятивное чтение недопустимы. Например, лишнее (спекулятивное) чтение регистра данных COM- порта «выдернет» байт из принимаемого потока. Операция чтения регистра состояния может быть неявным подтверждением сброса какого-либо признака (например, запроса прерывания), и она изменяет состояние устройства. Записи в канал данных (и управления) также нельзя опускать (для «ускорения»).

Каждый байт (ячейка памяти, порт, регистр) имеет собственный уникальный *физический адрес*. Этот адрес устанавливается на системной шине процессором, когда он инициирует обращение к данным ячейке или порту. По этому же адресу к этой ячейке (порту, регистру) могут обращаться и другие активные компоненты системы — так называемые *мастера шины*.

В семействе x86 и PC-совместимых компьютерах пространства адресов ячеек памяти и портов ввода-вывода разделены. Это предусмотрено с обеих сторон: процессоры позволяют, а компьютеры используют данное разделение. Нынешние 32-битные процессоры имеют разрядность физического адреса памяти 32 и даже 36 бит, что позволяет адресовать до 4 и 64 Гбайт соответственно. Пространство ввода-вывода использует только младшие 16 бит адреса, что позволяет адресовать до 65 384 однобайтных регистров. Адреса «исторических» системных устройств PC не изменились с самого рождения — это дань совместимости, которая без разделения пространств вряд ли бы обеспечивалась столько лет. Пространства памяти и портов ввода-вывода неравнозначны не только по объему, но и по способам обращения. Способов адресации к ячейке памяти в x86 великое множество, в то время как для адресации ввода-вывода их существует только два. К памяти возможна (и широко используется) виртуальная адресация (см. 7.3), при которой для программиста, программы и даже пользователя создается иллюзия оперативной памяти гигантского размера. К портам ввода-вывода обращаются только по реальным адресам; правда, и здесь возможна виртуализация, но уже чисто программными средствами операционной системы. И, наконец, самое существенное различие пространств памяти и портов ввода-вывода: процессор может считывать инструкции для исполнения только из пространства памяти. Конечно, через порт ввода можно считать фрагмент программного кода (что и происходит, например, при считывании данных с диска), но для того чтобы этот код исполнить, его необходимо записать в память.

Регистры различных устройств могут быть приписаны как к пространству портов ввода-вывода, так и к пространству памяти. Под портом устройства, как правило, подразумевают регистр, связанный с этим устройством и приписанный к пространству портов ввода-вывода. Точность приведенной терминологии, конечно же, относительна. Так, к примеру, ячейки видеопамяти (тоже память!) служат в основном не для хранения информации, а для управления свечением элементов экрана. Понятие *Memory Mapped I/O* означает регистры периферийных устройств, отображенные на пространство памяти (то есть занимающие адреса именно в этом пространстве, а не в пространстве ввода-вывода).

Разделение пространств памяти и ввода-вывода было вынужденной мерой в условиях дефицита адресуемого пространства 16-битных процессоров и сохранилось во всех процессорах x86. В процессорах ряда других семейств такого разделения нет, и для нужд ввода-вывода используется выделенная область единого адресного пространства. Тенденция изживания пространства ввода-вывода наблюдается в современных спецификациях устройств и интерфейсов для PC.

1.4. Подсистемы памяти и хранения данных

Память компьютера предназначена для кратковременного и долговременного хранения информации — кодов команд и данных. В памяти информация хранится в массиве ячеек. Минимальной адресуемой единицей является байт — каждый байт памяти имеет свой уникальный адрес. Память можно рассматривать как иерархическую систему, простирающуюся от кэш-памяти процессора до ленточных архивов.

Со времени появления больших (по размерам) компьютеров сложилось деление памяти на внутреннюю и внешнюю. Под внутренней подразумевалась память, расположенная внутри процессорного «шкафа» (или плотно к нему примыкающая). Сюда входили и электронная и магнитная память (на магнитных сердечниках). Внешняя память представляла собой отдельные устройства с подвижными носителями — накопители на магнитных дисках (а сначала — на барабанах) и ленте. Со временем все устройства компьютера удалось «поселить» в один небольшой корпус, и прежнюю классификацию памяти применительно к PC можно переформулировать так:

- *внутренняя память* — электронная (полупроводниковая) память, устанавливаемая на системной плате или на платах расширения;
- *внешняя память* — память, реализованная в виде устройств с различными принципами хранения информации, чаще всего с подвижными носителями; в настоящее время сюда входят устройства магнитной (дисковой и ленточной) памяти, оптической и магнитооптической памяти; устройства внешней памяти могут размещаться как в системном блоке компьютера, так и в отдельных корпусах, достигающих иногда размеров небольшого шкафа.

Для процессора непосредственно доступной является *внутренняя память*, доступ к которой осуществляется по адресу, заданному программой. Для внутренней памяти характерен одномерный (линейный) адрес, который представляет собой одно двоичное число определенной разрядности. Внутренняя память подразделяется на *оперативную*, информация в которой может изменяться процессором в любой момент времени, и *постоянную*, информацию в которой процессор может только считывать. Обращение к ячейкам оперативной памяти может происходить в любом порядке, причем как по чтению, так и по записи, поэтому оперативную память называют памятью с произвольным доступом (Random Access Memory, RAM) — в отличие от постоянной памяти (Read Only Memory, ROM).

Внешняя память адресуется более сложным образом — каждая ее ячейка имеет свой адрес внутри некоторого *блока*, который, в свою очередь, имеет многомерный адрес. В ходе физических операций обмена данными блок может быть считан или записан только целиком. В случае одиночного дискового накопителя физический адрес блока является трехмерным — он состоит из номера поверхности (головки), номера цилиндра и номера сектора. В современных накопителях этот трехмерный адрес часто заменяют линейным номером — логическим

адресом блока, а его преобразованием в физический адрес занимается внутренний контроллер накопителя. Поскольку дисковых накопителей в компьютере может быть множество, в адресации дисковой памяти участвуют также номер накопителя и номер канала интерфейса. С такой сложной системой адресации процессор справляется только с помощью программного драйвера, в задачу которого в общем случае входит копирование некоторого блока данных из оперативной памяти в дисковую и обратно. Название «дисковая память» широко применяется для *внешней памяти с прямым доступом*; словосочетание «прямой доступ» подразумевает возможность обращения к блокам (но не к его ячейкам!) с чередованием операций чтения и записи в произвольном порядке. Память с *последовательным доступом* накладывает ограничения на свободу: в ней невозможны произвольное чередование операций чтения/записи и произвольность адресов. Ряд устройств запись вообще не выполняют (например, CD- ROM). Последовательный метод доступа используется в ленточных устройствах, а также в большинстве оптических дисков (CD, DVD). С такими неудобствами обращения мирятся только из-за того, что устройства последовательного доступа обеспечивают самое дешевое хранение больших объемов информации, к которой не требуется оперативного доступа:

Ниже перечислены наиболее важные параметры подсистемы памяти.

- *Объем хранимой информации.* Нет необходимости объяснять, что чем он больше, тем лучше. Максимальный (в принципе — неограниченный) объем информации хранят ленточные и дисковые устройства со сменными носителями, за ними идут дисковые накопители, и завершает этот ряд оперативная память.
- *Время доступа* — усредненная задержка начала обмена полезной информацией относительно появления запроса на данные. Минимальное время доступа имеет оперативная память, за ней идет дисковая, после нее — ленточная.
- *Скорость обмена* при передаче потока данных (после задержки на время доступа). Максимальную скорость обмена имеет оперативная память, за ней идет дисковая, после нее — ленточная.
- *Удельная стоимость хранения единицы данных* — цена накопителя (с носителями), отнесенная к единице хранения (байту или мегабайту). Минимальную стоимость хранения имеют ленточные устройства со сменными носителями, их догоняют дисковые накопители, а самая дорогая — оперативная память.

Помимо этих имеется и ряд других характеристик — энергонезависимость (способность сохранения информации при отключении внешнего питания), устойчивость к внешним воздействиям, время хранения, конструктивные особенности (размер, вес) и т. п. У каждого типа памяти есть различные реализации со своими достоинствами и недостатками.

Внутренняя и внешняя память

Внутренняя и внешняя память используются существенно различными способами. Внутренняя (оперативная и постоянная) память является хранилищем

программного кода, который непосредственно может быть исполнен процессором. В ней же хранятся и данные, также непосредственно доступные процессору (а следовательно, и исполняемой программе). Внешняя память обычно используется для хранения файлов, содержимое которых может быть произвольным. Процессор (программа) имеет доступ к содержимому файлов только опосредованно, через отображение их (полное или частичное) на некоторую область оперативной памяти. Исполнить программный код или обратиться к данным непосредственно на диске процессор не может в принципе. То же относится, естественно, и к ленточной памяти.

Однако реальная жизнь многообразнее этой упрощенной схемы, и на практике дисковая и оперативная память переплетаются сложным образом. Главные недостатки дисковой памяти — большое время доступа и низкая скорость обмена — устраняются с помощью *виртуального диска*, представляющего собой своеобразно используемую область оперативной памяти. В этой области хранятся файлы, и с точки зрения операционной системы (и, тем более, прикладной программы) она выглядит как обычный, но очень быстрый диск. Конечно же, его объем ограничен, и этот объем вычитается из объема физически установленной памяти, доступной процессору в качестве обычной оперативной. Кроме того, виртуальный диск в отличие от реального не является энергонезависимым. Более того, информация на нем не переживет даже перезагрузки операционной системы. Однако несмотря на эти ограничения виртуальный диск во многих случаях может повысить эффективность работы компьютера при интенсивном дисковом обмене. В операционной системе виртуальный диск реализуется загрузкой программного драйвера, как правило, с именем RAMDRIVE.SYS (в некоторых версиях — VDISK.SYS). Другим способом решения проблем быстродействия дисковой памяти за счет оперативной является *кэширование дисков* — хранение образов последних из использованных блоков дисковой памяти в оперативной в надежде на то, что вскоре будет следующий запрос к ним, который удастся удовлетворить из памяти. В Windows 9x/NT кэширование возложено на операционную систему, в MS-DOS кэшированием дисков занимается загружаемый драйвер SMARTDRV.EXE, но даже и без этого драйвера «неглубокое» кэширование выполняет операционная система (ОС), и этим процессом можно управлять с помощью строки BUFFERS=xxx файла CONFIG.SYS. Если затребованный с диска блок уже находится в одном из буферов, ОС не будет «беспокоить» диск, а удовлетворит запрос из буфера. Чем больше значение xxx, тем больше блоков может держать ОС в оперативной памяти, но область памяти для буферов, естественно, уменьшает объем памяти, доступной программам.

Основной недостаток оперативной памяти заключается в том, что конструктивно достижимый объем ее во много раз меньше, чем дисковой (пока что это было справедливо на всех ступенях технического прогресса). Решить проблему увеличения объема оперативной памяти за счет дисковой позволяет *виртуальная память*, которую можно считать кэшированием оперативной памяти на диске. Суть ее заключается в том, что программам предоставляется виртуальное пространство оперативной памяти, по размерам превышающее объем физически установленной оперативной памяти. Это виртуальное пространство разбито на

страницы фиксированного размера, а в физической оперативной памяти в каждый момент времени присутствует только часть из них. Остальные страницы хранятся на диске, откуда операционная система может их «подкачать» в физическую память на место предварительно выгруженных на диск страниц. Процесс замещения страниц называется *свопингом* (swapping), а области дисковой памяти, выделяемые для этих целей, — файлами подкачки, или своп-файлами (swap file). Для прикладной программы этот процесс прозрачен (если только она не критична ко времени обращения к памяти). Для пользователя этот процесс заметен по работе диска в те моменты, когда не требуется обращение к файлам. Расплатой за почти безмерное увеличение объема доступной оперативной памяти является снижение средней производительности обращений к памяти и некоторый расход дисковой памяти на файл подкачки. Естественно, размер виртуальной памяти не может превышать суммы размеров ОЗУ и дисковой памяти. Виртуальная память с подкачкой страниц реализуется операционными системами защищенного режима (например, OS/2, MS Windows) на основе аппаратных средств 32-разрядных процессоров (386 и выше), а теперь и 64-разрядных.

Таково в общих чертах устройство подсистемы памяти. Здесь для упрощения изложения опущена система кэширования оперативной памяти, которая для современных процессоров чаще всего является двухуровневой. Об этом подробнее можно прочесть в главе 7. Кроме того, не упоминалось автономное кэширование дисковых устройств, реализуемое в самих накопителях и их высокопроизводительных контроллерах. Эта тема раскрывается в главе 9.

В общем случае в подсистему памяти обязательно входят оперативная память и энергонезависимая память, хранящая, по крайней мере, программу первоначальной загрузки компьютера. Дисковая память как таковая может и отсутствовать. Однако внешняя память с прямым доступом в том или ином виде — будь то действительно дисковые накопители, флэш-диск (не имеющий круглых, а тем более вращающихся деталей) или сетевой диск, отображающий часть диска физически значительно удаленного компьютера-сервера, — является обязательным атрибутом персонального компьютера. Благодаря применению дисковой памяти компьютер становится универсальным устройством, способным выполнять великое множество прикладных программ, интересующих пользователя. Эти программы загружаются в память именно с дисков. Без внешней памяти компьютер вырождается в узкоспециализированное устройство с ограниченным набором функций (например, функций эмуляции терминала или функций интерпретатора языка Basic), «зашитых» в его постоянную память, объем которой не может быть большим по технико-экономическим причинам.

Диски и файловые системы

Дисковая память в компьютере используется для хранения *файлов* — цепочек байтов, несущих любую информацию. Для данного обсуждения содержимое файлов несущественно — в них могут быть исполняемые программные модули, данные (числовые, текстовые, мультимедийные) и их смесь. Способ размещения файлов на диске и возможности манипуляций с файлами определяются

файловой системой. Каждый файл имеет набор атрибутов, состав которого зависит от используемой файловой системы. «Прожиточный минимум» — это имя файла, его длина и время (и дата) последней модификации; в простейшей файловой системе MS-DOS к этому минимуму добавляются атрибуты системного, скрытого, архивированного и только читаемого файла. Имя файла может задаваться как в классической форме «8.3» (8 символов на имя плюс 3 символа на тип), понимаемой всеми операционными системами, так и в почти произвольной длинной форме, характерной для более сложных ОС (OS/2, Windows 9x/NT/200x/XP, Unix...). Файловая система включает в себя систему каталогов и системы размещения файлов на диске, простейшей из которых можно считать FAT в MS-DOS. Эти системы определяют возможности и эффективность манипулирования файлами — создания, записи, чтения, поиска, модификации, удаления, восстановления удаленных файлов, — а также средства восстановления файловой системы после сбоев, вызванных неисправностями и некорректными действиями пользователей или программ.

Для каждой операционной системы характерны свои файловые системы (одна или несколько), которые она «понимает». Компьютер с незагруженной ОС «не понимает» ни одной файловой системы (в этом его универсальность). Изначально были приняты соглашения, позволяющие «голому» компьютеру загрузить ОС с так называемого *системного диска*, причем самая первая фаза загрузки выполняется без какой-либо файловой системы.

1.5. Устройства ввода-вывода и коммуникаций

Устройства ввода-вывода связывают компьютер с внешним миром, без них он был бы «вещью в себе». Список устройств, делающих компьютер «вещью для нас», практически не ограничен. К ним относятся дисплеи (устройства отображения, то есть вывода), клавиатура и мышь (устройства ввода), принтеры и сканеры, плоттеры и дигитайзеры, джойстики, акустические системы и микрофоны, телевизоры и видеокамеры и прочие устройства в великом множестве их разновидностей. Любопытно, что в этих парах обычно лидируют устройства вывода, появившиеся в компьютерах раньше соответствующих устройств ввода. Благодаря фантазии и техническому прогрессу появляются все новые и новые устройства; так, например, шлем виртуальной реальности из области фантастики перешел в производственно-коммерческую область. К компьютеру можно подключать датчики и исполнительные устройства технологического оборудования, различные приборы — в общем, все, что в итоге может вырабатывать электрические сигналы и/или ими управляться.

Консолью компьютера называют его «выступающую часть», обращенную к пользователю. В PC стандартной консолью являются клавиатура (устройство ввода) и дисплей, позволяющий отображать символьную информацию (устройство вывода). На консоль выводятся все системные сообщения; с консоли можно управлять компьютером — запускать и принудительно завершать приложения,

выполнять перезагрузку ОС и управлять процессом загрузки. Вместо стандартных устройств роль консоли могут играть и иные устройства, способные выводить и вводить символы и имеющие необходимую программную поддержку.

Коммуникационные устройства связывают компьютеры (и другие устройства) в сложные системы, составные части которых могут находиться довольно далеко друг от друга. Коммуникационные устройства обеспечивают передачу информации самого разного назначения. К этим устройствам относятся модемы, адаптеры локальных и глобальных сетей. Соответствующий набор устройств ввода-вывода и коммуникаций позволяет превратить персональный компьютер, например, в факс-машину, аппарат IP-телефонии (голосовой) или видеоконференцсвязи.

1.6. Адаптеры, контроллеры и иерархия подключений периферийных устройств

Компоненты компьютера соединяются друг с другом иерархией средств подключения, наверху которой стоят *интерфейсы системного уровня подключения.* Для этой группы интерфейсов характерно то, что в их транзакциях фигурируют *физические адреса пространства памяти* и (если есть) *пространства ввода-вывода.* Группа связанных между собой интерфейсов системного уровня образует логическую *системную шину компьютера.* Системную шину составляют следующие физические интерфейсы:

- шина подключения центрального процессора (или нескольких процессоров в сложных системах) — FSB (Front Side Bus — фасадная шина)[1];
- шина подключения контроллеров памяти, оперативной и постоянной; собственно шина памяти (memory bus) системной уже не является, поскольку в ней фигурируют не системные адреса, а адреса физических банков памяти;
- шины ввода-вывода, обеспечивающие связь между центральной частью компьютера и периферийными устройствами.

Типичные представители шин ввода-вывода в IBM PC — шина ISA (отмирающая), а также шины PCI (развивающаяся в PCI-X) и PCI-E (PCI Express)[2]. Через шины ввода-вывода проходят все обращения центрального процессора (ЦП) к периферии. К шинам ввода-вывода подключаются *контроллеры* и *адаптеры* периферийных устройств или их интерфейсов.

Адаптер является средством сопряжения какого-либо устройства с какой-либо шиной или интерфейсом компьютера. *Контроллер* служит тем же целям сопряжения, но при этом подразумевается его некоторая активность — способность к самостоятельным действиям после получения команд от обслуживающей его

[1] Это понятие в ряде источников отождествляют с системной шиной, но в данной книге будем пользоваться более широким толкованием понятия «системная шина».

[2] Систему двухточечных соединений PCI Express шиной называть не совсем корректно, но она обеспечивает ту же функциональность, что и предшествующие ей шины расширения.

программы. Сложный контроллер может иметь в своем составе и собственный процессор. На эти тонкости терминологии не всегда обращают внимание, и понятия «адаптер» и «контроллер» считают почти синонимами. Для взаимодействия с периферийными устройствами процессор обращается к регистрам контроллера (адаптера), «представляющего интересы» подключенных к нему устройств.

Часть периферийных устройств (ПУ) совмещена со своими контроллерами (адаптерами), как, например, сетевой адаптер Ethernet, подключенный к шине PCI. Другие же ПУ подключаются к своим контроллерам через промежуточные *периферийные интерфейсы*, находящиеся на нижнем уровне иерархии подключений. Периферийные интерфейсы — самые разнообразные из всех аппаратных интерфейсов. К периферии, подключаемой через промежуточные интерфейсы, относится большинство устройств хранения (дисковые, ленточные), устройств ввода-вывода (дисплеи, клавиатуры, мыши, принтеры, плоттеры), ряд коммуникационных устройств (внешние модемы).

Для взаимодействия с программой (с помощью процессора или сопроцессоров) адаптеры и контроллеры обычно имеют *регистры* ввода-вывода, управления и состояния, которые могут располагаться либо в адресном пространстве памяти, либо в пространстве портов ввода-вывода. Кроме того, используются механизмы аппаратных прерываний для сигнализации программе о событиях, происходящих в периферийных устройствах. Для обмена информацией с устройствами применяют также механизмы прямого доступа к памяти (Direct Memory Acces, *DMA)* и прямого управления шиной. Контроллер, который способен инициировать транзакции на системной шине, является активным компонентом компьютера. С помощью транзакций он может обращаться к другим устройствам (точнее, их контроллерам или адаптерам), обеспечивая равноранговое взаимодействие. Чаще всего ограничиваются взаимодействием контроллера с системной памятью (это проще).

1.3. Программное обеспечение

Ранее в общих чертах было рассмотрено устройство компьютера (естественно, подразумевается наличие корпуса с блоком питания). Однако этот набор «железок» не имеет практической ценности без программного обеспечения (ПО), которое в компьютере имеет многоуровневую организацию.

Неотъемлемой от компьютера частью программного обеспечения является *базовая система ввода-вывода* (Basic Input-Output System, BIOS), которая хранится в постоянной (энергонезависимой) памяти ROM BIOS (ПЗУ базовой системы ввода-вывода). В ROM BIOS находится *программа инициализации*, называемая POST (PowerOn Self Test — самотестирование по включению), которая обеспечивает тестирование и запуск компьютера при включении, а также загрузку операционной системы. В ROM BIOS содержатся процедуры для работы со стандартными устройствами, реализующие связь операционной системы и прикладных программ с аппаратными средствами компьютера. BIOS предоставляет такие сервисы, как ввод символа с клавиатуры, вывод на экран или принтер, чтение-запись сектора на диске и ряд других (см. главу 5). BIOS нахо

дится на самым нижнем уровне ПО, который обеспечивает изоляцию вышестоящих уровней от подробностей реализации аппаратных средств компьютера. В ROM BIOS имеется также утилита CMOS Setup, обеспечивающая настройку аппаратных средств компьютера.

Следующий уровень — *операционная система* (ОС), основным назначением которой является загрузка прикладных программ и предоставление им некоторых сервисов. Сервисы ОС функционируют на более высоком уровне — если BIOS работает с физическими устройствами, то ОС предоставляет возможность работы на логическом уровне. Сервисы ОС, обслуживающие стандартные устройства, могут обращаться к соответствующим сервисам BIOS; они расширяют функциональность сервисов BIOS, а также выполняют обработку ошибок физических устройств. ОС может работать с системными устройствами и в обход BIOS, через собственные драйверы. В MS-DOS все сервисы ОС обслуживались через BIOS, что обеспечивало высокий уровень совместимости и переносимости ПО с машины на машину, но ценой невысокой эффективности. Более современные ОС работают в обход BIOS — это эффективнее с точки зрения производительности, но осложняет переносимость ПО. ОС ведает распределением системных ресурсов. На устройствах хранения она организует файловую систему. Операционная система, как правило, загружается с устройства внешней памяти (локального или сетевого диска), хотя для специальных применений (во встроенных компьютерах) встречаются и так называемые резидентные ОС, «зашитые» в ПЗУ. Для загрузки ОС требуется специально подготовленный *системный диск*. В самом начале системного диска располагается *загрузчик* — короткая программа, загружающая несколько файлов ядра операционной системы в память и передающая им управление. Эти файлы находятся на том же системном диске в месте, известном загрузчику (он должен найти файлы еще до того, как будет обеспечена поддержка файловой системы со стороны ОС). Программный код загрузчика привязан к загружаемой ОС и файловой системе диска, но сам загрузчик для любых ОС и дисков запускается единым способом (см. 9.6). Если загрузчик на своем диске находит необходимые файлы операционной системы (например, IO.SYS и MSDOS.SYS), он загружает их в оперативную память и передает управление по определенному адресу. С этого момента работой компьютера управляет ОС, она загружает все свои компоненты, выполняет нужные настройки и подготавливается для загрузки и исполнения приложений, предоставляя им сервисы файловой системы (см. 9.11). Естественно, чем сложнее ОС, тем больший объем памяти ей требуется. Первые ОС помещались на гибких дисках небольшого объема (180 Кбайт) и довольно быстро с них загружались. ОС Windows (и другие современные ОС) занимают десятки (а то и сотни) мегабайт и загружаются довольно долго (минуты) даже на самых быстрых ПК. Но, к счастью, на гибком диске вполне умещается минимальный вариант ОС, достаточный для «понимания» файловой системы Windows и запуска приложений и утилит восстановления. Так что на случай аварии «большой» ОС можно иметь «спасательную» дискету, достаточную для начала «аварийно-спасательных работ».

И наконец, верхний уровень иерархии ПО — *прикладное программное обеспечение*, ради исполнения которого и «городился весь этот огород». Прикладные программы могут пользоваться сервисами ОС, BIOS, а также обращаться к аппаратным средствам компьютера напрямую, адресуясь к портам и ячейкам памяти. Чем ближе прикладная программа к «железу», тем эффективнее она может с ним работать (вызовы сервисов BIOS, а тем более ОС вносят дополнительные накладные расходы на организацию программных интерфейсов). Однако использование сервисов высокого уровня (BIOS, а тем более ОС) страхует от возможных проблем совместимости программного обеспечения с аппаратными средствами компьютера. Прикладные программы, как и ОС как правило, загружаются с устройств внешней памяти. Именно возможность загрузки любой прикладной программы в совокупности с неограниченным ассортиментом подключаемых устройств и позволяет считать персональный компьютер универсальным инструментом с неограниченными возможностями.

Важными компонентами программного обеспечения являются *драйверы* (driver — буквально, «водитель») — программные модули, содержащие процедуры работы с устройствами. Необходимость выделения драйверов в отдельные модули вполне очевидна: устройство определенного назначения (к примеру, дисплейный адаптер) может иметь самые разные реализации — от MDA до современных видеокарт с трехмерными акселераторами. Если бы не было драйверов, то программист, разрабатывающий прикладную программу, должен был бы включать в нее множество аппаратно-зависимых процедур, причем для всех известных ему моделей дисплейных адаптеров. На написание этих процедур он потратил бы массу времени, программа неимоверно «распухла» бы, а появление новых дисплейных адаптеров потребовало бы модернизации прикладной программы. К этому добавим еще и богатые возможности для ошибок (надо заметить, что ошибки на стыке программ и «железа» довольно трудно «ловить»). Выделение драйверов в отдельные модули избавляет от этих и других неудобств. Драйвер хорошо «знает» программную модель и особенности эффективной работы со своим устройством. Для прикладных программ или операционной системы драйвер представляет *набор сервисов* с интерфейсом, понятным «потребителю». Для каждого сервиса известны способ вызова (программное прерывание или точка входа в процедуру), а также местоположение входных и выходных данных. Если говорить о дисплейных адаптерах, то базовые сервисы (например, очистка экрана, вывод символа в определенную позицию в телетайпном режиме) вызываются через прерывание Int 10h, которое обслуживает BIOS, а параметры передаются через регистры процессора. В зависимости от типа установленного дисплейного адаптера этот сервис обслуживается либо системной микросхемой BIOS, либо микросхемой ROM BIOS, расположенной на графической карте (см. главу 10). Драйверы, обслуживающие данный сервис, специально загружать не требуется — они подставляются автоматически на этапе инициализации BIOS. Однако более сложные сервисы графической карты, которыми, например, пользуется ОС Windows, реализуются отдельными загружаемыми драйверами, «предъявляемыми» на этапе установки ОС. Плохой драйвер может быть источником самых разных неприятных эффектов, вплоть до «зависания» компьютера и даже разрушения ОС. А если речь идет о драйве

рах устройств хранения, то под угрозой оказываются и файлы, хранящиеся на этих устройствах. Плохие драйверы часто появляются из-за желания производителей побыстрее выпустить новый продукт, пускай даже в еще «сыром» виде. Качественный драйвер работает без побочных эффектов, а очень качественный — еще и быстро. Хорошим тоном для производителей устройств является поддержка своих изделий для постоянно меняющихся версий операционных систем и прикладных программ. К сожалению, некоторые модели со временем оказываются заброшенными, и их не удается использовать с новыми продуктами не потому, что последние для них непригодны, а только из-за отсутствия нужного драйвера.

ГЛАВА 2

Устройство персонального компьютера

Персональный компьютер общего назначения имеет как минимум три составные части — системный блок, клавиатуру и дисплей. Клавиатуру с дисплеем можно назвать одним словом — *консоль* («выступающая» часть компьютера, обращенная к оператору-пользователю). Этот минимум может расширяться дополнительными устройствами, в том числе манипуляторами (мышь, трекбол, джойстик), устройствами вывода (принтер, плоттер), устройствами ввода (сканер, считыватели штрих-кодов и магнитных карт), мультимедийными устройствами (аудио и видео), дополнительными устройствами хранения данных (стример, дисковые устройства), коммуникационными устройствами (модем, адаптер локальной сети, телефон) и рядом других. Эти дополнительные устройства либо встраиваются в системный блок, либо являются отдельными «коробками», подключаемыми к системному блоку.

Устройства, подключаемые через интерфейсы последовательных шин (USB и FireWire), физически (кабелями) могут подключаться и к дисплею (или подставке под него), который теперь иногда выполняет и функцию кабельного центра (хаба). Это удобно, поскольку системный блок не обязательно держать под рукой (его можно спрятать под стол), в то время как дисплей всегда должен быть перед глазами. Хотя логически все эти устройства остаются подключенными к системному блоку, кабельное хозяйство становится «элегантнее» и переключения выполняются проще. Некоторые периферийные устройства (чаще всего принтеры) могут иметь беспроводное соединение с системным блоком — через инфракрасный порт или по радиосвязи Bluetooth.

Системный блок является центральным блоком компьютера, определяющим его основные характеристики — производительность (тип и тактовую частоту) процессора, объем оперативной и дисковой памяти, графическую систему, аудиосистему и ряд других. Системный блок включает ряд обязательных компонентов:

- системную плату (см. главу 6);
- дисплейный адаптер (см. главу 10);
- устройства дисковой памяти (см. главу 9);
- набор разъемов для подключения внешних устройств;

- блок питания, систему охлаждения или просто вентиляции (см. главу 3), особо актуальную для высокочастотных процессоров, высокооборотных винчестеров и графических адаптеров с мощными акселераторами.

Дисплейный адаптер и дисковая память в некоторых случаях могут и отсутствовать. В слоты системной платы могут быть установлены дополнительные карты расширения. Системный блок может иметь разные конструктивные исполнения, и перечисленные элементы в них компонуются по-разному.

2.1. Настольные компьютеры

Конструктивные решения, заложенные в первую модель IBM PC образца 1981 года, без каких-либо революционных изменений дошли до наших дней. В классическом варианте исполнения PC состоит из системного блока, к которому подключаются клавиатура, видеомонитор и все периферийные устройства. В *системном блоке* (рис. 2.1) расположена *системная* (system board), или *материнская* (motherboard), *плата* с установленными на ней центральными компонентами компьютера — процессором, оперативной памятью, вспомогательными схемами и щелевыми разъемами-слотами, в которые можно устанавливать платы расширения. В корпусе системного блока имеются *отсеки* (bay) для установки дисковых накопителей и других периферийных устройств трех- и пятидюймового формата, а также блок питания. На задней стенке корпуса имеются отверстия для разъемов клавиатуры и некоторых других, а также щелевые прорези, через которые из корпуса выходят внешние разъемы, установленные на платах расширения. Плата (карта) расширения имеет краевой печатный разъем, которым она соединяется со слотом шины ввода-вывода, и металлическую скобу, закрепляющую плату на корпусе. На этой скобе могут быть установлены внешние разъемы.

Габаритные и присоединительные размеры плат, способы их крепления и шины ввода-вывода унифицированы, что превращает персональный компьютер в увлекательный конструктор, в который «играют» миллионы пользователей.

ИМАНИЕ----------

Играя с этим конструктором, не забывайте закреплять платы расширения полагающимся для этого винтом — небрежность в этом деле может дорого обойтись: выпадение плат, равно как и их установка и изъятие при включенном питании, обычно приводит к выгоранию большого числа компонентов (проверено!).

Унификация системных плат, корпусов и плат расширения обеспечивается следующими конструктивными соглашениями:

- стандартизацией размеров, количества контактов и электрического интерфейса слотов шин расширения;
- фиксированным расстоянием от слота до задней кромки платы;
- фиксированным шагом между соседними слотами, а также их привязкой к крепежным точкам и положению внешних разъемов (на платах XT и AT — привязка только к разъему клавиатуры);

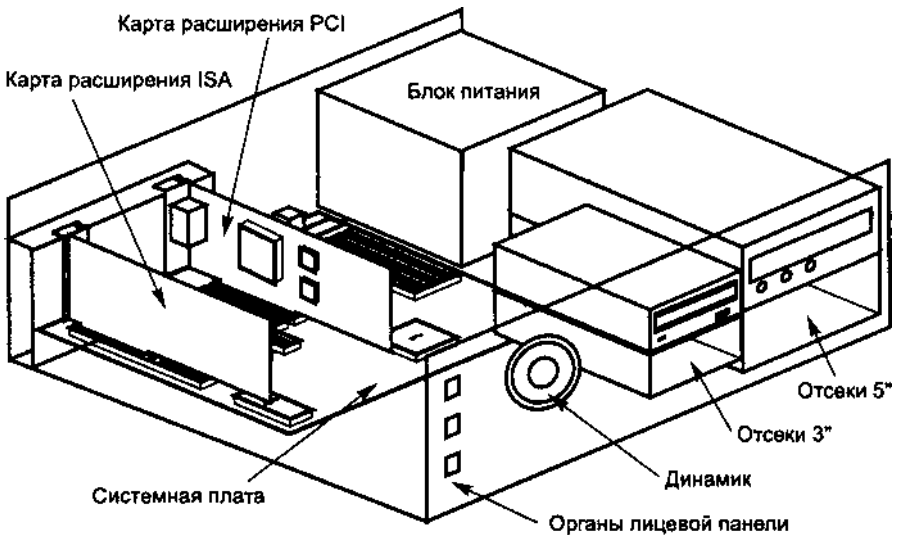


Рис. 2.1. Компоновка традиционного системного блока

- определением максимальных габаритов (длина и высота) карт расширения;
- определением геометрии нижнего края платы расширения, формы и размеров фиксирующей скобки.

Крепежные скобки вместе с некоторым разбросом размеров корпуса иногда доставляют немало забот. Может оказаться, что затягивание винта приводит к выдергиванию платы слота. В этом случае нужно попробовать сместить крепежную скобку относительно платы или подогнуть ее отогнутый конец. Любым способом надо обеспечить вхождение платы в слот до упора и фиксацию винтом в правильном положении. Бывают еще и неприятности из-за неправильного закрепления системной платы — при попытке плотно вставить плату адаптера системная плата прогибается, причем из слотов могут выскочить соседние платы.

Изначально системный блок ставился на стол горизонтально, и этот тип корпуса называется *desktop* (настольный). Корпуса были довольно громоздкие, но со временем за счет уменьшения площади системной платы удалось сократить их длину. Так появился формат корпуса (и системной платы) *baby-AT* («детка»), а традиционные корпуса и платы получили титул *full-AT* (полноразмерные). В настоящее время под корпусом desktop подразумевается корпус длиной около 35 см (чуть длиннее, чем baby). Сверху на настольные корпуса часто устанавливают монитор (хотя при этом его экран оказывается слишком высоко), а перед корпусом располагается клавиатура. Вся эта композиция занимает слишком много места, особенно в глубину, и на обычном столе помещается плохо (оттого и появилась «компьютерная» мебель). Позже догадались поставить корпус «на попа», слегка изменив расположение отсеков внешних устройств. Так появился тип корпуса *tower* (башня), наиболее популярный в настоящее время. В него можно устанавливать системные платы и карты расширения тех

же форматов, что и в desktop, но конструктивно он лучше и удобнее за счет наличия жесткого скелета-шасси. Корпуса типа tower могут иметь разные размеры, в зависимости от которых их устанавливают на стол либо рядом со столом на полу или какой-либо подставке. При напольной установке могут возникнуть проблемы с длиной кабелей подключения клавиатуры и монитора, но эти проблемы разрешимы с помощью специальных удлинителей. В принципе, не возбраняется и укладка корпуса tower на стол горизонтально, тогда на него можно поставить не очень тяжелый монитор. Однако при этом, в отличие от корпуса desktop, отсеки для накопителей окажутся расположенными вертикально. В таком положении могут возникнуть трудности с использованием привода CD/ DVD, если этот привод не рассчитан на работу в таком положении.

Корпус *mini-tower* является самой маленькой «башней» — он имеет высоту около 35 см, ширину 17-18 см (чуть шире отсека 5"), глубину около 40 см и всего два отсека формата 5". Из трех-четырех отсеков 3" на лицевую панель могут выводиться всего два.

Корпус *midi-tower* несколько больше — он имеет высоту около 40 см и по крайней мере три отсека формата 5".

Корпус *big-tower* имеет высоту около 60 см и пять-шесть отсеков формата 5". Эти корпуса обычно шире (для устойчивости и лучшего охлаждения внутренних устройств). Есть и более емкие корпуса — *super big-tower* и другие, предназначенные для компьютеров-серверов.

Корпуса, иногда жаргонно называемые кейсами (case), могут иметь различные конструктивные особенности и дополнительные элементы — например, запираемые или просто пылезащитные дверцы на отсеках накопителей, элементы блокировки несанкционированного доступа, средства контроля внутренней температуры и т. п. Блоки питания широко распространенных корпусов имеют унифицированный конструктив, но, в зависимости от размера корпуса, различные мощность и количество разъемов для питания накопителей.

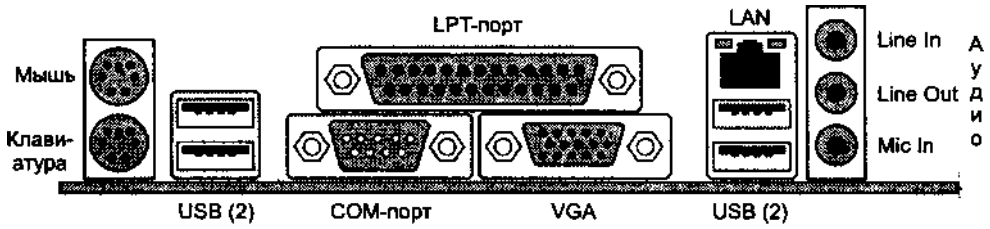


Рис. 2.2. Пример окна с периферийными разъемами платы ATX

В конце 90-х годов был принят стандарт на конструктив системной платы и корпуса — *ATX*, отличающийся от «классического» конструктива AT. Этот конструктив появился в связи с тенденцией расположения максимального числа периферийных контроллеров на системной плате, что затруднило вывод их внешних разъемов. В ATX на задней стенке корпуса предусмотрено окно, в которое выходят разъемы периферии, установленной на системной плате (рис. 2.2),

в корпусах AT было только отверстие для разъема клавиатуры. Кроме того, формат ATX способствует порядку и во внутренних соединениях системного блока, а также имеет другой интерфейс блока питания. Системная плата ATX без проблем устанавливается только в корпус ATX (но не AT), а любые «гибридные» варианты (AT-ATX) проблематичны.

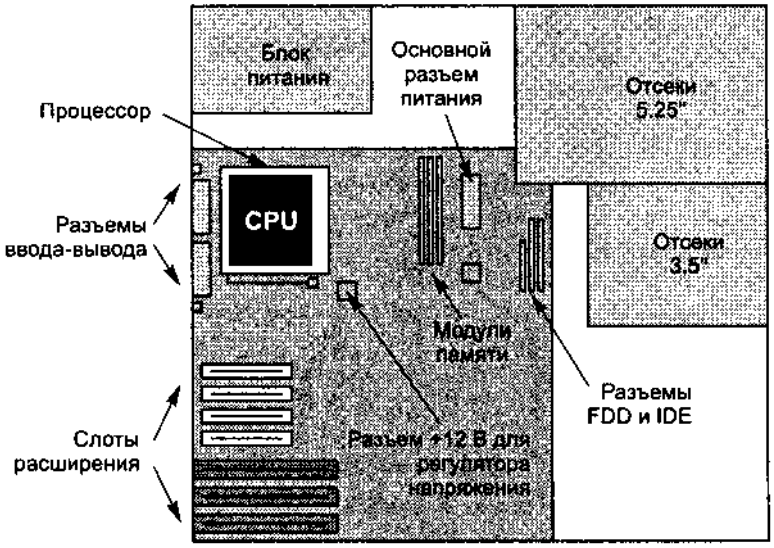


Рис. 2.3. Компоновка элементов в корпусе ATX tower

В конце 2003 года фирма Intel выпустила *спецификацию интерфейсов BTX* (Balanced Technology Extended). Цель спецификации — обеспечить стандартизацию интерфейсов и конструктивов настольных компьютеров с учетом электрических, тепловых и механических характеристик компонентов. В спецификации учитываются и балансируются особенности электрической разводки печатных проводников, тепловых потоков, габаритов (высоты) различных компонентов системной платы и компьютера в целом. Спецификация позволяет конструировать корпуса разного размера, в том числе и специфические для «развлекательных» компьютеров (entertainment PC). Примеры компоновок компьютеров в конструктивах ATX и BTX приведены на рис. 2.3 и 2.4. Как видно из этих примеров, расположение компонентов по сравнению с ATX изменилось на почти зеркальное (конечно, «сбалансированная технология» этим изменением не исчерпывается). Для подключения блока питания используются те же разъемы, что и для ATX12V.

Для настольного исполнения существуют различные модели корпусов уменьшенных размеров. Главным образом стремятся уменьшить высоту, которая для горизонтально расположенных корпусов определяется допустимой высотой плат расширения. В *низкопрофильных корпусах* (slim line) платы расширения располагают в плоскостях, параллельных плоскости системной платы. Они устанавливаются в специальную *переходную плату* (riser card). На рис. 2.5 пока-

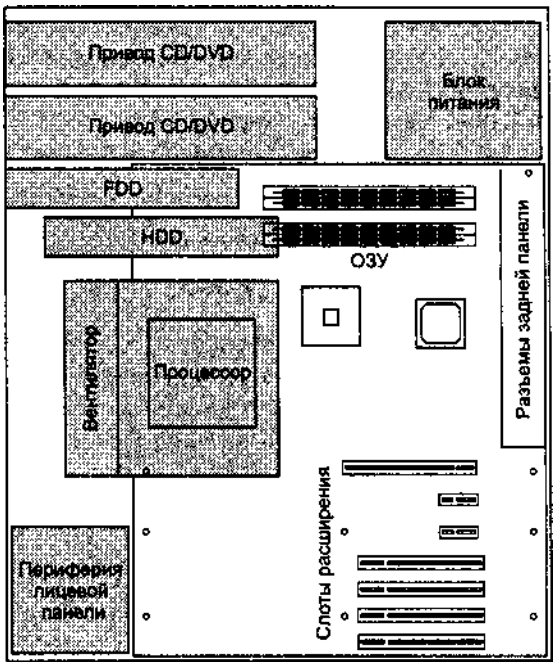


Рис. 2.4. Компоновка элементов в конструктиве BTX tower

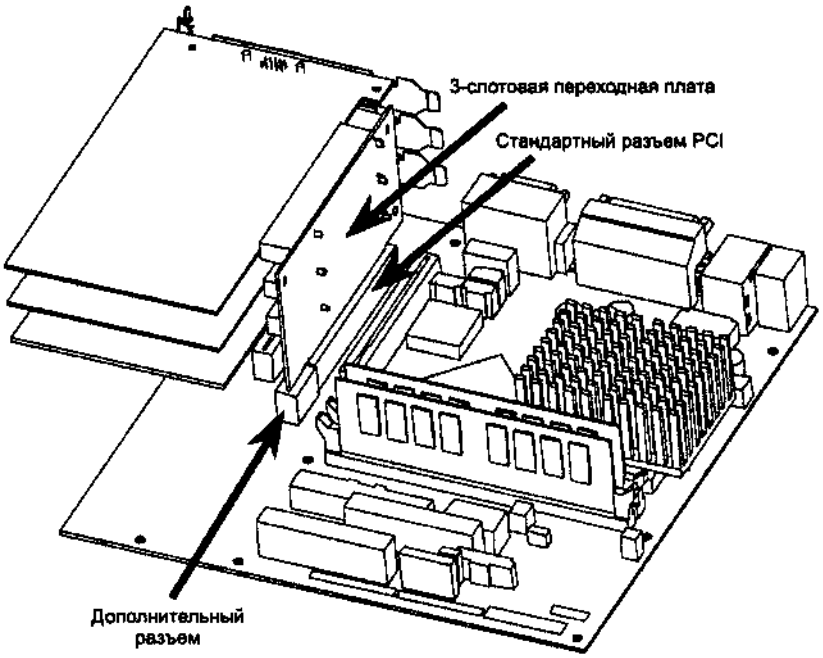


Рис. 2.5. Плата ATX, переходная плата и карты расширения

зана переходная плата для карт PCI, устанавливаемая в специальный слот системной платы формата ATX. От обычного слота PCI он отличается наличием дополнительной 22-контактной секции, через контакты которой выводятся специфические сигналы PCI, идущие к каждому слоту индивидуально. Такой слот устанавливается на системную плату, специально предназначенную для низкопрофильных корпусов. Однако на практике подобные платы встречаются редко, обычно используются рядовые системные платы. При этом индивидуальные сигналы на слоты переходной платы снимаются со свободных слотов системной платы. Возникающие при этом проблемы и способы борьбы с ними описаны в 14.8.

Все вышеперечисленные типы корпусов позволяют использовать стандартные платы расширения и довольно широкий ассортимент системных плат, то есть «конструктор» является универсальным и возможности модернизации не упираются в необходимость приобретать изделия одного производителя. Однако существуют и «фирменные» типы корпусов, в которые могут устанавливаться только «родные» им системные платы. Что касается карт расширения, то обычно они все-таки универсальны, хотя попадаются системы, замкнутые на себя. Существуют корпуса экзотических форм — например, в виде прямоугольного сектора цилиндра, предназначенные для установки в угол (фирма Packard Bell). Есть и компьютеры-моноблоки, в которых системный блок и монитор расположены в общем корпусе. Имеются так называемые мультимедийные корпуса со встроенными стереофоническими акустическими системами.

2.1. Малогабаритные компьютеры

Помимо настольных (напольных) стационарных PC давно уже выпускаются и их портативные варианты. Первые из них были довольно громоздкими. Переносная машина *IBM PC Portable* была скомпонована в корпусе обычного настольного размера, но на ее переднюю панель выходил экран небольшой электронно-лучевой трубки монитора. Клавиатура пристегивалась к передней панели и при переноске являлась крышкой. Вес машины был внушительным (из-за прочного стального корпуса), а питание могло подаваться только от сети. Несколько позже появились компьютеры класса *Laptop* («наколенные»), которые имели вид небольшого портфеля-дипломата. Они уже были оборудованы плоскими жидкокристаллическими мониторами и имели возможность работы от встроенных аккумуляторов. Каждый разработчик делал эти машины по-своему, поэтому об их открытости и модернизируемости говорить не приходится.

Более компактны машины класса *Note Book (блокнотные ПК*, или *ноутбуки*), линии которых успешно развиваются в настоящее время. Свое название они получили за форму, напоминающую раскрытый блокнот: нижняя часть — системный блок с клавиатурой, верхняя (откидная) часть — матричный (ЖК) экран с типовым размером 14—15 дюймов. Так что габариты их (примерно 25 х 30 см в плане) соответствуют довольно большому блокноту. Есть и *субноутбуки* — их

размеры уменьшены примерно до 18 х 25 см. В блокнотных ПК уже достигнута унификация модулей их функционального расширения в виде стандарта PC Card, который ранее назывался PCMCIA. Для них существуют специальные малогабаритные винчестеры и приводы CD/DVD, а также малогабаритные модули памяти. Компоненты, используемые в этих ПК, отличаются пониженным энергопотреблением, которое достигается либо ценой снижения производительности, либо за счет более дорогих технологий. Для подключения внутренних периферийных адаптеров существуют стандартные конструктивы Mini PCI и Small PCI. По характеристикам блокнотные компьютеры не слишком отстают от своих настольных собратьев, но они дороже в несколько раз (главным образом, из-за дисплея). Важными параметрами блокнотных ПК являются габариты, масса и длительность автономной работы. Последние два параметра определяются уровнем технологии энергосбережения и применяемого аккумулятора. Для энергосбережения используются технологии динамического управления потреблением и производительностью. За энергосбережение приходится расплачиваться некоторой медлительностью (задержками реакции, но не низкой производительностью) этих компьютеров.

В блокнотных компьютерах системный блок, дисплей, клавиатура и манипулятор совмещены в одном корпусе, дисплеем является откидная крышка. В качестве манипулятора раньше использовали трекбол, в современных моделях применяют малогабаритную сенсорную панель (touch pad), чувствительную к прикосновению. В корпус, как правило, встроен привод CD/DVD и, конечно же, малогабаритный винчестер, причем немалой емкости. Дисковод для гибких дисков (3,5") на современные компьютеры уже не ставят, при необходимости можно воспользоваться внешним устройством, подключаемым к USB (или к разъему LPT-порта). Для расширения функциональных возможностей имеется одно или несколько гнезд PC Card, их спецификации могут быть различными (см. 14.11). В формате PC Card популярны модемы, адаптеры локальных сетей, карты с флэш-памятью (электронные диски объемом в десятки и сотни мегабайт), контроллеры SCSI для подключения внешней периферии и другие устройства. Для PCI Express предусмотрен также формат гнезд Express Card. Современные компьютеры имеют богатый набор внешних интерфейсов: шины USB и FireWire, проводные (Ethernet) и беспроводные (Wi-Fi) интерфейсы локальной сети и подключения периферийных устройств (радиоинтерфейс BlueTooth, инфракрасный порт IRDA), разъем подключения телефонной линии (для встроенного модема). Из традиционных интерфейсов присутствует LPT-порт; COM-порты и разъемы клавиатуры и мыши PS/2 встречаются уже редко (их заменяет USB). Обычно на компьютере устанавливают разъем (VGA) для подключения внешнего монитора или проектора. Для мультимедийного оборудования компьютеры снабжают соответствующими аудио- и видеоразъемами.

Для длительной работы в стационарных условиях многие пользователи предпочитают стандартные внешние клавиатуры и мыши — эти устройства все-таки удобнее, чем их компактные версии, встроенные в блокнотные ПК. Для упрощения подключения стационарной периферии можно использовать *переходник*

Port Bar - это блок со стандартными периферийными разъемами, подключаемый к блокнотному ПК всего одним (но нестандартным, фирменным) разъемом.

Многие блокнотные ПК можно подключать к специальным *док-станциям*, которые доводят возможности блокнотных ПК до уровня обычных настольных. Док-станция является блоком расширения — она имеет обычные слоты расширения с шинами ISA и PCI, периферии у которых больше, — и гораздо доступнее по цене. Шины док-станции соединяются с блокнотным ПК через специальные мосты. Док-станция не имеет своего процессора и оперативной памяти — работает «центр» блокнотного ПК. Она может быть оборудована дополнительными устройствами внешней памяти, которые расширяют объем доступной дисковой памяти ПК. Док-станции часто подключают к локальной сети офиса.

Дальнейшая миниатюризация компьютеров привела к появлению их совсем маленьких версий — *Palm-Top* («наладонные»), которые, как и следует из названия, умещаются на ладони или в кармане — КПК (карманные ПК). Эти компьютеры, конечно, имеют не очень много общего с «классической» архитектурой PC, но вполне способны исполнять специально под них адаптированные офисные приложения. Существует также класс специализированных устройств *PDA* (Personal Digital Assistant — персональный электронный секретарь) — например, электронные словари, записные книжки и т. п. Очень удобно, когда эти миниатюрные компьютеры можно подключать к настольным или блокнотным для обмена данными. В карманных компьютерах нет места для клавиатуры, ее заменяет сенсорный экран - комбинация дисплея и сенсорной панели. Эта панель чувствительна к прикосновению специальным пером (stilus). Для ввода текста можно вывести на экран изображение клавиатуры и набирать текст прикосновением к соответствующим нарисованным «клавишам». Возможен и графический ввод текста: пользователь пишет буквы на панели, их образ распознается и преобразуется в текстовые символы.

Планшетные ПК (*TabletPC*) — нечто среднее (по размеру) между КПК и блокнотными ПК. У них уже довольно большой экран, совмещенный с сенсорной панелью. Такие компьютеры предназначены для чтения электронных книг, для просмотра веб-страниц, для рисования — хорошие сенсорные панели распознают не только факт прикосновения, но и степень нажатия на панель. Как правило, планшет сопровождается специальным электронным пером, которое приходится держать в руках, что не всегда удобно. У чисто планшетных ПК клавиатуры нет (есть всего несколько кнопок управления), как нет винчестера и приводов оптических дисков. Эти внешние устройства становятся доступными при наличии док-станции, к которой подключается планшетный ПК (если предусмотрено конструкцией). Выпускаются и компьютеры-гибриды, а также трансформеры, которые могут рассматриваться как планшеты с клавиатурой или как блокнотный ПК с сенсорным экраном. Расплатой за многофункциональность экрана, сочетающего в себе ЖК-матрицу и сенсорную панель, как правило, является уменьшение угла обзора экрана (и так не очень широкого в ЖК-панелях).

2.2. Промышленные и инструментальные компьютеры

Компьютеры для промышленного применения обобщенно называются *Industrial PC*. Здесь, конечно же, под PC понимается не персональный (как таковой) компьютер, а компьютер, совместимый с IBM PC. Такие компьютеры предназначены для особых (не офисных) условий эксплуатации.

Промышленному компьютеру по роду службы приходится располагаться поблизости от подконтрольного объекта, в той или иной степени разделяя его условия существования (в противном случае компьютер можно было бы установить в уютном офисе с обычными условиями эксплуатации). Условия эксплуатации могут быть тяжелыми в смысле климата — температуры, влажности, пыли, осадков и т. п. Компьютер может подвергаться механическим воздействиям — вибрации, ударам, ускорению. Химическое воздействие подразумевает, например, агрессивные пары и газы. Неблагоприятное соседство с электропотребителями (мощными контакторами, сварочными аппаратами, печами) вызывает как электромагнитные возмущения, так и осложнения с питанием. Уже перечисленных невзгод достаточно, чтобы испугаться за «здоровье» нежного настольного компьютера, попавшего в такие условия. Добавим еще, что может потребоваться подключение к промышленному компьютеру большого числа цепей связи с объектом, для которых на задней панели PC просто не хватит места под разъемы, а на системной плате не хватит слотов для интерфейсных карт сопряжения. И наконец, конструкция должна обеспечивать минимальное время поиска и устранения неисправностей, которые неизбежны даже при самом высоком уровне надежности. К *инструментальным компьютерам*, в основном предназначенным для сбора и обработки информации о каком-либо сложном объекте (например, экспериментальной установке), предъявляются похожие требования; правда, внешние условия, как правило, помягче.

Для соответствия этим требованиям конструктив PC должен быть заметно преображен. В PC объединение модулей (интерфейсных карт) осуществляется через системную плату, на которой сейчас размещают практически все основные и жизненно важные компоненты, от процессора до большинства стандартных интерфейсных адаптеров. И эта сложнейшая плата оказывается на самом дне корпуса, «погребенная» под установленными в нее интерфейсными картами и подсоединенными кабелями. Если она откажет, то для замены или ремонта компьютер придется разобрать полностью, что делается не так-то быстро. Чтобы избежать таких затруднений, в промышленных и инструментальных компьютерах функцию объединения модулей выполняет *пассивная кросс-плата* (passive backplane). Точный перевод названия указывает на местоположение этой платы в конструктиве — заднюю плоскость. На такой плате устанавливают только разъемы подключения функциональных модулей и блока питания. Все функциональные модули устанавливаются в блок спереди и объединяются между собой магистральной шиной кросс-платы. Внешние подключения к модулям осуществляют либо со стороны лицевой панели модулей, либо с задней стороны кросс-платы через контакты разъемов, не используемых под магист

ральные шины. Функциональные модули могут иметь различное назначение, но главным является, конечно же, процессорный модуль. Современные процессорные модули функционально идентичны традиционным системным платам с интегрированной периферией. На них устанавливают процессоры от 386 до Pentium II/III, «золотой серединой» являются экономичные и эффективные процессоры классов 486 и Pentium. Периферийные модули выполняют функции аналогового и цифрового ввода-вывода, и из широкого ассортимента выпускаемых модулей всегда можно набрать комплект, «персонально» подходящий к компьютеризуемому объекту.

Как и для традиционных (настольных) ПК, в данной отрасли существуют стандарты на конструктивы и, конечно же, стандарты на объединительные шины.

Модульная система «Евромеханика» широко применяется для приборов промышленного назначения и инструментальных систем. Это международный стандарт на типоразмеры и конструктивы печатных плат, модулей, субблоков, блоков и 19-дюймовых шкафов и стоек. В зависимости от сложности устройств стандарт позволяет выбрать подходящий размер модулей и плат (рис. 2.6). Модуль представляет собой плату с некоторым внешним оформлением — передней панелью и, возможно, кожухом. Модули устанавливаются в каркасы блоков и с помощью коннекторов (разъемов), установленных на задней стороне их плат, соединяются с кросс-платой. Обычно на кросс-плате имеется *шина* (bus), объединяющая модули и подводящая к ним стандартные напряжения питания.

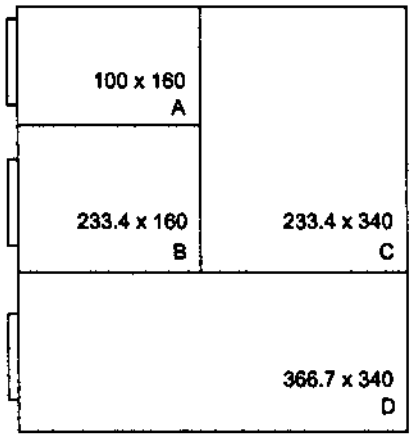


Рис. 2.6. Форматы модулей «Евромеханика»

В евромодулях используются стандартизованные магистральные шины, из которых для PC-совместимых компьютеров сейчас популярны шина Compact PCI и ее расширение PXI [7]. Распространенная шина VME ориентирована совсем не на «PC-шные» процессоры Motorola.

«Евромеханика» является мощной конструктивной базой для построения сложных устройств, но есть варианты компоновки модульных компьютеров попроще (и подешевле). Самое простое решение для создания конструктива инстру-

ментальных и промышленных компьютеров — использование стандартной шины карты ISA (половинной или полноразмерной). Все компоненты с традиционной системной платы переносят на карту ISA, получая одноплатный компьютер, называемый *микроPC* (microPC, mPC). На такой карте содержатся процессор, память, графический адаптер, контроллеры портов и дисковые интерфейсы, иногда на нее же ухитряются поместить и дополнительные контроллеры цифрового и аналогового ввода-вывода. Для подключения к модулям (картам) расширения используют пассивную кросс-плату с обычными разъемами ISA. Если требуется более высокопроизводительный канал, задействуют и шину PCI. Кросс-плата для таких систем становится неоднородной: у нее одна часть слотов имеет разъемы PCI, другая часть — ISA, расположенные на обычных местах, а место для системного контроллера оборудовано обоими разъемами. Достоинством такого конструктива является его совместимость с обычными картами расширения для PC, но оно оборачивается и недостатком — остается все то же ненадежное крепление и мало места под внешние разъемы.

На базе плат mPC (чаще половинного формата) делают и модульные конструктивы. Верхнюю сторону платы снабжают панелью, которая оказывается лицевой. При этом изменяется система крепления и подвода внешних цепей — верхняя (длинная) сторона платы становится доступной для установки внешних разъемов. Появляется вторая точка крепления, так что платы при малейших внешних усилиях самопроизвольно не вылезают из слота, как морковка из грядки. Конечно же, такие модули не располагают всей мощью платы «Евромеханика» (большой размер, надежные разъемы, подключение через заднюю панель), но вполне пригодны для устройств средних размеров.

Для «самых маленьких» встраиваемых контроллеров существует другой конструктив с шиной PC/104. В ее названии присутствует число контактов коннектора, на который выводятся сигналы шины ISA. От обычной шины ISA шина PC/104 отличается только типом коннектора и нагрузочными характеристиками линий. Основой контроллера является плата mPC с разъемом (розеткой) PC/104 (рис. 2.7, *а)*. Если требуется подключение платы расширения, она своим разъемом PC/104 (вилкой) вставляется в плату контроллера. Помимо вилки на плате расширения имеется и розетка PC/104 (коннектор двухсторонний), так что можно собирать «бутерброд» из нескольких плат. Если плат более трех, то сверху «бутерброда» устанавливают терминатор. Для фиксации плат стандартизовано расположение крепежных отверстий, и платы скрепляются несущими стоечками (длинными винтами с втулками). Конечно, такой конструктив удобен только для небольших систем с двумя-тремя платами, для которых он и предназначается. Возможна и иная компоновка — установка нескольких модулей на одной (большой) кросс-плате. С широким использованием процессоров Pentium и выше в модуль ввели еще и шину PCI — так появился стандарт PC/104-Plus. Расположение коннекторов и габариты платы PC/104-Plus иллюстрирует рис. 2.7, *б.* Отметим особенности коннекторов: J1 — коннектор шины ISA-8, J2 — его расширение до ISA-16; эти коннекторы обычно имеют дюймовый шаг контактов (2,54 мм), но могут встречаться и метрические с шагом 2,5 мм (они взаимно несовместимы). Обратим внимание и на специфическую

нумерацию рядов и номеров контактов (у J2 нумерация начинается с нуля). Коннектор PCI имеет шаг контактов 2 мм.

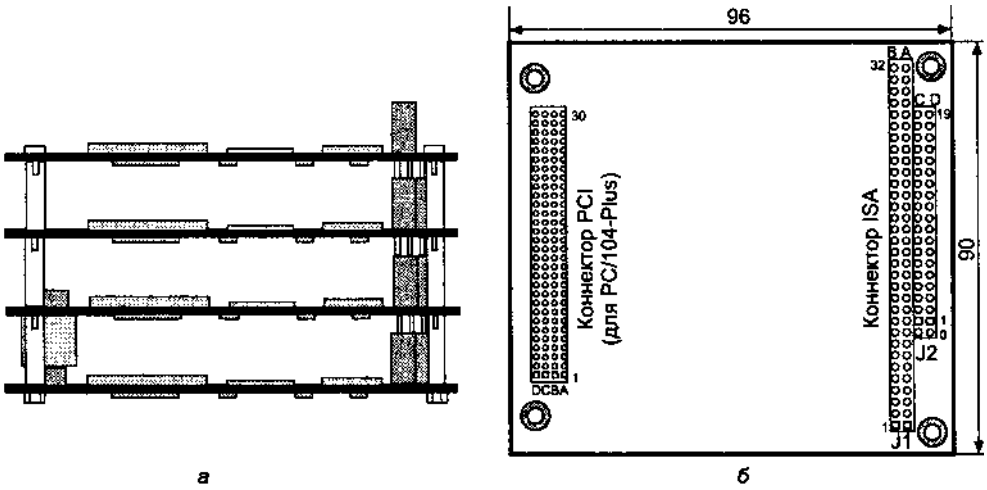


Рис. 2.7. Компьютер с шиной PC/104: а — стопка плат, б — расположение системных коннекторов

2.4. Периферийные устройства

Периферийные устройства, подключаемые к системному блоку, весьма разнообразны. В этом разделе приводится краткий (и неполный, поскольку нельзя объять необъятное) обзор устройств с описанием назначения, возможностей и способов подключения.

Дисплей — основное устройство вывода текстовой, графической и видеоинформации; подключается к выходному разъему графического адаптера. К компьютеру может быть подключено и более одного дисплея, что допускают современные видеокарты с двумя интерфейсами. Можно также установить два графических адаптера (и подключить к ним по два дисплея). В качестве дисплея (основного или дополнительного) может быть использован и обычный телевизор, если графический адаптер имеет соответствующий интерфейс (или есть специальный конвертор). Подробнее о дисплеях, графических адаптерах и их сопряжении с телевизионной техникой см. главу 10.

Клавиатура — самое привычное устройство ввода символьной информации — подключается к специализированному интерфейсу системной платы или же шине USB. Принцип работы, устройство, аппаратный и программный интерфейсы клавиатуры рассмотрены в 11.1. Здесь отметим, что клавиатура является самым быстрым устройством ввода текстовой информации и команд от пользователя. Кроме того, она может использоваться и необычным образом — например, с ее помощью можно управлять объектами в играх, а также исполнять музыкальные произведения, если назначить ее примитивным MIDI-кон-

троллером (см. 12.3). Клавиатур (при наличии USB) можно подключать несколько, но все потоки набираемых символов со всех клавиатур сольются в один поток.

Устройства хранения данных (внешняя память) в основном «прячутся» в системном блоке. Компьютер общего назначения должен иметь как минимум один жесткий диск (винчестер) для основной работы. Для переноса информации небольшого объема используют (в настоящее время все реже и реже) дискеты (1,44 Мбайт) с соответствующим дисководом. В качестве удобных средств переноса информации популярны устройства хранения на флэш-памяти с интерфейсом USB (см. 9.10). Для установки современного ПО, а также исполнения ряда приложений (особенно игр), прослушивания музыки (аудио-CD и диски с файлами формата MP3), просмотра фильмов компьютер должен иметь привод CD или DVD (который читает и CD). Для выпуска собственной продукции, а также архивирования данных и копирования CD/DVD к компьютеру подключают CD/DVD-рекордер. Для архивирования и переноса больших объемов данных применяют магнитооптические диски, устройства ZIP, JAZ, стримеры. Устройства хранения подключаются к шинам ATA (только внутренние), SATA, SCSI, SAS, USB, FireWire, а также к LPT-порту. Подробнее об этих устройствах можно узнать в главе 9.

Устройство-указатель — *мышь* или *трекбол* (шар) — служит для позиционирования курсора на экране, а также подачи команд нажатием нескольких кнопок. Мышь подключается к специализированному интерфейсу системной платы, COM-порту или шине USB. Подробно мыши рассмотрены в 11.2. В блокнотных ПК вместо мыши используют сенсорную панель, чувствительную к прикосновению. Как правило, к компьютеру подключают лишь одну мышь (и указатель тоже один), но при желании можно подключить несколько устройств-указателей, возможно, и разных типов. Все они будут управлять одним курсором.

Принтеры и *плоттеры* — устройства вывода текстовой и графической информации на «твердые» носители (бумагу, пленку). Эти устройства подключаются к LPT-порту, к COM-порту или к шине USB, и подробно описаны в 11.5. Современные принтеры в графическом режиме обеспечивают высокую геометрическую точность вывода изображений, ими можно пользоваться, например, для создания фотошаблонов печатных плат или шаблонов раскроя материалов. Это, естественно, относится и к плоттерам.

Дигитайзер (планшет) — устройство для оцифровки изображений или рисования (см. 11.3). В этих устройствах лист изображения закрепляется на специальном планшете, и, подводя специальный указатель (в виде пера или «оптического прицела») к элементам изображения, по нажатии кнопки в компьютер вводят точные координаты элемента. Дигитайзеры большого размера служат, например, для ввода чертежей (формата А3 и более). Малогабаритные дигитайзеры (например, формата А4) с указателем-пером используются художниками-оформителями — с их помощью можно рисовать привычными движениями (рисовать мышью или трекболом очень неудобно). Дигитайзеры являются векторными устройствами ввода, подключаются к COM-порту или USB. Дигитай

зер можно применять и как устройство-указатель, но не наоборот: мышь сообщает только приращения координат, а дигитайзер — абсолютные координаты.

Сканеры — растровые устройства ввода графической информации. Сканеры бывают различных конструкций (см. 11.4). Наиболее широко распространены *планшетные сканеры*, в которых лист с вводимым изображением укладывается на прозрачный стол, а под столом проезжает каретка со считывающим устройством. Планшетные сканеры являются высокоточными (в смысле геометрии) устройствами ввода; дорогие сканеры обеспечивают и точную цветопередачу. Более дешевые сканеры — ручные, в них отсутствуют стол и привод. Пользователь прокатывает считывающий блок по изображению, при этом точность ввода, естественно, страдает. Сканеры-ручки предназначены для построчного ввода текста. Сканеры подключаются через LPT-порт либо через шину SCSI или USB.

Графическое изображение текста, введенное со сканера, может быть преобразовано в символьный формат с помощью программ распознавания текста (Optical Character Recognizer, OCR). Современные программы на современных компьютерах позволяют выполнять эти преобразования быстро и довольно верно, автоматически выделяя текст в возможном окружении графических рисунков, выявляя таблицы и другие элементы оформления. Символьное хранение текстов гораздо компактнее и информативнее графического. Текст может распознаваться при разном начертании букв, вплоть до рукописных, но для этого программа распознавания должна быть достаточно совершенна и самообучаема. Еще несколько лет назад ввод текста гораздо быстрее и качественнее выполняла квалифицированная машинистка, но сейчас ситуация уже иная. Есть *сканеры-ручки,* которые сами распознают текст и до передачи в компьютер могут хранить введенный текст во внутренней флэш-памяти. Есть и универсальные «ручки», позволяющие вводить не только готовый текст сканированием, но и текст, тут же записываемый этой самой ручкой; кроме того, возможен ввод фотоизображений и даже видео — правда, с невысокой частотой смены кадров. Подобные ручки связываются с компьютером по беспроводной связи Bluetooth.

К сканерам относятся и *устройства считывания штрих-кодов*. Поскольку выходные данные — не очень длинный набор цифр, для подключения этих устройств может использоваться обычный COM-порт. Применение модема позволяет при необходимости значительно отдалить сканер от компьютера.

К растровым устройствам ввода относятся и *цифровые фотокамеры*, как правило, подключаемые к USB (реже — к FireWire). В отличие от сканера, вводящего изображение с плоского оригинала, фотокамеры вводят объемное изображение, «видимое» через объектив.

Мультимедийные устройства — это устройства общения компьютера с человеком через разные среды (multi media — множество сред). Поскольку основная информация воспринимается зрительно, главным устройством вывода является дисплей. Помимо текста и графики на дисплей мультимедийного компьютера может выводиться видеоизображение — воспроизводиться видеозапись (с CD и DVD), приниматься поток видеоданных по сети (видеоконференцсвязь), воспроизводиться «живое» видео от внешних источников сигнала (каме

ры, телеприемника, видеомагнитофона). Для этого от дисплея дополнительных способностей (кроме общего качества) практически не требуется. Воспроизводить поток сжатых видеоданных (для проигрывания фильмов и обслуживания телеконференций) может практически любой дисплейный адаптер современного компьютера, но дополнительные аппаратные средства ускорения декодирования и масштабирования позволят улучшить качество при снижении загрузки центрального процессора. Для работы с источниками видеосигнала требуются специальные аппаратные средства (видеобластер), позволяющие выводить видео в окно экрана, захватывать отдельные кадры для сохранения и обработки и даже захватывать поток кадров «живого» видео.

Для ввода-вывода видеопотока («живого» видео) используются различные интерфейсы: аналоговые (Composite Video или S-Video) и цифровые (USB, FireWire). Для компьютеров выпускают карты телевизионных приемников (тюнеров), способных принимать телепрограммы одного или нескольких стандартов вещания. Тюнер может входить и в состав графической карты — такой «комбайн» приобретает богатые функциональные возможности. Например, можно выводить в отдельные окна сигналы нескольких каналов (правда, не полнодвижущиеся, поскольку приемник один и его можно лишь переключать с канала на канал, запоминая картинку). Полезным свойством графической карты является выход на телевизионный приемник (монитор) — с его помощью можно выводить изображение на большой экран и даже на несколько экранов, что удобно на презентациях и тому подобных мероприятиях. К графическому адаптеру могут подключаться и специальные проекционные аппараты, выводящие изображение на большой экран. Относящаяся к видео часть мультимедийного компьютера рассмотрена в главе 10.

Неотъемлемой частью мультимедийного компьютера является *аудиосистема* (см. главу 12), которая может быть картой расширения (звуковая карта ISA или PCI) или встраиваться в системную плату (интегрированный звук). Современные звуковые карты имеют *аудиокодек* — средство цифровой записи и воспроизведения аудиосигналов, обеспечивающее качество на уровне аудио-CD и выше. Кодек позволяет озвучивать приложения (звук в играх), проигрывать звуковые файлы с различными форматами данных, записывать (оцифровывать) внешние аудиосигналы (можно, например, переписывать грампластинки в цифровой формат), организовывать голосовую связь по сети и выполнять ряд других операций. Кодек совместно с программными средствами позволяет синтезировать и распознавать речь. Текст (например, из файла) может проговариваться компьютером, причем «диктором» можно управлять — подобрать приятный голос, темп, попросить подождать и повторить и т. п. Распознавание речи позволяет подавать команды голосом. Очень эффективен компьютер для обучения языкам. Карта обычно имеет синтезатор, с помощью которого можно проигрывать MIDI-файлы, исполнять MIDI-команды с внешнего порта и просто играть на клавиатуре (обычной компьютерной или виртуальной, нарисованной на экране). Звуковая карта имеет микшер, позволяющий смешивать сигналы от разных источников — от кодека, микрофона, аудиовыхода привода CD/DVD, линейного входа, синтезатора. К карте подключают стереонаушники, колонки,

внешние усилители с акустическими системами. При этом можно использовать разные схемы расположения акустики: от обычной стереофонической (и монофонической, конечно, тоже) до объемной, вплоть до 6- или 8-колоночной схемы Dolby Digital AC-3. Внешние интерфейсы аудиосистемы постепенно мигрируют от традиционных аналоговых к цифровым. Аудиосистема ПК может обходиться и без специальных аппаратных средств — звуковые устройства (микрофон, акустические системы) подключаются к компьютеру по шине USB или FireWire, а все операции с цифровыми аудиоданными выполняются чисто программно — современные компьютеры на это уже вполне способны.

Для мультимедийных применений требуются средства хранения данных, которыми, как правило, являются приводы CD/DVD. Хранить мультимедийные данные только на винчестере слишком накладно — даже сжатые данные, особенно видео, занимают много места. Исключение составляют разве что MIDI- файлы — этот формат позволяет небольшое музыкальное (но чисто инструментальное!) произведение уместить на обычной дискете.

Итак, мультимедийный компьютер должен иметь, как минимум, графический адаптер, звуковую карту (или интегрированный звук) и привод CD/DVD. Помимо этого минимума возможны игровые «излишества». С самого своего появления персональные компьютеры в значительной степени были ориентированы на игры, что в первую очередь отражается на стремительном прогрессе графических средств. На данном этапе «игровой» ПК должен иметь графический адаптер с 3D-ускорителем — современные игры на простых картах если и работают, то не так эффектно. Помимо трехмерного изображения игры производят и «трехмерный» звук, правда, не так убедительно, как графику. Для трехмерного звука в аудиосистему добавляют специальные средства, программные и аппаратные. К изображению и звуку для игр требуется и более «объемные» средства ввода. С первых моделей PC к ним можно подключить *джойстик* или иное устройство аналогового ввода, для чего предназначен игровой порт (GAME- порт, см. 11.6). Эти устройства (например, руль и педали автомобиля) создают у игрока реальные ощущения управления каким-либо игровым объектом. С помощью этих устройств «играют» и в серьезные игры — компьютер легко превратить в тренажер для обучения вождению автомобиля или самолета. Для большего реализма в эти устройства можно вводить и сигналы обратной связи, позволяющие физически (тактильно) прочувствовать поведение объекта в разных ситуациях. Современные игровые манипуляторы, особенно с механической обратной связью, подключают к шине USB.

И наконец, для полного отрыва от реальной действительности появились *шлемы виртуальной реальности* — комбинированные аудиовидеосистемы. В отличие от обычного монитора, эти шлемы выводят изображения раздельно для каждого глаза, что обеспечивает объемное восприятие трехмерного изображения. Изображения формируются малогабаритными жидкокристаллическими дисплеями. Шлем подключается к выходу графического адаптера, который программно заставляют поочередно выводить изображения для левого и правого глаз, а также к аудиовыходу. Существуют и более простые средства стереоскопического зрения — *стереоочки* с оптическими затворами, поочередно откры

ваемыми синхронно со сменой кадров графического адаптера. Через эти очки смотрят на обычный дисплей, который опять-таки выводит чередующиеся кадры. Однако такие очки хотя и дают объемное восприятие, сильно утомляют глаза. Существуют системы стереопоказа и с пассивными очками (неуправляемыми), но они требуют более сложных устройств вывода.

Коммуникационные устройства позволяют связывать компьютеры между собой и с «внешним миром» — например, с Интернетом. Цели связи: обмен данными, использование общих информационных ресурсов, общей периферии, общей внешней памяти, совместная работа над общим проектом, совместные игры. К коммуникационным устройствам относят модемы и адаптеры локальных сетей. *Модем* позволяет связываться с другими компьютерами и сетями по телефонной сети общего пользования или по специальным выделенным линиям. Модем в ПК может выполнять множество функций: пересылать данные (в том числе получать все услуги Интернета, включая интернет-телефонию и видеосвязь), принимать и передавать факсы, работать как автоответчик, телефонный секретарь и др. Модемы устанавливаются в слоты расширения (ISA, PCI, PC-Card) или подключаются внешне, к COM-, к LPT-порту или к шине USB. Адаптеры локальных сетей, проводных и беспроводных, позволяют обмениваться данными с гораздо более высокими скоростями, но на меньших расстояниях (в пределах здания). Локальные сети используют для совместного доступа нескольких ПК к общей периферии (принтеры, плоттеры, устройства хранения данных), обеспечения связи для клиент-серверных приложений, коммуникаций между пользователями (включая сетевые игры). Более подробные сведения о сетях приведены в [4], а в главе 13 рассматриваются вопросы телекоммуникаций, непосредственно связанные с самим компьютером.

Электронные ключи — устройства, с помощью которых возможно ограничение незаконного распространения (тиражирования) ПО. Разработчики ПО, отстаивая свои законные права на получение вознаграждения за свой труд, предпринимают ряд мер, препятствующих бесконтрольному тиражированию их продукции. Эти меры могут быть как организационными (необходимость ввода ключа — последовательности символов, который можно получить «только» у законного продавца, требование соблюдения лицензионных соглашений и т. п.), так и техническими. Несостоятельность организационных мер пояснять не требуется. Технические меры могут быть двоякими — защита от копирования или применение электронных ключей. Как известно, любую защиту от копирования (будь то нестандартный формат ключевой дискеты или CD-ROM) можно «расколоть», поскольку она проверяется и создается теми рке самыми программными средствами, которые доступны и взломщику. Электронные ключи представляют собой устройства, без подключения которых к компьютеру защищаемое приложение работать не будет. Конечно, и здесь возможен взлом, но его вероятность зависит от сложности ключа и системы привязки. Первые системы были довольно простыми, и зачастую в приложении можно было найти и «выкусить» кусок кода, проверяющего присутствие ключа (так, например, «ломали» пакет p-CAD, предназначенный для разводки печатных плат). Более сложные ключи внутри себя содержат некий «фрагмент кода», используемый в работе приложения. «Выкусить» его из приложения, не имея исходных кодов программы,

очень трудно, а снять копию с ключа может быть и технически невозможно. Есть, например, устройства энергонезависимой памяти (см. 8.5), которые не позволяют считать свое содержимое, но могут дать только результат сравнения — совпал приложенный ключ (многобитный) с записанным образцом или нет. При большой длине ключа (и небыстром обмене с такой памятью) перебор вариантов просто не завершится при жизни взломщика. Электронные ключи подключают к портам (COM или LPT) или шине USB; сетевые приложения могут использовать и сетевые ключи — ключ должен быть подключен к одному из компьютеров локальной сети, на котором работает приложение (так, например, защищается популярный бухгалтерский пакет фирмы 1С). Конечный пользователь приобретает ключ у продавца защищаемого программного продукта. У производителя ключи («болванки») приобретают разработчики программного обеспечения, и именно они привязывают ключи к своей продукции. Говоря о ключах, вспомним о привязке ПО к конкретному компьютеру — этот метод защиты одно время использовался для недорогих продуктов. Суть его заключается в записи ключевой информации в какую-либо скрытую область энергонезависимой памяти компьютера — CMOS (см. 4.6) или на жесткий диск либо в привязке продукта к серийному номеру какого-либо компонента (процессора, винчестера и т. п.). Конечно же, защита условна — зная скрытые, но доступные места хранения информации, скопировать ключ большого труда не составит. Этот способ некорректен, поскольку при обслуживании компьютера, модернизации (замене ключевого элемента), ремонте и в других ситуациях пользователь может лишиться законного права на использование продукта (к этому моменту поставщик «защищенного» ПО может уже стать недоступным). Кроме того, две разные программы могут писать свой ключ в одно и то же место — выживает последняя устанавливаемая. Электронные ключи этих недостатков не имеют — они не копируемы, но переносимы и зачастую «прозрачны»: ключ для COM- или LPT-порта является проходным, через него к порту может быть подключено обычное периферийное устройство или еще один ключ. На компьютере, где работают множество защищенных пакетов, можно увидеть целые гирлянды ключей. Конечно же, сложность (и связанная с ней стоимость) ключа должна соответствовать защищаемому продукту — смешно ставить противоугонное устройство за 500 долларов на детский велосипед (иногда и без педалей).

В системах, связанных с защитой данных от несанкционированного доступа, применяют различные средства аутентификации (подтверждения личности) пользователя. Когда простейших административных средств (введение имени и пароля) оказывается недостаточно, применяют, например, биометрические датчики. Эти датчики анализируют какие-либо неизменяемые (неподделываемые) признаки. Существуют, например, устройства, считывающие дактилоскопические отпечатки, для чего достаточно лишь приложить палец к специальной площадке. Такие устройства подключаются, как правило, к обычным интерфейсам (COM-порт, USB).

Для промышленных и инструментальных компьютеров периферия (подключаемые объекты управления и наблюдения) содержит аналоговые и дискретные датчики и исполнительные устройства. Для их подключения выпускают раз

личные карты сопряжения (и отдельные устройства внешнего исполнения), содержащие аналого-цифровые и цифроаналоговые преобразователи, порты ввода и вывода дискретных сигналов с различными параметрами. С помощью этих устройств и надлежащего программного обеспечения компьютер можно научить всему чему угодно — от медицинской диагностики до управления военной и космической техникой.

2.3. Интерфейсы подключения периферийных устройств

Большинство периферийных устройств подключаются через промежуточные *периферийные интерфейсы*, находящиеся на нижних уровнях иерархии подключений (на верхнем уровне — системная шина, см. 1.6). Периферийные интерфейсы — самые разнообразные из всех аппаратных интерфейсов. К периферии, подключаемой через промежуточные интерфейсы, относятся большинство устройств хранения (дисковые, ленточные), устройств ввода-вывода (дисплеи, клавиатуры, мыши, принтеры, плоттеры), ряд коммуникационных устройств (внешние модемы). По назначению периферийные интерфейсы можно разделить на специализированные и универсальные, выделенные и разделяемые:

- *Специализированные* интерфейсы ориентированы на подключение устройств определенного узкого класса, и в них используются сугубо специфические протоколы передачи информации. Примеры — популярнейший интерфейс мониторов VGA, интерфейс накопителя на гибких дисках, традиционные интерфейсы клавиатуры и мыши, IDE/ATA и ряд других.
- *Универсальные* интерфейсы имеют более широкое назначение, их протоколы обеспечивают доставку данных, не привязываясь к специфике передаваемой информации. Примеры — коммуникационные порты (COM), интерфейс SCSI, шины USB и FireWire.
- *Выделенные* интерфейсы позволяют подключить к одному порту (точке подключения) адаптера (контроллера) лишь одно устройство; число подключаемых устройств ограничено числом портов. Примеры — COM-порт, интерфейс VGA-монитора, порт AGP, интерфейс Serial SCSI.
- *Разделяемые* интерфейсы позволяют подключить к одному порту адаптера множество устройств. Варианты физического подключения разнообразны: шина (жесткая, как ISA или PCI; кабельная шина SCSI и IDE/ATA), цепочка (daisy chain) устройств (SCSI, IEEE 1284.3), логическая шина на хабах (USB) или встроенных повторителях (IEEE 1394 FireWire).

Виды передаваемой информации

Информация (данные), которую следует передавать по интерфейсам, может быть разной природы:

- Аналоговая информация отображает процесс, непрерывный во времени и произвольный по величине (может принимать любое из бесконечного числа значений, пусть и в ограниченном интервале). Пример: звуки, которые мы слышим (в том числе речь), представляют собой непрерывное изменение давления. Передача такой информации осуществляется, например, при подключении микрофона (устройства, преобразующего изменения давления в изменения электрического напряжения) к компьютеру.
- Дискретная информация отображает процесс конечным числом значений. Элементарная единица дискретной информации — 1 бит, который может принимать лишь одно из двух логических значений: 1 (истина, «да») или 0 (ложь, «нет»). Одним битом, к примеру, можно отобразить состояние кнопки мыши — нажата или нет. Дискретная двоичная информация является «родной» для большинства компьютеров, поскольку ее проще всего получать, обрабатывать, хранить и передавать. Дискретная информация может быть не только двоичной — интересны, например, и троичные системы; состояние одного трита[1] можно трактовать как «да», «нет», «не знаю».
- Цифровая информация представляет собой последовательность (набор) чисел, имеющих ограниченную разрядность (и соответственно, конечное число возможных значений). Пример — оцифрованный звук, являющийся последовательностью отсчетов мгновенных значений давления, взятых через равные интервалы времени.

Дискретную и цифровую информацию не всегда корректно различают (и не всегда это требуется), поскольку «выглядит» она похоже: в двоичной системе та и другая представляет собой наборы ноликов и единичек. Важным отличием цифровых данных является осмысленность сравнения значений по условиям «больше-меньше». Цифровая информация является особым видом дискретной. Для передачи данных по различным интерфейсам наиболее существенно разделение на аналоговые (непрерывные) и дискретные данные.

Для того чтобы передавать данные, их нужно представить в виде *сигнала* — физического процесса (электрического, оптического, электромагнитного, хотя возможны и другие). Сигналы могут быть различных типов: аналоговые (непрерывные), дискретные, цифровые. Заметим, что тип сигнала может и не соответствовать типу передаваемых данных. Так, аналоговый сигнал телефонного модема несет дискретные (цифровые) данные. Тип и природа используемого сигнала определяются требованиями к интерфейсу: дальностью связи, скоростью передачи данных, надежностью, достоверностью, безопасностью, стоимостью, удобством подключения, энергопотреблением и др.

Параллельные и последовательные интерфейсы

Для компьютеров и связанных с ними устройств наиболее распространенной является задача передачи дискретных данных, и, как правило, в значительных объемах (не один бит). Самый распространенный способ представления дан-

[1] Вариант названия одного троичного разряда.

ных сигналами — двоичный: например, условно высокому (выше порога) уровню напряжения соответствует логическая единица, низкому — логический ноль (возможно и обратное представление). Один двоичный сигнал за один квант времени передает один бит информации. Как говорилось ранее, процессор с периферийными устройствами обменивается байтами (8 бит)[1], словами (в мире x86 — 16 бит), двойными словами (32 бита) данных. Для того чтобы передавать группу битов, существует два подхода к организации интерфейса:

- *Параллельный интерфейс* — для каждого бита передаваемой группы имеется своя сигнальная линия (обычно с двоичным представлением), и все биты группы передаются одновременно за один квант времени, то есть продвигаются по интерфейсным линиям параллельно. Примеры: параллельный порт подключения принтера (LPT-порт, 8 бит), интерфейс ATA/ATAPI (16 бит), SCSI (8 или 16 бит), шина PCI (32 или 64 бита).
- *Последовательный интерфейс* — используется лишь одна сигнальная линия, и биты группы передаются друг за другом по очереди; на каждый из них отводится свой квант времени (битовый интервал). Примеры: последовательный коммуникационный порт (COM-порт), последовательные шины USB и FireWire, интерфейсы локальных и глобальных сетей.

На первый взгляд, организация параллельного интерфейса проще и нагляднее (не надо выстраивать биты в очередь на передачу и собирать байты из принятой последовательности битов). Также, на первый взгляд, параллельный интерфейс обеспечивает более быструю передачу данных, поскольку биты передаются сразу пачками. Очевидный недостаток параллельного интерфейса — большое количество проводов и контактов разъемов в соединительном кабеле (по крайней мере, по одному на каждый бит). Отсюда громоздкость и дороговизна кабелей и интерфейсных цепей устройств, но с этим мирятся ради вожделенной скорости. У последовательного интерфейса приемно-передающие узлы функционально сложнее, зато кабели и разъемы гораздо проще и дешевле. Понятно, что на большие расстояния тянуть многопроводные кабели параллельных интерфейсов неразумно (и невозможно), здесь гораздо уместнее последовательные интерфейсы. Эти рассуждения были основополагающими при выборе типа интерфейса примерно до начала 1990-х годов. Тогда выбор был прост: на ближних расстояниях (максимум — до пары десятков метров) при требованиях к высокой скорости использовали параллельные интерфейсы, а на дальних расстояниях или в случае неприемлемости параллельных кабелей — последовательные, жертвуя скоростью передачи.

Теперь поточнее разберемся со скоростью передачи данных. Очевидно, что она равна числу бит, передаваемых за квант времени, деленному на длительность кванта. Для простоты можно оперировать *тактовой частотой интерфейса* — величиной, обратной длительности кванта. Это понятие естественно для синхронных интерфейсов, у которых имеется сигнал синхронизации (clock), определяющий возможные моменты возникновения всех событий (смены состояния). Для асинхронных интерфейсов можно пользоваться эквивалентной

[1] Бывают и не 8-битные байты.

тактовой частотой — величиной, обратной минимальной длительности одного состояния интерфейса. Теперь можно сказать, что максимальная (пиковая) скорость передачи данных равна произведению тактовой частоты на разрядность интерфейса. У последовательного интерфейса разрядность 1 бит, у параллельного — столько, сколько имеется параллельных сигнальных цепей для передачи битов данных. Остаются вопросы о достижимых тактовой частоте и разрядности. И для последовательного, и для параллельного интерфейсов максимальная тактовая частота определяется достижимым (при разумных цене и затратах энергии) быстродействием приемопередающих цепей устройств и частотными свойствами кабелей. Здесь уже проглядывают преимущества последовательного интерфейса: для него затраты на построение высокоскоростных элементов не приходится умножать на разрядность интерфейса, как в случае параллельного интерфейса.

В параллельном интерфейсе есть явление *перекоса* (skew), существенно влияющее на достижимый предел тактовой частоты. Суть его в том, что сигналы, одновременно переданные с одного конца интерфейсного кабеля, доходят до другого конца не одновременно из-за отклонений характеристик цепей. На время прохождения влияют длина проводов, свойства изоляции, соединительных элементов и т. п. Очевидно, что перекос (разница во времени прибытия) сигналов разных битов должен быть явно меньше кванта времени, иначе биты будут искажаться (путаться с одноименными битами предшествующих и последующих посылок). Вполне понятно, что перекос ограничивает и допустимую длину интерфейсных кабелей: при одной и той же относительной погрешности скорости распространения сигналов на большей длине «набегает» и больший перекос. Перекос сдерживает и увеличение разрядности интерфейса: чем больше параллельных цепей, тем труднее добиться их идентичности. Из-за этого даже приходится «широкий» (многоразрядный) интерфейс разбивать на несколько «узких» групп и для каждой группы использовать свои управляющие сигналы. В 90-х годах в схемотехнике приемно-передающих узлов стали осваиваться частоты в сотни мегагерц и выше, то есть длительность кванта стала измеряться единицами и долями наносекунд. Достичь соизмеримо малого перекоса можно лишь в пределах жестких компактных конструкций (печатная плата), а для связи отдельных устройств кабелями длиной в десятки сантиметров пришлось остановиться на частотах до десятков мегагерц. Для того чтобы ориентироваться в числах, отметим, что за 1 наносекунду сигнал пробегает по электрическому проводнику порядка 20-25 сантиметров.

Для повышения пропускной способности параллельных интерфейсов с середины 90-х годов стали применять *двойную синхронизацию* (Dual Data Rate, DDR). Ее идея заключается в выравнивании частот переключения информационных сигнальных линий и линий стробирования (синхронизации). В «классическом» варианте данные информационных линий воспринимаются только по одному перепаду (фронту или спаду) синхросигнала, что удваивает частоту переключения линии синхросигнала относительно линий данных. При двойной синхронизации данные воспринимаются и по фронту, и по спаду, так что частота смены состояний всех линий выравнивается, что при одних и тех же физических

параметрах кабеля и интерфейсных схем позволяет удвоить пропускную способность. Волна этих модернизаций началась с интерфейса ATA (режимы UltraDMA) и прошла уже и по SCSI (Ultral60 и выше), и по памяти (DDR SDRAM). Кроме того, на высоких частотах применяется *синхронизация от источника данных* (source synchronous transfer): сигнал синхронизации, по которому определяются моменты переключения или действительности данных, вырабатывается самим источником данных. Это позволяет точнее совмещать по времени данные и синхронизирующие импульсы, поскольку они распространяются по интерфейсу параллельно в одном направлении. Альтернатива — синхронизация от общего источника (common clock) — не выдерживает высоких частот переключения, поскольку здесь в разных (географически) точках временные соотношения между сигналами данных и синхронизации будут различными.

Повышение частоты переключений интерфейсных сигналов, как правило, сопровождается *понижением уровней* сигналов, формируемых интерфейсными схемами. Эта тенденция объясняется энергетическими соображениями: повышение частоты означает уменьшение времени, отводимого на переключения сигналов. Чем больше амплитуда сигнала, тем большие требуются скорость нарастания сигнала и, следовательно, выходной ток передатчика. Повышение выходного тока (импульсного!) нежелательно по разным причинам: большие перекрестные помехи в параллельном интерфейсе, необходимость применения мощных выходных формирователей, повышенное тепловыделение. Тенденцию снижения напряжения можно проследить на примере порта AGP (3,3/1,5/0,8 В), шин PCI/PCI-X (5/3,3/1,5 В), SCSI, шин памяти и процессоров.

В *последовательном* интерфейсе явление перекоса отсутствует, так что повышать тактовую частоту можно вплоть до предела возможностей приемнопередающих цепей. Конечно, есть ограничения и по частотным свойствам кабеля, но изготовить хороший кабель для одной сигнальной цепи гораздо проще, чем для группы цепей, да еще и с высокими требованиями к идентичности. А когда электрический кабель уже «не тянет» требуемые частоту и дальность, можно перейти на оптический, у которого есть в этом плане огромные, еще не освоенные «запасы прочности». Устраивать же параллельный оптический интерфейс — слишком дорогое удовольствие[1].

Приведенные соображения объясняют тенденцию перехода на последовательный способ передачи данных.

Сигналы и среда передачи

Самым «модным» физическим процессом, используемым для передачи сигналов интерфейсов, являются электромагнитные колебания различных частотных диапазонов. Наиболее привычные *электрические сигналы* — это электромагнитные колебания сравнительно низкочастотного диапазона (до десятков и сотен мегагерц), передаваемые по электрическим проводам. Передатчик такого сигна

[1] В 10-гигабитной версии технологии Ethernet есть параллельно-последовательный вариант.

ла посылает в электрическую линию связи (кабель) сигнал в виде определенных уровней напряжения или тока, приемник на другом конце линии получает сигнал, в той или иной степени похожий на переданный. Волновые явления заставляют применять для передачи сигнала специальные конструкции электрических кабелей — коаксиальные кабели, витые (скрученные) пары проводов и некоторые другие. Назначение этих конструкций — максимально сохранить форму передаваемого сигнала, не выпустить его за пределы кабеля и, по возможности, не впустить внешние помехи. Последние два пункта имеют особое значение для обеспечения безопасности передачи информации (в плане конфиденциальности) — недопущения (осложнения) ее «подслушивания» и злонамеренного искажения извне. Проводная передача электрического сигнала реализована в подавляющем большинстве интерфейсов периферийных устройств, обеспечивая дальность передачи в единицы, десятки и сотни метров со скоростями до единиц гигабит в секунду; она же доминирует и в компьютерных сетях.

Электромагнитные колебания с частотами в десятки и сотни мегагерц пригодны и для *беспроводной радиопередачи сигналов*. Для беспроводной связи широко используется микроволновый диапазон частот около 2,4 ГГц. В этом диапазоне радиоволны распространяются по прямой (нет эффекта огибания, свойственного длинным волнам), с некоторым затуханием проходя сквозь стены зданий. Осложняет связь отражение сигнала от различных предметов, в результате которого приемник получает не только прямой сигнал от передатчика, но и отраженные сигналы, приходящие с некоторой задержкой относительного прямого. Из-за этого эффекта многолучевого приема в некоторых точках пространства на определенных частотах связь оказывается невозможной, но достаточно немного сместить приемник (или передатчик) или изменить частоту, как связь появляется. С замиранием сигнала из-за многолучевого приема борются разными способами. Беспроводные интерфейсы привлекательны отсутствием кабелей и разъемов, которые нужно прокладывать и соединять для организации связи, — соединяемым устройствам достаточно лишь оказаться в зоне действия. Однако это имеет и обратную сторону — среда передачи полностью открыта, в том числе для злоумышленников, которые могут перехватывать сигнал для съема информации и передавать свой сигнал для вредоносных действий. Эти проблемы безопасности имеют решения, выходящие за рамки уровня физической передачи сигнала. А вот обратная сторона медали: высокая и постоянно растущая «заселенность» радиоэфира, вызывающая интерференцию (нежелательное взаимодействие) излучений различной аппаратуры (беспроводных и сотовых телефонов, аппаратуры беспроводных локальных сетей, микроволновых печей и других устройств).

Если двигаться дальше в сторону повышения частоты электромагнитных колебаний, то мы попадаем сначала в инфракрасный диапазон, к которому примыкает и видимый оптический диапазон. Эти диапазоны также используются для оптической передачи сигналов как по проводам (оптоволокну), так и без проводов.

Инфракрасный порт — стандартный IrDA и его фирменные предшественники HP-SIR и ASK IR — уже долгие годы используется для беспроводного подклю-

чения периферии (принтеров и других устройств) к компьютерам. Особенно эффектно это подключение выглядит с малогабаритными устройствами, которые соизмеримы кабелям и разъемам традиционных интерфейсов (а то и меньше их). Малая (по сравнению с радиоинтерфейсом) зона охвата не всегда является недостатком — ее проще контролировать на предмет несанкционированных подключений, будучи уверенным, что из-за стенки никто не подключится и не подслушает.

В *проводной оптической связи* световые импульсы инфракрасного диапазона передаются по стеклянному или пластиковому оптоволокну. Стеклянное волокно в основном используется в телекоммуникациях, где требуется дальность связи, измеряемая сотнями метров и десятками (и даже сотнями) километров. Недостаток стеклянной оптики — дороговизна оконечных устройств (приемопередатчиков) и соединительной аппаратуры, сам же кабель может быть дешевле медного. В интерфейсах, не требующих больших расстояний (до десятков метров), с успехом применяется пластиковое волокно, для которого и кабели, и разъемы существенно дешевле. Примеры оптического интерфейса в современном персональном компьютере — Toslink (оптическая версия цифрового аудиоинтерфейса S/PDIF) и Fibre Channel (FCAL), с помощью которого подключают устройства хранения данных.

Говоря об оптических и радиоинтерфейсах, следует отметить, что они обеспечивают полную *гальваническую развязку* соединяемых устройств. Кроме того, оптический интерфейс нечувствителен к электромагнитным помехам. В ряде случаев эти свойства имеют решающее значение, например при соединении оборудования на энергетических объектах, в производственных помещениях с сильными источниками помех и т. п. Проводные оптические интерфейсы — наиболее защищенные от несанкционированного подключения. Съем информации без механического вмешательства в кабельное хозяйство практически невозможен, при необходимости можно организовать мониторинг состояния линии и своевременно засечь попытку подключения.

Гальваническая развязка устройств

Гальваническая развязка означает, что «схемные земли» соединяемых устройств не имеют электрической связи друг с другом через интерфейсные цепи. При этом устройства (их «схемные земли») могут иметь существенно различающиеся потенциалы.

В большинстве электрических интерфейсов гальваническая развязка отсутствует. Так, например, «схемные земли» устройств, соединенных кабелями с COM- или LPT-портами PC, оказываются связанными со «схемной землей» компьютера (и между собой). Если между устройством и компьютером до подключения интерфейсного кабеля была разность потенциалов, то по общему проводу интерфейса потечет уравнивающий ток, что плохо по целому ряду причин. Падение напряжения на общем проводе, вызванное протеканием этого тока, приводит к смещению уровней сигналов, а протекание переменного тока приводит к сложению полезного сигнала с переменной составляющей — помехой. К этим помехам особенно чувствительны ТТЛ-интерфейсы; в то же время

в RS-232C смещение и помеху в пределах 2 В поглотит зона нечувствительности. В случае обрыва общего провода или плохого контакта, а гораздо чаще — при подключении и отключении интерфейсов без выключения питания устройств разность потенциалов прикладывается к сигнальным цепям, а протекание уравнивающих токов через них часто приводит к пиротехническим эффектам. В аудиотехнике уравнивающие токи ведут к слышимым помехам (фону).

Гальваническая развязка сигналов интерфейса от «земли» устройства осуществляется с помощью оптоэлектронных приборов (интерфейсы MIDI, «токовая петля») или трансформаторов (шина FireWire, сетевые интерфейсы Ethernet). Иногда развязку по постоянному току осуществляют с помощью разделительных конденсаторов (дешевые варианты интерфейса FireWire).

Разность потенциалов устройств, соединяемых интерфейсом с гальванической развязкой, ограничена допустимым для данного интерфейса *напряжением изоляции.* Так, например, адаптеры Ethernet (для витой пары) должны выдерживать напряжение до 1,5 кВ, развязка на оптронах — 500-1000 В, конденсаторная развязка в FireWire — до 60 В. Оптоволоконные интерфейсы обеспечивают развязку с напряжением до тысяч и даже миллионов вольт. Гальваническую развязку обеспечивают и любые беспроводные интерфейсы.

Достоверность, надежность передачи и управление потоком

Контроль достоверности передачи данных — это возможность обнаружения, а иногда и исправления ошибок, возникающих при передаче. Этот контроль реализован далеко не во всех интерфейсах: где-то достоверность не очень важна, где-то вероятность возникновения ошибок пренебрежимо мала. В новых интерфейсах контролю достоверности уделяется серьезное внимание, поскольку они, как правило, рассчитываются на экстремальные условия работы (высокие частоты, большие расстояния, наличие помех).

Проверка на четность (parity check) — простейший способ обнаружения ошибок. Здесь к каждому передаваемому элементу информации (как правило, байту или слову) добавляется бит четности (parity), дополняющий число единичных информационных битов до четного (even parity) или нечетного (odd parity). Приемник проверяет количество единичных битов, включая контрольный, на четность (или нечетность, в зависимости от соглашения) и в случае несоответствия считает принятые данные искаженными. Проверка четности — самый примитивный и неэффективный вариант контроля достоверности; при заметных накладных расходах (обычно 1 бит на каждый байт) в ходе этой проверки не выявляются все ошибки четной кратности (искажения четного числа битов). Проверка четности применяется в последовательных интерфейсах (COM-порт), шине SCSI; раньше она применялась и для памяти.

Дублирование информации — еще более расточительный (но и более надежный) способ контроля, применяемый для небольших объемов информации. Здесь каждый информационный элемент (обычно битовое поле длиной в несколько битов) повторяется дважды, причем одна из копий может передаваться в инверс

ном виде. Несовпадение принятых копий считается ошибкой. Так, например, защищаются идентификаторы пакетов в USB. Развитие этой идеи — трехкратное повторение блока: если из трех принятых копий две совпали, то их считают верными (это можно считать и исправлением ошибки). Данный способ применяется в радиоинтерфейсе Bluetooth.

Более сложный, но и более эффективный вариант контроля — вычисление *циклического избыточного кода* (Cyclic Redundant Code, CRC) и добавление его к передаваемой информации. Так, 16-битный CRC-код способен с очень высокой вероятностью обнаружить ошибки в блоках данных размером до 4 Кбайт. Подсчет CRC удобно выполнять при последовательной передаче данных — для этого требуются несложные аппаратные схемы (регистр сдвига с обратными связями). При параллельной передаче (и программно на процессоре общего назначения) подсчет CRC трудоемок. Тем не менее CRC-контроль применяется и в параллельном интерфейсе IDE/ATA, но только в режимах UltraDMA (в других режимах передачи на этой шине никак не контролируются).

Для исправления ошибок передачи применяют *коды с исправлением ошибок* (Error Checking and Correction, ECC). Идея заключается в подсчете нескольких проверочных битов, каждый из которых вычисляется по правилам контроля четности для определенных групп информационных битов. Специальное разделение на группы (они пересекаются) позволяет по принятым информационным и проверочным битам обнаруживать и даже исправлять ошибки. Число проверочных битов зависит от числа информационных битов и желаемой кратности (числа искаженных битов) исправляемых и обнаруживаемых ошибок. Так, для исправления однократных и обнаружения всех двукратных ошибок (и подавляющего большинства ошибок большей кратности) для 8 информационных битов требуется 4 проверочных, для 16 — 5, для 32 — 6, для 64 — 7 проверочных битов. ECC-коды широко применяются для контроля памяти (особенно кэшпамяти) и в ряде интерфейсов (например, PCI-X).

Под обеспечением *надежности передачи* понимается доведение до инициатора транзакции сведений о состоянии ее выполнения (успешно-неуспешно), что позволяет ему в случае неуспеха предпринять какие-то специальные действия (например, попытку повтора). Ряд интерфейсов (и протоколов) не обеспечивает надежности: так, на шине ISA возможно даже обращение к несуществующему устройству. При этом операции записи идут просто «в никуда», а операции чтения обычно возвращают «пустые» данные (FFh), которые инициатор не может отличить от настоящих. Шина PCI является надежной: инициатор всегда знает судьбу своих транзакций; достоверность передач проверяется (по четности или ECC).

Во многих случаях возникает задача согласования темпа работы устройств, связанных интерфейсом, которая решается с помощью механизмов квитирования или/и управления потоком.

Квитирование — это взаимное подтверждение отдельных шагов протокола обоими участниками транзакции, что позволяет согласовать темп работы инициатора и целевого устройства. Квитирование широко применяется в параллельных

интерфейсах (в том же LPT-порте, шинах расширения), для чего используются специальные интерфейсные линии.

Управление потоком — это уведомление источника (передатчика) данных о возможностях их приема противоположной стороной: если приемник не успевает обрабатывать приходящие данные, он «просит» передатчик приостановить передачу на определенное время или до особого разрешения.

В тех интерфейсах, где имеется квитирование, отдельная задача управления потоком, как правило, не возникает (квитирование обеспечивает и согласование темпа). В последовательных интерфейсах без управления потоком в общем случае не обойтись; в COM-порте имеются даже два варианта протокола управления потоком.

Асинхронные, синхронные и изохронные передачи

Теперь обсудим требования, которые могут предъявляться ко времени и темпу выполнения транзакций, используемых для организации асинхронных, синхронных и изохронных передач данных. Сразу заметим, что связь типов интерфейсов (асинхронных и синхронных) и типов передач не является жесткой.

В *асинхронных передачах данных* и *интерфейсах* участники не имеют друг перед другом никаких особых обязательств по времени: инициатор в любой момент может начать транзакцию, а целевое устройство, как правило, может ее приостановить в случае своей неготовности. Асинхронная передача применима для всех устройств, не связанных с реальным временем: принтеров, сканеров, устройств хранения и т. п.

Синхронная передача данных — это передача с *постоянной мгновенной скоростью.* Она требуется, например, для мультимедийных данных, в частности — для передачи оцифрованного звука в формате ИКМ (он же PCM — передача отсчетов сигнала через равные промежутки времени). В телефонии отсчеты (8 бит) передаются с частотой 8 кГц (итого получаем скорость 64 Кбит/с), а для высококачественного звуковоспроизведения в аудио-CD — с частотой 44,1 кГц по 16 бит на стереоканал (около 1,4 Мбит/с). Нарушение синхронности ведет к потере данных — искажениям, помехам, провалам звука. Синхронная передача данных требует выделенного синхронного интерфейса для каждого подключаемого устройства (или сложных систем мультиплексирования).

Изохронная передача данных — это передача с *постоянной средней скоростью:* за определенный (фиксированный) интервал времени должен быть передан определенный объем данных, но сама скорость (мгновенная), с которой данные передаются, не оговаривается. Конечно, мгновенная скорость должна быть, по крайней мере, не ниже средней. Обычно мгновенная скорость (пропускная способность интерфейса) выбирается намного выше требуемой средней скорости. Это позволяет использовать один интерфейс для подключения множества устройств и организовывать множество одновременных изохронных каналов передачи (с суммарной скоростью несколько меньшей, чем пропускная способность интерфейса). В одном интерфейсе изохронные передачи спокойно уживаются

с асинхронными. Вопросами распределения полосы пропускания интерфейса занимается *диспетчер изохронных ресурсов* — отдельная функция программной поддержки. Изохронные передачи требуются мультимедийным устройствам — аудио- и видеоаппаратуре. В устройствах имеется буферная память, в которую складываются поступающие пакеты изохронных передач, а «расходование» этих данных (например, на звуковоспроизведение) внутри устройств происходит уже с постоянной мгновенной скоростью (обратная передача происходит похожим образом). Изохронные передачи удобны и в мультимедийных приложениях с переменной скоростью (когда используется сжатие данных, скорость их поступления может колебаться, но, конечно, до известного предела). Изохронные передачи поддерживаются шинами USB, FireWire, радиоинтерфейсом Bluetooth; изохронный трафик могут нести и обычные сети передачи данных с достаточно высокой пропускной способностью. Поддержка изохронного обмена введена и в новые интерфейсы системного уровня: AGP 3.0, PCI Express. Точная синхронизация изохронных устройств имеет свои особенности: устройствам приходится задействовать собственные (очень точные!) тактовые генераторы, поскольку непосредственной синхронизации (как в синхронных интерфейсах) у них нет. Для решения задачи синхронизации используются, например, механизмы обратной связи, позволяющие устройствам корректировать уход своих «часов».

2.4. Карты, сокеты, слоты, джамперы

Для большей ясности дальнейшего изложения определим некоторые термины, относящиеся к аппаратным средствам современных компьютеров. Поскольку персональные компьютеры, увы, имеют иностранное происхождение (опала, в которую попала кибернетика в нашей стране, не позволила удержать приоритеты в этой области), приходится мириться с рядом иностранных слов, вошедших в технический русский язык в виде даже не всегда правильных транслитераций. Во многих случаях для профессионалов они понятнее — много информации по данной теме черпается из зарубежных, чаще англоязычных источников. Рассмотрим варианты названий основных элементов компьютера.

Системной (system board), или *материнской* (mother board) *платой* называют основную печатную плату, на которой устанавливаются процессор, оперативная память, ROM BIOS и некоторые другие системные компоненты (см. главу 6).

Картой (*платой) расширения* (expansion card) называют печатную плату с краевым разъемом, устанавливаемую в слот расширения. Карты расширения, привносящие в PC какой-либо дополнительный интерфейс, называют *интерфейсными картами* (interface card). Поскольку интерфейсная карта представляет собой «приспособление» для подключения какого-либо устройства, к ней применимо и название *адаптер* (adapter). К примеру, дисплейный адаптер (display adapter) служит для подключения дисплея-монитора. Названия «интерфейсная карта», «адаптер» и «контроллер» зачастую считаются синонимами (разница между ними поясняется в главе 1), хотя адаптеры и контроллеры могут размещаться и на системной плате. Рис. 2.8 и 2.9 иллюстрируют легкую различи-

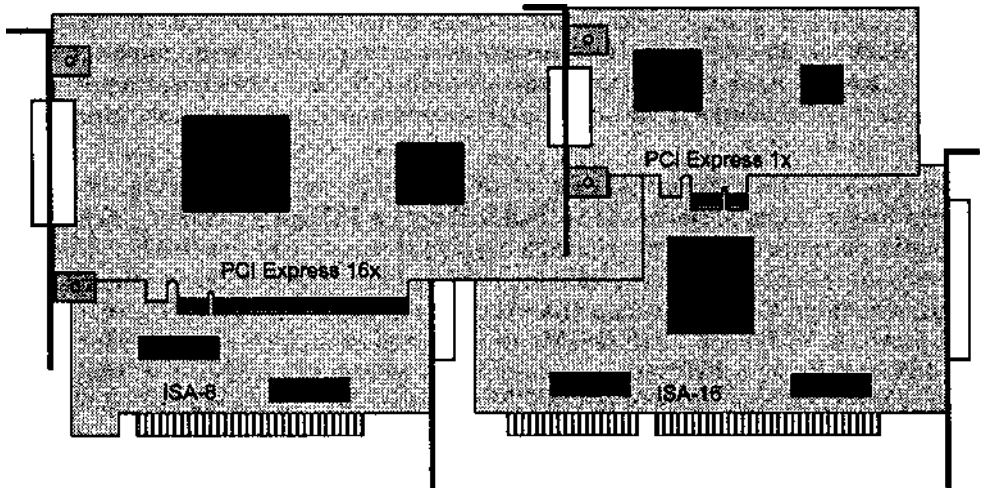


Рис. 2.8. Карты расширения ISA и PCI-E

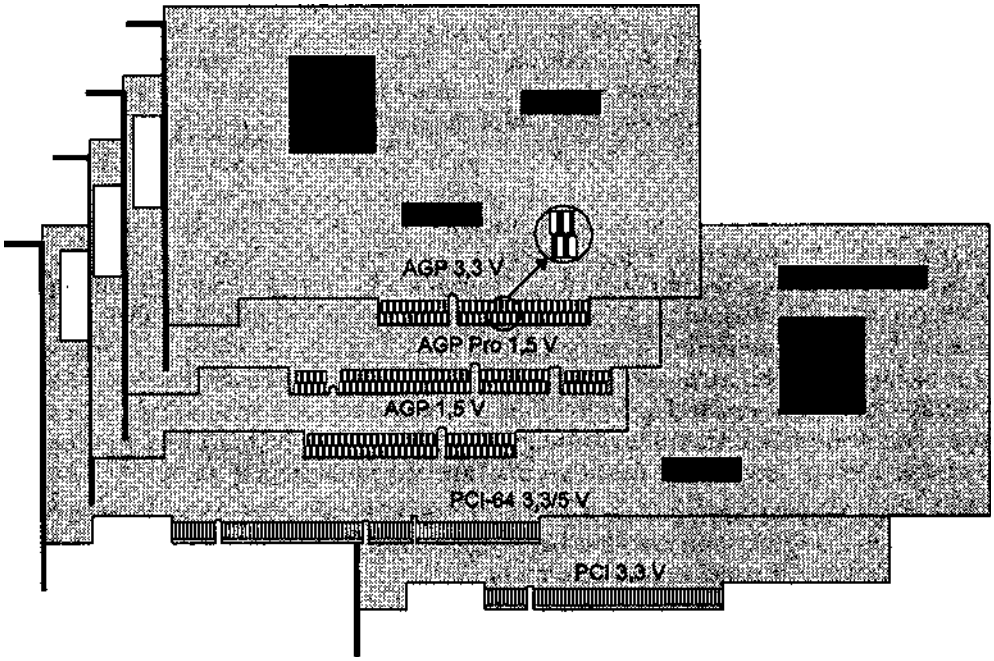


Рис. 2.9. Карты расширения PCI и AGP

мость карт ISA, PCI, AGP и PCI-E (PCI Express) по виду и расположению краевых разъемов. Допустимая длина карты может быть ограничена особенностями корпуса и компоновкой элементов системной платы (иногда ее установке мешают высокие элементы). Максимальная длина карты расширения составля

ет 335 мм, при этом ее передний край должен входить в направляющие полозья, установленные на корпусе. Полные длину и высоту (full size) имеют только очень старые или достаточно сложные адаптеры. Большинство адаптеров короче и ниже, встречаются и фигурно вырезанные платы — этим экономится расходуемый материал.

Слот (slot) — это щелевой разъем, в который устанавливается какая-либо печатная плата. *Слот расширения* (expansion slot) в PC представляет собой разъем шины расширения ввода-вывода в совокупности с прорезью в задней стенке корпуса компьютера — то есть является посадочным местом для установки карты расширения. Слоты расширения имеют разъемы шин ISA/EISA, PCI, AGP, PCI-E, MCA, VLB и PC Card (PCMCIA), о которых подробно рассказано в главе 14. Внутренние слоты используются и для установки модулей оперативной памяти (DIMM), кэш-памяти (COAST), определенных типов процессоров (Pentium II/III, Athlon), а также процессорных модулей в некоторых моделях PC.

У карт PCI, AGP и PCI-E, в отличие от ISA/EISA и VLB, компоненты (микросхемы и другие детали) расположены на левой стороне печатной платы (на рис. 2.8 и 2.9 платы изображены со стороны компонентов). Для экономии места на системной плате используют так называемый разделяемый слот (shared slot). На самом деле это окно на задней стенке корпуса, которое может использоваться либо картой ISA, либо картой PCI. Таким образом, максимальное суммарное количество доступных адаптеров оказывается на единицу меньшим, чем видимое количество слотов на системной плате.

Сокет (socket) представляет собой гнездо, в которое устанавливаются микросхемы. Его контакты рассчитаны на микросхемы со штырьковыми выводами в корпусах DIP, PGA во всех модификациях или же микросхемы в корпусах SOJ и PLCC с выводами в форме буквы «J». Сокет *ZIP-Socket* (Zero Insertion Force — нулевое усилие вставки) предназначен для легкой установки при высокой надежности контактов. Эти гнезда имеют замок, открыв который, можно без усилий установить или изъять микросхему. Для работы после установки замок закрывают, при этом контакты сокета плотно обхватывают выводы микросхемы. Сокет для процессоров Pentium 4 в корпусе LGA отличается от своих предшественников тем, что штырьковые контакты установлены в самом сокете, а на процессоре для них размещены плоские контактные площадки.

Джампер (jumper) представляет собой съемную перемычку, устанавливаемую на торчащие из печатной платы штырьковые контакты (рис. 2.10, *а*). Джамперы используются для конфигурирования различных компонентов как выключатели или переключатели, для которых не требуется оперативного управления. Джамперы переставляют с помощью пинцета, что рекомендуется делать только при выключенном питании, поскольку есть опасность уронить их в неподходящее место или закоротить пинцетом близко расположенные контакты.

DIP-переключатели (DIP switches) представляют собой малогабаритные выключатели в корпусе DIP (рис. 2.10, *б),* применяемые для тех же целей, что и джамперы. Их преимущество в более легком переключении, которое удобно

производить шариковой ручкой. Недостатками переключателей являются большее, по сравнению с джамперами, занимаемое на плате место и более высокая цена. Кроме того, несмотря на название, они обычно являются только выключателями, что делает их применение менее гибким, чем применение джамперов.

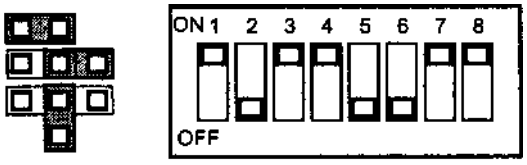


Рис 2.10. Аппаратные средства конфигурирования: а — джампер, б — DIP-переключатель

В современных компонентах стремятся сокращать количество переключателей или джамперов, стараясь переложить все конфигурационные функции на программные компоненты. Платы (карты), в которых удается изжить джамперы полностью (но которые требуют конфигурирования), называют *jumperless cards* (карты, свободные от джамперов). Компоненты, которые после установки конфигурируются автоматически, относят к классу PnP (Plug and Play — вставляй и играй).

Чип (chip) — это полупроводниковая микросхема, причем обычно неявно подразумевается ее функциональная сложность. В данной книге это слово больше не встретится — автору больше нравится слово «микросхема». *Чипсет* (chip set) — это «набор интегральных схем, при подключении которых друг к другу формируется функциональный блок вычислительной системы» (формулировка из толкового словаря по вычислительным системам; к ней можно добавить слово «специализированных»). Чипсеты широко применяются в системных платах, графических контроллерах и других сложных узлах, функции которых в одну микросхему заложить не удается.

2.7. Кабели и разъемы

Для соединения устройств и узлов PC применяются различные разъемы, из которых здесь мы упомянем лишь несколько.

Разъемы *D-типа* (рис. 2.11) используются для подключения внешних устройств — мониторов, принтеров, модемов, манипуляторов и т. п. Розетки (sockets, female, в просторечии «мамы») обозначаются как *DB-xxS* или *DB-xxF*, где *xx* — количество контактов. Вилки (plug, male, они же «папы») обозначаются как *DB-xxP* или *DB-xxM*. Ключом является D-образный кожух, однако трехряд- ные разъемы кабелей мониторов почему-то довольно легко удается вставить «вверх ногами». Назначение разъемов, выходящих на заднюю стенку PC, стандартизовано (табл. 2.1).

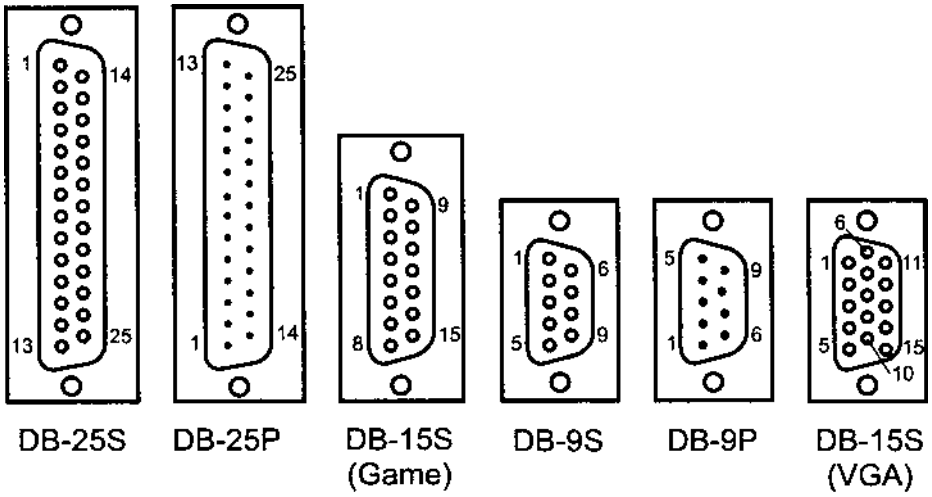


Рис. 2.11. Разъемы D-типа (вид с наружной стороны)

Таблица 2.1. Назначение разъемов D-типа

Тип разъема	Назначение
Вилка DB-9P	COM-порт
Розетка DB-9S	Выход на монитор Mono, CGA, EGA
Розетка DB-15S (двухрядный)	Game-порт, MIDI
Розетка DB-15S (трехрядный)	Выход на монитор VGA/SVGA
Вилка DB-25P	COM-порт
Розетка DB-25S	LPT-порт

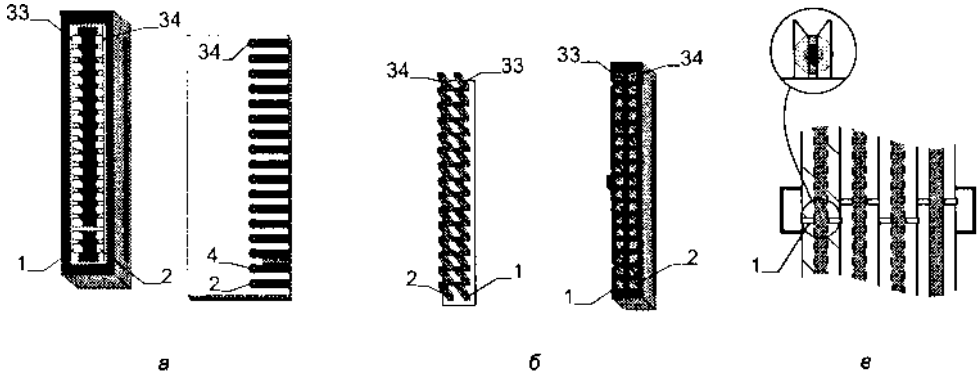


Рис. 2.12. Разъемы IDC: *а* — краевые, *б* — штырьковые, *в* — заделка проводов

Разъемы *IDC* (Insulation-Displacement Connector — разъем, смещающий изоляцию) получили название от способа присоединения кабеля. Контакты этих разъемов со стороны, обращенной к кабелю, имеют ножи, подрезающие и смещающие изоляцию проводников кабеля. Эти разъемы предназначены в основ-

ном для ленточных кабелей-шлейфов, хотя в них возможна заделка и одиночных проводников. Для заделки кабелей в эти разъемы существуют специальные инструменты-прессы, но при необходимости можно обойтись плоской отверткой (и умелыми руками). Разъемы IDC существуют как ответные части для краевых печатных разъемов (рис. 2.12, а) и штырьковых контактов (рис. 2.12, *б).* Разъемы могут иметь ключи: для печатных разъемов это прорезь и соответствующая ей перемычка, расположенная ближе к первым контактам. Для штырьковых разъемов ключом является выступ на корпусе, но этот ключ сработает, только если ответная часть имеет пластмассовый бандаж с прорезью. Дешевые варианты штырьковых разъемов бандажа не имеют. Ключом может являться и отсутствующий штырек — на разъеме для него не оставляют отверстия. На ленточном кабеле крайний провод, соединяемый с контактом «1», маркируют цветной краской. На печатной плате штырек «1» обычно имеет отличающуюся от формы других (квадратную) форму контактной площадки. Разъемы IDC и ленточные кабели-шлейфы применяют для соединений внутри корпуса — подключения накопителей, а также для соединения внешних разъемов с системной платой и картами расширения.

Разъемы типа *Centronics* (рис. 2.13) имеют надежные пружинистые контакты и проволочные петли-фиксаторы. Свое название они получили в честь фирмы Centronics Data Corporation, выпускавшей первые широкодоступные матричные принтеры для ПК. На этих принтерах устанавливались 36-контактные разъемы такого типа. Существуют разъемы типа Centronics и с другим числом контактов, а также в малогабаритных вариантах. Они применяются в интерфейсе SCSI и некоторых других.

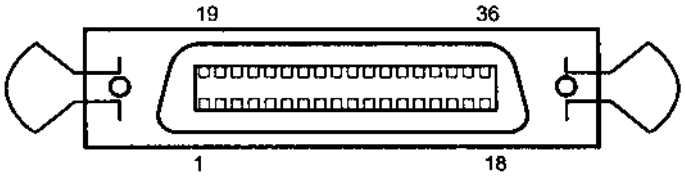


Рис. 2.13. Разъемы типа Centronics

О других типах разъемов и кабелей будет рассказано при описании подсистем, в которых они применяются.

ГЛАВА 3

Питание компьютеров и периферийных устройств

Эта глава посвящена «энергетическому интерфейсу» компьютера: в ней рассматривается, откуда берутся питающие напряжения и как их правильно подать на компоненты компьютера; куда девать высвобождающуюся тепловую энергию (иной физической работы, кроме нагревания окружающей среды, компьютер, увы, не производит). В этой главе также рассматриваются «здоровые отношения» компьютера, пользователя и питающей сети (условия их мирного сосуществования).

3.1. Схемотехника блоков питания

Блоки питания аппаратуры, предназначенные для питания от сети переменного тока, в зависимости от назначения и мощности могут быть выполнены по различным схемам. Схема простейшего *блока питания с трансформаторным входом* приведена на рис. 3.1.

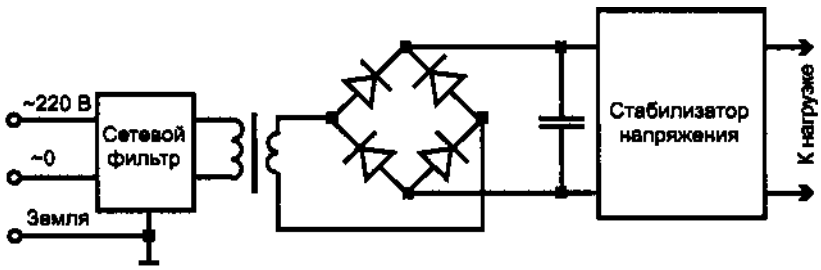


Рис. 3.1. Блок питания с трансформаторным входом

Здесь понижающий трансформатор, работающий на частоте питающей сети 50/60 Гц, обеспечивает требуемое напряжение и гальваническую развязку питаемых цепей от сети переменного тока. Выходное напряжение может стабилизироваться непрерывным или импульсным низковольтным стабилизатором напряжения. Главный недостаток такого блока — большие габариты низкочастотного силового трансформатора. Трансформатор блока питания, рассчитан-

ный на частоту 60 Гц (зарубежные питающие сети), на частоте 50 Гц (наши сети) может ощутимо нагреваться. Естественно, от сети постоянного тока (такие изредка встречаются) такой блок работать не может. Блоки питания с трансформаторным входом применяются при небольшой выходной мощности, чаще всего — в выносных адаптерах (старых моделей), обеспечивающих питание модемов, хабов и прочих маломощных устройств внешнего исполнения. Такие блоки довольно часто монтируются прямо на вилке питания.
В *блоках питания* с *бестрансформаторным входом* понижающий трансформатор работает на высокой частоте — в десятки и даже сотни килогерц, что позволяет уменьшить габариты и вес блока питания. В этом случае входное напряжение сразу выпрямляется и после фильтрации поступает на высокочастотный преобразователь. Высокочастотные импульсы преобразователя попадают на понижающий импульсный трансформатор, который обеспечивает гальваническую развязку выходных и входных цепей. Преобразователь чаще всего делают управляемым, так что на него возлагаются еще и функции регулирующего элемента стабилизатора напряжения. Управляя шириной импульса, можно изменять величину энергии, поступающей через трансформатор в выпрямитель,
и, следовательно, регулировать (стабилизировать) его выходное напряжение. В зависимости от мощности стабилизатор строится по однотактной или двухтактной схеме. Однотактная схема несколько проще (рис. 3.2), ее применяют в блоках питания, где мощность обычно не превышает сотни ватт (например, в мониторах). В мониторах частоту импульсного блока обычно синхронизируют с частотой генератора строчной развертки во избежание видимых помех. В настоящее время выпускается широкий ассортимент управляющих микросхем со встроенным ключевым транзистором и развитыми функциями защиты и управления. Блоки питания на их основе получаются предельно простыми и компактными; маломощные блоки могут размещаться прямо в вилках-адаптерах.

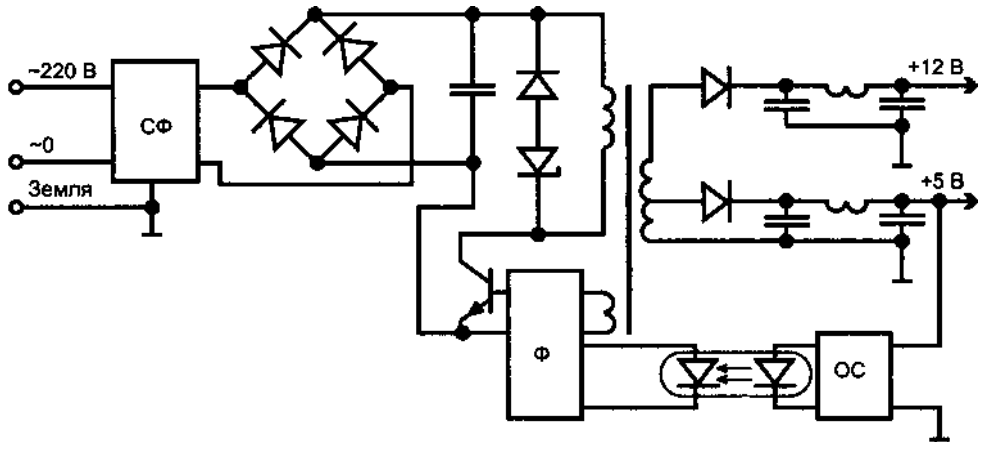


Рис. 3.2. Однотактный блок питания (СФ — сетевой фильтр, Ф — формирователь импульсов, ОС - усилитель обратной связи)

Двухтактные преобразователи сложнее, но они обеспечивают большую выходную мощность. Такие блоки широко используются в блоках питания PC (см. 3.2).

Если блок питания должен вырабатывать несколько выходных напряжений, сам преобразователь может стабилизировать лишь одно из них. Остальные напряжения могут быть стабилизированы дополнительными выходными стабилизаторами, но часто их оставляют нестабилизированными. При этом появляется взаимозависимость: чем больше нагрузка но основной (стабилизированной) цепи, тем выше напряжения на остальных шинах.

Импульсные блоки питания имеют малые габариты, но компактный трансформатор представляет собой довольно сложное изделие. Импульсные помехи, которые могут проникать как в питаемые, так и в питающие цепи, подавляют тщательно разработанными фильтрами. Внешнее излучение подавляется металлическим экраном, в который заключают весь блок.

Импульсные блоки питания не критичны к частоте сети (50 или 60 Гц), могут работать от постоянного тока и часто в широком диапазоне входных напряжений. Современные блоки, у которых указано свойство Autoswitching Power Supply, работают в диапазоне 110-230 В без переключателя напряжения. Такие блоки применяются в большинстве современных мониторов.

ВНИМАНИЕ----------

Самый тяжелый режим функционирования элементов блока питания возникает в момент включения. После выключения блока питания (любой конструкции) включать его повторно рекомендуется не раньше, чем через 10 с. Несоблюдение этой рекомендации может сократить жизнь блока питания.

Наличие выпрямителя и накопительного конденсатора на входе бестрансформаторного блока питания обусловливает ярко выраженную динамическую нелинейность входной цепи. На рис. 3.3 приведены осциллограммы напряжения сети и потребляемого тока, которые иллюстрируют эту нелинейность. Пока мгновенное значение напряжения ниже напряжения на накопительном конденсаторе выпрямителя, ток практически не потребляется. На верхушках синусоиды ток резко возрастает, так что в его спектре очень сильно выражена 3-я гармоника. Для питающей сети такой характер нагрузки нежелателен, но с ним приходится мириться. Конечно, нелинейность имеется и в трансформаторном блоке питания, но она несколько сглаживается низкочастотным трансформатором.

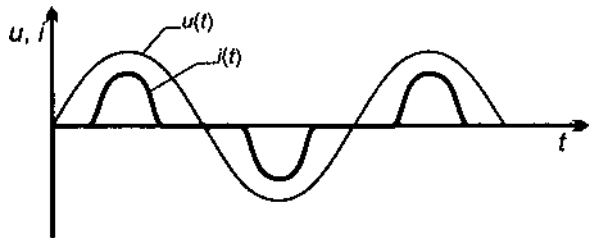


Рис. 3.3. Нелинейность входной цепи бестрансформаторного блока питания

3.2. Блок питания PC

Блок питания PC обеспечивает напряжениями постоянного тока системный блок со всеми его сложными и часто привередливыми устройствами. С самых первых моделей PC здесь применяется двухтактная схема преобразователя с бестрансформаторным входом, без революционных изменений эта схема дошла и до наших дней (ее упрощенный вариант приведен на рис. 3.4). Входное напряжение после высокочастотного фильтра выпрямляется и поступает на накопительные конденсаторы (C1 и C2), являющиеся главными хранителями энергии на случай кратковременного провала питающего напряжения. Мощные высоковольтные транзисторы T1 и T2 и конденсаторы C1 и C2 образуют полу- мостовую схему генератора-преобразователя, нагрузкой которого является высокочастотный импульсный силовой трансформатор Тр2. Этот трансформатор обеспечивает и гальваническую развязку выходных и входных цепей. Преобразователь является регулирующим элементом стабилизатора напряжения основного источника: +3,3 В для ATX (и более новых конструктивов) или +5 В (PC/AT). Остальные напряжения могут быть стабилизированы дополнительными выходными стабилизаторами, но чаще их оставляют нестабилизированными. При этом чем больше нагрузка блока по основной (стабилизированной) цепи, тем выше напряжения на остальных шинах. Убедиться в этом просто — понаблюдайте за вентилятором блока питания, который питается от цепи +12 В, изменяя нагрузку по основной цепи, например подключая и отключая систем-

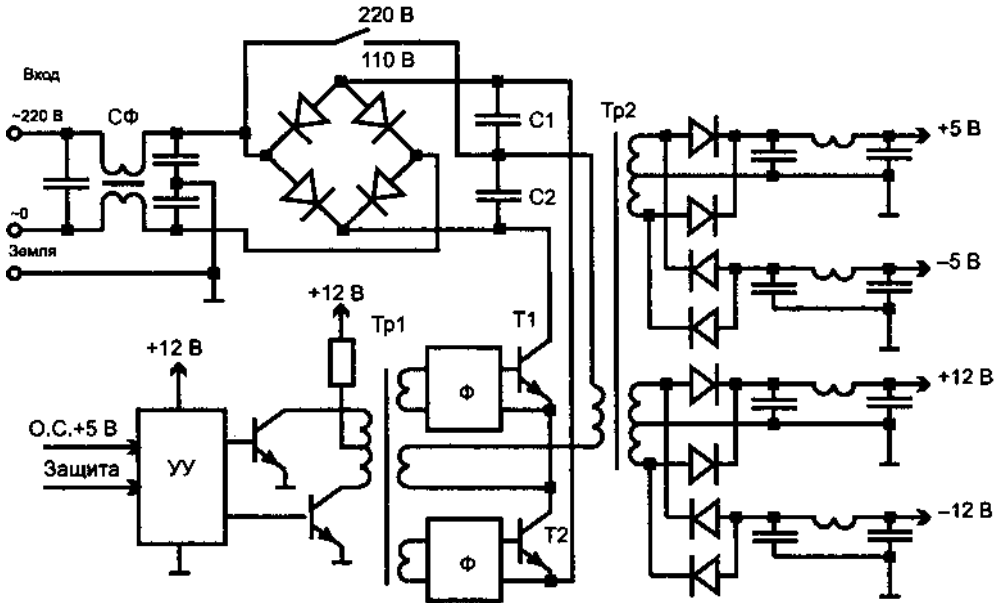


Рис. 3.4. Двухтактный блок питания: СФ - сетевой фильтр, УУ — устройство управления, Ф — формирователи импульсов, Тр1 — трансформатор развязки цепей управления, Тр2 — силовой трансформатор

ную плату. При подключении нагрузки скорость вращения вентилятора повышается. Это происходит потому, что с повышением тока нагрузки преобразователь вырабатывает более широкие импульсы, а выходное напряжение нестабилизированных выпрямителей при постоянной нагрузке пропорционально ширине этих импульсов. По этой причине уровни напряжения на не основных выходах большинства блоков питания соответствуют номиналам лишь при номинальной и сбалансированной нагрузке. Однако, как правило, потребители этих напряжений не требуют особой точности напряжения, а стабильность обеспечивается относительным постоянством нагрузки основной цепи.

Двухтактные блоки питания многих поколений PC строились на основе управляющей микросхемы TL494CN или ее аналогов. Эта микросхема содержит встроенный генератор и управляет ключами выходных транзисторов, воспринимая сигнал обратной связи из цепи +5 В и сигнал отключения по токовой перегрузке. Для определения перегрузки по току последовательно с первичной обмоткой силового трансформатора включают еще и трансформатор тока (на рис. 3.4 для упрощения не показан), с выхода которого сигнал через пороговую схему подается на вход управляющей микросхемы. Интересная особенность блоков питания, построенных на микросхеме TL494CN, заключается в идеологии управления выходными ключами. Вопреки ожиданиям, связанным с эксплуатацией импульсных блоков питания, например ЕС ЭВМ, эта микросхема управляет *запиранием* выходных ключей, а не активным отпиранием. Благодаря такому подходу упрощается процесс запуска источника (в тех же блоках ЕС для запуска применялся источник служебного напряжения). При включении блока питания PC симметричный мультивибратор, образованный выходными транзисторами совместно с трансформатором, начинает плавно возбуждаться. Когда выходное напряжение цепи + 12 В, от которого питается и управляющая микросхема, достигает уровня нескольких вольт, микросхема приступает к исполнению своих сдерживающих регулировочных обязанностей и блок выходит в рабочий режим, управляемый генератором микросхемы. Заметим, что некоторые блоки не запускаются без нагрузки.

Для мощных блоков питания обеспечить работу в широком диапазоне питающих напряжений довольно сложно, и на них устанавливают переключатель входного напряжения:

- 230 В — напряжение в диапазоне 180-265 В;
- 115 В — напряжение в диапазоне 90-135 В.

Переключение диапазона входного напряжения легко осуществляется переключателем, который преобразует мостовую схему выпрямителя в схему выпрямителя с удвоением для питания от сети 115 В. При включении в сеть 220 В блока, предназначенного для работы при напряжении 110 В, часто выходят из строя ключевые транзисторы или диоды. Блоки, у которых указано свойство Autoswitching Power Supply, работают в диапазоне 110-230 В без переключателя. В них применяют силовые компоненты с большим запасом по допустимым напряжению и току.

ВНИМАНИЕ

Встречаются и «упрощенные» блоки питания (китайского производства), у которых сетевой фильтр отсутствует (конденсаторов нет, а дроссели заменены перемычками). Эта экономия оборачивается большим уровнем помех, попадающих от данного блока в сеть, и повышенной чувствительностью компьютера к помехам из сети. Эти помехи могут приводить к сбоям, зависаниям или внезапным перезагрузкам компьютера и даже к самопроизвольному включению компьютеров с блоком питания ATX (см. далее).

Поскольку большинство цепей блока питания находится под высоким напряжением, ремонт блока требует соответствующей квалификации и знаний техники безопасности. Не вдаваясь в подробности, можно дать несколько практических рекомендаций по ремонту блока:

- Для проверки и ремонта блока питания полезно иметь нагрузку — мощные резисторы — по крайней мере, для основной цепи (+3,3 или +5 В). Резистор 5 Ом, 5 Вт обеспечит ток, вполне достаточный для проверки работоспособности. Использование в качестве нагрузки системной платы или накопителей чревато их выходом из строя в процессе ремонта блока.
- Если блок питания не включается, отключите его от сети и разрядите накопительные конденсаторы (C1 и C2 на рис. 3.4). После этого проверьте омметром диоды и транзисторы — чаще всего выходят из строя высоковольтные диоды и транзисторы. Заменять неисправные элементы желательно однотипными.
- После замены неисправных элементов не торопитесь подавать питание — какая-нибудь незамеченная «мелочь» может снова вывести из строя замененные детали. Не подключая сетевое напряжение, подайте от внешнего источника напряжение 10-12 В на шину +12 В. Если генератор управляющей микросхемы исправен, он «заведется», а по форме импульсов на базах выходных ключевых транзисторов можно судить об исправности большинства цепей формирования управляющих импульсов или о характере неисправности. Питание от сети на ремонтируемый блок следует подавать только после проверки его силовых цепей (диодов и транзисторов) и базовых цепей выходных ключей.

Блок питания PC обычно имеет стандартный конструктив и набор жгутов с разъемами питания системной платы и периферийных устройств. На задней стенке блока устанавливается входной разъем питающего кабеля, а также может присутствовать транзитный выходной разъем для питания монитора. Подключение монитора к этому разъему не только сокращает количество вилок, включаемых в розетку питания, но и обеспечивает связь «земель» монитора и системного блока. В ряде типов блоков питания транзитный разъем может и отсутствовать. При этом монитор включают в дополнительную розетку, и хорошо, если при этом соблюдают правила заземления. На задней стенке устанавливается также переключатель диапазона питающего напряжения, если таковой присутствует в блоке. Выключатель питания в старых конструктивах располагался на боковой или задней стенке блока питания. Позже его вынесли с блока

питания на лицевую панель корпуса и стали присоединять к блоку кабелем со съемными контактами. К этому кабелю, проходящему через весь системный блок, следует относиться со вниманием, поскольку он является источником и опасности, и помех. В конструктиве ATX главный выключатель питания вернулся на блок питания, а с передней панели блоком питания управляют с помощью кнопки и низковольтных цепей системной платы. Таким образом, провода с напряжением питающей сети удалось убрать из корпуса компьютера, и теперь высокое напряжение присутствует только внутри корпуса блока питания.

Мощность блока питания зависит от назначения корпуса системного блока и лежит в диапазоне от 150-450 Вт для обычных компьютеров до 350-750 Вт для мощных серверов. В настольных компьютерах основными потребителями мощности являются системная плата с процессором и памятью, а также графический акселератор. Чем выше тактовые частоты, тем «прожорливее» эти компоненты, и мощность блока питания выбирается именно под них. С учетом «аппетитов» процессоров 6-8-го поколений мощность 350 Вт не является излишней. У серверов значительное потребление может иметь подсистема хранения данных.

Вентилятор блока питается от цепи +12 В и обеспечивает охлаждение всего системного блока. В традиционных блоках питания вентилятор работает на отсос воздуха из корпуса системного блока. В современных качественных блоках питания устанавливают так называемое устройство Fan Processor, регулирующее скорость вращения вентилятора в зависимости от температуры. Это позволяет увеличить ресурс вентилятора и снижает шум при нормальной температуре окружающего воздуха.

Блок питания для корпусов AT

Блок питания AT вырабатывает основное стабилизированное напряжение +5 В при токе до 10-50 А; +12 В при токе 3,5-15 А для питания двигателей устройств и интерфейсных цепей; -12 В при токе 0,3-1 А для питания интерфейсных цепей; -5 В при токе 0,3-0,5 А (обычно не используется, присутствует только для соблюдения стандарта ISA Bus). Как говорилось ранее, уровни напряжений + 12 В, -12 В, -5 В обычно пропорциональны нагрузке цепи +5 В. Для регулировки выходного напряжения обычно имеется подстроечный резистор, хотя для доступа к нему может потребоваться разборка блока питания. Если старые системные платы хорошо себя чувствовали при номинале питания 5,0-5,1 В, то платы для процессоров 4-5-го поколений иногда лучше работают при напряжении питания 4,9-4,95 В.

Помимо питающих напряжений, блок вырабатывает сигнал P.G. (Power Good) — питание в норме. Этот сигнал с уровнем в 3-6 В появляется через 0,1-0,5 с после включения питания при нормальных выходных напряжениях блока. При отсутствии этого сигнала на системной плате непрерывно вырабатывается сигнал аппаратного сброса процессора, появление сигнала «выпускает» систему в нормальную работу. Этот сигнал должен сброситься раньше, чем пропадет напряжение +5 В при отключении блока. Отсутствие должной задержки сигнала при включении и запаздывание при выключении могут приводить к потере ин

формации в CMOS-памяти и ошибкам при загрузке по включении питания. Нажатие кнопки Reset по действию почти эквивалентно замыканию сигнала P.G. на «схемную землю».

Выходные цепи блоков питания выводятся гибкими жгутами проводов со стандартным набором разъемов (рис. 3.5). *Разъемы для питания накопителей* имеют ключи, исключающие возможность неправильного соединения. Однако иногда встречаются блоки с ошибочно собранными разъемами; в результате на шину питания +5 В попадает +12 В, чего устройства, как правило, не выдерживают. В практике автора такая ошибка привела к выходу из строя подряд двух дисководов формата 3" — ошибку в цепях питания стали искать лишь после поломки второго дисковода. Традиционные *разъемы питания системной платы* PS-8, PS-9 всегда устанавливаются рядом так, чтобы четыре черных провода GND шли подряд. Их ключи весьма условны, а ошибка подключения чревата выгоранием системной платы. Цвета проводов в жгутах стандартизованы:

- GND — черный;
- -12V — коричневый;
- +5V — красный;
- -5V — голубой;
- +12V — желтый;
- P.M. — белый (питание в норме).

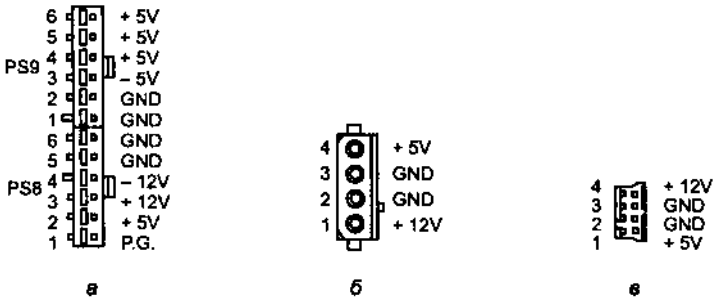


Рис. 3.5. Выходные разъемы блока питания AT: а — подключения системной платы, б — подключения периферийных устройств ATA и SCSI, *в* — подключения дисководов 3,5"

Блок питания ATX и ATX12V

Блок питания ATX предназначен для питания системных плат одноименных конструктивов. Он значительно отличается от блоков AT как по габаритным размерам, так и по электрическому интерфейсу.

В соответствии с тенденцией к снижению напряжения питания, в блоке ATX появился источник напряжения +3,3 В. Основным источником (по которому выполняется стабилизация) в первых версиях ATX был источник +5 В. В последующих версиях мощность источника +3,3 В была увеличена, и он стал основным. Для стабилизации напряжения +3,3 В предусмотрена подача внешнего

сигнала обратной связи (цепь +3,3VSense). Это позволяет скомпенсировать падение напряжения на проводах и контактах (обратная связь снимается с потребителя напряжения). В *спецификации ATX12V* увеличена мощность цепи +12 В, поскольку основное питание (в том числе и питание регулятора напряжения процессора) выгоднее брать от более высокого напряжения. Здесь появился дополнительный (независимый) выход +12V2, выведенный на отдельный разъем. Первоначально в блоке присутствовал источник -5 В, в ATX12V версии 2.01 (2004 г.) его изъяли.

Положительные напряжения поддерживаются с точностью ±5 %, отрицательные — с точностью ±10 %. Цепи +3,3V, +5V и +12V должны иметь защиту от превышения напряжения (4,2, 6,3 и 15,0 В соответственно): при превышении напряжения блок должен отключаться.

В интерфейс блока питания введен управляющий сигнал PS-ON#, включающий основные источники +5, +3,3, +12, -12 и-5 В (рис. 3.6). Эти источники вырабатывают напряжения только при удержании сигнала PS-ON# на низком логическом уровне. При высоком уровне или свободном состоянии цепи источники отключены. О нормальном напряжении питания сигнализирует сигнал PW-OK (Power O'Key), по действию аналогичный сигналу P.G. традиционных блоков. Интерфейс управления питанием позволяет чипсету системной платы выполнять программное отключение питания.

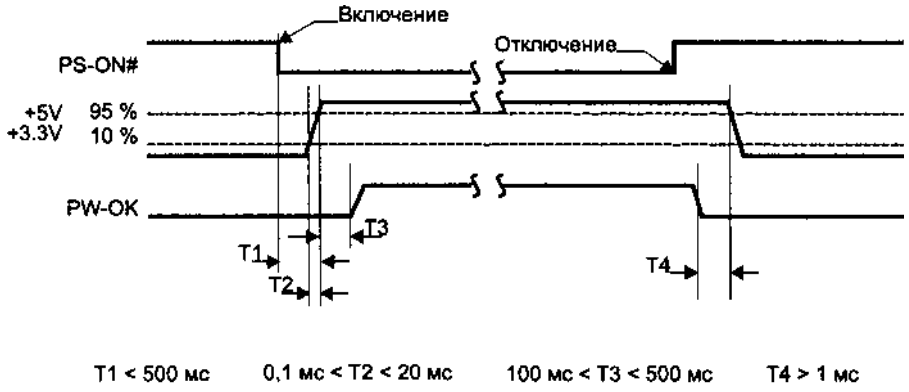


Рис. 3.6. Временная диаграмма интерфейса управления питанием ATX

Блок ATX имеет *дополнительный* «*дежурный*» (standby) маломощный источник с выходной цепью +5VSB, который включается сразу при подаче сетевого напряжения. Он предназначен для питания цепей управления энергопотреблением и устройств, активных и в спящем режиме (например, факс-модема, способного при поступлении входящего звонка «разбудить» машину). В первой версии спецификации ATX он был совсем маломощным (ток 10 мА); впоследствии допустимый ток был увеличен до 1 А, а в ATX12V — уже до 2,5 А, что позволяет расширить число функций, которые можно выполнять в дежурном режиме.

Все питающие и сигнальные провода от блока ATX к системной плате подключаются одним основным разъемом с надежным ключом. В первоначальной версии (рис. 3.7, *а)* использовался 20-контактный основной разъем, в ATX12V версии 2.0 его заменили 24-контактным (рис. 3.7, *б).* Ключевой является форма обрамления каждого контакта (квадратная или со скошенными углами), так что ни сместить, ни перевернуть разъем не удастся. Отдельный 4-контактный разъем для +12 В (рис. 3.7, *в)* появился в ATX12V. Дополнительный 6-контакт- ный разъем (рис. 3.7, *г)* не обязателен. Сигнал +3,3VSense первоначально планировалось подавать через дополнительный разъем (рис. 3.7, д), от которого отказались. Теперь для этого сигнала используется контакт 11 основного 20-кон- тактного разъема (в 24-контактном этот контакт имеет номер 13), кроме толстого провода питания к нему подходит дополнительный тонкий провод обратной связи. На традиционных разъемах подключения накопителей, естественно, сохранилось традиционное назначение контактов (рис. 3.7 е, *ж).* Для Serial ATA блок питания имеет специальные разъемы (рис. 3.7, з).

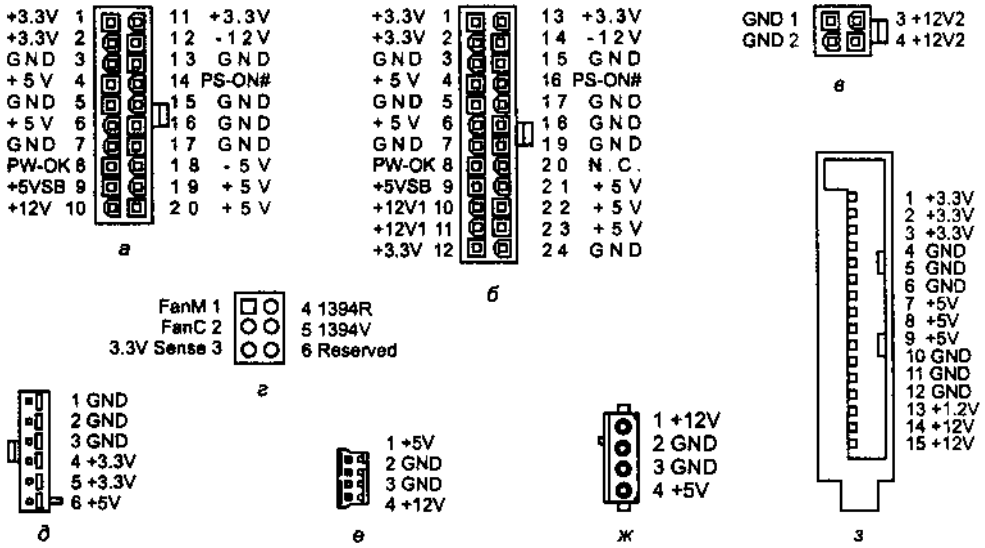
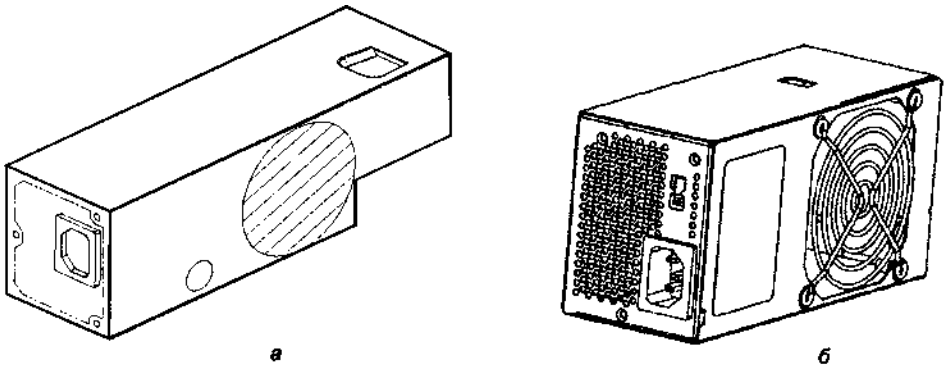


Рис. 3.7. Разъемы блока питания ATX: а, *б* — основные разъемы, *в* — разъем +12 В, г, — дополнительные разъемы, е - питание НГМД, ж — питание устройств ATA и SCSI, з — питание SerialATA

Блоки ATX выпускаются с различной номинальной мощностью. В табл. 3.1 приведены нагрузочные параметры для некоторых номиналов. Блоки выпускаются и в разном конструктивном исполнении, например CFX12V (Compact Form Factor), LFX12V (Lowprofile Form Factor), SFX (для microATX и Flex ATX), TFX12V (Thin Form Factor). Некоторые из них изображены на рис. 3.8.



а *б*

Рис. 3.8. Малогабаритные варианты блоков питания: а - LFX12V, *б* - TFX12V

Таблица 3.1. Токи нагрузки блоков питания ATX

Цепь	Блоки ATX			Блоки ATX12V		
	160 Вт	250 Вт	300 Вт	250 Вт	350 Вт	450 Вт
+3,3V	14	16	20	14	20	22
+5V	18	25	30	12	12	15
+5VSB	1,5	1,5	1,5	2,5	2,5	2,5
+12V(1)	6	10	12	8	10	14
+12V2	–	–	–	13	13	16
–12V	0,8	0,8	0,8	0,3	0,3	0,3
–5V	0,3	0,3	0,3	–	–	–

Расширенная спецификация для блока питания ATX предусматривает передачу информации от датчиков вентилятора на системную плату, что обеспечивает контроль скорости вращения и температуры воздуха. Для этих целей предназначен дополнительный (необязательный) жгут с разъемом, изображенный на рис. 3.7, *е*. На дополнительном разъеме также имеются контакты 1394V (+) и 1394R (-) изолированного от «схемной земли» источника напряжения 8-48 В для питания устройств шины IEEE-1394 (FireWire). В спецификациях ATX12V этот разъем не фигурирует.

Цепи блоков питания ATX имеют стандартизованную цветовую маркировку:

- COM - черный (соответствует цепи GND традиционных блоков);
- +5V — красный;
- +12V — желтый;
- -5V — белый;
- -12V — синий;
- +3.3 V — оранжевый;
- +3.3V Sense — коричневый;
- +5VSB — малиновый (purple);
- PS-ON — зеленый;

- PW-OK-серый.

Дополнительный разъем:

- +3.3V Sense — белый с коричневыми полосками;
- FanC — белый с синими полосками;
- FanM — белый;
- 1394V — белый с красными полосками;
- 1394R — белый с черными полосками.

Питание блокнотных ПК

Питание блокнотных ПК значительно отличается от питания настольных — у блокнотных ПК имеется встроенный аккумулятор, обеспечивающий автономную работу в течение нескольких часов. От аккумулятора (напряжение 12-20 В) питается импульсный преобразователь, поддерживающий необходимые уровни питающих напряжений. *Внешний блок питания* (адаптер питания) дает напряжение 12-20 В (зависит от аккумулятора ПК) при питании от сети переменного тока или бортовой сети автомобиля (12 В). В зависимости отмоде блок обеспечивает зарядку аккумулятора и работу ПК при наличии питающей сети. Заметим, что такая схема питания от сети (при исправном аккумуляторе) избавляет от необходимости использования источников бесперебойного питания в тех местах, где питающая сеть ненадежна. Возможна работа компьютера и без аккумулятора (но и без защиты от перебоев и провалов напряжения). Внешний блок обеспечивает гальваническую развязку компьютера от питающей сети.

Аккумуляторы для блокнотных и других малогабаритных ПК — довольно сложные и дорогостоящие компоненты: от них требуется большая емкость при ограниченных массе и габаритах. Аккумуляторы имеют внутренние средства контроля уровня заряда, что позволяет оптимизировать процесс их зарядки и прогнозировать момент отключения компьютера из-за разрядки батарей при отсутствии внешнего питания.

Для оптимизации работы блокнотного ПК на нем должна быть установлена программная *система управления потреблением* (например, PowerGear), отслеживающая состояние внешнего питания и уровень заряда аккумулятора. Эта система управляет потреблением основных подсистем ПК: процессора, памяти, графического адаптера, винчестера, дисплея, а также информирует пользователя о перспективах отключения. У этих подсистем есть возможность работать быстро *(с* максимальным потреблением) или экономно (с минимальной производительностью), возможны также промежуточные варианты. Политику управления питанием выбирает и настраивает пользователь.

3.3. Питание процессоров

Процессоры первых поколений (до 486) использовали напряжение питания 5 В (за исключением некоторых процессоров для портативных компьютеров). Развитие технологии привело к необходимости и возможности снижения напряже

ния питания до 3,3 В и ниже. Стандартный блок питания AT для процессора обеспечивает только питание 5 В, блок питания ATX выдает еще и 3,3 В. На системных платах для процессоров с пониженным напряжением питания стали устанавливать дополнительные регуляторы напряжения (Volt Regulation Module, VRM). В качестве регуляторов для сравнительно маломощных процессоров (4-5-го поколений) использовались микросхемы линейных стабилизаторов напряжения фиксированного или управляемого уровня. Эти стабилизаторы просто гасили избыточное напряжение, выделяя при этом избыточную мощность в виде тепла. Для питания относительно мощных процессоров микросхема устанавливается на радиаторе, на некоторых системных платах для процессоров 486 в качестве теплоотвода присутствует медная площадка под микросхемой на самой печатной плате. Для процессоров с раздельным питанием на плате должны стоять два регулятора. На плате ATX регулятор может быть один, поскольку для питания интерфейса процессора 3,3 В может использоваться непосредственно дополнительная шина источника +3,3 В. Для процессоров 6-8-го поколений линейный стабилизатор был бы слишком неэкономичным — он бы выделял мощность, соизмеримую с потреблением процессора, и требовал бы такого же мощного охлаждения. Низкое напряжение для таких процессоров получают с помощью импульсных преобразователей из более высокого. Сначала его получали из основного напряжения +5 В, в системных платах с блоками питания ATX12V его получают из +12 В (это позволяет уменьшить токи ключевого транзистора и потери мощности).

Напряжение управляемых регуляторов для процессоров 4-5-го поколений задается джамперами. Иногда их делают красного цвета, указывая на то, что они отвечают за жизненно важный параметр — напряжение, подаваемое на процессор. Для процессоров 6-8-го поколений предусмотрена возможность автоматической установки питающего напряжения в зависимости от имеющегося процессора. Здесь напряжение регулятора задается несколькими сигналами VIDx, входящими в интерфейс процессора, — некоторые из них заземлены в процессоре, чем и выставляется код требуемого напряжения. Таким образом, с пользователя снимается забота об установке напряжения, но вместе с тем он лишается возможности им управлять, например, для разгона. Ряд системных плат позволяют выбирать между автоматической и ручной установкой напряжения.

Установленное значение питающего напряжения должно соответствовать номиналу процессора. Слишком низкое напряжение приводит к неустойчивой работе, слишком высокое может вывести процессор из строя. Системные платы выпускают с регуляторами, поддерживающими номиналы питания для процессоров, «модных» в текущее время. Чем новее процессоры, тем более низкое питающее напряжение, как правило, им требуется, и невозможность его обеспечения регулятором может стать одной из причин отказа от применения нового процессора в относительно старой системной плате. Эта проблема может решаться путем установки процессоров в специальные переходники, содержащие регулятор напряжения (например, «слот-сокет», или даже «сокет-сокет»). Недостаточная мощность регулятора может вызвать проблемы у особо «прожорливых» процессоров. Сведения о номиналах питающих напряжений для различных процессоров приводятся в соответствующих разделах главы 6.

3.4. Охлаждение компонентов системного блока

Системный блок современного компьютера содержит ряд компонентов-«печек», выдающихся по энергопотреблению, а следовательно, и тепловыделению (в итоге вся потребляемая мощность выделяется в виде тепла, больше никакой «полезной» работы компьютер не производит). К таким компонентам относятся процессор, память повышенного быстродействия, графический акселератор, винчестер с высокой скоростью вращения, магнитооптические накопители, устройства записи оптических дисков, мощные дисковые контроллеры. Возникает задача отвода тепла от самих элементов и вывода тепла из корпуса системного блока. В компьютерах на процессорах первых трех поколений с этой задачей легко справлялся единственный вентилятор блока питания, высасывающий воздух из корпуса системного блока. Поскольку применялись микросхемы со сравнительно невысокой (по нынешним меркам) степенью интеграции, каждая из них выделяла умеренную мощность, а благодаря большим габаритам корпусов микросхем тепло с них отводилось в воздух естественной конвекцией. С повышением степени интеграции и уменьшением размеров корпусов микросхем, а также с повышением рабочих частот вопросы отвода тепла от корпусов микросхем и из самого системного блока ставятся все острее, несмотря на успехи технологий, позволивших во много раз снизить мощности, рассеиваемые отдельными элементами. В первую очередь проблемы охлаждения актуальны для процессоров, поэтому на этом вопросе остановимся подробнее.

Вопрос *охлаждения процессора* стал актуальным, начиная с моделей класса 486. Процессор 486SX-33 еще не требовал установки специального радиатора. Однако с повышением тактовой частоты мощность, рассеиваемая процессором, возрастает. Кроме того, потребляемая мощность зависит от интенсивности работы процессора: разные инструкции задействуют различные части процессора, и при увеличении доли «энергоемких» инструкций мощность, рассеиваемая процессором, повышается. Существуют даже специальные тестовые программы для проверки теплового режима, способные перегреть процессор с недостаточным охлаждением и довести его до сбоев и даже разрушения.

Для охлаждения процессоров применяют *радиаторы* (heat sink). Радиатор эффективно работает, только если обеспечивается его плотное прилегание к верхней стороне корпуса процессора (даже тонкий воздушный зазор значительно снижает теплопроводность). Весьма эффективно использование теплопроводной мастики, которую наносят тонким слоем на корпус процессора, после чего радиатор «притирают» к процессору. Хорошие результаты дает и приклеивание радиатора к процессору двусторонней «самоклейкой» — но только специально предназначенной для этих целей, поскольку обычные «липучки» термостойки и имеют большое тепловое сопротивление. Когда пассивного теплоотвода, обеспечиваемого радиатором, рассчитанным на естественную циркуляцию воздуха внутри корпуса компьютера, оказывается недостаточно, применяют *активные теплоотводы* (Cooler, Fan). Они имеют *вентиляторы*, устанавливаемые на радиатор процессора или на сам процессор. Вентиляторы обычно являются

съемными устройствами, питающимися от источника +12 В через специальный переходной разъем. Размеры (габаритные и установочные) вентиляторов и радиаторов для процессоров 486, Pentium (они разные для процессоров 60-66, 75-180 и 200-233 МГц), Pentium Pro, Pentium II/III, Celeron различаются — чем новее процессор, тем больше радиатор и вентилятор. Процессор Pentium 4 только подтверждает это правило.

Для особо горячих процессоров (в основном для их разгона) применяют и полупроводниковые холодильники на модулях, использующих эффект Пельтье. *Холодильник Пельтье* работает тепловым насосом: он отбирает тепло с одной стороны модуля и выделяет его на другой стороне, обеспечивая разность температур до нескольких десятков градусов. При этом он и сам потребляет значительную мощность, соизмеримую с потребляемой мощностью охлаждаемого элемента (то есть десятки ватт), и выделяет ее в виде тепла. Таким образом, вентилятор, обдувающий радиатор холодильника Пельтье, должен выносить из корпуса компьютера значительно больше тепла, чем выделяет сам процессор. Это является расплатой за возможность охлаждения отдельных элементов до температуры, меньшей, чем температура окружающего воздуха. Здесь имеются и побочные эффекты — на холодной части может конденсироваться влага, что чревато утечками тока (замыканием проводников). Холодильник питается либо от общего блока питания компьютера (по линии +5 В), либо от отдельного источника питания. Холодильник может быть управляемым и неуправляемым: в управляемом холодильнике имеется термодатчик, который включает холодильник лишь при определенном пороге температуры охлаждающей стороны; неуправляемый холодильник может заморозить процессор (до зависания) при переходе процессора в энергосберегающий режим. Отключенный (вышедший из строя) холодильник представляет собой теплоизолятор, под которым процессор, работающий на полной мощности, может сгореть. Цена холодильника зависит от его мощности и составляет несколько десятков долларов.

Применяют и *системы водяного охлаждения*: один радиатор с каналами для циркуляции воды устанавливается на процессор. Он парой гибких трубок соединяется с другим, более крупным радиатором, который может быть вынесен из корпуса компьютера. В комплект входит насос, обеспечивающий циркуляцию воды, и большой вентилятор, обдувающий выносной радиатор. Цена такого комплекта соизмерима с ценой процессора.

Стандарт конструктива ATX предусматривает установку процессора прямо под блоком питания, при этом для обдува радиатора могут использоваться: внутренний вентилятор блока питания, дополнительный внешний вентилятор, устанавливаемый снаружи блока питания, вентилятор процессора. Теоретически все они должны работать согласованно — на обдув воздухом радиатора процессора. В противном случае их суммарная эффективность падает. При наличии большого радиатора в корпусе ATX можно обойтись и без отдельного вентилятора на процессоре.

Процессоры P6 имеют внутренний датчик температуры, аварийно останавливающий процессор в случае перегрева. Для измерения температуры процессоры P6 имеют термодиод, его анод и катод выведены на контакты процессора.

В процессоре Xeon к термодиоду подключен встроенный электронный термометр, который при перегреве вырабатывает сигнал, используемый для генерации прерывания.
Вентиляторы современных процессоров могут иметь датчик вращения, вырабатывающий пару импульсов за один оборот. Сигнал датчика выведен на разъем питания вентилятора, обработка сигнала возлагается на компоненты системной платы. Системная плата со встроенными средствами мониторинга позволяет программно измерять температуру процессора (по термодиоду), частоту вращения вентиляторов, а в критической ситуации вырабатывать прерывание для оповещения ОС и пользователя.
Большинство современных процессоров допускают температуру до +85 °С (Pentium — до +70 °С). Температура измеряется в центре верхней стороны корпуса процессора (не радиатора!) в установившемся рабочем режиме. Процессоры для мобильных применений обычно имеют меньшую потребляемую мощность и более высокую допустимую температуру корпуса. Существуют специальные исполнения процессоров, допускающие расширенный температурный диапазон. Они, естественно, дороже обычных и в PC применяются довольно редко.
Остальные компоненты, требующие отвода тепла, охлаждаются аналогично процессорам — радиаторами, вентиляторами, а то и холодильниками Пельтье. Общие соображения по тепловому режиму системного блока довольно просты:

- С помощью просторного системного блока проще обеспечить нормальный режим охлаждения всех компонентов.
- На пути воздушных потоков не должно быть препятствий в виде непроходимых «джунглей» проводов и шлейфов. Вентиляционные отверстия в корпусе не должны быть перекрыты.
- Два и более вентиляторов, гонящих воздух по одному пути, должны работать согласованно (не гнать воздух навстречу друг другу).
- Сильно нагревающиеся компоненты следует по возможности отдалять от других, особенно от чувствительных к нагреву.
- Периодически следует чистить компьютер — пыль, оседающая на компонентах (в том числе радиаторах), препятствует их охлаждению. Нельзя допускать попадания посторонних предметов (обрывков бумаги, а также проводов и шлейфов) в лопасти вентиляторов.

Системная плата может иметь входы для подключения датчиков температуры, датчик на гибком кабеле должен входить в комплект поставки платы. Установив датчик на критичном устройстве (винчестере, графической карте), можно наблюдать за его температурой с помощью утилиты CMOS Setup или специальной загружаемой утилиты. Если позволяет ПО, то можно настроить и порог предупреждения о критической температуре.
Вентилятор как электромеханическое устройство имеет принципиально меньшую надежность (срок жизни), чем процессор и другие электронные компоненты. С вентиляторами могут быть связаны неприятности разной степени тяжести —

от повышенного шума при работе до отказа (остановки). От повышенного шума помогает периодическая смазка оси вентилятора. Для смазки вентилятор приходится снимать с радиатора (или корпуса блока питания) и открывать место смазки подшипника, закрытое наклейкой-шильдиком. Смазывать подшипник можно обычным машинным маслом (жидким). Для чистки вала и подшипников приходится еще и снимать ротор, для чего необходимо снять фиксирующую шайбу, находящуюся под той же наклейкой. Снизить шум от вибрации вентилятора можно смягчением его крепления — установкой демпфирующих шайб и другими «домашними» методами. Вентилятор на пониженных оборотах шумит меньше (но и дует слабее) — на этом основано «интеллектуальное» управление (fan processing), реализуемое довольно простыми средствами. Частой причиной остановки вентилятора является касание лопастями вентилятора внутренних соединительных проводов (интерфейсных шлейфов дисков и кабелей для подключения кнопок и индикаторов лицевой панели). Поэтому рекомендуется после сборки компьютера подвязывать провода к шасси корпуса — целее будут и вентилятор, и провода. Существуют вентиляторы с сигнализацией о неисправности: они имеют датчик вращения и простенькую вмонтированную плату электроники. Эта плата включается между разъемом стандартного динамика PC и самим динамиком. При остановке вентилятора динамик начинает пищать. Признаком наличия такого устройства является характерная «мелодия», звучащая при включении питания (ее невозможно спутать с однотональными «писками» диагностического теста POST).

Современные вентиляторы, используемые для охлаждения блоков питания, процессоров и других компонентов, способны работать в системах автоматического управления. Для этого они снабжаются тахометрическими датчиками (для обратной связи) и управляющим входом. Сигналы управления и обратной связи выводятся на стандартные разъемы вентиляторов, которые могут быть трех- или четырехконтактными (рис. 3.9).

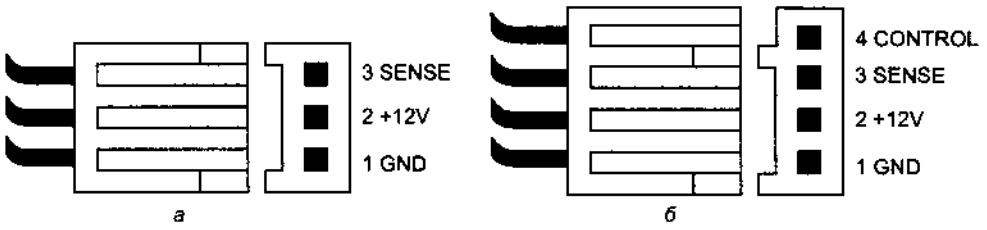


Рис. 3.9. Разъемы вентиляторов: а — неуправляемого, *б* — *с* ШИМ-управлением

Сигнал Sense — выход типа «открытый коллектор» от тахометрического датчика вентилятора, вырабатывающего два импульса на каждый оборот ротора. Этот сигнал на системной плате должен быть «подтянут» к цепи +12 В. С помощью данного сигнала можно определять остановку вентилятора, а также измерять скорость вращения.

Сигнал Control — входной с уровнем ТТЛ, на который подаются импульсы с частотой 25 кГц (допустимо 21-28 кГц). Скорость вращения вентилятора опре

деляется относительной длительностью импульса, которая может составлять от 20 до 100 % периода. При минимальной длительности (20 %) скорость вентилятора не должна превышать 30 % полной скорости, которая достигается при длительности в 100 %. При малой длительности поведение вентилятора может быть различным: он может сохранять минимальную скорость при уменьшении длительности вплоть до нулевой, а может и останавливаться при каком-то пороговом значении.

В интерфейсе блоков питания ATX первых версий фигурирует дополнительный разъем (в последующих версиях его изъяли), на котором присутствуют сигналы вентилятора:

- FanM — выход типа «открытый коллектор» от тахометрического датчика вентилятора блока питания, вырабатывающего два импульса на каждый оборот ротора;
- FanC — вход для управления скоростью вентилятора путем подачи *напряжения* в диапазоне от 0 до +12 В при токе до 20 мА. Если уровень напряжения выше +10,5 В, вентилятор работает на максимальной скорости. Уровень ниже +1 В означает запрос от системной платы на остановку вентилятора. Промежуточные значения уровня позволяют плавно регулировать скорость. Внутри блока питания сигнал FanC подтягивается к уровню +12 В, так что если дополнительный разъем оставить неподключенным, вентилятор будет всегда работать на максимальной скорости.

3.5. Общие вопросы электропитания и заземления

Рассмотрим правила подключения к питающей сети с точки зрения безопасности человека и компьютера.

Практически каждый блок питания компьютера или периферийного устройства имеет сетевой фильтр (рис. 3.10). Конденсаторы этого фильтра предназначены для шунтирования высокочастотных помех питающей сети на землю через провод защитного заземления и соответствующие трехполюсные вилку и розетку. «Земляной» провод соединяют с контуром заземления, но допустимо его соединять и с «нулем» силовой сети (разница ощущается только в особо тяжелых условиях эксплуатации). При занулении необходимо удостовериться в том, что «нуль» не станет фазой, если кто-нибудь вдруг перевернет вилку питания. Если же «земляной» провод устройства никуда не подключать, на корпусе устройства появляется напряжение порядка 110 В переменного тока (рис. 3.11): конденсаторы фильтра работают как емкостной делитель напряжения, и поскольку их емкость одинакова, 220 В делится пополам.

Конечно, мощность этого «источника» ограничена — ток короткого замыкания $I_{К.З.}$ на «землю» составляет доли миллиампера, причем чем мощнее блок питания, тем больше емкость конденсаторов фильтра. При емкости конденсатора C = 0,01 мкФ этот ток будет около 0,7 мА. Заметим, что здесь мы учитываем лишь частоту питающей сети. Для высокочастотных (импульсных) помех, при

ходящих как по сети, так и от входного преобразователя блока питания, те же конденсаторы дают во много раз меньшее сопротивление, и ток короткого замыкания может многократно возрастать.

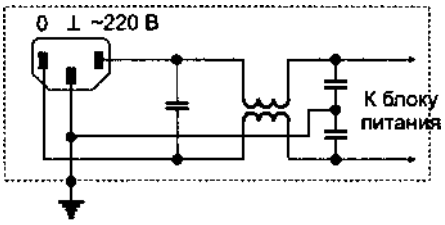


Рис. 3.10. Входные цепи блока питания

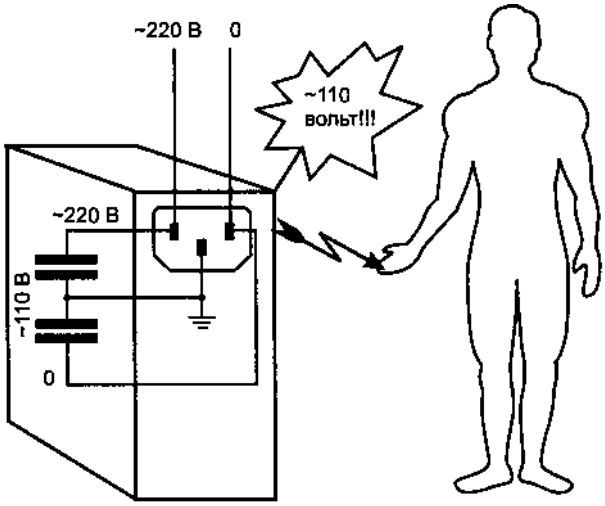


Рис. 3.11. Образование потенциала на корпусе компьютера

Такие напряжение и ток опасны для человека. Попасть под напряжение можно, прикоснувшись одновременно к неокрашенным металлическим частям корпуса компьютера и, например, к батарее отопления. Это напряжение является одним из источников разности потенциалов между устройствами, от которой страдают интерфейсные схемы.

Посмотрим, что происходит при соединении двух устройств (компьютера и принтера) интерфейсным кабелем. Общий провод интерфейсов последовательных и параллельных портов связан со «схемной землей» и корпусом устройства. Если соединяемые устройства надежно заземлены (занулены) через отдельный провод на общий контур, проблемы разности потенциалов не возникает. Если же в качестве заземляющего провода использовать нулевой провод питания при разводке питающей сети с трехполюсными розетками двухпроводным кабелем, на нем будет набегать разность потенциалов, вызванная падением напряжения от протекающего силового тока. Если в эти же розетки включать устрой

ства с большим энергопотреблением, разность потенциалов и импульсные помехи при включении-выключении оказываются ощутимыми. Поскольку обычно сопротивление интерфейсного кабеля больше питающего, через общий провод интерфейса потечет ток, существенно меньший, чем силовой. Но при нарушении контакта в нулевом проводе питания через интерфейсный провод может протекать и весь ток, потребляемый устройством. Он может достигать нескольких ампер, что влечет за собой выход устройств из строя. Невыровненные потенциалы корпусов устройств являются также источником помех в интерфейсах.

Если оба соединяемых устройства не заземлены, в случае их питания от одной фазы сети разность потенциалов между ними оказывается небольшой (вызванной разбросом емкостей конденсаторов в разных фильтрах). Если незаземленные устройства подключены к разным фазам, разность потенциалов между их несоединенными корпусами будет порядка 190 В, при этом уравнивающий ток через интерфейс может достигать десятка миллиампер. Когда все соединения/разъединения выполняются при отключенном питании, для интерфейсных схем такая ситуация почти безопасна. Но в случае коммутации при включенном питании возможны неприятности: если контакты общего провода интерфейса соединяются позже (разъединяются раньше) сигнальных, разность потенциалов между «схемными землями» прикладывается к сигнальным цепям и они выгорают. Самый тяжелый случай — соединение заземленного устройства с незаземленным, особенно когда у последнего мощный блок питания.

Для устройств, блоки питания которых имеют шнуры с двухполюсной вилкой, эти проблемы тоже актуальны. Такие блоки питания зачастую имеют сетевой фильтр, но с конденсаторами малой емкости (ток короткого замыкания довольно мал).

Весьма коварны сетевые шнуры компьютеров с двухполюсной вилкой, которыми подключаются блоки питания с трехполюсным разъемом. Пользователи, подключающие свои компьютеры в бытовые розетки, могут столкнуться с проблемами из-за отсутствия заземления.

Локально проблемы заземления решает применение сетевых фильтров типа «Pilot» и им подобных. Питание от одного фильтра всех устройств, соединяемых интерфейсами, решает проблему разности потенциалов. Еще лучше, когда этот фильтр включен в трехполюсную розетку с заземлением (занулением). Однако заземляющие контакты (обжимающие «усики») многих розеток могут неплотно соприкасаться с вилкой вследствие своей слабой упругости или заусениц в пластмассовом кожухе. Кроме того, эти контакты не любят частых вынимания и вставки вилок, так что обесточивание оборудования по окончании работы лучше выполнять выключателем питания фильтра (предварительно выключив устройства).

ВНИМАНИЕ ----------

Настоятельно рекомендуется отключать питание при подключении и отключении интерфейсных кабелей. Небольшая разность потенциалов, которая практически исчезнет при соединении устройств общими проводами интерфейсов, может пробить входные и выходные цепи сигнальных линий, если в момент присоединения разъема контакты общего провода соединятся позже сигнальных. От такой последовательности обычные разъемы не страхуют.

К помехам, вызванным разностью потенциалов «схемных земель» (корпусов) устройств, наиболее чувствительны параллельные порты. У последовательных портов зона нечувствительности шире (пороги ±3 В); еще меньшую чувствительность имеют интерфейсы локальных сетей, где обычно присутствует гальваническая развязка сигнальных цепей от схемной земли с допустимым напряжением изоляции порядка 1000 В.

Правила заземления в документации к импортной аппаратуре приводятся не всегда, поскольку подразумевается, что трехполюсная вилка *всегда* должна включаться в соответствующую розетку с заземлением, а не в двухполюсную с рассверленными отверстиями. В нашей стране распространены так называемые «евророзетки» (трехполюсные). Для заземления, как правило, используются контакты-усики, а не центральный заземляющий штырь.

Проблемы разводки электропитания и заземления стоят особенно остро в локальных сетях, поскольку здесь, как правило, имеется большое количество устройств (компьютеров и коммуникационного оборудования), соединенных между собой интерфейсными кабелями и значительно разнесенных в пространстве (локальная сеть может охватывать и многоэтажное здание). Подробнее о решении проблем питания и заземления в сетях см. в [4]. Там же приведены и более подробные пояснения и оценки разностей потенциалов и уравнивающих токов.

3.6. Средства улучшения качества электропитания

Электронное оборудование, питающееся от сети переменного тока, подвергается различным негативным воздействиям со стороны этой питающей сети. Стандартным требованием к питающей сети является напряжение питания 220 В с допустимыми отклонениями от -15 до +10 % от номинала (187-242 В) при частоте 50±1 Гц. Возмущения со стороны сети могут приводить к сбоям (импульсным помехам и провалам питающего напряжения), самопроизвольному отключению или перезапуску устройств и даже к выходу их из строя под воздействием импульсных напряжений или длительных перенапряжений. Поскольку большинство блоков питания имеют импульсный преобразователь с бестрансформаторным входом, к отклонениям частоты или формы напряжения они обычно почти нечувствительны. Однако последствия сбоев питания могут быть весьма тяжелыми, вплоть до потери данных на диске мощного и ответственного сервера (не считая выхода из строя аппаратуры).

Для защиты от воздействий сетевых возмущений применяется целый комплекс мер:

- *Сетевой LC-фильтр* задерживает высокочастотные помехи из сети и в сеть от импульсных блоков питания. Этот фильтр входит в состав практически любого блока питания, а также в сетевые колодки питания типа «Pilot» и им подобные.
- *Ограничитель перенапряжений* (surge protector) подавляет высоковольтные выбросы, как относительно длинные коммутационные (до 10 мс), возникаю

щие при переключениях мощных цепей, так и короткие — грозовые. Энергия импульсов перенапряжений поглощается полупроводниковым *варистором.* При хорошем подборе параметров варистор может спасать также от длительных (и значительных) повышений напряжения сети, например, из-за перекоса фаз. В этом случае варистор ограничивает напряжение, выделяя значительную мощность, что приводит к его пробою на короткое замыкание и отключению питания предохранителями токовой защиты (если они есть и рассчитаны на соответствующий ток).

На рис. 3.12 приведена схема фильтра, комбинированная с ограничителем перенапряжений.

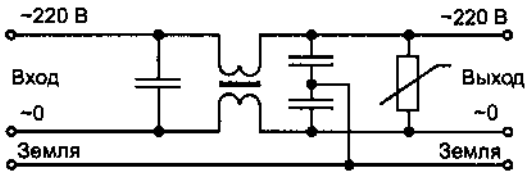


Рис. 3.12. Фильтр-ограничитель с варистором

От внезапного пропадания напряжения сети предохраняют *источники бесперебойного питания — ИБП* (Uninterruptible Power System, UPS). В их состав обязательно входят аккумуляторные батареи, выпрямитель входного напряжения и инвертор, обеспечивающий нагрузку напряжением переменного тока.

Источники бесперебойного питания различают по классам (режимам работы). Существуют блоки Off-Line (Stand-By), Line-Interactive и On-Line; их «полезность» (и цена) растут в порядке этого перечисления.

От класса, мощности устройства и емкости батарей, определяющей время автономной работы при максимальной нагрузке, существенно зависит цена ИБП.

При пропадании сетевого напряжения ИБП переключается на резервное питание и обычно подает звуковой сигнал. Для защиты данных компьютера устройство ИБП должно иметь возможность передать сигнал о грядущем отключении питания. Сигнал может подаваться аппаратным прерыванием через специальную плату сопряжения с PC или разъем PS/2 Mouse (как варианты у Smart UPS), через COM-порт или встроенный в ИБП адаптер ЛВС. Два последних варианта более универсальны и обеспечивают двунаправленный обмен развернутой управляющей и диагностической информацией. При восстановлении питания происходит обратное переключение, и батареи подзаряжаются. Если питание не восстановилось за время работы батарей, ИБП отключается, а его повторное включение после подачи напряжения может быть ручным или автоматическим.

Современные модели ИБП имеют в своем составе микроконтроллер, который в совокупности со специализированным ПО серверов и станций, поставляемым для конкретных моделей, может предоставлять широкий спектр услуг в зависимости от интерфейса связи ИБП с системой:

- *Телеметрия.* Информация о состоянии питающей сети, батареи и других узлов, температуре внутри ИБП, величине нагрузки и т. д. передается в систему сбора, обработки и отображения информации. Система может прогнозировать время работы от батарей и соответственно корректировать задержку закрытия сервера.
- *Телеуправление.* Двунаправленный интерфейс с ИБП обеспечивает подачу управляющих команд — отключение, запуск диагностических тестов и т. д.
- *Планирование включения и выключения.* Администратор может задать график работы сервера, указывая время включения и отключения питания на каждый день недели. Программа при наступлении времени отключения посылает предупреждение всем клиентам, через некоторое время инициирует закрытие сервера и программирует ИБП на отключение питания через определенный интервал времени и повторное включение в заданное время. После отключения по команде ИБП переходит в режим ожидания и своим внутренним таймером отсчитывает время до включения. В заданное время ИБП включает питание нагрузки, сервер автоматически загружается, и следующее запланированное отключение произойдет по инициативе программы, работающей на сервере.

Возможности взаимодействия по сети оператора с ИБП определяются ПО этого устройства. Популярные пакеты PowerChute («парашют») для Smart UPS фирмы APC, OnliNet Basic для ИБП фирмы EXIDE обеспечивают вышеперечисленные функции для различных ОС, они вполне удовлетворительны для систем с одним устройством ИБП. В системах с более сложным питанием желательно использовать сетевые варианты ПО, предоставляющие централизованное управление сетями ИБП. Для ИБП фирмы EXIDE это OnliNet Network, OnliNet NVX и др.

Простейшая программная поддержка ИБП должна обеспечивать оповещение о пропадании сетевого напряжения и принудительное завершение работы приложений и операционной системы, когда остается небольшой ресурс времени автономной работы (от аккумулятора). Сигнал о пропадании сетевого напряжения от ИБП к защищаемому компьютеру должен подаваться обязательно, он инициирует оповещение. Принудительное завершение может выполняться по дополнительному сигналу ИБП, когда устройство «чувствует», что батарей хватит только на определенное время. Возможна настройка ПО и на работу только от одного сигнала — принудительное завершение инициируется, если сигнал пропадания напряжения удерживается дольше заданного времени. Штатная служба UPS в Windows NT/9x позволяет использовать для сигнализации управляющие сигналы COM-порта: линия CTS — для сигнализации о пропаже питающего напряжения (power fail), DCD — для сигнализации о малом ресурсе батарей (battery low). Интерфейс настройки сервиса позволяет выбрать полярность сигналов, а также использовать только первый сигнал и инициировать завершение по тайм-ауту. Некоторые модели ИБП указанные сигналы в двухполярном представлении, воспринимаемом COM-портом, не генерируют, а имеют интерфейс «сухой контакт». Событие отражается замыканием или размыканием этого контакта, гальванически не связанного ни с какими цепями. В этом

случае можно воспользоваться переходником, питающимся от выходных линий интерфейса RS-232C (рис. 3.13).

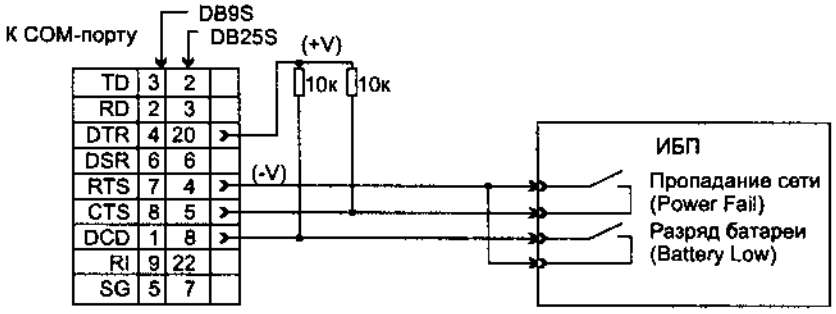


Рис. 3.13. Преобразование «сухого контакта» в сигналы RS-232C

Более точное представление о состоянии ИБП, а также планирование включения-выключения питания возможны только при полноценной двусторонней связи ИБП со специальным модулем ПО, функционирующим на защищаемом компьютере. Наиболее широко распространенный вариант связи — через COM- порт. Многие модели ИБП имеют разъем DB9, который обычно и используется интерфейсом RS-232. Однако зачастую назначение его контактов в значительной степени отличается от стандартного. На рис. 3.14 показаны схемы кабелей подключения UPS Fiskars и Smart UPS фирмы APC к COM-порту.

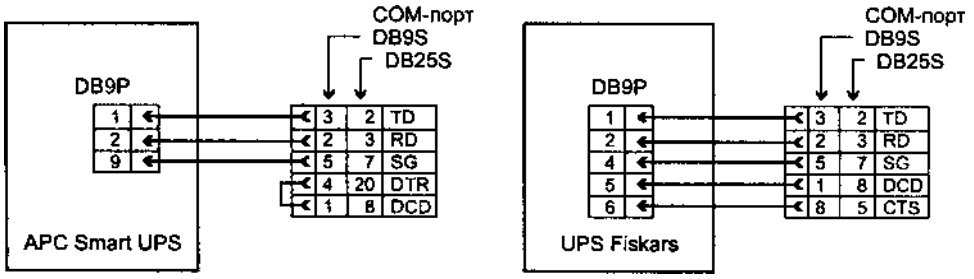


Рис. 3.14. Кабели подключения ИБП к COM-порту

ГЛАВА 4

Архитектура IBM PC-совместимого компьютера

Архитектурный облик PC-совместимого компьютера определяется рядом свойств, обеспечивающих возможность функционирования программного обеспечения, управляющего периферийным оборудованием. Программы могут взаимодействовать с устройствами разными способами:

- через вызовы функций операционной системы (прерывания DOS, API Windows и т. п.);
- через вызовы функций базовой системы ввода-вывода (BIOS);
- непосредственно взаимодействуя с известным им «железом» — портами и памятью устройств или контроллеров интерфейсов.

Такой «толстый пирог» из слоев совместимости существует благодаря изначальной открытости архитектуры первых IBM PC и сохранения имеющихся решений (пускай иногда и не самых лучших) в последующих моделях, обрастающих новыми узлами.

Облик PC-совместимого компьютера в значительной степени определяется разработчиками Microsoft и Intel. Для этих фирм уже стало традицией выпускать объемистый документ, диктующий разработчикам аппаратуры требования для получения вожделенного логотипа «Designed for Microsoft Windows». В спецификациях определяются требования к функциональности и производительности всех подсистем компьютера, включая периферийные устройства. Отдельные положения этих спецификаций упоминаются в разделах, посвященных конкретным подсистемам ПК.

4.1. Структурная схема

Структурная схема современного IBM PC-совместимого компьютера приведена на рис. 4.1. Ядром компьютера являются процессор (один или несколько), ОЗУ, ПЗУ с BIOS и интерфейсные средства, связывающие их между собой и с остальными компонентами. Эти средства на рисунке изображены в виде «обла

ка», поскольку их формы разнообразны (шины, хабы). Это «облако» обычно имеет интерфейсы одной или нескольких шин расширения (ISA, PCI/PCI-X, PCI-E), а также порта AGP (уже вытесняемого PCI-E). Стандартная архитектура PC определяет набор обязательных средств ввода-вывода и средств поддержки периферии, включая систему аппаратных прерываний (i8259A), систему прямого доступа к памяти (i8237A), трехканальный счетчик (i8254), интерфейс клавиатуры и управления (i8042), канал управления звуком, память и часы CMOS. На рисунке изображены лишь логические связи между этими устройствами; подразумевается, что с помощью средств того же «облака» они представлены своими стандартизованными регистрами в общедоступном пространстве ввода-вывода. Также подразумевается, что все компоненты получают требуемое питание, что превращает весь этот набор компонентов в работоспособный компьютер. Конечно же, он должен быть дополнен периферией: дисплеем со своим адаптером, подключаемым к порту AGP, шине расширения или прямо в «облако», контроллерами шин периферийных устройств (ATA, SATA, SCSI, SAS, USB, FireWire), интерфейсов портов (COM, LPT, GAME...), дисководов, аудиосредств и пр. «Облако» вместе со средствами ввода-вывода и поддержки периферии реализуется чипсетом системной платы (см. 6.1), который обычно включает в себя и перечисленные выше интерфейсы.

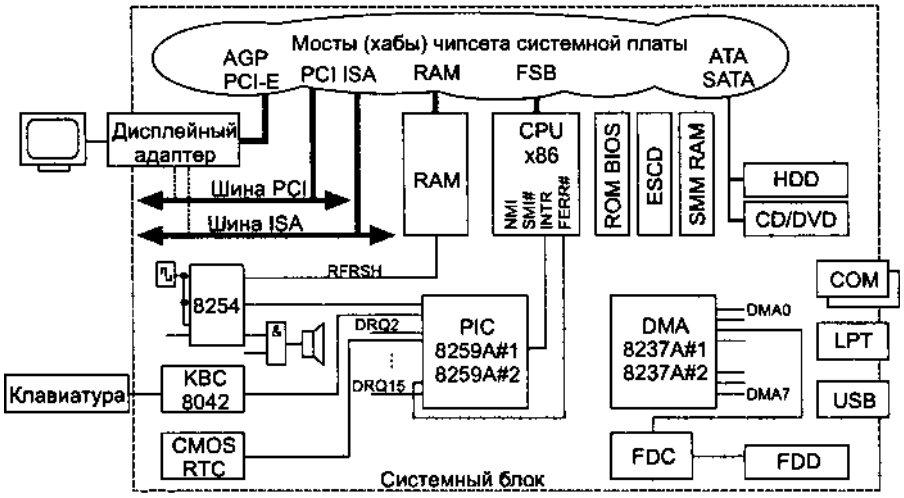


Рис. 4.1. Структурная схема компьютера

Любой PC-совместимый компьютер имеет следующие характерные черты:

- процессор, программно совместимый с семейством x86 фирмы Intel;
- специфическую систему распределения пространства адресов памяти;
- традиционное распределение адресов пространства ввода-вывода с фиксированным положением обязательных портов и совместимостью их программной модели;

- систему аппаратных прерываний, позволяющую периферийным устройствам сигнализировать процессору о необходимости исполнения некоторых обслуживающих процедур;
- систему прямого доступа к памяти, позволяющую периферийным устройствам обмениваться массивами данных с оперативной памятью, не отвлекая на это процессор;
- набор системных (стандартных) устройств и интерфейсов ввода-вывода;
- унифицированные по конструктиву и интерфейсу шины расширения (ISA, EISA, MCA, VLB, PCI/PCI-X, PCI-E, PC Card, Card Bus), состав которых может варьироваться в зависимости от назначения и модели компьютера;
- базовую систему ввода-вывода (BIOS), выполняющую начальное тестирование и загрузку операционной системы, а также имеющую набор функций, обслуживающих системные устройства ввода-вывода.

4.2. Распределение пространства памяти

Пространство памяти в PC-совместимых компьютерах используется для размещения собственно памяти (ОЗУ, ПЗУ), а также регистров (и областей локальной памяти) периферийных устройств. Распределение пространства памяти PC обусловлено особенностями системы адресации процессоров семейства x86 и требованиями обратной совместимости PC с ПО всех предшествующих поколений. Адресуясь в пространстве памяти, центральный процессор и активные устройства (мастера шин) могут обращаться и к памяти периферии, отображенной на это пространство. Отметим, что в логическом распределении памяти фигурирует физическая память (оперативная и постоянная), а кэш является лишь «прозрачным» средством повышения ее производительности и не представляет отдельно адресуемых областей.

Процессоры 8086/88, применявшиеся в первых моделях PC, имели доступное адресное пространство 1 Мбайт (20 бит шины адреса). Эти процессоры использовали *сегментную модель памяти*, унаследованную и позднейшими моделями в реальном режиме. Согласно этой модели, исполнительный (линейный) адрес вычисляется по формуле *Addr* = *Seg* • 16 + *Offset*, где *Seg* и *Offset* — содержимое сегментного и адресного регистров (16-разрядных). Таким образом обеспечивался доступ к адресному пространству *Addr* = 00000 - FFFFFh при помощи пары 16-битных регистров. Заметим, что при *Seg* = FFFFh и *Offset* = FFFFh данная формула дает адрес 10FFEFh, но ввиду 20-битного ограничения на шину адреса эта комбинация в физической памяти указывает на адрес 0FFEFh. Таким образом, адресное пространство как бы сворачивается в кольцо с небольшим «нахлестом». Начиная с процессора 80286, шина адреса была расширена до 24 бит, а впоследствии (386DX, 486 и выше) до 32 и даже 36 (у процессоров P6). В реальном режиме процессора, используемом в DOS, применяется та же сегментная модель памяти и формально доступен лишь 1 Мбайт памяти, что является недостаточным для большинства современных приложений. *Процессоры 80286*, с которых началась жизнь IBM PC/AT, эмулируют 8086 с ошиб

кой: та самая единица в бите A20, которая отбрасывалась в процессорах 8086/88, теперь попадает на шину адреса и в результате максимально доступный линейный адрес в реальном режиме достигает 10FFEFh. За эту ошибку с радостью ухватились разработчики PC, поскольку дополнительные (64 К - 16) байты оперативной памяти, адресуемой в реальном режиме, оказались подарком, позволяющим освободить дефицитное пространство оперативной памяти для прикладных программ. В эту область (100000h - 10FFEFh), названную *высокой памятью* (High Memory Area, HMA), стали помещать часть операционной системы и небольшие резидентные программы. Однако для полной совместимости с процессором 8086/88 в схему PC ввели вентиль линии A20 шины адреса — *GateA20*, который либо пропускает сигнал от процессора, либо принудительно обнуляет линию A20 системной шины адреса. Старшие биты такой «заботы» не требуют, поскольку переполнение при суммировании 16-битных компонентов адреса по данной схеме до них не распространяется. Управление этим вентилем подключили к свободному программно-управляемому выходному биту 1 контроллера клавиатуры 8042, ставшего стандартным элементом архитектуры PC, начиная с AT. Предполагалось, что этим вентилем часто пользоваться не придется. Однако жизнь внесла свои поправки, и оказалось, что переключение вентиля в многозадачных ОС, часто переключающих процессор между защищенным режимом, реальным режимом и режимом V86, контроллером клавиатуры выполняется слишком медленно. Так появились альтернативные методы быстрого переключения вентиля, специфичные для различных реализаций системных плат (например, через порт 92h). Кроме того, иногда использовали и аппаратную логику быстрого декодирования команды на переключение бита, поступающую к контроллеру клавиатуры. Для определения способа переключения в утилиту CMOS Setup ввели соответствующие параметры (см. 6.6), позволяющие выбрать между стандартным, но медленным способом и менее стандартизованным, но быстрым, в зависимости от используемого ПО.

Поскольку ошибка эмуляции 8086 была радостно принята и широко использовалась, ее повторили в 386 и в следующих моделях процессоров. А для упрощения внешних схем в процессоры, начиная с 486, ввели и вентиль GateA20 с соответствующим внешним управляющим выводом.

Распределение памяти PC, физически адресуемой процессором, проиллюстрировано рис. 4.2 и представляется следующим образом:

- Адреса 00000h-9FFFFh (640 Кбайт) — *стандартная*, или *базовая*, память (conventional, или base, memory). Доступна DOS и программам реального режима. В некоторых системах с видеоадаптером MDA верхняя граница сдвигается к AFFFFh (704 Кбайт). Иногда верхние 128 Кбайт стандартной памяти (область 80000h-9FFFFh) называют расширенной базовой памятью (extended conventional memory).
- Адреса A0000h-FFFFFh (384 Кбайт) — *верхняя* память (Upper Memory Area, UMA). Зарезервирована для системных нужд. В ней размещаются области буферной памяти адаптеров, подключенных к шине ISA (например, видеопамять), и постоянная память (BIOS с расширениями). Эта область,

обычно используемая не в полном объеме, ставит архитектурный барьер на пути непрерывной (нефрагментированной) памяти, удобной для программного применения.

- Память выше 100000h — *дополнительная*, или *расширенная*, память (extended memory). Непосредственно доступна только в защищенном (и в «большом реальном») режиме для компьютеров с процессорами 286 и выше. В ней выделяется область 100000h-10FFEFh — *высокая* память (HMA) — единственная область расширенной памяти, доступная 286+ в реальном режиме при открытом вентиле *Gate A20.*

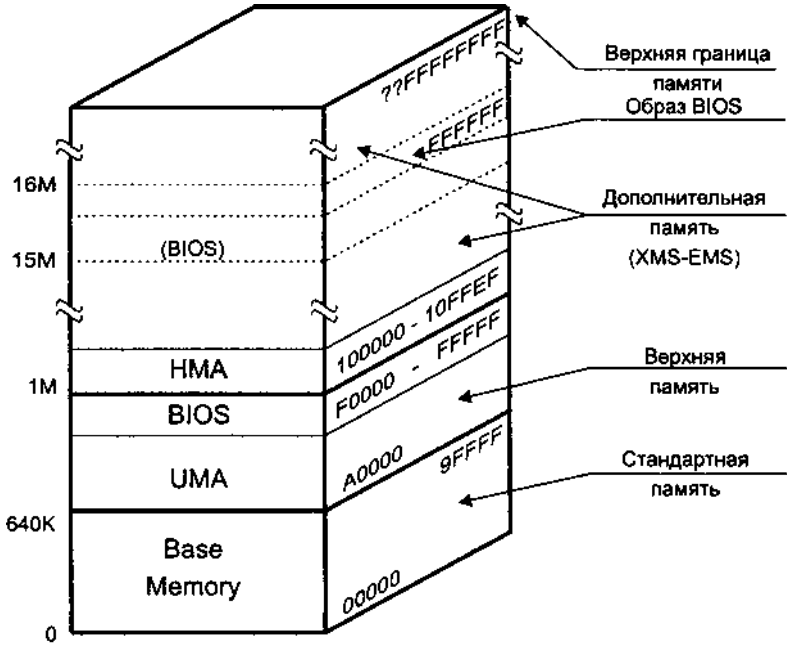


Рис. 4.2. Распределение памяти PC

Область памяти выше первого мегабайта в различных источниках называется по-разному. Ее современное английское название — extended memory — пересекается с названием одной из спецификаций ее использования — extended memory specification. В то же время название другой спецификации — expanded memory specification — в прямом переводе на русский язык неотличимо от перевода предыдущего термина (оба термина, и «extended» и «expanded», переводятся как «расширенный»). Будем придерживаться терминологии, укрепившейся в литературе, выпущенной издательством «Питер», и область всей физической памяти, расположенной в адресном пространстве выше первого мегабайта, назовем *дополнительной памятью.* Ее объем указывается строкой Extended Memory xxxxx Kbyte в таблице, выводимой после прохождения теста POST, и в меню стандартной конфигурации CMOS Setup. В современных компьютерах указывается общий объем оперативной памяти

Верхняя граница адресуемой памяти определяется разрядностью шины адреса процессора и системной шины; эти разрядности могут и не совпадать (ограничение дает компонент с минимальной разрядностью). В старших адресах памяти находится образ ПЗУ BIOS: в нем располагается программа начального запуска компьютера (POST), стартующая с фиксированного адреса. Оперативная память начинается с младших адресов, что обусловлено фиксированным положением таблицы векторов прерываний в реальном режиме (она начинается с нулевого адреса). Области пространства памяти, отводящиеся для отображения периферии, находятся в местах, не занятых оперативной и постоянной памятью.

Для первых компьютеров на процессорах 8086/88 с 20-битной шиной адреса верхняя граница адресуемой памяти — 0F FFFFh. Область ПЗУ BIOS расположена по адресам 0E 0000h-0F FFFFh; для оперативной памяти доступны область стандартной памяти (640 К) и некоторые области UMA. Память периферийных устройств может располагаться только в UMA.

Для компьютеров класса AT-286 с 24-битной шиной адреса верхняя граница адресуемой памяти — FF FFFFh. Область FE 0000h-FF FFFFh содержит ПЗУ BIOS (ROM BIOS Area), обращение к этой области эквивалентно обращению к ROM BIOS по адресам 0E 0000h-0F FFFFh. В этих компьютерах для оперативной памяти доступна и область дополнительной памяти, максимальный размер ОЗУ может достигать 15,9 Мбайт. Однако последний мегабайт (кроме области BIOS) может быть отдан для областей памяти периферии (дополнительно к UMA), так что объем ОЗУ окажется меньше 15 Мбайт.

Для процессоров 386+ и 32-битной шины адреса верхняя граница адресуемой памяти — FFFF FFFFh (4 Гбайт). Здесь образ BIOS находится в адресах FFFE 0000h-FFFF FFFFh, для ОЗУ и памяти периферии остается почти 4 Гбайт. Для обеспечения совместимости BIOS дополнительно проецируется и в адреса E 0000h-F FFFFh (для программ, вызывающих сервисы BIOS по фиксированным адресам). Для периферии доступна область UMA, не занятая BIOS, и область, находящаяся выше границы ОЗУ (но ниже границы 4 Гбайт). Периферия, расположенная на шине PCI и ее «родственниках», может быть приписана к любым адресам (на PCI доступно все адресное пространство). Периферия на шине ISA с ее 20-разрядным адресом может располагаться только в пределах первых 16 Мбайт: в UMA или в 16-м мегабайте памяти. Для адаптеров ISA в CMOS Setup предусмотрен параметр Memory Hole At 15-16M, его установка запрещает отображение на эти адреса оперативной памяти. В современных версиях BIOS эта «дырка» не мешает использованию ОЗУ объемом свыше 15 Мбайт.

Современные процессоры с 64-битным расширением, как и 32-разрядные процессоры с 36-битной шиной адреса, позволяют адресовать память и выше 4-ги- габайтной границы. Объем установленного ОЗУ также может превышать 4 Гбайт, но для периферийных устройств предусмотрено «окно» под границей

4 Гбайт. В процессорах с 64-битным расширением есть пара специальных регистров, определяющих нижние границы адресов для устройств ввода-вывода,

отображенных на память, для двух областей: под границей 4 Гбайт и под границей физически адресуемой памяти (зависящей от модели процессора).

Иногда (в некоторых версиях BIOS для 32-разрядных процессоров) в CMOS Setup можно включить проекцию BIOS на область FE 0000h-FF FFFFh (как в AT-286). Особого смысла в этом нет (программы реального режима задействуют образ в E 0000h-F FFFFh), однако включение этого параметра может создать трудности для использования более 16 Мбайт ОЗУ (система воспринимает только найденную непрерывную область оперативной памяти).

Объем установленной оперативной памяти определяется тестом POST при начальном включении (перезагрузке) компьютера, начиная с младших адресов. Натолкнувшись на отсутствие памяти (ошибку), тест останавливается и сообщает системе объем реально работающей памяти. Установленные в Setup «дырки» под 16-м мегабайтом современные версии BIOS успешно обходят. Современные системные платы позволяют установить ОЗУ, объем которого исчисляется уже гигабайтами. Возможность использования тех или иных областей оперативной памяти определяется типом операционной системы: ОС реального (MS-DOS и аналогичные) или защищенного режима (Windows, Unix, Linux).

Физическое распределение адресного пространства выполняется программированием регистров чипсета системной платы и мостов шин расширения. Северный хаб (или мост) чипсета определяет диапазоны адресов, которые обслуживает контроллер памяти (с «вырезами» в области UMA и другими «дырками»). Распределением оставшейся части занимаются мосты иерархии шин PCI (в эту иерархию входят мосты AGP, PCI-X и PCI-E), к которым могут подключаться и мосты старых шин (ISA).

Память для режима SMM

Компьютеры, использующие режим системного управления (System Management Mode, SMM), поддерживаемый большинством процессоров последних поколений, имеют еще одно адресное пространство памяти — *SMRAM* (System Management RAM). Это адресное пространство «параллельно» пространству обычной памяти и при работе доступно процессору только в режиме SMI. Память SMRAM может представлять собой часть физической оперативной памяти, хотя может быть реализована и отдельной микросхемой памяти. Объем памяти для режима SMM может варьироваться в диапазоне от 32 Кбайт (минимальные потребности SMM) до 4 Гбайт. SMRAM располагается, начиная с адреса SMIBASE, и распределяется следующим образом:

- SMIBASE+(FE00h-FFFFh) — область сохранения контекста процессора (распределяется, начиная со старших адресов по направлению к младшим). По прерывании SMI здесь сохраняются почти все регистры процессора, но сохранение регистров FPU/MMX не производится.
- SMIBASE+8000h — точка входа в обработчик (SMI handler).
- SMIBASE+(0-7FFFh) — свободная область.

После сброса процессора устанавливается SMIBASE = 0003 0000h, и первое же прерывание SMI вызовет сохранение контекста в область 0003 FE00h - 0003 FFFFh и запуск обработчика по адресу 0003 8000h. При этом процессор генерирует специальный выходной сигнал SMIACT#, означающий доступ к памяти SMRAM. Код обработчика, естественно, должен быть помещен в эту область до вызова SMI. Если для SMRAM используется системное ОЗУ, то область SMRAM перемещают в старшие адреса. Это можно сделать только обработчиком прерывания SMI, заменив образ регистра SMIBASE в сохраненной области контекста. После выхода из прерывания SMI это назначение вступит в силу, и следующий вход в SMM уже будет производиться по новым адресам. Область SMRAM должна исключаться из области ОЗУ, доступной операционной системе (обработчик SMI является более низкоуровневой процедурой, чем драйверы ОС).

Верхняя память — UMA

Верхняя память имеет области различного назначения, которые могут быть заполнены буферной памятью адаптеров, постоянной памятью или оставаться незаполненными. В первое время эти «дыры» не использовали из-за сложности «фигурного выпиливания» адресуемого пространства. С появлением механизма страничной переадресации (у процессоров 386 и выше) их стали по возможности заполнять «островками» оперативной памяти, названными блоками верхней памяти (Upper Memory Block, UMB). Эти области доступны DOS для размещения резидентных программ и драйверов через драйвер EMM386, который отображает в них доступную дополнительную память.

Стандартное распределение верхней памяти выглядит следующим образом (рис. 4.3):

- Адреса A0000h-BFFFFh (128 Кбайт) — *видеопамять* (Video RAM, VRAM). Обычно используется не полностью.
- Адреса C0000h-DFFFFh (128 Кбайт) — резерв для адаптеров (*adapter ROM* и *adapter RAM),* использующих собственные модули ROM BIOS или/и специальное ОЗУ, разделяемое с системной шиной.
- Адреса E0000h-EFFFFh (64 Кбайт) — свободная область, иногда занятая под системные модули BIOS *(system BIOS).*
- Адреса F0000h-FFFFFh (64 Кбайт) — системные модули BIOS.
- Адреса FD000h-FDFFFh — *ESCD* (Extended System Configuration Data) — область энергонезависимой памяти, используемая для конфигурирования устройств Plug and Play. Эта область имеется только при наличии PnP BIOS, ее положение и размер жестко не заданы.

В области UMA практически всегда присутствует графический адаптер. В зависимости от модели он занимает следующие области:

- MDA RAM - B0000h-B0FFFh.
- CGA RAM - B8000h-BBFFFh.
- EGA ROM - C0000h-C3FFFh/C7FFFh.

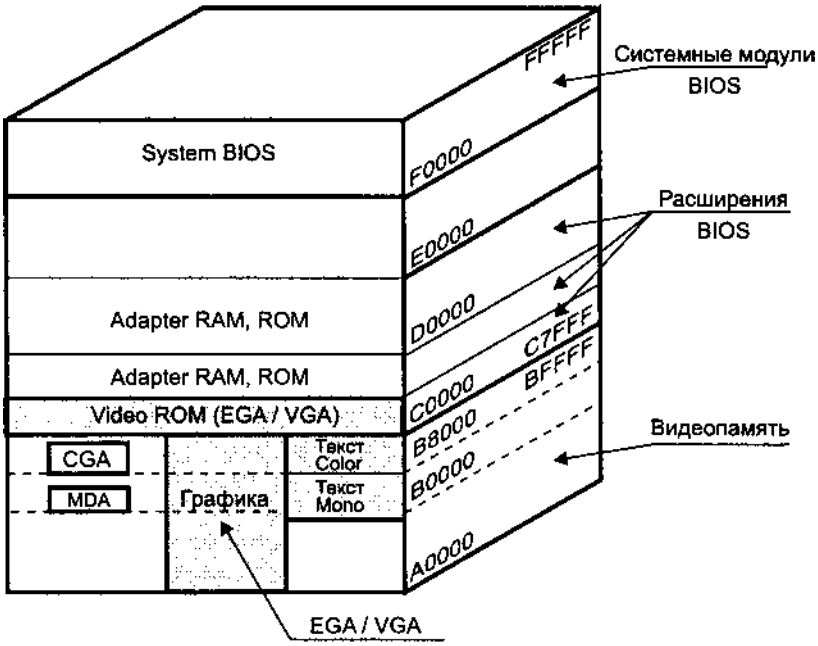


Рис. 4.3. Распределение верхней памяти (UMA)

- VGA ROM - C0000h-C7FFFh.
- EGA, VGA RAM — A0000h-BFFFFh, в зависимости от видеорежима используются следующие области:
  - Graphics — A0000h-AFFFFh;
  - Color Text - B8000h-BFFFFh;
  - Mono Text — B0000h-B7FFFh.

Также распространенным потребителем UMA являются расширения ROM BIOS, расположенные на платах дисковых контроллеров, а еще микросхемы удаленной загрузки (boot ROM) на платах адаптеров ЛВС. Обычно они занимают область C8000h — CBFFFh/C9FFFh/C8FFFh (для дисковых контроллеров), но могут и перемещаться при конфигурировании адаптеров.

Размер области, занимаемой системными модулями ROM BIOS, колеблется от 8 Кбайт у PC/XT до 128 Кбайт, однако разумное значение — 64 Кбайт. Большая область использовалась «на радостях» от появления микросхем ROM и флэш-памяти объемом 1 Мбит (128К х 8), но при этом размер доступной памяти UMA сократился. Тогда стали микросхемы того же (и большего) объема отображать только на область F0000h-FFFFFh (64 Кбайт), а иногда и меньшую. Это оказалось возможным, поскольку не все содержимое микросхемы ROM BIOS должно быть доступно одновременно. Таким образом удалось примирить интересы пользователей UMB с необходимостью расширения объема BIOS, связанной с усложнением технических средств.

Теневая память — Shadow ROM и Shadow RAM

В области верхней памяти UMA обычно располагаются устройства с медленной памятью: системная память BIOS (system ROM BIOS), расширения BIOS на графическом адаптере (video ROM BIOS), на контроллерах дисков и интерфейсов (adapter ROM), ПЗУ начальной загрузки на сетевой карте (boot ROM), видеопамять (video memory buffer). Они, как правило, реализованы на 8- или 16-битных микросхемах с довольно большим временем доступа. Обращение к полноразрядному системному ОЗУ выполняется гораздо быстрее. Для ускорения обращений к памяти этих устройств применяется *теневая память* (shadow memory) — подмена памяти системным ОЗУ. Теневая память появилась на развитых моделях AT-286, где она была реализована аппаратно. Процессоры класса 386+ позволяют ее реализовать программно за счет страничной переадресации (см. 7.3). Затенение ОЗУ и ПЗУ устройств выполняется по-разному.

При инициализации теневого ПЗУ *(shadow ROM)* содержимое затеняемой области копируется в ОЗУ и при дальнейшем чтении по этим адресам подставляется ОЗУ, а запись в эту область блокируется.

При использовании теневого ОЗУ *(shadow RAM)* запись производится одновременно в физическую память затеняемой области и в системное ОЗУ, наложенное на эту область. При чтении затененной области обращение идет только к системной памяти, что гораздо быстрее. Особенно велик эффект от затенения видеопамяти старых графических адаптеров, которая по чтению бывает доступна только во время обратного хода развертки, и процессору приходится долго ждать этого момента. Однако затенение областей разделяемой памяти, модифицируемых со стороны адаптеров, недопустимо — эти изменения не воспринимаются процессором. К разделяемой относятся буферная память сетевых адаптеров, видеопамять адаптеров с графическими сопроцессорами (акселераторами). Из этого следует, что затенение видеопамяти применимо только к примитивным графическим картам, устанавливаемым в слот ISA, и то не во всех режимах.

Обычно теневая память включается через CMOS Setup отдельными областями размером по 16 Кбайт или более крупными, и для каждой области указывается режим затенения (shadow ROM или shadow RAM). Возможно ее включение и драйверами ОС (например, драйвером EMM386). На современных системных платах затенение области системной микросхемы BIOS выполняется всегда, на старых платах затенением этой области можно было управлять. Затенение BIOS видеоадаптера (video BIOS shadowing) для работы в среде Windows с «родными» драйверами графического адаптера может и не дать прироста производительности.

Оперативная память для MS-DOS

Для операционных систем реального режима (MS-DOS) *оперативная память* может размещаться в стандартной области (640 Кбайт), дополнительной памяти и в некоторых областях UMA.

Стандартная память является самой дефицитной в PC, на ее небольшой объем (типовое значение — 640 Кбайт) претендуют и BIOS, и ОС реального режима, а остатки отдаются прикладному ПО. Стандартная память распределяется следующим образом:

- 00000h-003FFh — *векторы прерываний* (interrupt vectors), всего 256 двойных слов;
- 00400h-004FFh — *область переменных BIOS* (BIOS data area);
- 00500h-00xxxh — *область DOS* (DOS area);
- 00xxxh-9FFFFh — *память, предоставляемая пользователю* (user RAM), всего до 638 Кбайт (для PS/2 Mouse область 9FC00h-9FFFFh используется как расширение области переменных BIOS, поэтому размер пользовательской памяти меньше).

Дополнительная память и UMA могут использоваться в качестве оперативной только со специальными программными интерфейсами, для которых были разработаны спецификации EMS и XMS.

Спецификация па отображаемую память (Expanded Memory Specification, EMS) — это программная спецификация на использование дополнительной памяти DOS-программами реального режима. Спецификация *LIM EMS* — соглашение фирм Lotus, Intel, Microsoft на применение EMS. С помощью специальных аппаратных или программных средств любая область дополнительной памяти может быть отображена на небольшие страницы, расположенные в области UMA. В первоначальном варианте можно было задействовать 4 страницы по 16 Кбайт, примыкающие друг к другу, обычно начиная с адреса D0000h (положение страниц можно менять в пределах свободных областей UMA). Обращение прикладных программ к памяти EMS осуществляется через диспетчер памяти, вызываемый по прерыванию Int 67h. Программа, нуждающаяся в дополнительной памяти, должна сначала запросить выделение области, указав ее размер в 16-килобайтных страницах. В ответ на этот запрос (если имеется свободная память) диспетчер сообщает программе номер дескриптора EMS (EMS handler), по которому программа в дальнейшем будет ссылаться на выделенную ей область при управлении отображением. Далее программа через диспетчер назначает отображение требуемой логической страницы из выделенной ей области дополнительной памяти на выбранную физическую страницу, расположенную в области UMA. После этого любые программные обращения процессора к физической странице, расположенной в пределах первого мегабайта, будут в действительности работать с логической страницей дополнительной памяти, расположенной выше первого мегабайта, причем без переключения в защищенный режим. Для работы с иной логической страницей требуется вызов диспетчера для переназначения отображения. В режиме EMS 4.0, эмулируемом на процессорах 386+, появилась возможность увеличения числа доступных физических страниц и отображения дополнительной памяти не только на фиксированные области UMA, но и на любые области памяти.

Для поддержки EMS поначалу требовались специальные аппаратные средства. В компьютерах на процессорах 386 и выше появилась возможность программной эмуляции EMS, которую в MS-DOS 5+ выполняет драйвер EMM386.EXE.

Система EMS в основном предназначена для хранения данных — для исполняемого в данный момент программного кода она неудобна, поскольку требует программного переключения страниц через каждые 16 Кбайт. Ее используют для создания виртуальных дисков, хранения очередей заданий для печати, а также для хранения данных и даже программного кода некоторых резидентных программ (в целях экономии стандартной памяти).

Спецификация на расширенную память (extended Memory Specification, XMS) — это иная программная спецификация на использование дополнительной памяти DOS-программами, разработанная компаниями Lotus, Intel, Microsoft и AST для компьютеров на процессорах класса 286 и выше. Эта спецификация позволяет программе получить в распоряжение одну или несколько областей дополнительной памяти, а также задействовать область HMA. Распределением областей ведает диспетчер расширенной памяти — драйвер HIMEM.SYS. Диспетчер позволяет захватить или освободить область HMA (65 520 байт, начиная с 100000h), а также управлять вентилем линии адреса A20. Функции собственно XMS позволяют программе:

- определить размер максимально доступного блока памяти;
- захватить или освободить блок памяти;
- копировать данные из одного блока в другой, причем участники копирования могут быть блоками как стандартной, так и дополнительной памяти в любых сочетаниях;
- заблокировать блок памяти (запретить копирование) и разблокировать его;
- изменить размер выделенного блока.

В ответ на запрос выделения области диспетчер выдает 16-битный номер дескриптора блока (XMS handler), по которому выполняются дальнейшие манипуляции с этим блоком. Размер блока может достигать 64 Мбайт. Спецификация XMS позволяет программам реального режима устраивать «склады» данных в дополнительной памяти, которая им непосредственно недоступна, копируя в нее и из нее данные доступных областей первого мегабайта памяти. Доступ к диспетчеру XMS осуществляется через прерывание 2Fh. Заботу о переключении в защищенный режим и обратно для получения доступа к дополнительной памяти берет на себя диспетчер. По умолчанию HIMEM.SYS позволяет использовать до 32 дескрипторов блоков, но это число можно увеличить, задав параметр /NUMHANDLES=xx в строке загрузки драйвера HIMEM.SYS.

Помимо дополнительной памяти спецификация XMS определяет пару функций и для работы с блоками UMB — захватить блок требуемого размера (или определить максимально доступный блок) и освободить его.

Как видно, спецификации EMS и XMS различаются по принципу действия: в режиме EMS для доступа к дополнительной памяти выполняется отображение (страничная переадресация) памяти, а в режиме XMS — копирование блоков данных. На компьютерах с процессорами 386+ эти спецификации мирно сосуществуют при запуске драйвера HIMEM.SYS, поверх которого может быть загружен драйвер EMM386.EXE, пользующийся памятью XMS для эмуляции памяти EMS. Память, доступная EMS и XMS, может выделяться динамически из

числа дополнительной. Ключ NOEMS в строке запуска EMM386 запрещает выделение памяти для режима EMS.

Виртуальная память

Для ОС защищенного режима (в том числе Windows) доступна вся оперативная память, причем без каких-либо ухищрений вроде описанных режимов EMS и XMS. Более того, объем памяти, доступной приложениям, благодаря механизму виртуальной памяти может быть больше размера физической оперативной памяти.

Виртуальная память (virtual memory) представляет собой программно-аппаратное средство расширения пространства памяти, предоставляемой программе в качестве оперативной. Как уже было сказано в главе 1, эта память физически реализуется в оперативной и дисковой памяти под управлением соответствующей операционной системы. Виртуальное пространство памяти разбито на страницы фиксированного размера, а в физической оперативной памяти в каждый момент времени присутствует только часть из них. Остальные страницы хранятся на диске, откуда операционная система может «подкачать» их в физическую память, предварительно выгрузив на диск часть не используемых в данный момент модифицированных страниц. Обращение процессора к ячейке виртуальной памяти, присутствующей в физической памяти, происходит обычным способом. Если же затребованная область в данный момент не отображена в физической памяти, процессор вырабатывает исключение (внутреннее прерывание), по которому операционная система программно организует замещение страниц, называемое свопингом (swapping). Виртуальную память поддерживают процессоры, работающие в защищенном режиме, начиная с 80286, но реально ее широко стали применять только в операционных системах и оболочках для 32-разрядных процессоров (80386+). Виртуальная память используется лишь при наличии дополнительной памяти.

Суммарный объем виртуальной памяти, доступной всем приложениям, определяется объемом ОЗУ и файлов подкачки (их может быть и несколько). Объем файла подкачки может быть постоянным или же изменяться динамически по мере изменения потребностей системы. Для того чтобы приложениям хватало памяти, на диске, несущем динамический файл подкачки, должно быть достаточно свободного пространства (десятки и сотни мегабайт). В принципе, файл подкачки может располагаться и на сетевом диске, но при этом трафик сети оказывается напряженным. Конечно же, важен и объем установленной физической памяти — ее нехватка может быть принципиальным ограничением на запуск ряда приложений или установку операционных систем. При малом объеме ОЗУ свопинг (подкачка страниц) оказывается слишком интенсивным, в результате чего скорость работы приложений существенно снижается (обращения к диску выполняются на несколько порядков медленнее, чем к ОЗУ). Приложения реального времени (например, аудио- и видеопроигрыватели и тем более кодеры) могут стать неработоспособными именно из-за малого объема ОЗУ. Поскольку файл подкачки изменяет свой размер в процессе работы, важно следить за фрагментацией диска, несущего этот файл, — обращение к фрагментированному файлу выполняется медленнее, чем к нефрагментированному.

При выборе диска для размещения файла подкачки следует учитывать его быстродействие — время доступа и скорость передачи данных. При использовании приложений реального времени, интенсивно обменивающихся с дисками (те же проигрыватели и кодеры, а также программы, записывающие компакт-диски), файл подкачки и данные, с которыми работают эти приложения, по возможности следует размещать на разных дисках.

ВНИМАНИЕ ----------

Если на компьютере под управлением ОС защищенного режима (Windows, Unix, OS/2...) перестают запускаться приложения с сообщениями о недостаточном объеме оперативной памяти — проверьте, есть ли свободное место на жестких дисках, используемых для подкачки.

Увеличение физического объема оперативной памяти в ряде случаев может привести к неожиданному снижению производительности компьютера. Это возможно, когда системная плата (или процессор с вторичным кэшем) не способна кэшировать весь объем ОЗУ. У многих системных плат для процессоров Pentium кэшировались только первые 64 Мбайт ОЗУ; у первых процессоров Pentium II кэшировались только 512 Мбайт. Память, выходящая за размеры кэшируемой области, конечно же, доступна, но ее производительность гораздо ниже производительности кэшируемой. ОС Windows 9x распределяет доступную память, начиная с ее верхней границы, причем наверх попадает ее ядро, скорость работы которого существенна для многих приложений. Если после увеличения ОЗУ ядро попадает в некэшируемую область, можно наблюдать снижение производительности. Для «лечения этого недуга» можно воспользоваться условно бесплатной программой W2CACHE.COM, которая запускается в начале загрузки Windows и, оставаясь резидентной, «съедает» верхнюю часть памяти, заставляя ядро Windows загружаться в нижнюю, кэшируемую область. После окончания загрузки Windows программа освобождает занимаемую память, и ОС отдает ее в распоряжение приложений.

Вопросы организации виртуальной памяти в основном относятся к области системного программного обеспечения (ядра операционной системы). Аппаратные механизмы, обеспечивающие поддержку виртуальной памяти процессором, рассмотрены в главе 7. Здесь отметим, что использование виртуальной памяти с подкачкой страниц порождает проблему *согласованного видения памяти* разнообразными активными компонентами ПК: процессором, графическим акселератором и контроллерами ввода-вывода (адаптерами локальных сетей, контроллерами устройств хранения, шин USB и FireWire). Суть проблемы в том, что программа (и программист) оперируют *виртуальными адресами*, а устройства на системной шине оперируют *физическими адресами*. Что из этого следует, подробнее рассматривается в 5.1.

4.3. Пространство ввода-вывода

Процессоры семейства x86 имеют раздельную адресацию памяти и портов вво- да-вывода. Это разделение обеспечивается выделением специальных инструкций ввода-вывода, с помощью которых возможна передача данных между пор

тами и регистрами процессора (или портами и памятью). Инструкции ввода- вывода порождают шинные циклы обмена, в которых вырабатываются сигналы чтения из порта и записи в порт. На шине ISA это сигналы IORD# и IOWR# соответственно, они и отличают пространство ввода-вывода от пространства памяти, где соответствующие операции чтения и записи вырабатывают сигналы MEMRD# и MEMWR#. На шине PCI разделение памяти и пространства ввода-вывода происходит иначе — здесь тип операции кодируется 4-байтной командой в зависимости от типа инструкции, выполняемой процессором.

В инструкциях ввода-вывода используется 16-битная адресация, что обуславливает размер пространства ввода-вывода в 64 кбайт. Для дешифрации адресов портов в оригинальном ПК из 16 бит использовались только младшие 10 (A0-A9), что обеспечивает обращение к портам в диапазоне адресов 0-3FFh. Старшие биты адреса хотя и поступают на шину, но устройствами игнорируются. В результате обращения по адресам, к примеру, 378h, 778h, B78h и F78h, воспринимаются устройствами одинаково. Это упрощение, нацеленное на снижение стоимости как системной платы, так и схем плат адаптеров, для шины ISA никто не отменял. Устаревшие адаптеры *(legacy card)* для шины ISA для старших битов адреса не имеют даже печатных ламелей на своем краевом разъеме. Впоследствии перешли к 12-битной адресации устройств шины ISA, но ее приходится применять с оглядкой на возможное присутствие устройств с 10-битной адресацией. В адаптерах для шин MCA и PCI используются все 16 бит адреса, полная дешифрация адреса применяется и в современных системных платах. Карта распределения адресов ввода-вывода стандартных устройств PC приведена в табл. 4.1. Эта карта подразумевает 10-битную дешифрацию адреса. Естественно, что в конкретном компьютере реально присутствуют не все перечисленные устройства, но в то же время там могут оказаться другие, не попавшие в таблицу.

Таблица 4.1. Стандартная карта портов ввода-вывода

AT и PS/2	PC/XT	Назначение
000–00F	000–00F	Контроллер DMA #1 8237
010–01F		PS/2 – расширение DMA #1
020–021	020–021	Контроллер прерываний #1 – 8259A
040–05F	040–043	Таймер (PC/XT: 8253, AT: 8254)
060	060	Диагностический регистр POST (только запись)
	060–063	Системный интерфейс 8255
060, 064		Контроллер клавиатуры AT 8042
061		Источники NMI и управление звуком
070–07F		Память CMOS и маска NMI
080		Диагностический регистр
080–08F	080–083	Регистры страниц DMA
090–097		PS/2-микроканал, арбитр
	0A0	Маска NMI
0A0–0BF		Контроллер прерываний #2 – 8259A

AT и PS/2	PC/XT	Назначение
0C0–0DF		Контроллер DMA #2 8237A-5
0F0–0FF		Сопроцессор 80287
100–1EF		PS/2-управление микроканалом
170–177		Контроллер НЖМД #2 (IDE#2)
1F0–1F7		Контроллер НЖМД #1 (IDE#1)
200–207	200–20F	Игровой адаптер
	210–217	Блок расширений
238–23F		COM4
278–27F	278–27F	Параллельный порт LPT2 (LPT3 при наличии MDA)
	2A2–2A3	Часы MSM48321RS
2C0–2DF	2C0–2DF	EGA #2
2E0–2E7		COM4
2E8–2EF		COM4
2F8–2FF	2F8–2FF	COM2
300–31F		Плата прототипа
	320–32F	Жесткий диск XT
338–33F		COM3
370–377		Контроллер НГМД #2
376–377		Порты команд IDE#2
378–37F	378–37F	Параллельный порт LPT1 (LPT2 при наличии MDA)
380–38F	380–38F	Синхронный адаптер SDLC/BSC #2
3A0–3AF	3A0–3A9	Синхронный адаптер BSC #1
3B0–3BB	3B0–3BB	Монохромный адаптер (MDA)
3B4–3C9		PS/2-видеосистема
3BC–3BF	3BC–3BF	Параллельный порт LPT1 платы MDA
3C0–3CF	3C0–3CF	EGA #1
3C0–3DF	3C0–3DF	VGA
3D0–3DF	3D0–3DF	CGA/EGA
3E0–3E7		COM3
3E8–3EF		COM3
3F0–3F7	3F0–3F7	Контроллер НГМД #1
3F6–3F7		Порты команд IDE#1
3F8–3FF	3F8–3FF	COM1

Выбор базовых адресов стандартных устройств (3F8h, 3F0h, 378h и т. п.) объясняется стремлением к экономии. Эти адреса выбирались так, чтобы в их дешифрации участвовало максимальное количество единиц и минимальное — нулей (логические схемы «И-НЕ» применяются чаще, чем схемы «ИЛИ», а применение более сложных схем для дешифрации было нежелательно).

Каждой шине назначается своя область адресов ввода, и дешифратор адресов, расположенный на системной плате, при чтении открывает соответствующие буферы данных, так что реально считываться будут данные только с одной шины. При записи в порты данные (и сигнал записи) могут распространяться

и по всем шинам компьютера (широковещательно). В стандартном распределении адреса 0h-0FFh отведены для устройств системной платы. При наличии (и разрешении работы) периферийных устройств на системной плате чтение по этим адресам не распространяется на шины расширения. Для современных плат со встроенной периферией и несколькими шинами (ISA, PCI) распределением адресов управляет BIOS через регистры конфигурирования чипсета, а в иерархии шин PCI (и PCI-E) диапазоны адресов задаются программированием мостов (см. 14.6).

4.4. Аппаратные прерывания

Аппаратные прерывания обеспечивают реакцию процессора на события, происходящие асинхронно по отношению к исполняемому программному коду. По возникновении такого события адаптер (контроллер) устройства формирует запрос прерывания, который поступает на вход контроллера прерываний. Контроллер прерываний формирует общий запрос прерывания для процессора, а когда процессор подтверждает получение этого запроса, контроллер сообщает процессору вектор прерывания, по которому выбирается программная процедура обработки прерываний. Процедура должна выполнить действия по обслуживанию данного устройства, включая сброс его запроса, и отправить команду завершения в контроллер прерываний, что дает возможность отреагировать на следующие события. Логика работы системы прерываний и программная модель контроллера прерываний, совместимого с микросхемой i8259A, являются важной частью стандартизации архитектуры PC-совместимых компьютеров. Прерывания в процессорах x86 рассмотрены в 7.2.

Немаскируемые прерывания

Немаскируемые прерывания (Non-Maskable Interrupt, NMI) в PC используются для сигнализации о фатальных аппаратных ошибках. На немаскируемое прерывание процессор реагирует всегда (если завершено обслуживание предыдущего немаскируемого прерывания); этому прерыванию соответствует фиксированный вектор 2. Сигнал на линию NMI (вход процессора) приходит от схем контроля памяти (четности или ECC), от линий контроля шины ISA (IOCHK) и шины PCI (SERR#). Сигнал NMI блокируется до входа процессора установкой в 1 бита 7 порта 070h, отдельные источники разрешаются и идентифицируются битами порта 061h:

- бит 2 R/W (ERP) — разрешение контроля ОЗУ и сигнала SERR# шины PCI;
- бит 3 R/W (EIC) — разрешение контроля шины ISA;
- бит 6 R (IOCHK) — ошибка контроля на шине ISA (сигнал IOCHK#);
- бит 7 R (PCK) — ошибка четности ОЗУ или сигнал SERR# на шине PCI.

Источниками *прерывания SMI* являются схемы чипсета, участвующие в управлении энергопотреблением, а также контроллер USB при эмуляции традиционных клавиатуры и мыши. Это прерывание имеет наивысший приоритет и обслуживается несколько иначе, чем «классические» прерывания. Здесь процессор

не выполняет вызов процедуры, описанной в таблице прерываний, а переходит в режим SMM, что сопровождается установкой сигнала SMIACT#, по которому вместо обычной памяти процессору становится доступной память SMRAM. Выход из режима SMM происходит по выполнении инструкции RSM, завершающей процедуру обработки SMI. После обработки SMI возможен рестарт (повторное исполнение) инструкции останова (HALT) и инструкций ввода-вывода. Возможность рестарта инструкции ввода-вывода используют, например, когда прикладная программа (или системный драйвер) пытается обратиться операцией ввода-вывода к периферийному устройству, находящемуся в «спящем» режиме. Системная логика в этом случае должна выработать сигнал SMI# раньше сигнала готовности RDY#, завершающего шинный цикл рестартуемой инструкции ввода-вывода. Обработчик SMI «будит» устройство, после чего операция ввода-вывода рестартует и прикладное ПО (или драйвер) «не замечает», что устройство пребывало в спячке. Таким образом, управление потреблением может быть организовано на уровне BIOS способом, совершенно прозрачным для программного обеспечения (в том числе и ОС). Аналогично SMI позволяет незаметно выполнять манипуляции с контроллером USB по обращении программы к порту контроллера клавиатуры (портам 60h и 61h).

Маскируемые прерывания

Маскируемые прерывания используются для сигнализации о событиях в устройствах. Реакция процессора на маскируемые прерывания может быть задержана сбросом его внутреннего флага IF (инструкция CLI запрещает прерывания, STI — разрешает). По возникновении события, требующего реакции, адаптер (контроллер) устройства формирует *запрос прерывания*, который поступает на вход *контроллера прерываний*. Задача контроллера прерываний — довести до процессора запрос прерывания и сообщить вектор, по которому выбирается программная процедура обработки прерываний. В IBM PC-совместимых компьютерах применяется два основных типа контроллеров прерываний:

- *Периферийный контроллер прерываний* (Peripheral Interrupt Controller, PIC) программно совместим с традиционным контроллером 8259A, использовавшимся еще в первых моделях IBM PC. Со времен IBM PC/AT применяется связка из пары каскадно соединенных контроллеров PIC, позволяющая обслуживать до 15 линий запросов прерываний.
  - *Усовершенствованный периферийный контроллер прерываний* (Advanced Peripheral Interrupt Controller, APIC) введен в компьютеры для поддержки мультипроцессорных систем на базе процессоров 4-5-го поколений (486 и Pentium) и используется поныне для более поздних моделей процессоров. Помимо поддержки мультипроцессорных конфигураций, современный контроллер APIC позволяет увеличивать число доступных линий прерываний и обрабатывать запросы прерываний от устройств PCI, посылаемые через механизм сообщений (MSI). Компьютер, оснащённый контроллером APIC, обязательно имеет возможность функционировать и в режиме, совместимом со стандартной связкой пары PIC. Этот режим включается по аппаратному

сбросу (и включению питания), что позволяет использовать старые ОС и приложения MS-DOS, «не знающие» APIC и мультипроцессирования.

Традиционная схема формирования запросов прерываний с использованием пары контроллеров PIC изображена на рис. 4.4.

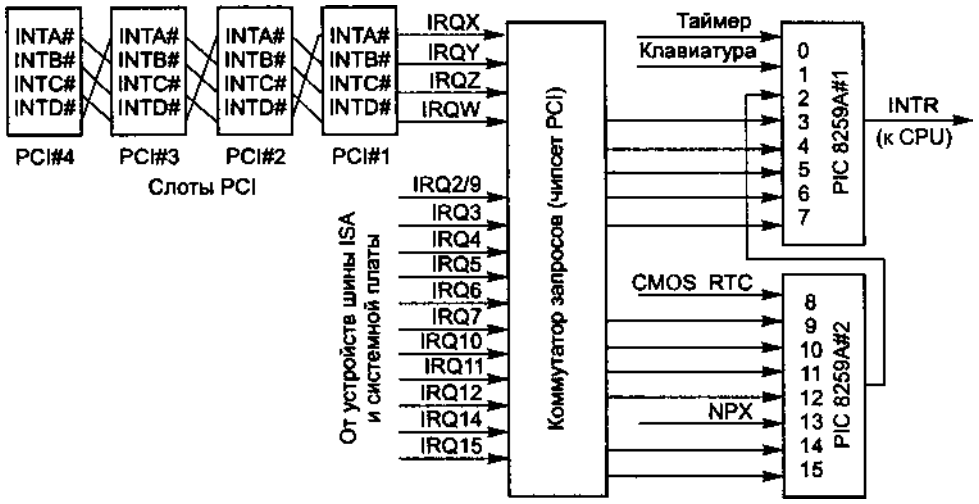


Рис. 4.4. Коммутация запросов прерываний

На входы контроллеров прерываний поступают запросы от стандартных устройств (клавиатура, системный таймер, CMOS-таймер, сопроцессор), периферийных контроллеров системной платы и от карт расширения. Традиционно все линии запросов, не занятые перечисленными устройствами, присутствуют на всех слотах шины ISA/EISA. Эти линии обозначаются как IRQx и имеют общепринятое назначение (см. далее). Часть этих линий отдается в распоряжение шины PCI.

Традиционные (не PnP) устройства ISA используют *запрос по положительному перепаду* сигнала IRQx, что делает невозможным их совместное (разделяемое) применение несколькими устройствами. Устройства ISA, поддерживающие механизм PnP, позволяют настраиваться на *прерывание по низкому уровню* сигнала; устройства PCI работают только по низкому уровню. Это дает возможность задействовать линии запросов в режиме разделения, что порождает определенные проблемы (см. далее). Линии IRQx, используемые шиной PCI, становятся недоступными для шины ISA. «Дележку» линий между шинами, а также управление чувствительностью (к перепаду или уровню) отдельных линий обеспечивают параметры CMOS Setup, а также механизм PnP. В параметрах CMOS Setup название «ISA» или «Legacy» подразумевает использование линий IRQx традиционными адаптерами шины ISA (статическое распределение), а название «PCI/PnP» — адаптерами шины PCI или адаптерами PnP для шины ISA (динамическое распределение).

Для запросов прерывания с шины PCI (см. 14.5) задействуют 4 линии запросов прерывания, которые обозначают как INTR A, B, C, D. Эти линии работают по низкому уровню, что дает возможность их совместного (разделяемого) использования. Линии циклически сдвигаются в слотах и независимо коммутируются на доступные линии IRQx с помощью конфигурационных регистров чипсета.

В табл. 4.2 запросы расположены в порядке убывания приоритета. Номера векторов, соответствующих линиям запросов контроллеров, система приоритетов и некоторые другие параметры задаются программно при инициализации контроллеров PIC или APIC. Эти основные настройки остаются традиционными для совместимости с программным обеспечением, но различаются для ОС реального и защищенного режимов. Так, например, в ОС Windows базовые векторы для ведущего и ведомого контроллеров — 50h и 58h соответственно.

Таблица 4.2. Аппаратные прерывания (в порядке убывания приоритета)

Имя (номер[1])	Вектор[2]	Вектор[3]	Контроллер/ маска	Описание
NMI	02h	02h	–	Контроль канала, четность памяти (в XT — сопроцессор)
IRQ0	08h	50h	#1/1h	Таймер (канал 0 8253/8254)
IRQ1	09h	51h	#1/2h	Клавиатура
IRQ2	0Ah	52h	#1/4h	XT – резерв, AT – недоступно (подключается каскад IRQ8–IRQ15)
IRQ8	70h	58h	#2/1h	CMOS RTC — часы реального времени
IRQ9	71h	59h	#2/2h	Резерв
IRQ10	72h	5Ah	#2/4h	Резерв
IRQ11	73h	5Bh	#2/8h	Резерв
IRQ12	74h	5Ch	#2/10h	PS/2-Mouse (резерв)
IRQ13	75h	5Dh	#2/20h	Математический сопроцессор
IRQ14	76h	5Eh	#2/40h	HDC – контроллер НЖМД
IRQ15	77h	5Fh	#2/80h	Резерв
IRQ3	0Bh	52h	#1/4h	COM2, COM4
IRQ4	0Ch	53h	#1/10h	COM1, COM3
IRQ5	0Dh	54h	#1/20h	XT – HDC, AT – LPT2, Sound (резерв)
IRQ6	0Eh	55h	#1/40h	FDC — контроллер НГМД
IRQ7	0Fh	56h	#1/80h	LPT1 — принтер

[1] Запросы прерываний 0, 1, 8 и 13 на шины расширения не выводятся.

[2] Указаны номера векторов при работе в реальном режиме процессора.

[3] Указаны номера векторов при работе в ОС Windows.

Каждому устройству, для поддержки работы которого требуются прерывания, должен быть назначен свой номер прерывания. Назначения номеров прерываний выполняются с двух сторон: во-первых, адаптер, нуждающийся в прерыва-

ниях, должен быть сконфигурирован на использование конкретной линии шины (джамперами или программно); во-вторых, программное обеспечение, поддерживающее данный адаптер, должно быть проинформировано о номере применяемого вектора. В процессе назначения прерываний может участвовать система PnP для шин ISA и PCI; для распределения линий запросов между шинами служат специальные параметры CMOS Setup. Современные ОС имеют возможность изменить назначение запросов относительно распределения, сделанного через CMOS Setup.

После того как конфигурирование системы прерываний произведено (проинициализирован контроллер прерываний, устройствам назначены линии запросов и установлены указатели на процедуры обработки), отработка маскируемых аппаратных прерываний происходит следующим образом:

1. Устройство по событию прерывания возбуждает назначенную ему линию запроса прерывания.
2. Контроллер принимает *сигналы запросов от источников прерываний* (сигналы IRQx) и при наличии незамаскированного запроса подает сигнал *общего запроса прерывания* (сигнал INTR) процессору x86.
3. Процессор, реагируя на запрос (когда прерывания флагом IF разрешены), сохраняет в стеке содержимое регистра флагов и адрес возврата, после чего формирует *шинный цикл INTA* (Interrupt Acknowledge — подтверждение прерывания), который доводится до контроллера прерываний.
4. В момент получения сигнала INTA контроллер прерываний фиксирует состояние своих входов запросов — к этому моменту их состояние могло измениться: например, появились новые запросы или пропал запрос от «нетерпеливого» устройства. Контроллер анализирует поступившие запросы в соответствии с запрограммированной схемой приоритетов и посылает процессору *вектор прерывания*, соответствующий самому приоритетному незамаскированному запросу, присутствующему на входе контроллера в момент подачи шинной команды *INTA*. При этом контроллер выполняет и некоторые действия в соответствии с установленной приоритетной политикой, учитывающие, какой именно вектор был послан (какой из запросов пошел на обслуживание).
5. Получив вектор прерывания, процессор по его номеру вызывает соответствующую процедуру обработки прерывания. Если данный вектор прерывания используется не только для аппаратных прерываний, но и для исключений и/или программных прерываний, то процедура в первую очередь должна определить, к какому из этих типов относится данное событие. Для этого процедура может обратиться к контроллеру PIC (прочитать регистр ISR) и проанализировать состояние регистров процессора.

   Дальнейшие шаги описывают случай аппаратного прерывания.
6. Процедура обработки прерывания идентифицирует источник прерывания — определяет устройство, его вызвавшее. При разделяемом использовании несколькими устройствами данного номера запроса (следовательно, и вектора) идентифицировать источник прерывания можно только последовательными обращениями к регистрам каждого из этих устройств. При этом следует учи

тывать возможность поступления запросов от нескольких устройств одновременно или в процессе обработки прерывания от одного из них.

7. Процедура обслуживает устройство-источник прерывания — выполняет «полезные» действия, связанные с событием, о котором и сигнализировало устройство. Это обслуживание должно обеспечить и *снятие сигнала запроса* прерывания от данного устройства. В случае разделяемых прерываний источников может быть несколько, и все они требуют обслуживания.

   Если обработка прерывания занимает значительное время, в течение которого требуется реакция системы на более приоритетные запросы, то после критической секции в обработчик включают инструкцию STI, устанавливающую флаг разрешения прерываний (IF) в процессоре. С этого момента возможны *вложенные прерывания*, прерывающие работу данного обработчика при поступлении другого, более приоритетного прерывания.
8. Процедура обработки прерывания посылает контроллеру *команду EOI* (End Of Interrupt — завершение обработки прерывания), по которой контроллер разрешает последующий прием сигнала с обслуженного входа и менее приоритетных. Это должно быть сделано после снятия сигнала прерывания от обслуженных устройств, иначе контроллер после *EOI* пошлет повторный запрос. Обработчик прерывания, для которого запрос поступил от ведомого контроллера, должен послать *EOI* как ведомому, так и ведущему контроллеру. Фрагмент обработчика от инструкции подачи команды *EOI* до завершения (инструкции IRET) должен быть *непрерываемым*, то есть являться *критической секцией.* Если обработчик разрешал вложенные прерывания, то перед инструкцией подачей команды *EOI* должна присутствовать инструкция CLI, запрещающая прерывания.
9. Завершается обработка прерывания инструкцией IRET, по которой процессор возвращается к выполнению прерванного потока инструкций, предварительно извлекая из стека содержимое регистра флагов (с установленным IF), и аппаратные прерывания снова оказываются разрешенными.

Эта последовательность описана применительно к обычному контроллеру прерываний (PIC), в системах с APIC меняется способ доставки вектора прерывания от контроллера к процессору, а в прерываниях MSI меняется способ доставки сигнала от устройства к контроллеру APIC. Эти нюансы описаны в последующих разделах.

Традиционный контроллер прерываний — PIC

Контроллер прерываний (PIC) 8259A является периферийным устройством, которое связано с процессором через ту или иную шину расширения ввода-вывода. По этой шине процессор может обращаться к регистрам контроллера, программируя его режимы и управляя им, а также получать от контроллера 8-бит- ный *вектор прерывания*, для чего в интерфейсе системной шины процессора и шины расширения имеется специальная команда подтверждения прерывания *(INTA).* Контроллер 8259A имеет 8 входов запросов от источников и один выход общего запроса. Каждому из входов соответствует свой вектор; программи

рованием регистров контроллера задается номер вектора для входа 0, остальным входам соответствуют последующие номера векторов. Каждый вход может быть программно *замаскирован* — тогда он не вызывает сигнал общего запроса. Контроллер занимает два адреса в пространстве ввода-вывода, программное обращение позволяет управлять режимами работы контроллера, а также приоритетами и масками запросов. Кроме того, контроллер позволяет работать в *режиме опроса* (poll mode), или *полиига* (polling), обеспечивая идентификацию источника прерывания (с учетом приоритетов) без выработки общего запроса.

С каждым входом запроса в контроллере связано по одному биту в регистрах IRR, IMR и ISR; бит 0 каждого из этих регистров относится ко входу 0, бит 1 — ко входу 1, бит 7 — ко входу 7:

- IRR (Interrupt Request Register) — регистр запросов прерываний. В этом регистре бит устанавливается при обнаружении сигнала прерывания на соответствующем входе, независимо от маски.
- IMR (Interrupt Mask Register) — регистр масок прерываний. Единичное значение бита означает замаскированность данного входа — по запросу с замаскированного входа общий запрос прерывания не генерируется.
- ISR (Interrupt Service Register) — регистр обслуживаемого прерывания.

Контроллер прерываний позволяет программировать свои входы на чувствительность к уровню или перепаду сигнала.

- *Чувствительность к уровню* (level sensitive) означает, что контроллер прерываний вырабатывает запрос прерывания процессора по факту обнаружения определенного уровня на входе IRQx. Если к моменту завершения обработки этого запроса (после записи команды *EOI* в регистр контроллера прерываний) контроллер снова обнаруживает активный уровень на том же входе DRQx, то он снова формирует запрос на прерывание процессора.
- *Чувствительность к перепаду* (edge sensitive) означает, что контроллер прерываний вырабатывает запрос прерывания процессора только по факту обнаружения перепада (на ISA — положительного) на входе IRQx. Повторно запрос по этому входу возможен только по следующему такому же перепаду, то есть сигнал предварительно должен вернуться в исходное состояние.

В любом случае сигнал запроса аппаратного прерывания IRQx должен удерживаться генерирующей его схемой, по крайней мере, до цикла подтверждения прерывания процессором — именно в этот момент PIC определяет самый приоритетный незамаскированный запрос и по нему формирует вектор. Если к этому моменту запрос окажется снятым, источник прерывания корректно идентифицирован не будет, и контроллер сообщит *ложный вектор прерывания* (spurious interrupt), соответствующий его входу с максимальным номером (IRQ7 для первого контроллера и IRQ15 для второго). Обычно периферийные устройства строят так, что сигнал запроса сбрасывается при обращении программы обслуживания прерывания к соответствующим регистрам адаптера, так что ложных прерываний возникать не должно.

Один контроллер PIC 8259A позволяет обслуживать 8 запросов прерываний; в PC/AT применяется *каскадное соединение двух контроллеров*, один из кото

рых является ведущим, другой — ведомым. *Ведущий контроллер 8259A#1* обслуживает запросы 0, 1, 3-7; его выход подключается ко входу запроса прерываний процессора. К его входу 2 подключен *ведомый контроллер 8259A#2*, который обслуживает запросы 8-15. При этом поддерживается вложенность приоритетов — запросы 8-15 со своим рядом убывающих приоритетов вклиниваются между запросами 1 и 3 ведущего контроллера, приоритеты запросов которого также убывают с ростом номера. В XT каскадирование не применялось, и один контроллер 8259A обслуживал все 8 линий запросов.

В IBM PC/XT/AT используется специальный режим вложенных прерываний с фиксированным приоритетом и автоматическим неспецифическим завершением. После инициализации (процедурой POST и при загрузке ОС) все неиспользуемые входы контроллеров замаскированы (на запросы прерываний не реагируют), а их векторы прерываний указывают на «заглушку» — процедуру с единственной инструкцией IRET. Для подключения обработчика прерывания от устройства первым делом следует загрузить обработчик в память и установить указатель на него в таблице прерываний. Далее следует демаскировать соответствующий ему вход в контроллере прерываний. Если обработчик прерывания удаляется из памяти, предварительно должен быть замаскирован соответствующий ему вход контроллера. Все изменения в таблице прерываний должны выполняться при замаскированных прерываниях, чтобы избежать попытки использования вектора в процессе его модификации (это приведет к «вылету» программы — обращению по некорректному адресу).

В современных системных платах функции контроллеров прерываний возлагаются на чипсет, который может иметь и более гибкие средства управления, чем пара контроллеров 8259A. Процедура инициализации контроллеров может и отличаться от традиционной, но ею занимается тест POST, который «знает» особенности системной платы. Однако в операционном режиме всегда сохраняется программная совместимость с 8259A. При работе с контроллером прерываний от программы требуется лишь управлять маской своего запроса (при инициализации программы нужно обнулить маску требуемого запроса) и корректно завершать обработку прерываний. Каждая процедура обработки аппаратного прерывания должна завершаться командой EOI (End Of Interruption), посылаемой контроллеру:

- для 1-го контроллера — посылка байта 20h по адресу 020h;
- для 2-го контроллера — посылка байта 20h по адресу 0A0h; программный вызов прерывания 0Ah — завершение для ведущего контроллера; для некоторых старых версий BIOS был необходим явный сброс маски запроса в регистре 2-го контроллера.

Некорректно завершенная процедура не позволит повторно использовать данный или другие запросы прерываний.

Улучшенный контроллер прерываний — APIC

Контроллер APIC в первую очередь предназначен для симметричных мультипроцессорных систем (см. 7.5), в которых все процессоры разделяют общие

устройства ввода-вывода и общие контроллеры прерываний. Однако APIC используется и в однопроцессорных системных платах. Система с APIC состоит из локальных контроллеров, установленных в процессорах, и контроллеров прерываний от ввода-вывода (одного или нескольких). Все контроллеры APIC соединены между собой локальной шиной, по которой они обмениваются друг с другом сообщениями. Задача каждого *локального контроллера* (local APIC) — трансляция сообщений, принятых по локальной шине, в сигналы, вызывающие все аппаратные прерывания своего процессора, — маскируемые (INTR), немаскируемые (NMI) и системного управления (SMI). Кроме того, локальные контроллеры APIC позволяют каждому процессору генерировать прерывания для других процессоров. Локальный контроллер имеет внутренний интервальный таймер, позволяющий вырабатывать прерывания через программируемый интервал времени. *Контроллер прерываний от ввода-вывода* (I/O APIC) преобразует запросы аппаратных прерываний от устройств в сообщения протокола локальной шины APIC. В мультипроцессорном режиме он отвечает за распределение прерываний по процессорам, для чего может потребоваться статическое или динамическое распределение. В случае статического распределения для каждого номера прерывания указывается номер процессора, который его обслуживает. В случае динамического распределения каждое прерывание направляется наименее приоритетному в данный момент процессору. Этот же контроллер отвечает за распространение сигналов о системных событиях (NMI, INIT, SMI) и межпроцессорных прерываний. Прерывания в мультипроцессорных системах подробно рассмотрены в документе «Intel Architecture Software Developer's Manual Volume 3: System Programming Guide», доступном на сайте http://www. intel.com. Здесь же ограничимся описаниями возможностей, предоставляемых контроллерами APIC для сигнализации прерываний ввода-вывода.

Контроллер I/O APIC является частью чипсета системной платы, например, он входит в хабы ICH2 и ICH3 чипсетов Intel. Доступны три режима обработки прерываний:

- *режим PIC* (PIC mode) — эмуляция пары PIC 8259A с традиционной передачей сигналов прерывания одному загрузочному процессору (Bootstrap Processor, BSP) по линиям INTR и NMI;
- *режим виртуальных проводов* (virtual wire mode) — то же, но с подачей сигналов прерывания по локальной шине APIC, при этом контроллер I/O APIC может работать совместно с PIC 8259A, обеспечивая дополнительные возможности (в частности, дополнительные входы запросов прерываний);
- *симметричный режим ввода-вывода* (symmetric I/O mode) — сообщения прерывания от устройств генерирует APIC; прерывания могут доставляться любому процессору; каждый вход запроса индивидуально программируется с помощью *таблицы перенаправления прерываний ввода-вывода* (I/O redirection table).

Первые два режима обеспечивают полную совместимость с системой прерываний PC/AT, с программной точки зрения они эквивалентны, различия лежат в области схемотехники. При аппаратном сбросе (и включении питания) система начинает работать в одном из этих режимов. Когда система подготовится

к переходу в мультипроцессорный режим (MP), APIC переводится в симметричный режим и активизирует таблицу перенаправления прерываний (предварительно программно инициализированную).

Для симметричных систем допустимы векторы в диапазоне 10h-FEh. Уровень приоритета прерывания определяется номером его вектора, деленным на 16. Самый приоритетный уровень — нулевой.

Контроллер I/O APIC позволяет вырабатывать значительное число запросов прерываний; каждому запросу соответствует свой элемент в таблице перенаправления, находящейся в APIC. Каждый элемент определяет способ реакции на свой запрос, вектор прерывания и процессор (процессоры) назначения, который должен его обработать. С запросами связаны индивидуальные входы INTINn; определенный уровень или перепад сигнала на этих входах вызывают соответствующие запросы. Чувствительность и вектор (а следовательно, и приоритет) для каждого запроса программируются индивидуально. Более совершенные модели I/O APIC позволяют вызывать прерывание записью номера входа в регистр контроллера, что, например, используется для поддержки прерываний MSI на шине PCI. При этом возможна и экономия сигнальных входов: APIC может иметь входы INTINn не для всех номеров запросов, вырабатываемых записью в этот регистр. Однако число запросов всегда ограничивается размером таблицы перенаправления.

Регистры контроллеров APIC отображаются на пространство памяти. Все *локальные контроллеры APIC* используют один и тот же диапазон адресов (по умолчанию базовый адрес — FEE0 0000h) — к их регистрам обращаются только программы, исполняемые на их же процессорах, и эти обращения не выводятся на системную шину. *Контроллеры I/O APIC* доступны всем процессорам, по умолчанию базовый адрес первого I/O APIC — FEC0 0000h, базовые адреса остальных контроллеров (если таковые имеются) назначаются последовательно с шагом 1000h.

Выделение для сообщений APIC отдельной локальной шины позволяет освободить системную шину процессора от трафика, связанного с обслуживанием прерываний (подачи подтверждений прерываний для получения вектора). В современных процессорах используется локальная шина, состоящая из трех сигнальных линий: PICD[1:0] — двунаправленная шина данных и PICCLK — сигнал синхронизации (тактовая частота). Протокол шины обеспечивает распределенный механизм арбитража и надежную доставку сообщений. Сообщения, передаваемые по локальной шине APIC, программно невидимы; реализация и протокол шины могут быть изменены производителями процессоров и чипсетов системных плат, но это не отразится на ПО.

Помимо использования последовательной локальной шины, есть и иной вариант доставки сообщений к локальным контроллерам APIC — в этом варианте происходят обращения к пространству памяти. Для этого локальные контроллеры APIC настраиваются на отслеживание операций записи по определенным адресам. Источник сообщений выполняет операцию записи в пространство памяти, в которой и адрес, и данные несут информацию о событии прерывания. В качестве источника сообщений может выступать *расширенный контроллер*,

называемый I*/O(x)APIC*. Вышеупомянутый хаб ICH3 имеет возможность работы в режиме I/O(x)APIC.

Проблема разделяемых прерываний

Линии запросов прерываний в компьютере, насыщенном периферийными устройствами, являются самым дефицитным ресурсом, поэтому приходится использовать эти линии совместно, то есть применять *разделяемые прерывания (shared interrupts)* между несколькими устройствами. Обработчики прерываний (программы) от разных устройств, имеющих одну линию запроса (и следовательно, общий вектор прерывания), должны быть выстроены в цепочку. В процессе обработки прерывания очередной обработчик в цепочке чтением известного ему регистра своего устройства должен определить, не это ли устройство вызвало прерывание. Если это, то обработчик должен выполнить необходимые действия и сбросить сигнал запроса прерывания от своего устройства, после чего передать управление следующему обработчику в цепочке; в противном случае он просто передает управление следующему обработчику. Чтобы прерывания, одновременно возникающие от нескольких устройств, не терялись, контроллер прерываний должен быть *чувствительным к уровню*, а не к перепаду на входе запроса. В соответствии со схемотехникой логики ТТЛ и КМОП активным уровнем должен быть низкий; выходной формирователь сигнала запросов у адаптеров должен обладать открытым коллектором (ТТЛ) или открытым стоком (КМОП); вход запроса у контроллера должен быть «подтянут» к высокому уровню резистором. Тогда непосредственное соединение этих выходов со входом контроллера («монтажное И») даст требуемый результат в аппаратном плане, а в программном плане необходимо корректно выстроить обработчики в цепочку.

Поясним, почему надежная разделяемость при чувствительности к перепаду на линии запроса невозможна. Если устройство 1 выработает сигнал запроса после того, как его выработает (но еще не снимет) устройство 2, то контроллер обработает только один запрос. Цепочка программных обработчиков окажется ненадежной: если обработчик устройства 1 в этой цепочке проверит свое устройство до возникновения прерывания, то прерывание будет потеряно. Поскольку прерывания по своей природе обычно асинхронны, работа этих устройств совместно с поддерживающими программами окажется загадочно нестабильной.

Для шины PCI с аппаратной точки зрения проблема разделения прерываний решена — здесь активным уровнем запроса является низкий, и контроллер прерываний чувствителен к уровню, а не к перепаду. Для шины ISA с ее запросами прерываний по положительному перепаду разделяемость прерываний невозможна. Исключения составляют системные платы и устройства с поддержкой ISA PnP, которые можно заставить работать и по низкому уровню.

После успешного решения аппаратной задачи обеспечения разделяемости линий запроса возникает задача *идентификации источника* каждого прерывания, что позволило бы запустить соответствующую процедуру обработки. Жела

тельно, чтобы эта задача решалась средствами ОС и с минимальными потерями времени. В первых версиях (до PCI 2.2 включительно) не было общепринятого способа программной индикации и запрета прерываний. К сожалению, в конфигурационных регистрах не нашлось стандартного места для бита, индицирующего введение запроса прерывания данным устройством, — тогда бы в прерываниях для PCI не было бы проблем с унификацией разделяемых прерываний. В каждом устройстве для работы с прерываниями используются свои специфические биты операционных регистров, относящихся к пространству памяти или ввода-вывода (иногда и к конфигурационному пространству). При этом определить, является ли данное устройство в текущий момент источником прерывания, может только его обработчик прерывания (Interrupt Service Routine, ISR), входящий в драйвер данного устройства. Таким образом, у ОС нет иной возможности диспетчеризации разделяемых прерываний, кроме как выстроить их обработчики в цепочку. За расторопность и корректность ISR отвечает его разработчик. В PCI 2.3 наконец-то появился фиксированный бит (Interrupt Status) в регистре состояния конфигурационного пространства устройства (функции), по которому ОС может определить источник разделяемого прерывания и вызвать только его ISR. Однако упоминание о поддержке PCI 2.3 в описаниях устройств и операционных систем встречается нечасто.

Обработчики прерываний устройств должны вести себя корректно, учитывая возможность попадания в цепочку обработчиков разделяемого прерывания. Встречается типичная ошибка обработчика прерываний: прочтя регистр состояния устройства и не обнаружив признака запроса, драйвер «на всякий случай» выполняет сброс всех источников запроса (а то и сброс всего устройства). Эту ошибку порождает незадачливый разработчик драйвера, не учитывающий возможности разделения прерываний и не доверяющий разработчикам аппаратных средств. Увидев в процессе отладки эту неожиданную ситуацию (прерывание вызвано, а источник не виден), он ее «учитывает» введением «вредного» фрагмента программного кода. Вредность заключается в том, что с момента чтения регистра устройства (не давшего признака запроса) и до выполнения этого ненужного сброса в устройстве может возникнуть запрос прерывания, который будет «вслепую» сброшен и, следовательно, потерян.

Однако и при корректности обработчиков, выстроенных в цепочку, разделяемые прерывания для разнотипных устройств в общем случае работоспособными считать нельзя — возможны потери прерываний от устройств, требующих быстрой реакции. Это может происходить, если обработчик такого устройства окажется в конце цепочки, а предшествующие ему обработчики будут «нерасторопными» (не самым быстрым способом обнаружат, что прерывание — чужое). Поведение системы в такой ситуации может меняться в зависимости от порядка загрузки драйверов. Для нескольких однотипных устройств (например, сетевых адаптеров на однотипных микросхемах контроллеров), пользующихся одним драйвером, механизм разделяемых прерываний работает вполне успешно.

Проявления конфликтов по прерываниям могут быть разнообразными. Сетевая карта не сможет принимать кадры из сети или будет их иногда терять (при этом

она может их успешно посылать); у устройств хранения доступ к данным будет поразительно медленным (иногда можно минутами ожидать, например, появления информации о файлах и каталогах) или даже невозможным; звуковые карты будут молчать или «заикаться»; на видеопроигрывателях изображение будет дергаться и т. д. Конфликты могут приводить и к внезапным перезагрузкам компьютера, например, по приходе кадра из сети или сигнала от модема. Спасением от проблем разделяемости может быть перестановка карт PCI в подходящий слот, в котором конфликты не наблюдаются (это не значит, что их нет). Однако попадаются «подарки разработчиков» интегрированных плат, у которых из нескольких слотов PCI неразделяемая линия прерывания есть только у одного (а то и нет вообще). Такие недуги без скальпеля и паяльника, как правило, не лечатся. Более радикальный способ — переход на сигнализацию прерываний через сообщения (MSI, см. 14.5).

4.5. Прямой доступ к памяти — DMA

Прямой доступ к памяти (Direct Memory Access, DMA) — это обмен между системной памятью (ОЗУ) и устройством, выполняемый без непосредственного участия процессора. Обмен осуществляет *контроллер прямого доступа*. Для устройств ISA в архитектуре PC/AT присутствует централизованный контроллер DMA, совместимый с «исторической» микросхемой i8237A. Для устройств шины PCI (и всех ее «родственников») контроллер является частью устройства — мастера шины (bus master). Штатного централизованного контроллера DMA, как это было в архитектуре ISA, для шины PCI нет[1].

Многоканальный контроллер DMA, программируемый по командам от центрального процессора, присутствует на системной плате PC-совместимого компьютера. Первоначально он использовался устройствами шины ISA, теперь он используется интегрированной традиционной периферией: контроллером НГМД, портами LPT и COM и некоторыми встроенными аудиосредствами. Процессор при обмене по DMA занят только инициализацией контроллера, которая сводится к записи в его регистры нескольких байтов, задающих начальный адрес и размер пересылаемого блока памяти, направление и режим обмена. Затем обмен производят системная шина и контроллер DMA, а инициатором обмена выступает устройство, сигнализирующее контроллеру DMA о своей готовности к обмену. Во время операций DMA процессор может продолжать работу, если выбранный режим обмена не занимает всей пропускной способности шин, задействованных процессором в данный момент (шины памяти, шины PCI, через которые подключается ISA в современных компьютерах). Контроллер DMA можно считать простейшим сопроцессором ввода-вывода, разгружающим центральный процессор от рутинных операций обмена.

В PC/AT доступны 7 каналов DMA: четыре 8-битных (номера 0-3) и три 16-битных (5-7), подключенные к первичному и вторичному контроллерам со

[1] Эмуляция централизованного контроллера, введенная на время перехода от ISA к PCI, в расчет не берется.

ответственно. Канал 4 требуется для каскадирования (соединения контроллеров), канал 2 — для контроллера НГМД. Контроллеры DMA программно совместимы с микросхемами i8237, используемыми в первых моделях PC/XT и AT. Стандартные каналы и адреса регистров приведены в табл. 4.3.

Таблица 4.3. Стандартные каналы прямого доступа к памяти

Номер канала DMA#	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
Разрядность, байтов	1				2 с четного адреса			
Макс. размер блока	64 Кбайт				128 Кбайт, четный			
Граница блока	Кратна 1 0000h				Кратна 2 0000h			
Регистр страниц	8 бит A16–A23				7 бит A17–A23			
Адреса регистров:								
— страниц	087	083	081	082	08F	08B	089	087
— адреса	000	002	004	006	0C0	0C4	0C8	0CC
— счетчика	001	003	005	007	0C2	0C6	0CA	0CE

16-битные каналы DMA 5-7 обеспечивают передачу только 16-битных слов по четным адресам. Они могут быть использованы и интеллектуальными устройствами для *прямого управления шиной* (bus mastering) ISA, при этом контроллер DMA, фактически, лишь играет роль арбитра шины. Интеллектуальный контроллер может выполнять более эффективные процедуры обмена, чем стандартный контроллер DMA. Однако архитектура шины ISA ограничивает доступное пространство памяти областью в 16 Мбайт, что по нынешним меркам маловато. «Заботливые» операционные системы (например, Novell NetWare) для таких адаптеров позволяют под буферы резервировать область в пределах младших 16 Мбайт.

В контроллерах 8237A регистры адресов и счетчики передач являются 16-раз - рядными. Для расширения разрядности адреса применяются внешние регистры страниц, загружаемые программно (и не модифицируемые автоматически). Это сковывает свободу использования контроллера: передача не должна пересекать границ 64-килобайтных (для 8-битных каналов) или 128-килобайтных (для 16-битных) страниц. Стандартные регистры страниц расширяют адрес только до 24 бит (адресуемо только 16 Мбайт ОЗУ). Ограничения на доступную память для режима DMA в новых машинах могут быть сняты применением расширенных регистров страниц, но об этих не совсем стандартных возможностях, конечно же, должно «знать» и программное обеспечение.

Для интерфейса ПУ каждый канал DMA представляется парой сигналов: запрос обмена — DRQx, подтверждение обмена — DACKx#. В операциях по каналу DMA адресуется текущая ячейка памяти, адрес порта не фигурирует, а используется только пара сигналов, соответствующая номеру канала.

Стандартный контроллер DMA на шине ISA с частотой 8 МГц работает медленно, даже в блочных передачах пропускная способность не превышает 1 Мбайт/с для 8-битных каналов и 2 Мбайт/с для 16-битных. Программно-управляемый ввод-вывод, выполняемый инструкциями REP INS/OUTS на той же шине, может работать в два раза быстрее. По этой причине для обмена данными с контролле

рами жестких дисков стандартные каналы DMA использовали редко, предпочитая обмен PIO.

С появлением шин MCA и EISA появился новый контроллер DMA, программно-совместимый с контроллером AT, но имеющий и дополнительные возможности конфигурирования. Он может работать в более производительных режимах (Type A, B, C) со скоростями до 33 Мбайт/с. В системах PCI для обмена с устройствами системной платы возможно использование контроллеров DMA с типом Type F (режим одиночной передачи или передачи по запросу и только с инкрементом адреса).

По стандартным каналам DMA возможно обращение как к системной памяти (расположенной на системной плате), так и к памяти, подключенной к шинам ISA и PCI. Обращение к 8-разрядной памяти, расположенной на адаптерах, допустимо только по 8-битным каналам (преобразование циклов выполнять некому). Однако ПУ, использующие эти каналы, могут располагаться лишь в слотах ISA/EISA или на системной плате (контроллер НГМД, LPT-порт в режиме *ECP* или *Fast Centronics*, аудиокодек). Если эти устройства системной платы используют каналы DMA, то данные каналы становятся недоступными для абонентов шины ISA. Контроллерам ATA, расположенным на системной плате, стандартные каналы DMA не требуются.

На время переходного периода, связанного с «изживанием» шины ISA, потребовалась возможность эмуляции каналов DMA для устройств шины PCI, что было обусловлено требованием совместимости звуковых карт PCI с существующим программным обеспечением, которое во многих случаях работает с аппаратными средствами напрямую, без каких-либо драйверов. Фактический стандарт на программную модель звуковой карты SB 16 подразумевает доставку цифрового потока по каналу DMA. Существуют два механизма эмуляции каналов DMA: PC/PCI и DDMA (см. 14.3). *Механизм PC/PCI* был разработан фирмой Intel для обеспечения возможности использования слотов ISA блокнотными ПК, подключаемыми к док-станции по шине PCI. Альтернативное решение — *механизм DDMA* (Distributed Direct Memory Access — распределенный прямой доступ к памяти). Как известно, контроллеры DMA для шины ISA располагаются на системной плате, и управление несколькими каналами выполняется через одни и те же регистры. DDMA позволяет «расчленить» стандартный контроллер и отдельные его каналы эмулировать средствами карт PCI. Оба этих механизма реализуемы только как часть моста между первичной шиной PCI и шиной ISA, поэтому их поддержка может обеспечиваться (или не обеспечиваться) лишь на системной плате и разрешаться в CMOS Setup. Для других карт расширения (например, адаптеров локальных сетей, контроллеров интерфейсов) задача эмуляции DMA не возникает, поскольку с ними ПО, как правило, работает через драйверы, поставляемые вместе с этими картами.

4.6. Аксессуары системной платы IBM PC

Системная плата первой модели PC содержала несколько функциональных узлов, которые благодаря открытому описанию приобрели надежный статус неприкосновенности, гарантируемый несчетным количеством программ и программных продуктов, их использующих. Эти узлы перечислены далее:

- Схемы предоставления системных ресурсов — памяти, ввода-вывода, прерываний, прямого доступа к памяти (см. 4.5).
- Микросхемы ROM BIOS с программным кодом начального тестирования, запуска и функций ввода-вывода.
- Системный таймер, реализованный на микросхеме 8253 и использовавшийся как генератор запросов регенерации памяти, интервальный таймер и тональный генератор для динамика. В AT те же функции выполняла аналогичная микросхема 8254.
- Системный порт AT, предназначенный для управления немаскируемыми прерываниями (см. 4.4) и звуком.
- Канал управления звуком (PC speaker) — логическая схема, использующая тональный сигнал таймера и программно-управляемые биты системного порта. На машинах AT такой «синтезатор» может исполнять даже записанную музыку и речь.
- Последовательный интерфейс клавиатуры и мыши, реализуемый на AT с помощью микроконтроллера 8042 (в конструктиве BTX его уже нет).
- Память конфигурации и часы-календарь — CMOS RTC — узел, появившийся с AT.

Со временем элементная база системной платы радикально изменилась, все функции отдельных контроллеров взял на себя чипсет, но программная модель этих узлов сохранилась. Рассмотрим их подробнее.

Системный таймер — 8253/8254

Во всех моделях PC используется трехканальный счетчик-таймер, выполняющий генерацию:

- прерываний от системных часов, вызывающих инкремент счетчика системного времени в ячейке 40:006E области переменных (data area) BIOS;
- запросов на регенерацию памяти;
- звуковых сигналов.

В качестве счетчиков-таймеров в XT применялась микросхема i8253, а в AT — более быстродействующая i8254, которая с процессорами 80286 могла работать без тактов ожидания. На современных системных платах те же функции берет на себя чипсет, сохраняя программную совместимость с 8253/8254. Микросхемы 8253 и 8254 представляют собой трехканальные программируемые счетчики-таймеры, функционально почти совпадающие, но имеющие разное быстродействие (со стороны системной шины) и разное назначение выводов. Внутренние счетчики микросхемы имеют разрядность 16 бит, но общение с ними возможно только 8-битными операциями. При этом можно задавать значение только младшего байта счетчика (LSB), только старшего (MSB) или обоих (LSB/MSB), причем сначала передается младший, а потом старший байт. Программирование микросхемы осуществляется записью байтов в управляющий регистр по отдельности для каждого канала. Назначение регистров счетчиков-таймеров иллюстрирует табл. 4.4. Все каналы работают в режиме генера

ции импульсов, в канале 2 используется управляющий вход GATE, высокий уровень которого разрешает счет (формирование выходного сигнала, см. далее). Счет для каналов 0 и 1 разрешен постоянно. Входная частота для всех каналов — 1,19318 МГц.

Таблица 4.4. Регистры счетчиков-таймеров

Порт, R/W	Назначение
040 RW	Счетчик 0 — системные часы (генерация сигнала IRQ0 каждые 54,936 мс — частота 18,206 Гц).
	Режим 011, LSB/MSB, Binary, константа счетчика = 0 (соответствует коэффициенту деления 65536)
041 RW	Счетчик 1 — регенерация памяти (DRQ0 для XT, логика регенерации — для AT).
	Режим 010, LSB, Binary, константа счетчика =12h (18)
042 RW	Счетчик 2 — генератор звука.
	Вход GATE от бита 0 порта B 8255 (061).
	Режим 011, LSB/MSB, Binary, значение счетчика определяет высоту тона
043 W	Управляющий регистр
	Биты 7, 6 — выбор счетчика 0, 1, 2.
	Биты 5, 4 — режим обращения:
	00 — защелка текущего значения;
	01 — LSB — только младший байт;
	10 — MSB — только старший байт;
	11 — LSB/MSB — сначала младший, затем старший байт.
	Биты 3–1 — режим счетчика:
	000 — прерывание по счетчику;
	001 — ждущий мультивибратор (одновибратор, у 8254 несколько отличается от 8253);
	x10 — генератор коротких импульсов заданной частоты;
	x11 — генератор меандра;
	100 — счетчик событий с разрешением;
	101 — счетчик событий с перезапуском.
	Бит 0 — 0 = Bin (двоичный счет), 1 = BCD (двоично-десятичный счет)

Канал управления звуком — PC speaker

Стандартный канал управления звуком PC speaker рассчитан на подключение высокоомного малогабаритного динамика. Логическая схема канала приведена на рис. 4.5. Звук формируется из тонального сигнала от второго канала таймера, работой которого можно программно управлять. Частоту сигнала (тон) можно изменять, программируя коэффициент деления счетчика. Кроме того, разрешая/запрещая формирование сигнала программно-управляемым битом 1 системного порта 61h, можно подавать сигналы определенной длительности. Такой способ формирования звука мало загружает даже процессор 8086/88 и позволяет исполнять незамысловатые мелодии, причем в фоновом режиме, посылая команды из очереди по прерываниям от системного таймера. А с учетом

физиологии слуха (инерционности восприятия) быстрым переключением частот можно достигать эффекта псевдомногоголосия. Таймер генерирует выходной сигнал при высоком уровне на входе GATE2 (при единичном значении бита 0 порта 61h). При низком уровне на входе GATE2 таймер формирует высокий уровень на выходе.

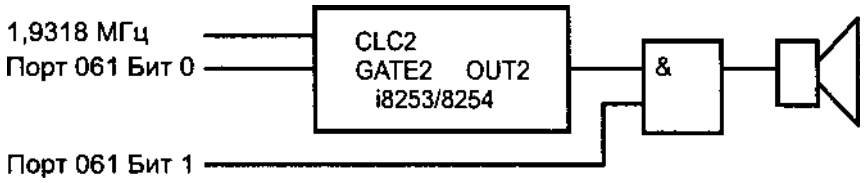


Рис. 4.5. Канал управления звуком

Более интересные звуки можно извлекать, используя принцип широтно-импульсной модуляции, программно осуществляемый через бит 1 порта 61h. При этом на входе GATE таймера должен быть низкий уровень (бит 0 порта 61h должен быть нулевым), чтобы на выходе OUT2 установился высокий уровень. В этом случае динамик играет роль фильтра нижних частот (инерционного звена) демодулятора. Процессоры, начиная с 80286, способны формировать такой поток управляющих сигналов, который позволяет воспроизводить музыкальный или речевой сигнал с качеством карманного приемника, что практически полностью загружает, например 286-й процессор. Кроме того, качество воспроизведения в значительной степени зависит от частотных свойств динамика. Предпочтительнее более крупные динамики, у которых лучше воспроизведение нижних частот, — с ними можно добиться даже разборчивости речи. Возможно также комбинированное управление обоими битами порта 61h одновременно с программированием коэффициента деления счетчика-таймера. Для Windows 3.x/95 существует даже драйвер, обеспечивающий извлечение звука через эти незамысловатые аппаратные средства, но в стандартную поставку ОС он не входит. Теперь для звуковоспроизведения (и звукозаписи) используется аудиокодек (см. главу 12). Роль стандартного звукового канала сводится к подачам гудков при загрузке, идентификации ошибок во время теста POST, когда сообщения на экран вывести еще нельзя, а также к сопровождению сообщений о системных ошибках.

Батарейная память и часы — CMOS Memory, RTC

В AT для хранения информации о конфигурации компьютера имеется специальная микросхема памяти КМОП небольшого объема, питание которой при выключенном компьютере осуществляется от батарейки. В той же микросхеме располагаются часы-календарь, тоже питающиеся от этой батарейки. Эти память и часы — *CMOS Memory and Real Time Clock* (CMOS RTC) — стали стандартным элементом архитектуры PC. Сначала содержимое этой памяти и дату модифицировали с помощью внешней загружаемой утилиты Setup, а позже эту утилиту встроили в BIOS. Микросхемы CMOS RTC имеют встроенную систему

контроля непрерывности питания, отслеживающую также факт разрядки батареи ниже допустимого уровня. Достоверность информации конфигурирования проверяется контрольной суммой.

Таймер синхронизируется от собственного генератора с кварцевым резонатором, как правило, на 32,768 кГц. Таймерная часть содержит:

- часы-календарь (год, месяц, число, час, минута, секунда);
- будильник, подающий сигнал в назначенные час, минуту и секунду;
- генератор меандра, позволяющий формировать запросы прерываний с заданной частотой (как правило, 1024 Гц).

Микросхема CMOS RTC является источником аппаратного прерывания с номером 8, прерывания могут возникать от будильника и генератора меандра, а также после смены времени в часах. Отдельные источники прерывания идентифицируются чтением ячейки 0Ch и разрешаются записью в ячейку 0Bh. Доступ к ячейкам CMOS RTC осуществляется через порты ввода-вывода 070h (индекс ячейки) и 071h (данные). Заметим, что бит 7 порта 70h используется и для блокировки NMI (см. 4.4), так что диапазон адресов памяти CMOS ограничен пределами 0-7Fh. Поскольку эта память имеет быстродействие порядка единиц микросекунд, необходима программная задержка между командами записи адреса и чтения-записи данных. Во время изменения состояния часов данные, считываемые из ячеек 0-9, могут оказаться некорректными. Признаком этой ситуации является единичное значение бита 7 ячейки 0Ah. Для определения момента окончания смены состояния часов можно пользоваться и разрешением соответствующего источника прерывания. Для работы с часами и будильником имеется сервис Int 1Ah BIOS (см. далее), который избавляет от необходимости программирования операций с учетом требуемых задержек, но может выдавать результаты и не с первого раза, если вызов попадет на момент изменения показаний часов. В этом случае вызов приходится повторять.

Назначение ячеек CMOS RTC приведено в табл. 4.5. Обратим внимание на формат представления даты, являющийся одним из источников проблемы «двух нулей» (проблемы 2000 года, кратко называемой и Y2K). Первоначально под год отводился лишь байт 09 (две младшие цифры), старшие подразумевались всегда равными 19. Впоследствии (в эпоху 386-х и 486-х машин) добавили еще один байт для века (32h или 37h), однако он автоматически (аппаратно схемой RTC) при переходе от 1999 к 2000 году инкрементировался не всеми таймерами. Не во всех версиях BIOS на XXI век был правильный календарь — дни недели, указанные в календаре CMOS Setup, могут не совпадать с реальными; неверный отсчет високосных годов может исказить дату.

Таблица 4.5. Стандартное назначение ячеек CMOS RTC

Индекс	Назначение
00h–09h, 32h	*Ячейки RTC в BCD-формате*:
(37 в PS/2)	00 — секунды;
	01 — секунды будильника;
	02 — минуты;

Индекс	Назначение
	03 – минуты будильника;
	04 – часы;
	05 – часы будильника;
	06 – день недели;
	07 – день месяца;
	08 – месяц;
	09 – год (2 младшие цифры);
	32h – век–1 (2 старшие цифры года);
	37h – век–1 (2 старшие цифры года) в PS/2
0Ah	*RTC Status Register A* (статус-регистр):
	Бит 7 – обновление времени: 0 – готов к чтению.
	Биты [6:4] – делитель частоты, для кварца на 32,768 кГц – 010.
	Биты [3:0]=0110 – выходная частота меандра 1024 Гц
0Bh	*RTC Status Register B* (статус-регистр):
	Бит 7 – остановка часов: 0 – нормальный ход.
	Бит 6 – разрешение периодических прерываний: 0 – запрещено.
	Бит 5 – разрешение прерывания от будильника: 0 – запрещено.
	Бит 4 – разрешение прерывания по окончании смены времени: 0 – запрещено.
	Бит 3 – разрешение выходного меандра (см. регистр 0Ah): 0 – запрещено.
	Бит 2 – BCD/BIN формат: 0 – BCD.
	Бит 1 – 12/24-часовой режим: 1 – 24-часовой.
	Бит 0 – зимнее/летнее время: 0 – переключение запрещено
0Ch	*RTC Status Register C – чтение флагов идентификаторов прерывания*:
	Бит 7 – IRQF (общий запрос прерывания).
	Бит 6 – PF (периодические прерывания).
	Бит 5 – AF (прерывание от будильника).
	Бит 4 – UF (прерывание по окончании смены времени).
	Биты [3:0] – зарезервированы
0Dh	*RTC Status Register D*:
	Бит 7 – питание: 1 – норма, 0 – батарея разряжена.
	Биты [6:0] – зарезервированы
0Eh	*POST Diagnostic Status Byte – диагностический байт состояния POST* :
	Бит 7 = 1 – Power Lost (терялось питание CMOS).
	Бит 6 = 1 – Checksum Bad (ошибка контрольной суммы CMOS).
	Бит 5 = 1 – Bad config (ошибка конфигурации).
	Бит 4 = 1 – RAM Size Error (несоответствие размера ОЗУ, определенного тестом, записи в CMOS).
	Бит 3 = 1 – HDD Error (ошибка при инициализации жесткого диска).
	Бит 2 = 1 – Time Valid (нет формальной ошибки часов-календаря – например, 30 февраля, 25 часов).
	Биты [1:0] – зарезервированы

продолжение

Таблица 4.5 (*продолжение*)

Индекс	Назначение
0Fh	*Shutdown Code* — используется POST для определения предыстории останова: 00 — аппаратный или программный сброс; 01 — размер памяти определен; 02 — тест памяти прошел; 03 — тест памяти выявил ошибку; 04 — тест POST завершен, идет загрузка системы; 05 — JMP FAR [0:0467h] с инициализацией контроллера прерываний; 06 — тест защищенного режима прошел; 07 — тест защищенного режима выявил ошибку; 08 — ошибка при определении размера памяти; 09 — перемещен блок Extended Memory (INT15h); 0A — JMP FAR [0:0467h] без инициализации контроллера прерываний; 0B — используется 80386
10h	*Типы НГМД*: Биты [7:4] — дисковод A; Биты [3:0] — дисковод B: 0 — нет, 1 — 360 Кбайт, 2 — 1,2 Мбайт, 3 — 720 Кбайт, 4 — 1,44 Мбайт
11h	*Зарезервирован*
12h	*Типы НЖМД*: Биты [7:4] — привод 0; Биты [3:0] — привод 1: 0 — нет, 1–Eh — типы 1–14, Fh — тип в байте 19h (для второго привода — в 1Ah)
13h	*Зарезервирован*
14h	*Установленное оборудование*: Биты [7:6] — количество НГМД: 00 = 1, 01 = 2. Биты [5:4] — тип первичного видеоадаптера: 00 — EGA или VGA; 01 — CGA, 40 столбцов; 10 — CGA, 80 столбцов; 11 — MDA, 80 столбцов. Биты [3:2] — зарезервированы. Бит 1 = 1 — есть математический сопроцессор. Бит 0 = 1 — есть НГМД
15h–16h	*Размер базовой памяти*, Кбайт (Low/High) 0280h — 640K
17h–18h	*Размер дополнительной памяти*, Кбайт (Low/High)
19h, 1Ah	*Расширенный тип дисков C, D* (в PS/2 — зарезервированы)
1Bh–2Dh	*Зарезервированы*
2Eh–2Fh	*Контрольная сумма CMOS с 10h по 20h* (High/Low)
30h–31h	*Реальный размер расширенной памяти*, Кбайт (Low/High)
32h–33h	*В PS/2 контрольная CRC-сумма CMOS с 10h по 31h* (High/Low)
33h	*Флаги POST*: Бит 7 — наличие 128 Кбайт ОЗУ под границей 1 Мбайт: 1 — есть, доступна теневая память. Бит 6 — флаг Setup: 1 — первая загрузка после выполнения Setup; обычно установлен 0

Индекс	Назначение
34h–3Fh	*Зарезервированы* (место привязки ПО к машине)
38h–3Fh	*В PS/2 пароль, доступ по несуществующим адресам 78h–7Fh*
40h–7Fh	*Используется для хранения настроек чипсета системной платы*, назначение ячеек не стандартизовано

Свободные ячейки CMOS RTC иногда использовали для привязки ПО к конкретному компьютеру (системной плате). Эта привязка выполняется в процессе установки ПО, и если не сохранить образ CMOS на диске, то при разрушении информации в CMOS (например, из-за разряда батарейки) пользователь потеряет право на использование ПО. По этой причине такой способ привязки нельзя считать корректным, поскольку изготовитель ПО ответственности за батарейку на себя, естественно, не берет.

Системная поддержка таймеров — Int 1Ah Int 15h BIOS

Сервисы Int 1Ah *BIOS* позволяют считывать и модифицировать значения системного *таймера*, *даты* и *времени*, а также служат для установки *будильника* часов реального времени CMOS RTC. Перечисленные ниже номера функций указываются при вызове в регистре AH:

- AH = 0 — чтение системного таймера (двойного слова по адресу 40:006Eh в области переменных BIOS), инкрементируемого по прерываниям от канала 0 счетчика-таймера 8253/8254 примерно раз в 55 мс. Таймер обнуляется при выполнении теста POST после аппаратного сброса. Функция возвращает значение таймера, в CX — старшую часть, в DX — младшую. AL = 0, если за последние 24 часа не было переполнения таймера. В современных версиях регистр AL возвращает счетчик переполнений таймера, хранящийся в ячейке 40:0070h (в старых версиях содержимое AL было флагом, а не числом).
- AH = 1 — установка системного таймера (CX — старшая часть, DX — младшая) и сброс флага (счетчика) переполнения таймера в ячейке 40:0070h. В случае ошибки устанавливается флаг CF = 1.
- AH = 2 — чтение времени из RTC. Возвращает в упакованном BCD-формате час (в регистре CH), минуту (CL), секунду (DH) и признак коррекции летне- го/зимнего времени (DL = 1 — коррекция, DL = 0 — нет коррекции). Признаком успешной операции является флаг CF = 0. Во избежание ошибок некоторых версий BIOS при вызове флаг CF должен быть сброшен.
- AH = 3 — установка времени в RTC (назначение регистров и признак результата аналогичны функции 2).
- AH = 4 — чтение даты из RTC. Возвращает в упакованном BCD-формате век (в регистре CH), две старшие цифры года (CL), месяц (DH) и день (DL). Признаком успешной операции является флаг CF = 0. Во избежание ошибок некоторых версий BIOS при вызове флаг CF должен быть сброшен.

- AH = 5 — установка даты в RTC (назначение регистров и признак результата аналогичны функции 4).
- AH = 6 — установка времени срабатывания будильника RTC. Возвращает в упакованном BCD-формате час (в регистре CH), минуту (CL) и секунду (DH). Если будильник уже установлен, переустановка не производится и возвращается флаг CF = 1. При срабатывании будильник вызывает прерывание Int 4Ah.
- AH = 7 — отмена установки будильника.

Функции Int 15h *BIOS* позволяют программировать *таймер CMOS RTC* — вводить задержку или запускать таймер установки флага, указывая время в микросекундах (CX — старшее слово, DX — младшее). Нулевое значение интервала не вызывает никаких действий. Достижимое разрешение в зависимости от производительности ПК может достигать единиц миллисекунд, максимальная выдержка — около 70 часов. Перечисленные ниже номера функций указываются при вызове в регистре AH или AX:

- AH = 86h — задержка на заданное время. Управление возвращается вызвавшему процессу только через указанный интервал. По окончании задержки устанавливается бит 7 в ячейке BDA 0040:00A0. Таймер может оказаться занятым, тогда вызов сразу возвратит флаг CF = 1 (при успехе возвратится флаг CF = 0, а в AL окажется маска, записанная в 8259A#2).
- AX = 8300h — запуск таймера, устанавливающего после указанной задержки бит 7 в ячейке, заданной регистрами ES:BX. При успешном запуске CF = 0; если таймер занят (он один) — CF = 1 и AL = 0. Управление возвращается процессу сразу, а флаг будет установлен через заданное время. Перед завершением программа, запускавшая таймер, должна его сбросить функцией 8301h (во-первых, чтобы освободить, во-вторых, чтобы снять «адскую машинку», которая неожиданно сама может изменить значение ячейки памяти, вполне возможно, уже задействованной другим, «ничего не подозревающим» процессом).
- AX = 8301h — сброс того же таймера.

4.7. Распределение системных ресурсов

Для того чтобы программы могли взаимодействовать со *своими* устройствами, не мешая другим (и не получая от них помех), все системные ресурсы — адреса памяти и ввода-вывода, запросы прерываний и каналы DMA — должны быть бесконфликтно распределены между устройствами, подключенными к системной шине.

Для адресов памяти и портов ввода-вывода бесконфликтность означает, что диапазоны соответствующих адресов всех имеющихся устройств не должны перекрываться. Это в первую очередь касается адресов, по которым выполняется чтение. Если устройства, конфликтующие по чтению, находятся на одной физической шине, то результат чтения из-за электрического конфликта оказывается неопределенным. Если конфликтующие устройства находятся на разных шинах, то будет прочитаны данные только одного устройства, но какого имен

но — зависит от настройки мостов, соединяющих шины. Конфликт по адресам для записи часто сознательно используют для одновременной передачи информации в несколько устройств (например, в PnP ISA, см. [6]). Информация при этом не искажается. Однако незапланированные конфликты по записи могут приводить к неожиданным побочным эффектам в работе устройств, не ожидающих данной записи. Для самого главного ведущего устройства — центрального процессора — безразлично, к какой из шин подключено устройство: он задает только адрес и тип операции. Операции записи могут выполняться широковещательно — распространяться по всем шинам. Операции чтения маршрутизируются — фактически, чтение по конкретному адресу памяти или порта обычно выполняется только с одной из шин. В иерархии шин PCI мосты выполняют маршрутизацию для всех транзакций.

Для линий запросов прерываний бесконфликтность трактуется несколько сложнее. В классической *системе ISA* одну линию запроса может использовать только одно устройство, все остальные варианты — конфликтные. В *системах PnP ISA* (имеющих PnP BIOS, более гибко программируемый контроллер прерываний и устройства ISA PnP) при корректных настройках устройства PnP аппаратно могут использовать разделяемые (общие для нескольких устройств) линии прерываний (см. 4.4). На обычные карты (устройства) ISA эта возможность, как правило, не распространяется. Однако и для устройств PnP возможны конфликты программ, работающих с этими устройствами, если в них не заложена возможность разделяемости прерываний. В *системах с PCI* разделяемость прерываний аппаратно предусмотрена, но опять-таки возможны программные конфликты (и некорректно спроектированные карты). В комбинированных системах ISA/PCI доступные линии запросов прерываний делятся между устаревшими (legacy) устройствами ISA и устройствами PnP ISA и PCI, во второй группе разделяемые прерывания, в принципе, допустимы (но при «правильном» ПО).

Для каналов DMA (контроллеров 8237A) бесконфликтным, как правило, является лишь монопольное использование канала одним устройством (хотя бывают редкие исключения). На устаревших устройствах каналы выбираются двумя джамперами — один для линии DRQx, другой для DACKx#. Естественно, они должны устанавливаться согласованно, на один и тот же номер канала.

Периферийные устройства могут быть встроены в системную плату, а также устанавливаться в слоты шин расширения. Системные ресурсы должны распределяться между всеми этими устройствами. В распределении всегда имеются относительно неизменная часть (устройства системной платы, установленные ее изготовителем) и переменная часть, определяемая составом карт, установленных пользователем. Настройкой CMOS Setup могут быть заданы ресурсы устройств системной платы, а часть из них может быть даже отключена, если вместо них используются адаптеры, установленные в слоты расширения. Правда, бывают случаи, когда штатное отключение (настройкой CMOS Setup) не помогает: отключают обычно неисправную периферию, но «сгореть» она может и вместе со своим «выключателем». В этом случае приходится неисправную пе

риферийную микросхему отключать физически (отпаивать; для микросхем CMOS по крайней мере отпаять вывод питания +5 В).

Ресурсы устанавливаемых карт задаются либо вручную, либо автоматически, в зависимости от возможностей шин, карт расширения и BIOS. Цель полной автоматизации — достичь идеала PnP (Plug-and-Play — «вставляй и играй»), когда от пользователя требуется лишь механически установить новое устройство, а дальше все распределения и установки драйверов выполняются без его участия. Однако у скептиков есть и другая расшифровка акронима PnP — Plug-and-Pray («включай и... молись»), имеющая под собой вполне реальную основу.

Наиболее распространенными для подключения карт расширения PC являются шины ISA и PCI. В шину PCI изначально были заложены возможности автоматического конфигурирования устройств, и она поддерживает механизм PnP в полном объеме (см. 14.7). Шина ISA не имела механизмов автоматического конфигурирования и распределения ресурсов, так что все заботы по конфигурированию устанавливаемых адаптеров и разрешению конфликтов ложились на пользователя. Задача конфигурирования осложнялась и из-за отсутствия общих средств автоматической передачи установленных параметров прикладному и системному программному обеспечению. После конфигурирования адаптеров, выполняемого обычно переключением джамперов (хорошо, если имелась документация с их описанием), установленные параметры заносились в какие-либо конфигурационные файлы, специфичные для каждого программного продукта. Для смены конфигурации (а такая необходимость обычно возникает только в процессе использования устройств или при добавлении новых) всю кропотливую работу по конфигурированию приходилось проводить повторно. При этом, естественно, возможны ошибки. Неподготовленному пользователю эта работа может показаться непосильной, и он зовет на помощь профессионала.

Некоторое облегчение конфигурирования принесло применение в адаптерах энергонезависимой памяти (NVRAM или ее разновидности — EEPROM), хранящей параметры, в том числе параметры использования системных ресурсов. Конфигурирование этих адаптеров выполняется программно специальной утилитой, а не с помощью джамперов. Отсюда и два их названия: *Software Con- figured* (программно конфигурируемые) или *Jumperless* (свободные от джамперов). Сейчас большинство карт ISA поддерживают спецификацию PnP ISA, благодаря чему во многих случаях с пользователя действительно снимаются заботы по конфигурированию.

Помимо «явно полезных» устройств конфигурированию подлежат и мосты PCI, соединяющие все шины современных ПК. При конфигурировании мостов им указывается распределение системных ресурсов по шинам, которые они связывают. Таким образом задаются пути транслирования управляющих сигналов по шинам с соответствующим управлением буферами данных. Конфигурирование мостов, как правило, происходит незаметно для пользователя, хотя некоторыми параметрами (выделением определенных ресурсов) можно управлять с помощью CMOS Setup.

4.8. Функционирование компьютера

Теперь, имея представление об архитектуре IBM PC-совместимого компьютера, рассмотрим, что в нем происходит во время работы — от включения и до выключения.

Начальный запуск и самотестирование

При включении блок питания вырабатывает сигнал аппаратного сброса, приводящего все узлы в исходное состояние. Этот же сигнал вырабатывается и при нажатии кнопки Reset. Во время действия сигнала аппаратного сброса процессор пассивен — он не управляет системной шиной. Процессор подготавливается к работе, воспринимая со своих выводов сигналы, задающие его конфигурацию (коэффициент умножения, роль в многопроцессорных системах и некоторые другие параметры). Внутренний кэш очищается, регистры (не все) приводятся в определенное состояние. Сигнал сброса поступает на все устройства (контроллеры и адаптеры), нуждающиеся в переводе в исходное состояние и находящиеся на системной плате или подключаемые к шинам расширения. После окончания сигнала процессор по определенному адресу выбирает из памяти и исполняет первую инструкцию — управление передается на точку входа в программу инициализации компьютера (POST). Первым делом необходимо выполнить инициализацию процессора — установить значения некоторых регистров. После сброса процессор всегда начинает работу в реальном режиме (совместимом с 8086/88). Далее происходят проверка работоспособности и инициализация подсистем компьютера. Эта «раскрутка» выполняется в несколько этапов, причем постепенно в работу вовлекаются протестированные подсистемы. Поначалу программа может смело пользоваться только регистрами процессора и постоянной памятью (которые тоже желательно проверить, например, подсчитать контрольную сумму). Если ПЗУ исправно, можно двигаться дальше, в противном случае лучше остановиться. Пока неизвестна работоспособность ОЗУ, оперативной памятью пользоваться нельзя и, следовательно, недоступны вызовы процедур и обработка прерываний (вызвать-то процедуру можно, а вот возврат не гарантируется, поскольку адрес возврата берется из стека, то есть из ОЗУ). Далее инициализируется ОЗУ (программируются регистры чипсета, заведующие настройкой контроллера памяти и регенерацией) и выполняется тестирование небольшого блока в начале ОЗУ. Если тест проходит успешно, то для дальнейшей работы уже можно пользоваться и вызовами, и прерываниями (не забыв проинициализировать таблицу прерываний), и задействовать память для хранения переменных — в таком окружении работать гораздо удобнее. Затем можно проинициализировать и протестировать дисплейный адаптер, и дальнейшая раскрутка пойдет уже с «ожившим» экраном. Далее тестируют ОЗУ в полном объеме, определяют наличие контроллеров и адаптеров, инициализируют их и тестируют. После этого программа POST узнает реальную конфигурацию компьютера и готова к загрузке операционной системы. Векторы прерываний, за которые отвечает BIOS, проинициализированы — ими можно пользоваться. В этот момент можно войти в меню

встроенной утилиты конфигурирования CMOS Setup и изменить настройку различных подсистем компьютера. После окончания работы этой утилиты тест POST приходится выполнять снова — конфигурация может стать уже иной. Программа POST завершается вызовом процедуры начальной загрузки. Подробнее о работе и диагностических сообщениях POST можно узнать в 5.2.

Для процессоров, начиная с шестого поколения, у POST есть дополнительная забота — загрузить «заплатки» микропрограмм самого процессора (см. 7.4). Эти заплатки позволяют исправлять некоторые ошибки, выявленные в процессорах определенных моделей и партий выпуска. Без них основные функции процессор, конечно, выполнять будет, но в современных сложных операционных системах и приложениях, как говорится, возможны варианты. Информация о том, какие заплатки требуются конкретному процессору, и сами заплатки хранятся в ROM BIOS. Если BIOS про установленный процессор (какой именно процессор установлен, POST может определить программно — см. 7.6) «ничего не знает», то и заплатки не загрузит. По этой причине может потребоваться обновление версии BIOS, что для современных системных плат выполняется довольно просто (если производитель своевременно выкладывает образ BIOS на своем веб-сайте).

В процессе выполнения программа POST отыскивает модули расширений ROM BIOS — сначала в известном месте (по адресу C0000h) ищется ROM BIOS графической карты, а позже ищутся модули карт расширения (в зоне C8000-F4000h). Найденные модули инициализируются, при этом могут переопределяться векторы прерываний, по которым вызываются некоторые сервисы BIOS.

К моменту окончания теста POST все стандартные устройства (клавиатура, дисплей, диски, порты) приводятся в состояние готовности к работе в стандартном режиме по умолчанию, часть настроек может выполняться в соответствии с выбранными установками CMOS Setup. Список обнаруженных устройств и их основные параметры могут отображаться в таблице на экране, но возможна и такая настройка CMOS Setup, при которой экран будет пустым до появления логотипа (или текстового сообщения) загружаемой ОС.

Загрузка ОС и прикладных программ

Последним шагом программы POST является выполнение *процедуры начальной загрузки* (bootstrap loader), которая вызывается как программное прерывание Int 19h BIOS (см. 5.2). Ее задача довольно скромная — с выбранного устройства загрузить в память один блок (сектор) данных, и если он похож на загрузчик — передать ему управление. Напомним, что «передать управление» означает выполнить инструкцию перехода на адрес точки входа в программу, загруженную в оперативную память. На этом деятельность POST заканчивается, и компьютер отдается во власть загружаемому ПО.

Процедура определяет первое готовое устройство из списка разрешенных и доступных загрузочных устройств. Загрузочным устройством может быть

НГМД, жесткий диск, диск LS-120, CD/DVD, флэш-память, сетевой адаптер. Список устройств, разрешенных для загрузки, а также порядок их опроса задаются параметрами CMOS Setup. При традиционном порядке опроса (A, C) сначала анализируется готовность НГМД попыткой прочитать загрузочную запись дискеты — первый сектор нулевой дорожки. Если дискета отсутствует (при этом дисковод не может прочитать никаких данных), то процедура переходит к попытке чтения главной загрузочной записи жесткого диска C. Если сектор с дискеты прочитать удалось, то по определенному признаку в его содержимом процедура определяет, имеется ли в нем загрузчик ОС. Если загрузчик имеется, то ему передается управление для загрузки операционной системы. Если в считанном секторе загрузчик не обнаружен, то компьютер останавливается с сообщением Non-system disk or disk error (несистемный диск или ошибка чтения), после которого остается лишь установить системную дискету или, наоборот, извлечь дискету и сделать «комбинацию из трех пальцев» — нажать клавиши Ctrl+Alt+Del. Это приведет к повторному запуску теста POST, но в сокращенном варианте, который завершится вызовом прерывания Int 19h — следующей попыткой загрузки. Такой способ перезапуска (перезагрузки) называется «теплым» (warm). Повторный запуск POST можно выполнить и «холодным» (cold) способом, нажав кнопку Reset, но при этом POST будет выполняться полностью, как после включения питания, что несколько дольше.

Загрузка с жесткого диска (в нашем случае — если не обнаружена готовность НГМД) тоже начинается с попытки чтения первого сектора нулевой головки нулевого цилиндра, но здесь ищется главный загрузчик, или главная загрузочная запись (Master Boot Record, MBR), с таблицей разделов диска (см. 9.6). Этот загрузчик должен найти описатель активного раздела, загрузить в память его первый сектор и, убедившись в том, что в нем находится загрузчик ОС, передать ему управление. Загрузчик ОС и должен загрузить операционную систему из выбранного (активного) раздела в память и передать ей управление. Главный загрузчик может являться и средством для выбора загружаемой ОС (boot manager) по желанию пользователя (из меню). На пути этого двухступенчатого загрузчика могут встречаться различные препятствия: не окажется MBR на штатном месте, не найдется описатель активного раздела, не найдется загрузчик в том секторе, который посчитали первым сектором активного раздела. Неприятности могут быть и у самого загрузчика ОС — он может не найти файлов, требуемых для загрузки ОС (это относится и к загрузчику дискеты). Наконец, может оказаться физически невозможным чтение какого-либо из требуемых секторов. Все эти препятствия отражаются соответствующими консольными сообщениями и приводят к остановке компьютера, после которой его можно перезапустить только «теплым» или «холодным» способом. Подробнее процедура загрузки описана в 5.2. Главный загрузчик универсален для всех операционных систем. Загрузчики активных разделов, как и загрузчик на дискете, ориентированы на загрузку только своих операционных систем. Если загрузчику не удается найти и загрузить файлы ОС, он, скорее всего, остановит компьютер с сообщением вида Missing Operation System (отсутствует операционная система). Раньше был шанс встретить машину, которая при невозможности загрузки ОС запускает интерпретатор языка BASIC, «зашитый» в ее ПЗУ. И сейчас некото

рые компьютеры при невозможности загрузки ОС «жалуются» на отсутствие интерпретатора BASIC, но и его присутствие дело бы не улучшило. Заметим также, что загрузчики проверяют то, что они загружают, весьма условно, так что возможна нештатная ситуация, когда под видом очередного фрагмента загрузчика или файлов ОС загружаются бессмысленные (в данном контексте) данные и этому загруженному «мусору» передается управление. В этом случае компьютер, скорее всего, «зависнет» безо всяких сообщений (а может и вывести на консоль абракадабру), и остается лишь попытаться его перезагрузить — возможно, с другого устройства. Такое поведение компьютера может быть признаком неисправного ОЗУ, дисковода и его контроллера, а также шин, их соединяющих.

Конечно, чаще всего загрузчику все же удается загрузить файлы операционной системы и передать управление по известному ему адресу памяти (иначе число компьютеров и их пользователей не росло бы так стремительно). Компьютер, у которого не загружается ОС, для рядового пользователя является лишь аппаратом, нуждающимся в ремонте. Для «продвинутого» пользователя — это поле деятельности, направленной на «оживление», то есть обеспечение возможности загрузки ОС. Если на «умершем» компьютере ранее занимались созидательной деятельностью (более серьезной, чем строительство городов в «борьбе империй»), то придется приложить усилия также и для сохранения информационного хозяйства, «нажитого непосильным трудом». Как именно хранится информация на дисках и каким образом она становится доступной, рассказывается в главе 9.

Как уже отмечалось, *последовательность опроса устройств*, с которых делается попытка загрузки операционной системы, задается параметрами CMOS Setup. «Классической» является последовательность A, C, и в старых компьютерах не было способа ее изменения. Позже появилась возможность ее изменения на обратную (C, A), а затем в нее стали включать и CD-ROM, и новые устройства LS120, и флэш-память, и даже обнаруженные устройства удаленной загрузки (Remote Program Loader, RPL), позволяющие загружать ОС по сети. Если первое по порядку загрузочное устройство не готово (не установлен сменный диск, на жестком диске нет активного раздела), то процедура обращается к следующему по списку устройству. Таким образом, современные компьютеры могут загружаться не только с первого жесткого диска (C), но и с другого, если на нем имеется активный раздел. Правда, активизировать раздел с помощью утилиты FDISK, как это обычно и делается, можно только на первом физическом жестком диске (таково ее ограничение), так что для этого приходится временно переставлять устройства. Механизм упорядочивания начальной загрузки подробнее рассмотрен в 5.2.

Для регулярной работы *попытка загрузки с гибкого диска вредна* по ряду причин, самой безобидной из которых является то, что на опрос готовности накопителя затрачивается излишнее время. Хуже то, что случайно оставленная в накопителе дискета может содержать вирус в загрузочном секторе, и попытка загрузки с такой дискеты, даже если на ней нет операционной системы, передаст управление коду этого вируса. При незагруженной операционной системе

вирусу аккуратно залезть в файловую систему затруднительно, но поселиться в загрузочном секторе жесткого диска (ищется легко — «первое место в первом ряду») он не поленится. Дальнейшее развитие событий зависит от злобности вируса и опытности пользователя (пока операционную систему не загрузили, этот вирус из загрузчика не вылезет). Запрет попытки загрузки с дискеты служит и средством защиты от несанкционированного доступа к данным. Если загрузка операционной системы закрыта паролем, а разрешена загрузка с дискеты, то, загрузив операционную систему, взломщик получит программный доступ к жесткому диску — по крайней мере, на физическом уровне. При заданной последовательности C, A загрузчик активного раздела жесткого диска не допустит передачи управления загрузчику с дискеты. Конечно, если изменение последовательности производится в целях защиты от несанкционированного доступа, вход в CMOS Setup также нужно закрыть паролем.

Загруженная *операционная система* выполняет инициализацию подведомственных ей программных и аппаратных средств. Она добавляет новые сервисы, вызываемые, как правило, тоже через программные прерывания, и расширяет некоторые сервисы BIOS. Под управлением операционной системы загружаются и исполняются *пользовательские приложения* и разные вспомогательные процессы и утилиты. Операционная система ведает распределением всех ресурсов компьютера — памяти (как оперативной, так и пространства на устройствах хранения данных), процессорного времени (в многозадачных системах), периферийных и коммуникационных устройств. Она же предоставляет пользователям интерфейс, с помощью которого пользователи запускают приложения, настраивают параметры ОС и выполняют иные действия.

Приложения и утилиты располагаются в виде файлов на устройствах внешней памяти — локальных и сетевых дисках или иных устройствах хранения. Пользователь запускает интересующие его приложения, указывая имя требуемой программы (или командного файла) с клавиатуры в ответ на приглашение (prompt) ОС или же «щелкая» на представляющем его значке. В *однозадачных системах* (типа MS-DOS) приложения запускаются поочередно: следующее приложение может запускаться только после завершения работы предыдущего. Правда, можно запускать еще так называемые резидентные программы — после запуска часть их программного кода и требуемая область данных «поселяются» в оперативной памяти, и ОС выдает приглашение для ввода очередной команды пользователя. Резидентные программы часто обозначают как *TSR* (Terminate and State Resident — завершить, оставив резидентно) по названию соответствующего сервиса DOS. Резидентными могут быть, например, программа фоновой печати, перекодировщик клавиатуры, калькулятор, отладчик, антивирусная программа и др. Резидентные программы «съедают» некоторый объем оперативной памяти, и программам, загружаемым после них, может не хватить свободной оперативной памяти. В ряде случаев резидентные программы удается разместить вне дефицитной стандартной памяти (см. 4.2). Управление резидентным программам передается по каким-либо событиям: аппаратным прерываниям (от устройств или таймера), вызовам определенных сервисов DOS или BIOS. Процессы в однозадачных системах могут быть и вложенны

ми — одно приложение, не будучи завершенным, может загрузить дочернее приложение и передать ему управление. В свою очередь, и дочернее приложение может быть родительским процессом для следующего вложенного. По завершении дочернего процесса управление возвращается родительскому процессу, так что активным все-таки является лишь один процесс. Возможности порождения процессов ограничиваются и малым объемом стандартной памяти, предоставляемой в распоряжение программам.

В многозадачных системах (Windows, Unix, Linux) одновременно могут запускаться множество программ (процессов), и они могут работать псевдопараллельно (см. 7.1). Если процессоров (физических или логических, см. 7.5) несколько и используется ОС с мультипроцессорной поддержкой, то процессы будут работать действительно параллельно. Многозадачные системы работают в защищенном режиме процессора и используют виртуальную память, что позволяет всем запущенным процессам расходовать даже больше оперативной памяти, чем реально установлено. Конечно же, в этих системах можно запускать и резидентные программы (процессы), а также порождать дочерние процессы.

Как пользователь взаимодействует со своей программой, мы здесь рассматривать не будем — на эту тему про каждую «фирменную» программу написано множество книг, а что требуют «самодельные» программы, зачастую знает (и то не всегда точно) только создатель. Программы могут обращаться к файлам данных и параметров: считывать файлы, выполнять какую-то обработку, записывать результаты. Для того чтобы не потерять измененные данные, программу следует корректно завершать (закрывать) предусмотренными в ней средствами. Принудительное завершение программы средствами ОС может вести к потере несохраненных данных.

По окончании работы современные ОС тоже требуют корректного *закрытия* (shutdown) — завершения работы приложений и сохранения информации, необходимой пользователю и операционной системе, на энергонезависимых носителях (на диске). Только после этого компьютер можно выключать или выполнять «теплый» или «холодный» перезапуск. «Комбинация из трех пальцев» (Ctrl+Alt+Del) для старых ОС (типа MS-DOS), как правило, безусловно приводит к «теплому» перезапуску компьютера с потерей всех несохраненных данных. Правда, программы могут перехватить обработчик прерывания `Int 19h` и предложить пользователю подтвердить свои намерения. В ОС защищенного режима неуместный перезапуск чреват неприятностями, и комбинация Ctrl+Alt Del обычно вызывает запрос о намерениях пользователя с предупреждением о возможных последствиях. Сервер ОС NetWare вообще игнорирует эту комбинацию. Если выключить или сбросить (в смысле нажать кнопку Reset) компьютер до завершения работы ОС, могут появиться проблемы при последующей загрузке: потеря несохраненных пользовательских данных или системных настроек, потеря фрагментов дисковой памяти (кластеров) и даже разрушение ОС, требующее ее переустановки. Компьютеры в конструктиве ATX (и BTX), а также портативные умеют сами выключать питание по завершении работы ОС, что упрощает правила поведения пользователя.

В старых ОС (MS DOS) «комбинация из трех пальцев» (Ctrl+Alt+Del) может приводить к малоприятным «чудесам», если используются контроллеры дисков или сетевые карты с прямым управлением шиной PCI или ISA. «Чудеса» обусловлены тем, что при «теплом» перезапуске не вырабатывается сигнал аппаратного сброса устройств, и мастер шины будет продолжать обмен данными между своим устройством и областью памяти, на которую он был предварительно «нацелен». Однако при перезагрузке в этой области могут оказаться уже совершенно не те данные. Если мастер выполняет запись в память (чтение диска или прием кадра из сети), неожиданные данные могут затереть область памяти, отведенную для других целей. Последствия могут быть различными — некорректная работа какой-либо загруженной программы, «зависание» и т. п. Если мастер выполняет чтение памяти (запись на диск или передачу кадра в сеть), то он может послать не те данные. В случае записи на диск последствия могут быть катастрофическими, если, например, на диск вместо таблицы размещения файлов запишется случайная информация. Передача ложных данных в сеть тоже нежелательна.

«Засыпание» и «пробуждение» компьютера

В работе персонального компьютера после загрузки ОС и запуска приложений часто возникают довольно длительные паузы (когда пользователь отвлекается на свои дела). В это время вхолостую расходуются электроэнергия (выделяется тепло), а также жизненные ресурсы некоторых устройств (например, зря выгорает люминофор монитора). Чтобы сократить эти напрасные потери, уже давно в компьютер ввели средства *управления энергопотреблением* (power management), а затем и улучшили их, обеспечив *расширенное управление энергопотреблением APM* (Advanced Power Management). Цель APM — переводить в «спящее» состояние (sleep) устройства, не требующиеся для работы в данный момент, и «будить» их (wake up) по первому требованию, по возможности незаметно для приложений и пользователей. Менеджер APM представляет собой часть системного ПО (BIOS и ОС), для работы ему требуется поддержка со стороны устройств. В спячку разные устройства можно вводить по-разному: у монитора можно погасить лучи и даже остановить генераторы развертки, у винчестера — остановить шпиндель, у процессора — остановить внутреннее тактирование или понизить эффективную тактовую частоту, память можно перевести в режим саморегенерации. С самого начала подобие спячки применяют для НГМД — мотор останавливается, если в течение 2 с нет обращений к дискете. Заснувшее устройство должно реагировать на свои интерфейсные сигналы, чтобы проснуться по сигналу от APM или по команде обычного обращения. В спящий режим устройства могут переходить по таймерам, отсчитывающим время от последнего обращения, прекращения активности пользователя (отсутствия сигналов от устройств ввода), а также по команде пользователя. Пробуждение системы выполняется по какому-либо внешнему событию — нажатию клавиши на клавиатуре (или специальной кнопки на системном блоке), движению мыши, специальному обращению по локальной сети (wake on LAN) или телефонному вызову модема. Конечно же, пробуждение устройства занимает

некоторое время, требующееся на разгон двигателя, нагрев трубки монитора и тому подобные операции, и первое обращение к заснувшему устройству выполняется относительно долго. Иногда это просто раздражает темпераментного пользователя, а в ряде случаев может приводить к неприятным последствиям. Например, при записи компакт-дисков (CD-R и CD-RW), которую в произвольный момент прерывать нельзя (испортится болванка), засыпание источника данных (винчестера или привода CD-ROM) неуместно. Пользователи, не желающие возиться с настройкой APM (установкой параметров CMOS Setup и ОС), часто просто запрещают работу APM, в результате их компьютер всегда готов к работе, но, возможно, потребляет больше энергии, чем необходимо. Вопрос энергосбережения стоит особо остро для мобильных (блокнотных) ПК при питании от аккумуляторов. Помимо энергопотребления, APM снижает шум работающего компьютера: шумят вентиляторы блока питания, процессора и видеокарт; шумят винчестеры и приводы CD и DVD, особенно высокоскоростные. Если охлаждаемые устройства переходят в энергосберегающий режим, то можно снизить и скорость вращения вентиляторов, а следовательно, и их шум. Современные винчестеры позволяют регулировать уровень шума: появляется выбор между быстрой, но шумной работой и тихой, но менее производительной.

В плане управления потреблением различают следующие состояния устройств (в порядке «углубления сна»):

- *On* — активная (нормальная) работа, полное потребление, максимальная производительность;
- *Standby* — отключение питания некоторых узлов, с возможностью быстрого (порядка секунды) перехода в активное состояние;
- *Suspend* — более глубокое отключение (например, строчной развертки и накала трубки монитора), выход из которого (*resume*) требует единиц — десятков секунд;
- *Off* — отключение питания всех узлов, кроме цепей, обеспечивающих последующее включение по команде.

Для ряда устройств (в том числе процессоров) применяют и иные названия состояния, например *sleep* (сон) и *deep sleep* (глубокий сон). Для других устройств применимо понятие *уровня активности (APM level),* который выражается численно: 0 — минимальное потребление, 255 — максимальная активность. Кроме того, имеется состояние механического отключения, когда устройство обесточено механическим выключателем и никакой менеджер APM его уже не включит.

Конечно, самый тихий и холодный компьютер — выключенный, но его «пробуждение» (включение) требует загрузки ОС и приложений, которые, по мере технического прогресса, требуют все больше времени на разгрузку. В принципе, можно процесс загрузки обойти, для чего достаточно сохранить все содержимое ОЗУ, а также содержимое всех регистров процессора и внутренних регистров (буферов памяти) всех устройств, например, на жестком диске. После этого можно обесточить компьютер, а по включении быстро восстановить запомненное состояние и продолжить работу с точки останова. Такой способ «усыпле

ния» называется «зимней спячкой» (*hibernate*), при этом сохранение и восстановление состояния компьютера занимает всего десятки секунд. Приложения, работа которых была приостановлена, продолжают работу с того места, в котором были остановлены. В принципе, пользоваться таким способом выключения компьютера можно сколь угодно большое число раз, но на практике периодически приходится перезагружать ОС и приложения «по-честному». К этому вынуждают и сбои работы ОС и приложений, и накапливающиеся «отходы» памяти, которые не всегда могут быть использованы до перезагрузки. Для подстраховки перед усыплением компьютера рекомендуется все-таки явно сохранять пользовательские файлы на диске — риск невозможности восстановления состояния после включения хоть и невелик, но есть. Менее радикальный способ «усыпления», называемый *Standby*, сводится к остановке винчестеров, выключению дисплея, максимальному торможению процессора и всех остальных устройств. При этом состояние подсистем компьютера сохраняется, но на своих местах (данные остаются в памяти, регистрах процессора и всех устройств). Переход в нормальный режим происходит гораздо быстрее (время уходит только на раскрутку диска и на прогрев монитора, и то если он традиционный электронно-лучевой). Однако в таком состоянии компьютер все-таки потребляет заметную мощность, вентилятор настольного ПК продолжает работать. Надежность сохранения состояния получается ниже: провал (скачок) питания, толчки, способные нарушить контакт в модулях памяти или карт расширения, и прочие возмущения могут привести к потере состояния (впрочем, как и в нормальном рабочем режиме).

Идею быстрого доступа к ПК продвигают Microsoft и Intel, выступая с инициативами OnNow PC (можно перевести как «PC всегда готов») и Instantly Available PC. Эти идеи отражены в спецификациях PC99 и PC2001, и для управления питанием требуется спецификация ACPI (Advanced Configuration and Power Interface — расширенный интерфейс конфигурирования и питания). Спецификация ACPI представляет собой описание довольно сложной комбинации функций, часть из которых раньше возлагалась на относительно независимые системы PnP (в части конфигурирования) и APM. Этот интерфейс должен по возможности разумно управлять питанием различных подсистем. Если раньше управляли лишь потреблением монитора, винчестера и процессора (приостанавливая синхронизацию), то теперь могут «останавливаться» и оперативная память, и любые отдельные подсистемы, не используемые в данный момент времени. Система и устройства в «спящем» состоянии должны казаться выключенными — не шуметь, не мигать индикаторами, но иметь индикатор режима. Включение-выключение питания должно быть программно-управляемым. Система должна уметь «засыпать» (sleep state) и программно отключаться (soft off). «Просыпаться» она должна по нажатию кнопки, внешнему сигналу от периферийного устройства (в том числе и на шине USB) или по заранее спланированному расписанию. Эта возможность может потребоваться, например, для автоматического запуска в ночное время утилит обслуживания дисков (дефрагментации), загрузки почты и т. п. Система должна иметь таблицу, описывающую методы управления всеми устройствами и шинами. В рамках ACPI требуется и контроль терморежима с управлением вентиляторами. Управление

засыпанием и пробуждением может выполняться как от специальных кнопок на системном блоке, так и от специальных клавиш на клавиатуре. Кнопка-выключатель питания, используемая в современных системных блоках (ATX) вместо механического выключателя, через CMOS Setup может быть запрограммирована на управление «спячкой».

Система управления потреблением настраивается параметрами CMOS Setup (раздел Power Management), а также средствами современных ОС, включая Windows 9x/200x/XP. Основной аппаратной базой управления являются системная плата и BIOS, поддерживающие спецификацию ACPI (а прежде APM). Возможность программного включения обеспечивает «дежурный» (standby) источник блока питания ATX; также важна поддержка APM и ACPI периферийными устройствами. Для быстрого запуска со стороны BIOS предусматривается сокращение времени выполнения начального теста POST — подробное тестирование может выполняться лишь при обнаружении проблем во время предыдущей загрузки. Время от включения до начала загрузки ОС стремятся сократить до нескольких секунд; правда, процесс может затянуться медленно запускаемыми дисками или необходимостью инициализации всего объема ОЗУ при обнаружении и коррекции ошибок (ECC). В BIOS должна быть возможность отключить визуализацию теста POST — тогда после включения ПК, если все в порядке, пользователь сразу увидит любимую заставку загружаемой им ОС.

ГЛАВА 5

Организация ввода-вывода и BIOS

В предыдущей главе была рассмотрена общая архитектура PC-совместимых компьютеров. В данной главе рассматриваются вопросы организации ввода-вывода — взаимодействия программ и аппаратных средств периферийных устройств.

Базовая система ввода-вывода *(BIOS)* предназначена для изоляции операционной системы и прикладных программ от специфических особенностей конкретной аппаратуры. BIOS находится в микросхемах энергонезависимой памяти, расположенных на системной плате; на картах расширения могут находиться дополнительные модули BIOS, поддерживающие функционирование этих карт.

ROM BIOS хранится в микросхемах ПЗУ, которые могут быть и перепрограммируемыми. Для изменения содержимого ПЗУ их обычно приходится извлекать из системной платы, стирать и перезаписывать на специальном устройстве-программаторе. Флэш-BIOS (*Flash-BIOS*) хранится в микросхемах флэш- памяти, допускающей перепрограммирование прямо на месте установки (см. 8.5). В нормальном режиме работы компьютера информация в микросхемах ROM и флэш-BIOS является постоянной. О возможностях и процедуре обновления можно подробнее узнать в 6.2.

5.1. Взаимодействие программ с периферийными устройствами

Периферийные устройства могут подключаться к интерфейсам системного уровня (ISA, PCI, PCI-X, PCI-Express, AGP, LPC) или к периферийным интерфейсам (порты COM, LPT, Game; шины USB, FireWire, SCSI). Абстрагируясь от конкретной реализации подключения на системном уровне, можно говорить о логической *системной шине* PC-совместимого компьютера[1] — интерфейсе со следующими базовыми свойствами:

[1] Другие платформы могут отличаться по набору свойств — например, не иметь выделенного пространства ввода-вывода.

- интерфейс обеспечивает транзакции обращения к пространствам памяти и ввода-вывода;
- в транзакциях фигурируют физические адреса пространств памяти и ввода-вывода;
- адресные пространства памяти и ввода-вывода являются «плоскими» — адрес выражается одним числом в диапазоне, определенном принятой разрядностью адресации, то есть любой адрес может принадлежать регистру (ячейке памяти) только одного устройства (или системной памяти, включающей ОЗУ и энергонезависимую память);
- транзакции могут инициироваться как центральным процессором (процессорами), так и активными устройствами (мастерами шины);
- все адресуемые элементы безусловно доступны центральному процессору; на адресуемость элементов со стороны мастеров шин могут накладываться специфические ограничения[1];
- устройства, подключенные к системной шине, могут посылать процессору (процессорам) запросы аппаратных прерываний.

Взаимодействие программ с *устройствами, подключенными к системной шине*, возможно следующими способами:

- через регистры устройств, отображенные на пространства памяти или ввода-вывода;
- через области адресов памяти, принадлежащей устройству (физически расположенной на контроллере или адаптере устройства);
- через регистры конфигурационного пространства PCI (для устройств, подключенных к PCI, PCI-X, PCI-Express, AGP);
- через области системного ОЗУ, доступные активным устройствам-мастерам шины (обмен с использованием DMA);
- через аппаратные прерывания, инициируемые устройствами по доступным им линиям IRQx (ISA) или INTx# (PCI), а также по сообщениям MSI (PCI, PCI-E).

Обращения к регистрам конфигурационного пространства PCI (также «плоского») не относятся к базовым свойствам системной шины, поскольку программно они реализуются операциями обращения к пространствам ввода-вывода и/или памяти.

Современные устройства PCI и PCI-E для размещения своих регистров в основном используют пространство памяти, поскольку оно достаточно велико и спецификации PCI позволяют перемещать занимаемые области в любую часть пространства. Использование пространства ввода-вывода не приветствуется: оно маленькое, к тому же «перегорожено» регистрами традиционных уст

[1] Например, мастеру шины ISA доступна системная память лишь в пределах первых 16 Мбайт; мастерам шин PCI безусловно доступна системная память, но могут быть недоступны другие устройства PCI, от которых их отделяет главный мост.

ройств и их псевдонимами, порожденными неполным использованием шины адреса в ISA.

С *устройствами, подключенными к интерфейсам периферийного уровня*, взаимодействие возможно только через их контроллеры (адаптеры), соединенные с системной шиной. На системной шине «видны» и доступны только эти адаптеры и контроллеры. Способы взаимодействия с устройствами определяются интерфейсом контроллера.

Программное обеспечение компьютера состоит из ряда компонентов: прикладного ПО (исполняемых модулей — exe-файлов), драйверов устройств, системных драйверов, динамически компонуемых модулей, BIOS. Эти компоненты имеют различные возможности взаимодействия с устройствами, состав используемых компонентов зависит от операционной системы.

Имеются три способа взаимодействия программ, выполняемых центральным процессором (хост-программ), с периферийными устройствами:

- программный обмен;
- прямой доступ к памяти;
- прерывания.

Программный обмен с устройством осуществляется с помощью инструкций ввода-вывода для портов устройства или инструкций обращений к областям памяти, находящейся в устройстве. Эти инструкции размещаются в прикладной программе или драйверах, которыми она пользуется. Реальное физическое взаимодействие с устройством и вызываемые этим изменения состояния устройства происходят в момент выполнения этих инструкций. Данный способ взаимодействия позволяет предельно упростить интерфейсную часть периферийного устройства. Расплатой за это упрощение является дополнительная нагрузка на центральный процессор. Программный обмен можно подразделить на два типа:

- При *программно-управляемом обмене* перед передачей очередного байта программно анализируется (и ожидается) готовность устройства, для чего считывается его регистр состояния. Такой обмен сильно загружает процессор и не позволяет достичь высоких скоростей передачи данных, особенно если программа формирует и управляющие сигналы обмена. Так, например, работает драйвер параллельного порта в стандартном режиме, когда строб данных формируется двумя инструкциями OUT.
- В случае *блочного обмена PIO* (Programmed Input/Output — программируемый ввод-вывод) цепочка байтов, слов или двойных слов между памятью и портом ввода-вывода пересылается с помощью одной инструкции REP INS/OUTS. Для этих инструкций задаются начальный адрес памяти, длина блока, адрес порта и направление изменения адреса памяти (инкремент или декремент). Инструкции блочной пересылки (и обмен PIO) появились с процессорами 80286, они обеспечивают более быстрый обмен, чем стандартный контроллер DMA (8237A). Передача в режиме PIO применяется для обмена с устройствами ATA (IDE) и LPT-портом (в режимах EPP и ECP). Для устройств ATA определен ряд режимов обмена *PIO Mode x* (см. 19.2) со скоро

стями от 3,3 (PIO Mode 0) до 22,2 Мбайт/с (PIO Mode 4). Готовность к обмену проверяется один раз перед передачей блока, готовое устройство обязано выдержать пересылку всего блока. Управление потоком возможно с использованием сигнала готовности, притормаживающего шинные циклы обмена.

Отметим, что применительно к шине PCI (и всем ее «родственникам») программный обмен (даже блочный) не позволяет приблизиться к декларированной высокой пропускной способности шины. Причиной тому является неспособность процессора породить длинные пакетные транзакции на шине PCI (см. 14.4). По этой причине следует избегать данного способа взаимодействия при интенсивном обмене данными.

Прямой доступ к памяти (DMA) минимизирует участие процессора в обмене данными с устройством. В зависимости от того, кто является инициатором обмена, различают два варианта прямого доступа: по инициативе хоста и по инициативе устройства:

- *DMA по инициативе хоста* (host initiated DMA). Задание на пересылку каждого блока формирует программа, исполняемая ЦП; она же сообщает контроллеру DMA параметры сеанса (начальный адрес, длину блока и направление передачи) записью в его регистры. Физические операции обмена синхронизируются с устройством — оно своими внутренними сигналами инициирует обмен и, если требуется, управляет потоком (вводит сигнал готовности). Этот вариант требует довольно простых аппаратных средств устройства, расплата за упрощение — необходимость привлечения ЦП к организации каждого сеанса (обычно по прерываниям). Это не очень эффективно при передаче больших объемов данных, которые могут располагаться на разных, несмежных страницах физической памяти (см. далее).
- *DMA по инициативе устройства* (target Initiated DMA). Хост-программа формирует в памяти *программу ввода-вывода* для устройства (обычно это связанный список дескрипторов передач) и указывает устройству на ее начало (начало списка). Контроллер устройства исполняет эту программу: считывает дескрипторы из ОЗУ и по ним организует сеансы передачи данных между устройством и буферами в ОЗУ, описанными дескрипторами передач. Формирование программы может быть статическим или динамическим. В первом случае хост-программа передает устройству указатель на готовый список дескрипторов и не имеет права его модифицировать до тех пор, пока устройство не отработает список до конца. Так, например, работает традиционный контроллер PCI шины ATA. При динамическом формировании хост может добавлять новые дескрипторы (в конец списка), постоянно «подбрасывая» контроллеру новые задания. Подобным образом работают контроллеры шин USB и FireWire, PCI-контроллеры локальных сетей и ряд других. Функционирование устройства по программе требует усложнения его контроллера, но эти затраты окупаются повышением производительности и эффективности ввода-вывода. При этом стараются минимизировать число прерываний центрального процессора, инициируемых устройством.

Прерывания (interrupts) — сигнализация от устройства (его контроллера) центральному процессору (процессорам в мультипроцессорных системах) о некоторых событиях, требующих программных действий хоста. Эти события асинхронны по отношению к программному коду, исполняемому процессором. Прерывания требуют приостановки текущего потока инструкций (с сохранением состояния) и запуска процедуры обработки прерывания (Interrupt Service Routine, ISR). Эта процедура первым делом должна идентифицировать источник прерывания (а их может быть и несколько), затем выполнить действия, связанные с реакцией на событие. Если события должны вызывать некоторые действия прикладной программы, то обработчику прерывания следует только подать через ОС сигнал, который запустит или пробудит поток инструкций, выполняющий эти действия. Собственно процедура ISR должна быть оптимизирована по затраченному времени. Обслуживание прерываний, особенно в защищенном режиме, в PC-совместимых компьютерах на процессорах x86 связано со значительными накладными расходами. По этой причине их число стараются сократить. Значительные хлопоты доставляет идентификация источника прерывания — в архитектуре PC-совместимых компьютеров для этого используются традиционные, но не эффективные механизмы. В ряде случаев прерывания от устройств заменяют *политом* — программно-управляемым опросом состояния устройств. При этом состояния множества устройств опрашивают по прерыванию от таймера.

В компьютерных системах с «интеллектуальной» системой ввода-вывода (Intelligent Input/Output, I_2O) помимо центрального процессора имеется *процессор ввода-вывода* (Input/Output Processor, IOP). Этот процессор обычно имеет сокращенную систему команд, ориентированную на задачи управления вводом-выводом. В круг этих задач входит пересылка блоков данных, подсчет четности (для дисковых массивов RAID 3 и 5), преобразование данных между форматами *Big Endian* (популярный в телекоммуникациях) и *Little Endian* (принятый в процессорах Intel). Процессор ввода-вывода может как работать в общем адресном пространстве, так и иметь свое обособленное адресное пространство для управляемой подсистемы ввода-вывода. Взаимодействие процессора ввода-вывода со своими устройствами ведется теми же тремя основными способами, что были описаны ранее.

В рядовых компьютерах обычно ограничиваются *прямым управлением шиной* (bus mastering), которое позволяет контроллерам ПУ (или их интерфейсам) самим обращаться к системным ресурсам, выполняя необходимые обмены данными и управляющей информацией. Для этого контроллер ПУ должен временно взять на себя роль инициатора транзакций на интерфейсе, связывающем его с центром (главным образом, с памятью). Поскольку традиционно этот интерфейс является шинным, такой активный контроллер называют *мастером шины* (bus master), даже если он подключается к двухточечному интерфейсу (порту AGP или PCI-E). Чаще всего прямое управление шиной требуется для прямого доступа к оперативной памяти (разновидности такого доступа были описаны ранее). Прямое управление шиной может использоваться и для сигнализации прерываний (MSI на шине PCI, см. 14.5). В новых версиях шины PCI-X и в

PCI Express появилась возможность *однорангового взаимодействия устройств* (без участия процессора) — *обмена сообщениями*. При этом в адресации сообщений не фигурируют адреса пространства памяти или ввода-вывода — сообщения адресуются по *идентификатору устройства* (Device Identified Messages, DIM).

Архитектурный облик PC-совместимых компьютеров определяется свойствами используемых в них процессоров семейства x86. Современные процессоры x86, работающие в защищенном режиме, имеют довольно сложные механизмы виртуализации памяти, ввода-вывода и прерываний, из-за которых приходится различать физические и логические пространства (адреса памяти и ввода-вывода) и события (операции ввода-вывода, прерывания).

Физический адрес ячейки памяти или порта ввода-вывода — это адрес, формируемый на системной шине для обращения к данной ячейке. *Логический адрес* — это тот адрес, который формируется исполняемой программой (по замыслу программиста) для доступа к требуемой ячейке. Логический адрес в процессорах x86 состоит из двух компонентов: селектора сегмента и смещения внутри сегмента; из этих компонентов формируется *линейный адрес* — целое беззнаковое число. В большинстве современных ОС используется *плоская модель памяти,* в которой все доступные сегменты отображены на одно и то же адресное пространство. При этом программа не оперирует селекторами; программист адресует структуры данных в памяти по *линейным адресам* (для современных процессоров и приложений — 32-разрядным). *Физический адрес* формируется из логического с помощью блока страничной переадресации; трансляция адресов выполняется на страничном базисе, популярный размер страницы — 4 Кбайт. Благодаря страничной переадресации реализуется виртуальная память с подкачкой страниц. Переадресация выполняется на основе таблиц, формируемых в памяти операционной системой. Непрерывная область виртуальной памяти в общем случае представляется произвольно расположенными страницами физической памяти.

Физическая операция ввода-вывода или обращения к памяти — это процесс (шинный цикл), во время которого генерируются электрические сигналы, обеспечивающие доступ к данной ячейке (порту). *Логическая операция* — это исполнение программной инструкции (команды) обращения к интересующей ячейке. Логическая операция не всегда порождает ожидаемую физическую операцию: при определенных условиях она может блокироваться средствами защиты процессора, вызывая даже принудительное завершение программы, или же эмулироваться, создавая иллюзию физического исполнения.

Взаимодействие через пространство памяти

В реальном режиме (при отключенной страничной переадресации) физический адрес, фигурирующий на системной шине, совпадает с линейным адресом, формируемым прикладной программой. Тут все просто, правда, в стандартном (а не большом) реальном режиме доступен только первый мегабайт адресного пространства (то есть из устройств доступны только отображенные на область UMA).

В защищенном режиме, в принципе, доступно все физическое адресное пространство, но появляются проблемы, связанные с отображением линейных адресов на физические. Страничной переадресацией (поддержкой таблиц) ведает ОС, и у разных программных компонентов (приложений, драйверов, динамических модулей) имеются различающиеся возможности взаимодействия с системой управления памятью. Заметим, что у каждой задачи может быть своя карта адресов, в которой не обязательно будут присутствовать физические адреса всех устройств.

Для *обращения к регистрам устройства* (или к области памяти устройства), расположенным в пространстве памяти, программа должна узнать физический адрес данной области. Далее она должна запросить у ОС линейный адрес, на который отображается этот физический адрес, и обращаться к устройству по этому линейному адресу. Иного пути добраться до физического устройства у программы нет, и если ОС откажет в данном запросе, устройство окажется для этой программы недоступным. Для обращения к устройствам через пространство памяти у процессоров x86 предусмотрено большое число разнообразных инструкций — как выполняющих просто пересылку, так и работающих с операндами в памяти (то есть в устройстве). Инструкции, модифицирующие ячейки памяти, порождают на системной шине блокированные транзакции «чтение-модификация-запись». Этот тип транзакций не приветствуется с точки зрения эффективности использования времени шины, так что предпочтительно избегать таких инструкций при взаимодействии с устройствами. Заметим, что инструкции процессора обычно не порождают эффективных пакетных транзакций на шине PCI, они вызывают лишь одиночные транзакции[1]. Некоторые программные ухищрения, позволяющие повысить эффективность программноуправляемого обмена, описаны в 14.4.

При *организации прямого доступа к памяти* как по стандартным каналам DMA, так и при использовании ведущих устройств шин ISA и PCI возникает ряд проблем, связанных со страничным преобразованием адресов. Программе требуется организовать обмен данными между устройством и некоторым буфером данных в ОЗУ, с которым программа общается по линейным адресам, а устройство — по физическим. Отметим ряд существенных моментов:

- Программа должна запросить у ОС физический адрес, которому соответствует линейный адрес предполагаемого буфера обмена. Именно этот физический адрес должен задаваться устройству, осуществляющему DMA (или централизованному контроллеру DMA), при инициализации сеанса обмена (при указании начального адреса, длины блока и запуске канала).
- Физические страницы, к которым обращаются посредством DMA, должны быть зафиксированы: механизм замещения страниц не должен их затрагивать — по крайней мере, пока не завершится обмен посредством DMA.
- Если буфер данных не умещается в одной логической странице, возникает проблема пересечения границ. Обычный контроллер DMA работает по по

[1] Передача 32-разрядного слова по невыровненному адресу порождает пакетный цикл из двух передач, но эффективным (с точки зрения пропускной способности) его не назовешь.

следовательно изменяемым (инкрементируемым или декрементируемым) адресам. При пересечении границы логической страницы, возможно, потребуется скачок физического адреса, поскольку следующая логическая страница может быть физически отображена на произвольное (относительно предыдущей страницы) место ОЗУ. Чаще всего ОС оперирует страницами по 4 Кбайт, при этом пересылка больших блоков данных ведется «короткими перебежками», между которыми должна выполняться повторная инициализация контроллера DMA.

Проблема пересечения границ решается усложнением контроллеров DMA — применением «разбросанной записи» (scatter write) в память и «собирающего чтения» (gather read) памяти. В этом случае контроллеру DMA задается список описателей блоков (начальный адрес и длина), каждый из которых не пересекает границ логической страницы. Обработав очередной блок памяти, контроллер переходит к следующему, и так до конца списка. Такие возможности имеет, например, стандартный контроллер PCI IDE. Для передачи логически непрерывного блока данных его описатель может быть сокращен. Так, можно задать полный физический адрес начала блока, его длину и только список базовых адресов занимаемых им страниц. На каждой странице, кроме начальной, данные будут начинаться с нулевого адреса; на каждой странице, кроме последней, данные будут доходить до последнего адреса. Вместо длины блока можно задавать и физический адрес его конца. Такие варианты описаний используются, например, в хост-контроллерах шин USB и FireWire.

Проблема пересечения границ может решаться и иначе, без усложнения контроллера DMA. Для этого в памяти резервируется буфер значительного размера, отображенный на непрерывную область физической памяти, и обмен данными физическое устройство выполняет только с этим буфером. Однако рядовое приложение не может создать такой буфер, он может быть организован лишь драйвером устройства. Приложения могут лишь получать указатели на этот буфер и обмениваться с ним данными. Таким образом, по пути от приложения к устройству появляются дополнительная «перевалочная база» (буфер драйвера) и дополнительная пересылка данных, что приводит к дополнительным затратам времени.

Взаимодействие через пространство ввода-вывода

Для обращения программы к пространству ввода-вывода предназначены всего четыре инструкции процессора: IN (ввод из порта в регистр процессора), OUT (вывод в порт из регистра процессора), INS (ввод из порта в элемент строки памяти) и OUTS (вывод элемента из строки памяти в порт). Последние две инструкции, появившиеся с процессором 80286, могут использоваться с префиксом повтора REP, что обеспечивает быструю пересылку блоков данных между портом и памятью. Как уже отмечалось, обмен данными с портами, при котором применяют строковые инструкции ввода-вывода, получил название *PIO* (Programmed Input/Output — программируемый ввод-вывод).

Разрядность слова, передаваемого за одну инструкцию ввода-вывода, может составлять 8, 16 или 32 бита. В зависимости от выровненности адреса по границе слова и разрядности данных используемой шины это слово может передаваться за один или несколько циклов шины с указанием соответствующего нарастающего адреса в каждом цикле обращения к памяти. Инструкции ввода-вывода порождают шинные циклы обмена, в которых вырабатываются сигналы чтения из порта и записи в порт. Во избежание недоразумений и для экономии шинных циклов рекомендуется выравнивать адреса 16-битных портов по границе слова, а 32-битных — по границе двойного слова. Обращение по выровненным адресам выполняется за один цикл системной шины. Обращение по невыровненным адресам выполняется за несколько циклов, причем однозначная последовательность адресов обращений, которая зависит от модели процессора, не гарантируется. Так, одна инструкция вывода слова по нечетному адресу приведет к генерации двух смежных шинных циклов записи. При программировании обращений следует учитывать специфику устройств ввода-вывода. Если, например, устройство допускает только 16-разрядные обращения, то старший байт его регистров будет доступен лишь при вводе-выводе слова по четному адресу.

В реальном режиме процессора программе доступно все пространство адресов ввода-вывода. В защищенном режиме инструкции ввода-вывода являются привилегированными: возможность их исполнения зависит от текущего уровня привилегий. В защищенном режиме 32-разрядных процессоров (частным случаем которого является и виртуальный режим V86) имеется возможность программно ограничить доступное пространство ввода-вывода, определяя его максимальный размер (начиная с нулевого адреса и в пределах 64 Кбайт), а внутри разрешенной области доступ может быть разрешен или запрещен для каждого конкретного адреса. Размер области и *карта разрешенных портов ввода-вывода* (IO permission bitmap) задаются операционной системой в дескрипторе сегмента состояния задачи (Task State Segment, TSS). Карта разрешений влияет на исполнение инструкций ввода-вывода в зависимости от соотношения текущего и требуемого уровней привилегий ввода-вывода. При недостаточных привилегиях обращение по неразрешенному адресу вызывает исключение процессора, а поведение его обработчика определяется операционной системой. Возможно снятие задачи-нарушителя (знаменитое сообщение «Приложение... выполнило недопустимую операцию и будет закрыто»). Возможен и другой вариант, когда по обращении к порту монитор операционной системы выполняет некоторые действия, создавая для программы иллюзию реальной операции ввода-вывода. Таким образом, виртуальная машина по операциям ввода-вывода может общаться с виртуальными устройствами. Программа, выполняемая на нулевом уровне привилегий, безусловно может обращаться ко всем портам непосредственно.

Наиболее корректный (с точки зрения организации ОС) способ общения приложения с портами устройства требует помещения инструкций ввода-вывода в драйвер устройства, работающий на уровне привилегий ОС (на нулевом уровне). Обращение к портам непосредственно из приложения возможно, если

в карте разрешения портов бит для данного порта сброшен. Если бит установлен, то обращение к порту вызывает *исключение защиты*, которое обрабатывает диспетчер виртуальной машины (Virtual Machine Manager, VMM). В этом случае VMM вызывает процедуру, назначенную для данного порта операционной системой. Это может быть либо специальная процедура виртуального драйвера, установленного для данного порта, либо процедура, заданная по умолчанию. В первом случае ввод-вывод для данного порта доступен приложению только через виртуальный драйвер, вызов которого каждый раз будет приводить к издержкам переключения задач и смены уровня привилегий (от приложения на уровне 3 к драйверу нулевого уровня). Однако с точки зрения идеологии многозадачности и защиты это — естественное решение, обеспечивающее полную виртуализацию ввода-вывода. Процедура, заданная по умолчанию (в Windows 9x), открывает порт для данного приложения (сбрасывает бит в карте разрешений ввода-вывода) и выполняет собственно инструкцию ввода-вывода, возвращая приложению результат ввода. Таким образом, приложению Windows 9x станут доступными любые порты, для которых не установлен виртуальный драйвер. Правда, первое обращение к каждому порту произойдет медленно (через исключение), но последующие будут выполняться быстро. Если для взаимодействия с устройством задержка первого обращения критична, то при инициализации приложения можно выполнить «безобидные» обращения по адресам всех требуемых портов, чтобы открыть их для дальнейшей непосредственной работы (без издержек).

Заметим, что ОС Windows 9x не особо заботится о виртуализации и защите ввода-вывода: например, в Windows 9x из окна DOS можно обращаться к любым портам, даже к портам устройств, занятых операционной системой. В Windows NT/XP/200x защита ввода-вывода организована строже.

Синхронизация программ и устройств

Мы рассмотрели способы передачи потока данных, а теперь обсудим вопросы его *инициализации* и *синхронизации*. Инициатором обмена может выступать как программа, так и периферийное устройство. Программа ожидает какого-либо события в устройстве (например, установки бита готовности), периодически читая его регистр состояния. Такой способ называется *обменом по опросу готовности*. Время реакции на события может быть сведено до долей микросекунды, когда программа активно занимается опросом устройства в монопольном режиме. Однако при этом во время ожидания события процессор загружен в общем-то бесполезной работой. Другой подход — использование *аппаратных прерываний*, вырабатываемых устройством по событиям, требующим взаимодействия с программой. Программные обработчики аппаратных прерываний обычно инициируют любой из вышеперечисленных способов блочного обмена или выполняют одиночную операцию пересылки. Время реакции на запрос прерывания зависит от множества факторов, в том числе от режима работы процессора. Реакция на прерывания связана с интенсивным обменом с памятью, так что время отклика на прерывание может достигать десятков микросекунд. А если в обработке прерывания задействована и виртуальная память, то

счет идет на десятки и сотни миллисекунд. В реальном режиме процессора ответ на прерывание может быть получен за единицы (и даже доли) микросекунд.

Возможно и комплексное решение — *полинг*, то есть опрос готовности ряда устройств по периодическим прерываниям, например от системного таймера. Устройство, для которого обнаружена готовность, обслуживается, не готовое — пропускается до следующего прерывания. При этом процессор не выполняет многочисленных и, возможно, бесполезных циклов опроса готовности, а может заниматься другими задачами. Правда, накладные расходы на обслуживание прерываний остаются, а максимальное время реакции на событие не может быть меньшим, чем период прерываний от таймера. Так, например, работает утилита фоновой печати PRINT. Она не использует аппаратное прерывание от LPT-порта (хотя могла бы), а работает по таймеру.

Активное использование прерываний характерно для современных многозадачных операционных систем. Однако в особых случаях (например, при работе с несколькими адаптерами Gigabit Ethernet) полинг оказывается эффективнее (число прерываний и связанных с ними накладных расходов уменьшается).

Буферизация данных в устройствах

Каждое устройство имеет свою специфику характера обмена данными, определяемую природой его внешней (по отношению к компьютеру) стороны. По характеру обмена устройства можно разделить на три основных типа:

- Блочные устройства, например дисковые накопители. Обмен с ними возможен только блоками фиксированного размера — секторами. При обмене с физическим диском (например, через контроллер НГМД) нельзя останавливаться посреди передачи блока.
- Поточные устройства, например принтеры и сканеры. Принтеру посылают поток данных, которые он по мере своих электромеханических способностей выводит в виде изображения на бумагу. Поток можно приостановить в любой момент, а затем продолжить передачу без всяких побочных эффектов.
- Регистро-ориентированные устройства, как правило, не являются источниками или приемниками больших объемов данных. Программам обычно требуется знать текущее состояние данных устройств или/и формировать текущие управляющие воздействия. Пример регистро-ориентированного устройства — джойстик: программа в определенные моменты опрашивает текущее состояние кнопок и координатных датчиков. Регистроориентированными, как правило, являются различные устройства сопряжения с технологическим оборудованием, компьютеризованные измерительные комплексы и т. п.

Во многих устройствах присутствует смесь этих основных типов: так, даже принтер имеет регистроориентированную часть — помимо приема потока он передает сигналы текущего состояния (ошибка, конец бумаги).

Для того чтобы обеспечить некоторую свободу программам, обслуживающим устройства, относительно процессов формирования (потребления) данных уст

ройствами, применяют различные способы *буферизации данных* внутри устройств или их контроллеров. Буфер представляет собой набор внутренних ячеек памяти с определенными правилами доступа как со стороны устройства, так и со стороны компьютерного «центра». Размер буфера и дисциплина его обслуживания выбираются исходя из технических (скорость и объем информации, допустимые задержки) и экономических (цена) соображений.

Для блочных устройств обычно применяют буфер, минимальный размер которого равен размеру блока. Так, первые устройства IDE/ATA имели буфер объемом 512 байт на один сектор диска. Буфер поначалу был *однопортовым*: при чтении с диска встроенный контроллер сначала заполнял буфер данными сектора, и только после этого данные из буфера могли считываться в память компьютера. Позже размер буфера увеличили до объема нескольких секторов и стали применять *двухпортовый буфер*, допускающий практически одновременное обращение с двух сторон (портов) — со стороны устройства (обмен с носителем) и со стороны интерфейса (обмен по шине ATA). При считывании последовательности секторов как только очередной сектор целиком попадает в буфер и схемы контроля сообщают об отсутствии ошибок, контроллер диска может выдавать данные этого сектора на внешний интерфейс, а сам тем временем продолжать считывание последующих секторов в оставшуюся часть буфера. Более хитрый контроллер может использовать *кольцевой буфер*, продолжая считывание секторов с носителя в освободившееся (считанное по внешнему интерфейсу) начало буфера. Однопортовый буфер большого размера оказывается неэффективным, поскольку при длинных операциях он вносит большую *задержку в доставку* данных. Двухпортовость позволяет уменьшить задержку, а кольцевая организация делает буфер практически безразмерным (при условии своевременного освобождения).

Адаптеры локальных сетей тяготеют к блочным устройствам — они передают данные целыми пакетами, которые должны приниматься и посылаться с определенной скоростью (10, 100 или 1000 Мбит/с для трех поколений Ethernet). Для них объем и организация буфера зависят от скорости передачи данных в среде и производительности интерфейса (шины расширения), к которому они подключены. Максимальный размер пакета (кадра) для Ethernet — около 1,5 Кбайт (для технологий TokenRing и FDDI гораздо больше). Сетевые карты Ethernet на 10 Мбит/с для шин ISA/EISA имели буфер по крайней мере на один кадр, а полнодуплексные — на два. Значительно более эффективными были карты с большим объемом буфера (так, карты 3C509B были гораздо «шустрее», чем 3C509). Эти буферы могли также быть однопортовыми или двухпортовыми со всеми вышеописанными свойствами. Однопортовые буферы приводят к задержкам передачи на время приема целого кадра, что находит отражение в увеличении времени отклика сети. Для 100-мегабитных (Fast Ethernet) карт PCI с прямым управлением шиной оказалось возможным использование всего двух 64-байтных буферов (по одному на прием и передачу, что обеспечивает поддержку полного дуплекса). Каждый буфер поделен пополам, и половинки чередуются (ping-pong buffer): при приеме пакета из сети сначала заполняется первая половина, затем вторая. Как только первая половина заполнится, карта запрашивает управление шиной, и как только его получает, сама выгружает

данные из этой половинки в память. Как только заполнится вторая половина, карта переключается на заполнение первой (уже свободной) и делает следующий запрос на управление для выгрузки второй половины. По мере приема кадра эта «игра в пинг-понг» продолжается; передача кадра выглядит аналогично. Такое упрощение организации (это технически проще реализации двухпортового доступа к локальной памяти) стало возможным благодаря высокой пропускной способности шины и гарантированному времени предоставления доступа к шине (см. 14.2). Однако для карт Gigabit Ethernet этот вариант уже не проходит, и на них устанавливают буферы на полный кадр.

Для потоковых устройств часто применяют *буфер с дисциплиной обслуживания FIFO* (First In, First Out — «первым вошел, первым вышел»). Размер такого буфера, как правило, невелик (например, 16 байт). Буфер ставится между «центром» и устройством: с одной стороны он наполняется, с другой — опорожняется. Опорожняющая сторона может извлекать данные из буфера, лишь когда наполняющая сторона их туда положит. Попытка извлечения данных из пустого буфера является *ошибкой опустошения* (underflow), попытка помещения в заполненный — *ошибкой переполнения* (overflow). Система управления буфером следит за степенью его заполнения и сообщает «центру» о критических ситуациях. Когда «центр» (программа, исполняемая процессором) выводит данные через FIFO, система управления буфером следит за снижением степени его заполнения ниже *порога опустошения* и в случае такового сигнализирует (обычно прерыванием) о необходимости вывода следующей порции данных. Система управления также препятствует переполнению, отвергая попытки записи лишних данных и немедленно сообщая об ошибке (обычно через соответствующий программно-читаемый бит состояния). При вводе данных через буфер FIFO его система управления следит за наличием свободного места в буфере и при превышении *порога заполнения* также сигнализирует прерыванием. Аналогично, она не позволяет считать данные из пустого буфера и сообщает об этом соответствующим битом. Также система управления буфером должна позволять его очищать по инициативе процессора, сообщать о количестве (или хотя бы о наличии) данных в буфере по запросу процессора. Управляемость порогов позволяет программе в зависимости от внешнего темпа обмена данными, возможностей и текущей загруженности компьютера выбрать оптимальный режим обмена, позволяющий и «не суетиться по мелочам», и не допускать переполнения/опустошения буфера. У двунаправленных устройств, как правило, имеется пара буферов FIFO (в случае полного дуплекса), для симплексных устройств достаточно одного. Например, в контроллере НГМД (хотя это и блочное устройство, но медленное), имеется один переключаемый буфер. Буферы FIFO применяются в COM-портах, LPT-портах и ряде других потоковых узлов компьютера. Без буферов FIFO, например, невозможен фоновый обмен данными с модемами на скоростях выше 19,2 Кбит/с. Буферная память принтеров также является буфером FIFO, правда, не очень гибким и «общительным» (о степени своего заполнения по стандартному интерфейсу он не сообщает).

Буферы современных устройств внешней памяти имеют более сложную организацию, обеспечивающую кэширование данных; однако и они используют вышеописанные принципы организации. Однопортовые буферы большого объема,

как уже говорилось, могут вносить заметную задержку. Для потоковых применений (например, для воспроизведения мультимедийных файлов) эта задержка обычно не очень существенна и на производительность не влияет. Однако для приложений «петлеобразного» характера, когда буфер оказывается в цепочке «запрос-ответ», его задержка может приводить к снижению производительности. Так, передача данных по сети обычно представляет собой последовательность кадров данных, на каждый из которых передающая сторона ожидает кадр подтверждения. Если каждый кадр будет «просиживать» в буфере, естественно, производительность снизится. От этой беды спасает метод «скользящего окна», при котором передающая сторона допускает некоторое отставание приема подтверждений. Примерно та же идея реализована в синхронном режиме передачи на шине SCSI (см. 20.4).

5.2. Системный модуль ROM BIOS

Системный модуль ROM BIOS (System ROM BIOS) обеспечивает программную поддержку стандартных устройств PC, конфигурирование аппаратных средств, их диагностику и вызов загрузчика операционной системы. Системный модуль ROM BIOS в значительной степени привязан к конкретной реализации системной платы, поскольку именно ему приходится программировать все микросхемы чипсета системной платы. Функции BIOS разделяются на следующие группы:

- инициализация и начальное тестирование аппаратных средств — POST;
- настройка и конфигурирование аппаратных средств и системных ресурсов — CMOS Setup, см. 6.6;
- автоматическое распределение системных ресурсов — PnP BIOS;
- идентификация и конфигурирование устройств PCI — PCI BIOS, см. 14.7;
- начальная загрузка (первый этап загрузки операционной системы) — Bootstrap Loader;
- обслуживание аппаратных прерываний от системных устройств (таймера, клавиатуры, дисков) — BIOS Hardware Interrupts;
- отработка базовых функций программных обращений (сервисов) к системным устройствам — ROM BIOS Services;
- поддержка управляемости конфигурированием — DMI BIOS;
- поддержка управления энергопотреблением и автоматического конфигурирования — APM и ACPI BIOS.

Все эти функции (или их часть) исполняет системный модуль BIOS, хранящийся в микросхеме ПЗУ или флэш-памяти на системной плате. Большинство сервисных функций выполняется в 16-битном режиме, хотя некоторые новые функции могут иметь и альтернативные вызовы для 32-битного исполнения.

Системный модуль BIOS должен обслуживать по вышеуказанным функциям все компоненты, установленные на системной плате: процессор, контроллер памяти (ОЗУ и кэш), стандартные архитектурные компоненты (контроллеры

прерываний и DMA, системный таймер, системный порт, CMOS RTC), контроллер клавиатуры, а также набор стандартных периферийных контроллеров и адаптеров, даже если они и не установлены на системной плате. В этот набор входят графические адаптеры CGA и MDA, порты COM и LPT, контроллер НГМД, диски ATA (теперь уже обязательно двух каналов). Если на системной плате установлены дополнительные компоненты, например контроллер SCSI, графический адаптер SVGA, адаптер локальной сети, то их поддержка тоже должна быть в системном модуле BIOS.

Микросхема с системным модулем BIOS приписана к пространству памяти, и ее положение определяется двумя свойствами процессоров x86:

- После аппаратного сброса процессор выполняет первую инструкцию по адресу начала последнего параграфа физически адресуемой памяти: 8086/88 по адресу FFFF0h; 80286 и 386SX — по адресу FFFFF0h; 386DX и выше — по адресу FFFFFFF0h (правда, P6 можно сконфигурировать и на старт по адресу FFFF0h).
- В стандартном реальном режиме процессору доступна лишь память с адресами в пределах O-FFFFFh, следовательно, программный код и данные BIOS должны находиться в этом диапазоне. Векторы прерываний, ссылающиеся на сервисы BIOS, в реальном режиме могут адресоваться только к памяти в диапазоне адресов 0-0FFFFFh (0-10FFEF при открытом вентиле *Gate A20)*.

По этим соображениям в PC AT область системного модуля BIOS располагается под границей первого мегабайта памяти по адресам F0000-FFFFFh, занимая 64 Кбайт (целый сегмент). Копия этого образа на машинах 80286 и 386SX располагается по адресам FF0000-FFFFFFh под границей 16-го мегабайта. На машинах 386DX и выше копия образа BIOS находится в области FFFF0000-FFFFFFFFh, но для процессоров P6 она, в принципе, необязательна. Тем не менее, ее продолжают использовать (даже, например, в чипсете i820). Кроме того, для совместимости с AT/286 на некоторых платах могла присутствовать и дополнительная копия BIOS по адресам FF0000-FFFFFFh, если она разрешена настройкой CMOS Setup (в этом случае невозможно использовать более 16 Мбайт ОЗУ). В машинах XT системный модуль BIOS был компактным (8 Кбайт) и размещался в области FE000-FFFFFh. Когда появились микросхемы ПЗУ емкостью 128 Мбайт, на некоторых машинах AT системный модуль BIOS стал занимать область E0000-FFFFFh, но вскоре от этого «расширения» отказались, поскольку оно сокращало размер доступной верхней памяти (см. 4.2). Современный системный модуль BIOS имеет типовой объем 128 или 256 Кбайт, который проецируется в «окно» размером 64 Кбайт страницами. Это возможно, поскольку во время начальных стадий теста POST и выполнения утилиты CMOS Setup не требуется поддержки всех сервисов BIOS, а в рабочем режиме, наоборот, не нужен программный код POST и Setup.

Поскольку содержимое флэш-BIOS может быть изменено прямо в компьютере, возникает опасность полной потери работоспособности компьютера при занесении некорректной «прошивки» или под действием вируса. С разрушенным мо

дулем BIOS компьютер не может запуститься. Для предотвращения таких ситуаций применяют различные способы защиты.

Тест начального включения — POST

При включении питания, аппаратном сбросе от кнопки Reset или нажатии комбинации клавиш Ctrl+Alt+Del процессор переходит к исполнению кода начального самотестирования *POST* (PowerOn Self Test — самотестирование при включении), хранящегося в микросхеме BIOS. POST выполняет тестирование процессора, памяти и системных средств ввода-вывода, а также конфигурирование всех программно-управляемых аппаратных средств системной платы. Часть процедуры конфигурирования выполняется однозначно, часть управляется джамперами системной платы, но ряд параметров позволяет или даже требует конфигурирования по желанию пользователя. Для этих целей служит утилита *Setup*, встроенная в код BIOS. После тестирования и конфигурирования (включающего настройку устройств PnP) POST инициирует загрузку операционной системы.

При прохождении каждой секции POST записывает ее код (номер) в диагностический регистр. Этот регистр физически располагается на специальной диагностической плате *POST Card* (это не почтовая карточка), устанавливаемой в слот шины расширения. Плата содержит 8-битный регистр со световой (двоичной или шестнадцатеричной) индикацией состояния битов. В пространстве ввода-вывода регистр занимает один адрес, зависящий от архитектуры PC (точнее, версии BIOS): ISA, EISA — 80h, ISA-Compaq — 84h, ISA-PS/2 — 90h, MCA-PS/2 — 680h, некоторые модели EISA — 300h (часто пишут то же и в 80h). По индикаторам платы можно определить, на какой секции остановился тест POST, и выяснить причину неисправности. Однако для такой диагностики необходимы, во-первых, сама плата-индикатор и, во-вторых, «словарь» неисправностей — таблица, специфическая для версии BIOS и системной платы.

Во время выполнения POST может выдавать диагностические сообщения в виде последовательности коротких и длинных звуковых сигналов, а после успешной инициализации графического адаптера — в виде небольших текстовых сообщений на экране монитора.

Ниже представлена обычная последовательность шагов теста POST:

1. Тестирование регистров процессора.
2. Проверка контрольной суммы ROM BIOS.
3. Проверка и инициализация таймера 8253/8254, портов 8255. После этого шага становится доступной звуковая диагностика (табл. 5.1).
4. Проверка и инициализация контроллеров DMA 8237.
5. Проверка регенерации памяти.
6. Тестирование 64 Кбайт нижней памяти.
7. Загрузка векторов прерывания и стека в нижнюю область памяти.

8. Инициализация видеоконтроллера — на экране появляется заставка Video BIOS, обычно с указанием модели видеокарты и объемом установленной видеопамяти.
   После успеха этого шага изображение на экране сменяется заставкой системного модуля BIOS со счетчиком объема тестируемой динамической памяти. Теперь диагностические сообщения выводятся на экран (табл. 5.2). POST продолжает работу.
9. Тестирование полного объема ОЗУ.
10. Тестирование клавиатуры.
11. Тестирование CMOS-памяти и часов.
12. Инициализация COM- и LPT-портов.
13. Инициализация и тест контроллера НГМД.
14. Инициализация и тест контроллера НЖМД.
15. Сканирование области дополнительной памяти ROM BIOS.
16. Вызов Bootstrap (Int 19h) — загрузка операционной системы, при невозможности — попытка запуска ROM Basic (Int 18h), при неудаче — останов процессора с сообщением System Halted (система остановлена).

ІМЕЧАНИЕ ----------

На новых системных платах реализуется и речевая звуковая диагностика (voice diagnostics) — через динамик пользователю предлагают проверить установку модулей памяти, видеоадаптера, подключения кабеля винчестера и т. п. Язык сообщений можно выбрать в CMOS Setup (когда эту процедуру удастся запустить), правда, русского языка пока не встречается.

Таблица 5.1. Звуковая диагностика POST

Сигнал[1]	Ошибка	Возможные действия
1д 2к	Не обнаружен графический адаптер	Установить (переставить) адаптер
1д 3к	Не подключен монитор (для системных плат со встроенным графическим адаптером)	Подключить монитор, проверить включение терминаторов на мониторе
1д Хк	Ошибка графического адаптера (Х зависит от версии Video BIOS)	Установить (переставить) адаптер
1к	Ошибка регенерации DRAM — установлено некорректное значение периода регенерации или неисправен контроллер регенерации	Попытаться установить параметры Setup, предлагаемые по умолчанию, заменить DRAM. Если не помогает, значит, неисправность в самой системной плате
2к	Ошибка четности DRAM (отсутствует у плат, не поддерживающих контроль четности)	Заменить (переставить) память
3к	Ошибка в первых 64 Кбайт DRAM	Заменить (переставить) память
4к	Ошибка системного таймера	Ремонт системной платы
5к	Ошибка процессора	Заменить процессор
6к	Ошибка управления GateA20 (контроллер 8042)	Переустановить или заменить ИС контроллера клавиатуры
7к	Ошибка защищенного режима	Ремонт системной платы

продолжение

Таблица 5.1 *(продолжение)*

Сигнал[1]	Ошибка	Возможные действия
8к	Ошибка видеопамяти	Заменить видеопамять (графический адаптер)
9к	Ошибка контрольной суммы ROM BIOS	Заменить (перезаписать) BIOS
10к	Ошибка CMOS (обращения к ячейке 0Fh)	Ремонт системной платы
11к	Ошибка кэш-памяти	Заменить кэш-память, проверить ее быстродействие и настройку Setup при отключенном кэше

[1] 1д 2к — один длинный сигнал, за которым следуют два коротких.

Таблица 5.2. Диагностические сообщения POST

Сообщение	Причина и возможные действия
PRESS A KEY TO REBOOT	Предложение перезагрузки путем нажатия любой клавиши сопровождает сообщение об ошибке, обнаруженной POST
SYSTEM HALTED, (Ctrl+Alt+Del) TO REBOOT	Остановка компьютера из-за обнаружения серьезной ошибки. Возможна только перезагрузка путем нажатия клавиш Crtl+Alt+Del, аппаратного сброса или повторного включения питания
CMOS Battery State Low CMOS BATTERY HAS FAILED	Упало напряжение питания CMOS. Проверить напряжение на батарее при выключенном питании компьютера (должно быть выше 3 В), проверить установку джампера 2-3 на разъеме внешней батареи. Заменить батарею
CMOS Checksum Failure CMOS CHECKSUM ERROR	Ошибка контрольной суммы CMOS. Может быть вызвана проблемами с питанием CMOS, применением непригодной утилиты Setup, действием вируса. Выполнить «штатную» утилиту Setup
CMOS System Options Not Set	Не установлены параметры Setup. Выполнить настройку Setup
CMOS Time and Date Not Set	Не установлены часы-календарь. Выполнить настройку Setup, задав время и дату
Display Switch Not Proper DISPLAY SWITCH IS SET INCORRECTLY	Проверить положение переключателя типа графического адаптера (Color/Mono), имеющегося на большинстве старых системных плат
DISPLAY TYPE HAS CHANGED SINCE LAST BOOT	С момента предыдущей загрузки изменился тип графического адаптера (монитора). Выполнить настройку Setup, изменив (подтвердив новый) тип адаптера
Keyboard is locked ... Unlock it	Клавиатура заблокирована ключом. Повернуть ключ (если не помогает, проверить правильность подсоединения ключа к разъему системной платы)
Keyboard Error K/B Interface Error KEYBOARD ERROR OR NO KEYBOARD PRESENT	Ошибка клавиатуры. Проверить подключение разъема, проверить состояние переключателя XT/AT на клавиатуре, заменить клавиатуру. Проверку клавиатуры можно подавить установкой параметра Keyboard Not Installed в Setup (параметр имеется не во всех версиях, тот же эффект дает установка значения HALT ON ALL, BUT KEYBOARD параметра Halt on Error)
DISK BOOT FAILURE, INSERT SYSTEM DISK AND PRESS ENTER	Нет доступного загрузочного устройства (гибкий, жесткий диск, CD-ROM, сетевой адаптер с микросхемой BOOT ROM) с действительной загрузочной записью. Установить загрузочную дискету в дисковод A, проверить контроллер, конфигурацию и подключение диска C

Сообщение	Причина и возможные действия
Invalid Boot Diskette Diskette Boot Failure	Невозможно загрузить ОС с дискеты (нет загрузочного сектора). Заменить дискету
No ROM Basic	Нет устройства, с которого можно загрузить ОС (гибкий, жесткий диск, CD-ROM, сетевой адаптер с микросхемой BOOT ROM), а интерпретатор Basic в ROM отсутствует (был в первых моделях PC). Подключить и сконфигурировать загрузочное устройство
DISKETTE DRIVES OR TYPES MISMATCH ERROR — RUN Setup	Тип дисковода (A или B) не совпадает с записью в CMOS. Выполнить настройку Setup и задать правильные типы дисководов
FDD Controller Failure FLOPPY DISK CNTRLR ERROR OR NO CNTRLR PRESENT	Ошибка контроллера накопителей на гибких дисках (дисководов, кабелей). Проверку можно отменить, установив в Standard Setup для дисков A и B значение Not Installed (None). Если контроллера нет, должно быть установлено это значение
HDD Controller Failure ERROR INITIALIZING HARD DRIVE CONTROLLER	Ошибка контроллера накопителей на жестких дисках (дисководов, кабелей). Проверку можно отменить, установив в Standard Setup для всех жестких дисков (двух или четырех) значение Not Installed
C: (D:) Drive Error C: (D:) Drive Failure ERROR ENCOUNTERED INITIALIZING HARD DRIVE	Невозможно обращение к диску C (D). Неверно установлены параметры в Setup, джамперы на накопителях, интерфейсные кабели, не отформатирован диск или неисправен дисковод
CMOS Memory Size Mismatch MEMORY SIZE HAS CHANGED SINCE LAST BOOT	Несовпадение размера памяти, определенной POST, со значением, записанным в CMOS. Обычно происходит при добавлении или удалении дополнительных модулей памяти, но может указывать и на неисправность памяти. Запустить Setup, открыть раздел STANDARD Setup и выйти с сохранением результатов в CMOS. Для EISA может потребоваться выполнение ECU
On Board Parity Error Off Board Parity Error Parity Error Memory Parity Error at XXXX RAM PARITY ERROR — CHECKING FOR SEGMENT	Ошибка четности памяти, установленной на системной плате (On Board), плате расширения (Off Board) или без указания местонахождения. Сбойный адрес XXXX может быть определен не всегда. Сообщение может быть вызвано и вирусом
PRESS F1 TO DISABLE NMI, F2 TO REBOOT	Предложение продолжить работу с запрещенным контролем четности (запрещено NMI), нажав клавишу F1, или перезагрузить компьютер, нажав клавишу F2. Может появляться при обнаружении ошибки паритета памяти
Memory Address Error at XXXX Memory Verify Error at XXXX	Ошибка памяти по адресу XXXX. Локализовать и заменить модуль (микросхему) памяти
Address Line Short	Замыкание адресных линий микросхем или модулей памяти. Переставить (заменить) микросхемы или модули DRAM
Cache Memory Bad, do Not Enable Cache!	Ошибка кэш-памяти. Устранить ошибку (заменить или переставить микросхемы) или запретить внешний (External или L2) кэш в Setup
I/O Card Parity Error at XXXX	Ошибка, обнаруженная на плате расширения (сигнал подается по линии IOCHK)

продолжение

Таблица 5.2 (*продолжение)*

Сообщение	Причина и возможные действия
DMA Bus Time-out	Устройство в режиме DMA задерживает цикл шины более чем на 78 мкс. Причина — неисправность платы расширения или системной платы
EISA Configuration is Not Complete	Не полностью задана конфигурационная информация EISA. Система может быть загружена в режиме ISA для конфигурирования утилитой ECU (EISA Configuration Utility)
Invalid EISA Configuration	Конфигурационная информация EISA недействительна. Система может быть загружена в режиме ISA для конфигурирования утилитой ECU
EISA CMOS Checksum Failure EISA Configuration Checksum Error	Ошибка контрольной суммы дополнительной CMOS-памяти конфигурации устройств EISA — возможно, из-за батарейки. Система может быть загружена в режиме ISA для конфигурирования утилитой ECU
EISA CMOS Inoperational	Ошибка доступа (чтение-запись) к дополнительной CMOS-памяти конфигурации устройств EISA — возможно, из-за батарейки
Expansion Board not ready at Slot X	Плата расширения в слоте X (EISA) не готова. Проверить плату и конфигурацию
ID information mismatch for Slot X Wrong Board in Slot X	Идентификатор установленной платы расширения EISA не совпадает с записью в CMOS для этого слота
Slot X Should Be Empty But EISA Board Found Slot X Not Empty	Слот X шины EISA должен быть пустым, но обнаружена плата. Выполнить конфигурирование утилитой ECU
Slot X Should Have EISA Board But Not Found	Для слота X шины EISA назначена плата, но она не обнаружена. Выполнить конфигурирование утилитой ECU
Invalid Configuration Information for Slot X	Некорректная информация конфигурации для платы расширения EISA в слоте X. Выполнить конфигурирование утилитой ECU
BUS Timeout NMI at Slot X	Ошибка тайм-аута обращения по системной шине для платы в слоте X
Fail-Safe Timer NMI	Произошло прерывание от таймера, контролирующего предельное время растяжки шинного цикла
INTR #1 Error	Ошибка контроллера прерываний #1 (отвечает за линии IRQ 0–7)
INTR #2 Error	Ошибка контроллера прерываний #2 (отвечает за линии IRQ 8–15)
8042 Gate A20 Error!	Неисправность работы вентиля линии A20 (Gate A20) в микросхеме контроллера клавиатуры 8042. Можно обойти, установив в Setup для параметра Gate A20 Control значение Fast (управление от чипсета)
DMA #1 Error DMA Error	Ошибка контроллера DMA (может быть вызвана платами расширения)

В процессе работы POST используются ячейки CMOS: результаты прохождения тестов заносятся в ячейку 0Eh (Post Diagnostic Status Byte), в ячейке 0Fh (Shutdown Flag) находятся идентификаторы состояния перед началом теста. В BIOS DATA AREA [0:0472] задается тип рестарта (1234h = Ctrl+Alt+Del — «теплый» старт, 4321h — сброс с сохранением памяти). Это позволяет различать причины рестарта (перезагрузка, выход из защищенного режима 286 и т. д.) для обхода некоторых секций POST.

Начальная загрузка

Стандартная процедура начальной загрузки (bootsrap loader), вызываемая по прерыванию Int 19h BIOS в конце теста POST, выбирает устройство начальной загрузки (Initial Program Loader, IPL) — блочное устройство, поддерживающее функцию чтения секторов. С этого устройства процедура пытается загрузить в ОЗУ самый первый сектор, и если у него в конце имеется сигнатура загрузчика, ему передается управление. До выполнения начальной загрузки должны быть инициализированы перечисленные ниже *загрузочные устройства* (boot device), которыми пользуются сама эта процедура и загружаемые ею модули:

- *Устройство ввода* (input device) — как правило, клавиатура, с которой можно управлять загрузкой, отвечая на запросы. Это устройство должно поддерживать посимвольный ввод — сервис Int 09h BIOS.
- *Устройство вывода* (output device) — как правило, дисплей, на который выводятся сообщения загрузчика. Это устройство должно поддерживать посимвольный вывод — сервис Int 10h BIOS.
- *Устройство начальной загрузки* (IPL) — как правило, НГМД, НЖМД и другие устройства, поддерживающие функции блочного чтения — сервис Int 13h (02 или 42h) BIOS. Для краткости в дальнейшем его будем называть просто устройством загрузки.

Первый сектор с выбранного устройства загрузки функцией чтения Int 13h (02) BIOS загружается в память по адресу 0000:7C00h, и если в его конце (по адресу 0000:7DFE) обнаружена сигнатура загрузочного сектора (слово AA55h), управление передается на его начало (по адресу 0000:7C00h), где расположена точка входа в программу загрузки. На этом деятельность теста POST завершается, хотя вызовом прерывания Int 18h загрузчик может снова отдать ему управление в случае своих неудач, и POST попробует выполнить загрузку с другого устройства. Нормального возврата из загрузчика не предусмотрено (только вперед, на загрузку ОС!).

Прерывание Int 18h BIOS на старых машинах предназначалось для вызова встроенного интерпретатора BASIC при невозможности загрузки с диска. Позже это соглашение стали использовать для попыток загрузки с альтернативных устройств, и Int 18h BIOS определили как функцию аварийного возврата для начального загрузчика. По прерыванию Int 18h POST снова инициализирует стек и переходит к другому устройству загрузки.

Содержимое загруженного первого сектора зависит от того, с какого устройства он был загружен. На *загрузочной дискете* первый сектор содержит *загрузчик* — программу, загружающую операционную систему или только ее ядро. Этот загрузчик привязан к своей ОС и записывается на диск при форматировании его средствами этой ОС. Загрузочный сектор содержит собственно код загрузчика и необходимые ему параметры диска (см. 9.6). Для дисков DOS загрузчик, пользуясь этими параметрами, находит начало корневого каталога и ищет в его первых двух элементах знакомые имена файлов — IO.SYS и MSDOS.SYS. Найдя их, он считывает первые 3 сектора файла IO.SYS в память по адресу 0070:0000 (или 0000:0700, что одно и то же) и передает управление на его начало, сохранив

в регистре CH тип носителя, в регистре DL — номер привода, в регистрах AX и BX — старшую и младшую части линейного адреса начала корневого каталога. Сам файл IO.SYS гораздо длиннее трех секторов, но в дальнейшем процессе его загрузки и продолжения загрузки ОС данный загрузчик уже участия не принимает.

На *жестком диске* первый сектор содержит *главную загрузочную запись* (MBR). Он также загружается в память по адресу 0000:7C00h, и если в его конце обнаружена сигнатура загрузочного сектора (AA55h), управление передается на его начало. При исполнении главный загрузчик первым делом перемещает (копирует) свой код (и таблицу разделов) по адресу 0000:0600h и продолжает свое исполнение уже из новой области. Задача главного загрузчика — найти активный раздел, загрузить его первый сектор в память и, если он имеет сигнатуру загрузочного сектора, передать ему управление. Найдя описатель активного раздела, главный загрузчик загружает в память по адресу 0000:7C00h его первый сектор, при этом регистр SI указывает на описатель активного раздела (после перемещения таблица разделов оказывается в памяти по адресам 0000:07B0- 07FDh). Первый сектор загружаемого раздела ищется просто: в регистр DX заносится слово 0, а в CX — слово 2 из описателя активного раздела. После этого остается лишь задать адрес буфера в памяти (в ES:BX), функцию чтения одного сектора (AX = 0201h) и вызвать дисковый сервис Int 13h BIOS. Если считать сектор без ошибок не удается (делается до 5 попыток), главный загрузчик останавливается с сообщением «Error loading operating system».

Программный код загрузчика, расположенного на разделе жесткого диска, несколько отличается от дискетного, поскольку для работы с дискетой требуется инициализация ее таблицы параметров. Однако общая идея процесса загрузки у них одинакова. Для других операционных систем загрузчик выполняет иные действия, но цель та же — загрузить в память начальные модули и передать им управление.

Главный загрузчик инвариантен по отношению к загружаемым операционным системам и дискам, его программный код, как и таблица разделов, записывается утилитой FDISK при конфигурировании жесткого диска (см. 9.12). Однако традиционный главный загрузчик в принципе не способен загрузить раздел, находящийся дальше чем через 8,4 Гбайт от начала диска, поскольку пользуется исключительно трехмерным (CHS) описанием границ разделов. Для больших дисков главный загрузчик должен использовать линейные описания разделов.

Помимо вышеописанного штатного способа загрузки с диска, выполняемой традиционным системным модулем BIOS, имеется возможность загрузки, в принципе, с любого устройства, с которого можно загрузить в память требуемый блок данных. Так, возможна загрузка с CD-ROM, но для этого необходимо выполнение специальной процедуры, включающей эмуляцию диском CD-ROM дискеты или жесткого диска (см. 9.11). Возможна загрузка и с нестандартного устройства, подключаемого через карту расширения. Однако системный модуль BIOS сам этого не умеет, и такое загрузочное устройство должно иметь ПЗУ расширения BIOS (см. 5.3) с собственной процедурой загрузки. Для такого устройства процедура инициализации в ПЗУ расширения должна перехва

тить вектор Int 19h (чтобы стать первым загрузочным устройством) или Int 18h (чтобы получить управление, если загрузка со штатных устройств не удается). Этот перехваченный вектор должен указывать на специфическую процедуру загрузки с данного устройства, которая и будет выполняться тестом POST вместо или в качестве запасного варианта обычной загрузки. Такой способ применяется для *устройств удаленной загрузки* (Remote Program Loader, RPL) — например, адаптеров локальной сети, снабженных ПЗУ удаленной загрузки (boot ROM). Когда подобный адаптер установлен и работа ПЗУ удаленной загрузки разрешена, при каждой перезагрузке на консоли может появляться предложение о выборе между загрузкой с жесткого диска или по сети (загрузка с дискеты свой приоритет не потеряет).

Для сложных систем, которые содержат разные устройства, претендующие на роль загрузочных (НГМД, жесткие диски и CD-ROM ATA/ATAPI и SCSI, USB флэш-диски, сетевые адаптеры и т. п.), требуется механизм упорядочивания загрузочных устройств. Пользователь должен иметь возможность знакомиться со всем списком имеющихся устройств и выбирать порядок их опроса в загрузочной последовательности. Для этих целей фирмами Compaq, Phoenix и Intel в 1996 году была выпущена спецификация BIOS Boot Specification (BBS). Подробнее процедура загрузки, спецификация BBS и способы создания «самодельных» загрузочных устройств описаны в [6].

Сервисы и другие векторы прерываний BIOS

При инициализации таблицы прерываний BIOS отвечает за корректное заполнение части векторов, имеющих отношение к аппаратным средствам компьютера и сервисам BIOS. На некоторые из них могут быть просто установлены заглушки: вектор ссылается на код обработчика, содержащего единственную инструкцию возврата из прерывания — IRET. BIOS инициализирует *векторы прерываний* различных назначений:

- внутренних прерываний процессора (исключений), которые могут возникнуть в реальном режиме работы (об исключениях защищенного режима в основном заботится соответствующая операционная система);
- аппаратных прерываний, маскируемых и немаскируемых;
- вызовов функций ROM BIOS (16-битных сервисов);
- указателей на системные таблицы.

Внутренние прерывания:

- Int 00h — деление на 0;
- Int 01h — пошаговый режим;
- Int 03h — точка останова;
- Int 04h — переполнение;
- Int 06h — недопустимая команда 286+;
- Int 07h — вызов отсутствующего математического сопроцессора (Numeric Processor Unit, NPU).

Аппаратные прерывания:

- Int 02h — немаскируемое прерывание;
- Int 08h — таймер 8253/8254;
- Int 09h — клавиатура;
- Int 0Ah - IRQ2/9;
- Int 0Bh - IRQ3;
- Int 0Ch - IRQ4;
- Int 0Dh - IRQ5;
- Int 0Eh — IRQ6 (контроллер гибких дисков);
- Int 0Fh - IRQ7;
- Int 70h — CMOS-таймер;
- Int 71h — IRQ9 (перенаправлено на Int 0Ah);
- Int 72h - IRQ10;
- Int 73h- IRQ11;
- Int 74h— IRQ12 (контроллер мыши PS/2);
- Int 75h— IRQ13 (исключение сопроцессора);
- Int 76h— IRQ14 (контроллер жестких дисков);
- Int 77h- IRQ15.

ПРИМЕЧАНИЕ ----------
Прерывания Int 70h-77h имеют место только в AT.

Функции ROM BIOS (16-битные сервисы):

- Int 05h (F000:FF54h) — печать экрана;
- Int 10h — видеосервис (см. 10.6);
- Int 11h — чтение списка оборудования (слово из BDA 0040:0010h), возвращает в AX:
  - биты 15:14 — число обнаруженных LPT-портов (00 — 0, ..., 11 — 3);
  - бит 13 — резерв;
  - бит 12 — обнаружен игровой адаптер;
  - биты 11:9 — число обнаруженных COM-портов (000 — 0, ..., 111 — 7);
  - бит 8 — наличие контроллера DMA;
  - биты 7:6 — число обнаруженных НГМД (00 — 1,..., 11 — 4);
  - биты 5:4 — активный видеорежим (00 — резерв, 10 — 80-колоночный цветной, 01 — 40-колоночный цветной, 11 — монохромный);
  - биты 3:2 — размер ОЗУ на системной плате (теперь обычно 00);
  - бит 1 — присутствие математического сопроцессора;
  - бит 0 — присутствие дисководов;
- Int 12h — размер непрерывной памяти;

- Int 13h — дисковый сервис (блочный ввод-вывод, см. 9.11);
- Int 14h — обслуживание COM-портов (см. 16.1);
- Int 15h — AT-функции (системный сервис, функции определяются значением AH/AX):
  - 00-03h — управление и обмен данными с кассетным магнитофоном (были когда-то и такие «стримеры»!) на старых ПК;
  - 4fh — перехват событий клавиатуры (см. 11.1);
  - 53xxh — сервисы расширенного управления энергопотреблением (APM);
  - 8300h — запуск таймера, устанавливающего флаг в заданной ячейке (см. 4.6);
  - 8301h — сброс того же таймера;
  - 84h — джойстик (см. 11.6);
  - 86h — программируемая задержка (см. 4.6);
  - 87h — перемещение блока расширенной памяти;
  - 88h — получение размера расширенной памяти;
  - 89h — переключение в режим V86;
  - C0h — получение системной конфигурации, при успешном выполнении (CF = 0, AH = 0) ES:BX указывает на таблицу данных конфигурации (табл. 5.3);
  - 80-82h, 85h, 90h, 91h — функции многозадачных ОС (BIOS устанавливает заглушки);
- Int 16h — клавиатурный ввод-вывод (см. 11.1);
- Int 17h — обслуживание LPT-портов (см. 15.4);
- Int 18h — процедура восстановления при неудаченачальной загрузки (прежде — ROM-Basic);
- Int 19h — начальная загрузка;
- Int 1Ah — системное время, дата, будильник (см. 4.6) и 16-битные вызовы сервисов PCI (см. 14.7);
- Int 1Bh — обработчик нажатия клавиш Ctrl+Break;
- Int 1Ch — процедура User Timer Interrupt, вызываемая обработчиком Int 08h каждые 55 мс; BIOS устанавливает простую заглушку (IRET), но программы могут перехватывать это прерывание; на время отработки процедуры *все аппаратные прерывания запрещены* (кроме NMI);
- Int 33h — поддержка мыши;
- Int 4Ah — обработчик будильника пользователя,установленного функцией Int 1Ah (6) BIOS (см. 4.6); прерывание вызывается асинхронно, так что при возврате из процедуры все регистры и флаги должны быть в том же состоянии, что и при входе; BIOS ставит заглушку (IRET);
- Int 67h — EMS-функции.

Указатели на таблицы:

- Int 1Dh — видеопараметры;
- Int 1Eh — параметры дискет;
- Int 1Fh — знакогенератор CGA;
- Int 41h — параметры HDD 0;
- Int 46h — параметры HDD 1;
- Int 43h — знакогенератор EGA.

Как видно из приведенных списков, большинство векторов BIOS накладывается на область векторов 00-1Fh, зарезервированную фирмой Intel под внутренние прерывания и исключения процессоров. Во времена 8086 из них использовалось совсем малое количество, зарезервированной была объявлена вся указанная область. Тем не менее творцы IBM PC «влезли» вэту область, чтоосложни. ключения современными процессорами.

Таблица 5.3. Параметры системы

Смещение	Длина, байт	Поле
0	2	Длина таблицы, в байтах
2	1	Модель: FF — PC, FE или FB — XT, FD — PCjr, FC — AT, FF — неизвестная
3	1	Подмодель: PC, XT, PCjr, AT = 00; AT = 01, XT-286 = 02
4	1	Ревизия BIOS
5	1	Свойства: – бит 0 — резерв; – бит 1 = 0 — PC-type I/O channel; – бит 2 = 1 — Extended BIOS area allocated; – бит 3 — поддержка функций ожидания Int 15h(83xx, 86h); – бит 4 — вызов Int 15h (4Fh) обработчиком Int 09h; – бит 5 — наличие RTC; – бит 6 — наличие второго контроллера 8259A; – бит 7 — использование жестким диском канала DMA#3
6	4	Резерв

32-разрядные вызовы — BIOS32

Как говорилось ранее, традиционные сервисы BIOS работают в 16-разрядном режиме процессора, и ими можно пользоваться в реальном режиме, в режиме V86 и в малопривлекательном 16-разрядном защищенном режиме. Для процессоров 386+ оптимальным по эффективности является 32-разрядный защищенный режим. Для того чтобы из этого режима можно было пользоваться сервисами BIOS (правда, не всеми) без промежуточных переключений, по инициативе фирмы Phoenix ввели 32-разрядные вызовы BIOS32. Адрес точки входа BIOS- 32 заранее не известен, но известен способ его нахождения: в диапазоне адресов

памяти 0E0000-0FFFFFh на границе параграфов (младшие 4 бита адреса нулевые) ищется строка-сигнатура «_32_» (число 325F5F33h) заголовка, за которой следует физический адрес точки входа.

Сами сервисы вызываются дальними вызовами точки входа в сервис. Номер, параметры вызываемых функций и результаты передаются в регистрах процессора. Функции PCI BIOS вызываются с AX = B1xx (см. 14.7).

Области данных ROM BIOS — BDA

Помимо векторов прерываний BIOS в оперативной памяти имеет свою область данных — *BIOS data area*, начинающуюся с адреса 400h (сразу за таблицей прерываний). Этот адрес в сегментной модели адресации реального режима может быть представлен как 0000:0400h или 0040:0000h, что указывает на один и тот же физический адрес. BIOS может также использовать и расширенную область данных (Extended BIOS Data Area, *EBDA*), которая обычно располагается под верхней границей (640 Кбайт) стандартной памяти. На ее положение указывает слово по адресу 40:0Eh, а первый байт этой области идентифицирует ее размер в единицах килобайт. Эта область используется для различных семафоров и указателей, ее размер обычно не превышает 1 Кбайт.

В области памяти ROM BIOS имеется несколько стандартно расположенных ячеек, а также фиксированные точки входа в процедуры BIOS. Положение этих точек искусственно удерживается на тех же местах, где они были при рождении PC, но пользоваться ими как интерфейсом не рекомендуется. Вызов процедур по этим точкам позволяет обойти все перехваты векторов прерывания, в том числе и вирусные.

PnP BIOS

С появлением карт ISA PnP возникла необходимость упорядочивания возможностей их использования, и в 1994 году фирмы Compaq, Phoenix и Intel выпустили спецификацию Plug and Play BIOS Specification, описывающую следующие расширения возможностей традиционной BIOS:

- распределение ресурсов и разрешение конфликтов на этапе выполнения POST;
- слежение за перехватом вектора загрузки Int 19h (традиционные ПЗУ расширения BIOS могли его неконтролируемо переопределять);
- введение контролируемого механизма удаленной загрузки (RPL);
- поддержка конфигурирования в рабочем режиме;
- уведомления о динамическом изменении конфигураций (подключении и отключении устройств).

Флэш-BIOS

Флэш-память широко применяется в качестве носителя BIOS в современных компьютерах. В принципе, это позволяет даже конечному пользователю обновлять версию BIOS, не вызывая высокооплачиваемых специалистов и оператив

но получая необходимые файлы через Интернет. Наиболее эффективно применение флэш-памяти с выделенным блоком загрузчика. Блок загрузчика после программирования может быть аппаратно защищен от перезаписи и работать в режиме ROM. Это позволяет его использовать как неизменную часть BIOS, обеспечивающую минимальные условия для загрузки утилиты программирования основного блока. Основной блок хранит главную часть BIOS, которая при необходимости может заменяться новыми версиями. В случае некорректности новой запрограммированной версии всегда есть путь к отступлению, обеспечиваемый неизменяемым блоком загрузчика.

Микросхемы семейства Boot Block (или другие микросхемы с небольшим размером стираемого сектора) помимо BIOS в блоках параметров могут хранить и конфигурационную информацию (ESCD системы PnP, конфигурацию устройств EISA и MCA). Применение микросхем большого объема позволяет помимо кода BIOS хранить и дополнительный резидентный код. В портативных компьютерах во флэш-память может помещаться и ядро ОС, что позволяет экономить энергию за счет сокращения обращений к диску.

Гибкость системы, обретаемая при использовании флэш-памяти как носителя BIOS, имеет и негативные обратные стороны — возможность повреждения в случае неудачной записи или записи неподходящей версии и появление новой и достаточно благодатной почвы для вирусов, которые могут незаметно переписать код BIOS в своих диверсионных целях. В связи с этим актуальна защита BIOS *от несанкционированного изменения.* Разные поколения флэш-памяти имеют свою специфику организации защиты:

- микросхемы Bulk Erase имеют защиту от модификации всего объема подачей напряжения V_{PP} = 5 В (V_{PP} = 0 для микросхем с стиранием 5 В);
- микросхемы Boot Block 12 В позволяют иметь дополнительную защиту загрузочного блока, управляемую напряжениями на входах V_{PP}P# (табл. 5.4);
- микросхемы Boot Block 5/12 В SmartVoltage позволяют управлять защитой напряжениями на входах V_{PP} и WP# (табл. 5.5), причем при V_{PP} = 5 В стирание и программирование возможны;
- аппаратное управление защитой микросхем с одним напряжением питания осуществляется только сигналами WP#, но у ряда микросхем они отсутствуют.

Таблица 5.4. Защита флэш-памяти Boot Block 12 В

Уровень напряжения		Уровень защиты
V_{PP}	RP#	
V_{PPL}	X	Все блоки защищены
V_{PPH}	V_{IL}	Все блоки защищены (Reset)
V_{PPH}	V_{IH}	Защищен только блок Boot Block
V_{PPH}	V_{HH}	Все блоки доступны

V_{PPL} = 5 В, V_{PPH} = 12 В ± 10 %.

V_{IL} и V_{IM} - низкий (<1,4 В) и высокий (>2,4 В) уровни логических сигналов. V_{HH}- 12 В ± 10 %.

Таблица 5.5. Защита флэш-памяти SmartVoltage Boot Block

Уровень напряжения			Уровень защиты
V_{PP}	RP#	WP#	
V_{IL}	X	X	Все блоки защищены
V_{PPLK}	V_{IL}	X	Все блоки защищены (Reset)
V_{PPLK}	V_{HH}	X	Все блоки доступны
V_{PPLK}	V_{IH}	V_{IL}	Защищен только Boot Block
V_{PPLK}	V_{IH}	V_{IH}	Все блоки доступны

V_{PPLK} = 2,5 В; V_{IL} и V_{IH} – низкий (<1,4 В) и высокий (>2,4 В) уровни логических сигналов. V_{HH} = 12 В ± 10 %.

У микросхем, у которых нет вывода у сигнала WP#, этот сигнал имеет внутреннее соединение с низким логическим уровнем (что исключает последнюю строку таблицы).

Реальные варианты использования защиты зависят от схемных решений конкретной системной платы. Системные платы, допускающие применение в качестве BIOS как ПЗУ, так и флэш-памяти (возможно, различных моделей), имеют набор джамперов, коммутирующих сигналы, поступающие на входы RP#, WP#, Vpp и WE# флэш-памяти. Их применение (и рекомендации по установке, приводимые в документации) не всегда отвечает требованиям безопасности. Системы, ориентированные на защитные свойства блока загрузки, имеют джампер или переключатель для восстановления BIOS (Boot recovery) после неудачной модификации. В зависимости от схемных решений он может либо аппаратно переключать адреса, либо анализироваться программно. В случае с преобразованиями адресов при нормальной работе точка входа по сбросу процессора (FFFF0h) попадает в основной блок, а в режиме восстановления точка входа попадает в область блока загрузки. Простейший способ преобразования — инверсия адреса A16 в нормальном режиме и его прямая подача в режиме восстановления. В случае программного анализа состояния переключателя блок загрузки всегда находится в верхних адресах, а по результату считывания положения переключателя код основного блока либо получает управление (нормальный режим), либо игнорируется (режим восстановления). Применение микросхем с симметричной архитектурой и небольшим размером сектора (например, SST29EE010) позволяет размещать блок загрузки (группу защищенных секторов) в произвольном месте памяти.

5.3. Расширения ROM BIOS

Платы адаптеров, устанавливаемые в слоты шин расширения, могут иметь микросхемы ПЗУ своей программной поддержки — *Additional ROM BIOS* (дополнительные модули ROM BIOS), они же Expansion ROM. Их используют графические адаптеры EGA/VGA/SVGA, некоторые контроллеры жестких дисков, контроллеры SCSI, сетевые адаптеры с удаленной загрузкой и другие периферийные устройства. Для этих модулей в пространстве памяти зарезерви-

рована область C8000h-F4000h. POST сканирует эту область с шагом 2 Кбайт в поисках дополнительных модулей BIOS на завершающем этапе выполнения (после загрузки векторов прерываний указателями на собственные обработчики). Дополнительный модуль BIOS графического адаптера (EGA, VGA, SVGA...) имеет фиксированный адрес C0000 и инициализируется раньше (на шаге инициализации видеоадаптера).
Дополнительный модуль ROM BIOS должен иметь заголовок, выровненный по границе 2-килобайтной страницы памяти; формат заголовка ПЗУ приведен в табл. 5.6.

Таблица 5.6. Заголовок модуля дополнительного ПЗУ

Смещение	Длина	Назначение
0	2	Сигнатура (признак начала модуля): байт 0 = 55h, байт 1 = AAh
2	1	Длина, указанная в блоках по 512 байт
3	3	Точка входа процедуры инициализации, заканчивающейся дальним возвратом Ret Far (вызывается инструкцией Far Call во время теста POST). Обычно здесь располагается трехбайтная инструкция JMP на начало процедуры
6–17h		Резерв
18h	2	Указатель на структуру данных PCI (только для карт PCI)
1Ah	2	Указатель на структуру расширенного заголовка карт ISA PnP

В традиционном заголовке присутствовали только первые 3 поля, указатели на структуры PCI и ISA PnP ввели позже. Корректным считается модуль, начинающийся с признака AA55h (значения слова с учетом порядка следования байтов) и нулевой суммой (по модулю 256) всех байтов в объявленной области (реальная длина модуля может превышать объявленную, но байт контрольной суммы, естественно, должен входить в объявленную область).
В случае обнаружения корректного модуля POST дальним вызовом (Call Far) вызывает процедуру инициализации модуля, начинающуюся с 3-го адреса заголовка модуля. Ответственность за ее корректность полностью ложится на разработчика. Процедура может переопределять векторы прерываний, обслуживаемых BIOS. Переопределив на себя Bootstrap (Int 19h), можно получить управление при загрузке, что и использовалось, например, для удаленной загрузки компьютеров через локальную сеть (remote boot reset). Если стандартное продолжение процедуры загрузки не требуется, а дополнительный модуль представляет собой, например, управляющую программу для какого-либо оборудования, вместо процедуры инициализации в ПЗУ может находиться и основная программа, не возвращающая управление системной последовательности POST. С внедрением технологии PnP способ включения устройств с ПЗУ в процесс загрузки упорядочили [6].
Процедура инициализации и программная поддержка устройства в ПЗУ должны быть написаны таким образом, чтобы им были безразличны абсолютные адреса, по которым они размещаются в пространстве памяти. На картах расширения, как правило, имеются аппаратные средства изменения базового адреса,

а иногда и размера ПЗУ (джамперы или программно-управляемые переключатели). Это позволяет бесконфликтно разместить модули ПЗУ нескольких установленных карт.

По сравнению с традиционным способом использования ПЗУ, когда оно, будучи разрешенным, постоянно присутствует в области памяти, появился более рациональный способ подключения расширений ROM BIOS, основанный на модели *DDIM* (Device Driver Initialization Model — модель инициализации драйвера устройств). Этот способ, а также особенности ПЗУ для карт ISA PnP и PCI описаны в [6].

5.4. DMI BIOS

Интерфейс управления настольными компьютерами (Desktop Management Interface, DMI) служит для удаленного администрирования компьютеров. Поддержка DMI введена в BIOS большинства современных компьютеров.

Идеи централизованного управления рабочими станциями развиваются многими фирмами-производителями компьютеров и сетевого оборудования. В 1992 г. компании Digital, Hewlett-Packard, IBM, Intel, Microsoft, Novell, Sun и Synoptics организовали группу DMTF (Distributed Management Task Force — рабочая группа для решения задач распределенного управления компьютерами). Позже к ним примкнули фирмы Apple, AST, Compaq, Dell, Symantec и ряд других, и теперь под флагом DMTF объединены более 400 производителей компьютеров и программного обеспечения. В 1994 году была выпущена первая спецификация интерфейса DMI, которая была сугубо локальной и не предусматривала управления по сети. В 1996 году вышла спецификация DMI 2.0, в которую уже была включена возможность дистанционного управления по сети. Основная идея DMI — всеобщий учет и контроль (в смысле возможности принудительного управления).

Настольный компьютер представляет собой набор аппаратных средств (hardware), встроенного (firmware) программного обеспечения (например, ROM BIOS) и загружаемого (software) программного обеспечения (операционная система и прикладное ПО). Для того чтобы выполнить какие-либо административные действия (например, установить или обновить сетевое ПО), в общем случае может понадобиться информация по любому элементу этого набора. Интерфейс DMI позволяет администратору, не подходя к рабочему месту пользователя, узнать о его компьютере все. Приведем в качестве примера выдержки из списка параметров, сообщаемых DMI BIOS производства фирмы Award:

- ROM BIOS — название, версия, производитель, дата выпуска, размер, поддерживаемые шины, способ загрузки и т. д.;
- система (компьютер) — название, производитель, версия, серийный номер;
- системная плата — то же;
- корпус (шасси) — производитель, заводской и инвентарный номера;
- процессор — тип, семейство, идентификатор, версия, тип сокета, частота ядра (максимальная и текущая), частота шины;

- контроллер памяти — поддерживаемые типы памяти, допустимое количество модулей памяти (слотов), напряжение питания, быстродействие модулей, методы обнаружения и исправления ошибок;
- модули памяти — тип слота, используемые банки, скорость, тип памяти, размер, контроль/исправление ошибок, наличие обнаруженных ошибок;
- кэш-память — тип, размер (допустимый и текущий), скорость, допустимые типы памяти;
- порты (COM, LPT, Mouse...) — логические параметры, а также тип коннекторов, внутренних и внешних (DB-9 или DB-25 для COM-портов), надпись на шильдике;
- слоты шин расширения — тип (ISA, PCI...), разрядность шины, частота, напряжение, обозначение и т. д.;
- встроенная периферия (графический, аудио-, видеоконтроллер) — подробная информация;
- журнал системных событий.

Все устанавливаемые адаптеры и контроллеры должны сообщать подобную детальную информацию о себе. Также представляться должны и операционная система со всеми драйверами, и приложения, поддерживающие DMI.

Помимо полной инвентаризации DMI предоставляет возможность дистанционного запуска на пользовательском компьютере процедур, используя протокол RPC (Remote Procedure Call — удаленный вызов процедур). Удаленно могут запускаться, например, утилиты обслуживания дисков, антивирусные программы, но самое заманчивое — удаленный запуск процедур установки и обновления операционных систем и прикладного ПО.

Принудительное дистанционное администрирование желательно производить при отсутствии пользователя (во внерабочее время). Для этого, естественно, необходимо включить его компьютер, что можно сделать, не подходя к компьютеру. Современные сетевые карты (см. 13.2) могут выполнять удаленное пробуждение (remote wake up) или пробуждение по сети (wake on LAN). Принудительное администрирование может выполняться в ночные часы или в выходные даже без участия администратора — он может запускать процедуры через планировщик заданий. Пользователь, пришедший на работу на следующий день, получит массу впечатлений от обновленной версии ПО, если, конечно, накануне он предусмотрительно не обесточит свой компьютер механическим выключателем.

Интерфейс DMI-2 и удаленное пробуждение поддерживают многие сетевые карты, предназначенные для клиентских машин (для серверов такой сервис слишком опасен). В спецификации Microsoft на аппаратные средства (PC99 Hardware Design Guide) интерфейс DMI не упоминается, но там речь идет об инициативе WfM (Wired for Management — подключение проводами для управления), а за подробностями отсылают на сайты http://www.intel.com/ managedpc/spec.htm (версия 1.0) и http://developer.intel.com/ial/wfm/ (версия 2.0).

5.2. Интерфейс ACPI

Интерфейс ACPI (Advanced Configuration and Power Interface — расширенный интерфейс конфигурирования и питания) представляет собой довольно сложную комбинацию функций, часть из которых раньше возлагались на относительно независимые системы PnP (в части конфигурирования) и APM. Спецификация ACPI разработана фирмами Compaq, Intel, Microsoft, Phoenix и Toshiba для стандартизации механизмов OSPM (Operating System-directed configuration and Power Management), позволяющих операционной системе управлять конфигурированием и энергопотреблением устройств и компьютера в целом. ACPI определяет аппаратные и программные интерфейсы, а также наборы данных (таблицы).

В ACPI различают *глобальные состояния системы* (global system state) по следующим критериям: работают ли приложения, насколько длительна задержка их реакции на внешние события, каков уровень потребления, требуется ли перезагрузка ОС для возврата в рабочий режим, можно ли разбирать компьютер и можно ли входить в это состояние и выходить из него электронным способом (не механическим выключателем):

- *G3* (Mechanical Off) — механическое отключение. Приложения не работают, потребления нет (кроме как от батарейки CMOS RTC), для перевода в рабочее состояние требуются механическое включение питания и загрузка ОС. Компьютер можно смело разбирать только в этом состоянии.
- *G2/S5* (Soft Off) — программное отключение. Приложения не работают, потребление минимально, включиться в рабочее состояние может программно (и от кнопки), требуется загрузка ОС.
- *G1* (Sleeping) — «сон», в котором компьютер кажется выключенным. Пользовательские процессы не исполняются, потребление малое, но переход в рабочее состояние может и не требовать загрузки ОС. Для этого весь контекст компьютера (состояние всех устройств и памяти) должен быть сохранен (большая часть — аппаратно, остальное — программно). Системная плата при этом получает «дежурное» питание, ее обесточивание ведет к потере контекста.
- *GO,* (Working) — рабочее состояние, в котором компьютер работает на полную мощность. При этом периферийные устройства могут динамически менять свое состояние, балансируя потребление в соответствии с требованиями по производительности.

Состояния потребления устройств (device power state) различаются по потребляемой мощности, сохраняемой части контекста устройства, действиям драйвера, необходимым для приведения устройства в рабочее состояние, и времени перевода в полностью рабочее состояние. За контекст устройства отвечает ОС, которая может восстановить теряемую часть, а то и полностью проинициализировать устройство, выполнив его сброс:

- *D3* (Off) — полностью обесточенное и неработающее устройство, не хранящее никакого контекста и не декодирующее свой адрес. Для включения требует полной инициализации.

- *D2* и *D1* — состояния пониженного энергопотребления, специфичные для каждого класса устройств. Могут выполняться не все функции и сохраняться не весь контекст. В состоянии D2 потребление и объем выполняемых функций и сохраненного контекста меньше, чем в D1, а время, требуемое для перевода в рабочий режим — больше.
- *DO* (Fully-On) — полностью рабочее (активное) состояние, в котором устройство постоянно хранит весь свой контекст.

Глобальное состояние сна (G1) имеет набор градаций «глубины».

- *S1* — неглубокий сон с быстрым пробуждением, весь контекст хранится в своих устройствах.
- *S2* — состояние с почти таким же быстрым пробуждением, но контекст процессора и кэш-памяти в них самих не сохраняется (за его сохранение и восстановление отвечает ОС).
- *S3* — состояние с небольшой задержкой пробуждения, в котором на системной плате информация сохраняется только в ОЗУ (процессор, кэш, чипсет и устройства ничего не помнят).
- *S4* (Non-Volatile Sleep) — энергонезависимый сон. По команде перехода в это состояние выполняется полное сохранение контекста в файле или энергонезависимой памяти, сохраняется и маркер-указатель на сохраненный контекст. Когда устройство переходит в рабочее состояние из состояния *G3* или G2, в начале загрузки ОС проверяется, имеется ли корректный сохраненный контекст. Если контекст есть, и конфигурация компьютера не изменилась (не изменился объем ОЗУ и на месте все несъемные устройства), то вместо перезагрузки выполняется восстановление контекста — все задания начнут работать с той точки, в которой их «отправили спать». «Спать» компьютер с сохраненным контекстом может сколь угодно долго (питание выключено).
- 55 (Soft Off) — состояние программного отключения, формально называемое сном, но, в отличие от состояния *S4*, никакой контекст не сохраняется. Введение этого состояния используется как признак, по которому при включении питания определяется, нужно ли искать сохраненный контекст для восстановления (после *S4)* или выполнять полную загрузку ОС (после 55).

Состояния потребления процессора различаются по задержке реакции, хранению данных во внутреннем кэше и восприятию внешних циклов слежения за обращениями к памяти (для поддержания когерентности кэша и памяти):

- *CO* — исполнение инструкций на полной скорости.
- *C1* — понижение потребления с минимальной задержкой отработки обращений, такое что ОС и приложения не замечают отличия от *CO.*
- *C2* — еще большее снижение потребления, задержка отработки обращений существенна, и для перехода в рабочее состояние требуются средства ACPI.
- C3 — максимальное снижение потребления, состояние кэша сохраняется, но слежение не выполняется. Для обеспечения когерентности памяти после выхода из этого состояния требуются усилия со стороны ОС.

Для процессора и устройств различают *состояния уровня производительности:*

- *P0* — состояние максимальной производительности.
- *P1*... Pn — состояния с убывающим уровнем производительности. Устройства могут поддерживать различное число уровней (*n* < 16).

Интерфейс ACPI предоставляет операционной системе возможность прямого (и эксклюзивного) управления потреблением (OSPM) и конфигурированием устройств системной платы. При запуске OSPM забирает эти функции от старых интерфейсов BIOS (APM BIOS, PnP BIOS) и берет на себя ответственность за обработку событий конфигурирования устройств системной платы, управление питанием, производительностью и температурным состоянием системы в соответствии с предпочтениями пользователя и требованиями приложений. Ниже перечислены области, охватываемые спецификацией ACPI:

- Управление питанием системы *(G0 ... G2*). ACPI определяет механизмы, переводящие компьютер (целиком) в спящее состояние и из него. Также определяется общий механизм, которым любое устройство может разбудить компьютер.
- Управление питанием устройств *(D0 ... D3).* Таблицы ACPI описывают устройства системной платы, их состояния потребления, питание подключенных к ним устройств и управляют переводом устройств в различные состояния потребления. Это позволяет ОС переводить устройства в состояние малого потребления в соответствии с используемыми приложениями.
- Управление питанием процессора. Когда ОС находится в состоянии ожидания (но не спит), она использует команды ACPI для перевода процессора в состояние минимального потребления *(CO ... C3)*.
- Управление производительностью процессора и устройств. Когда система активна, OSPM управляет переводом устройств и процессора в различные состояния производительности *(P0* ... Pn), обеспечивая баланс между производительностью и энергосбережением с учетом различных требований (акустический шум, видимость изображения).
- Plug-and-Play. ACPI определяет информацию, используемую для нумерации и конфигурирования устройств системной платы. Эта информация организуется иерархически так, что для событий вроде подключения и отключения док-станции или съемных устройств ОС точно и заранее узнает, на какие устройства эти события повлияют.
- Системные события. ACPI обеспечивает общий механизм оповещения, используемый для таких событий, как изменение температуры (перегрев каких-либо устройств), управление потреблением, подключение к док-станции, установка и снятие съемных устройств и т. п. Механизм гибкий, он не задает жестких требований к способу сигнализации о событиях ядру логики чипсета.
- Управление батареями. Для этого устройство с батарейным (аккумуляторным) питанием должно иметь специальный интерфейс (Smart Battery или Control Methood Battery Interface), позволяющий следить за состоянием (уровнем зарядки/разрядки) батарей.

- Термоконтроль. Поскольку ОС управляет потреблением процессора и устройств, в ACPI имеется простая (но масштабируемая) модель, позволяющая разработчику определять температурные зоны со своими индикаторами и методы охлаждения этих зон.
- Встроенные контроллеры. ACPI определяет стандартные аппаратные средства и программный интерфейс взаимодействия между шинным нумератором ОС и встроенными контроллерами[1]. Таким образом, стандартный драйвер позволяет ОС и приложениям использовать специфические возможности, предоставляемые встроенными контроллерами.
- Контроллер SMBus (вспомогательной последовательной шины системного управления). ACPI определяет стандартные аппаратные средства и программный интерфейс взаимодействия между шинными драйверами ОС и контроллером SMBus. Это позволяет создать для ОС стандартный драйвер, который непосредственно взаимодействует с устройствами, подключенными к SMBus. В свою очередь, это дает возможность ОС и приложениям использовать возможности общения с устройствами по SMBus для специфических функций управления:

В режиме ACPI встроенное ПО устройств (firmware) и другое ПО не должны манипулировать конфигурированием системных ресурсов, потреблением, производительностью и термоконтролем независимо от OSPM. Всю ответственность за координацию в этих областях берет на себя OSPM. Однако для защиты от катастрофического перегрева (если быстродействия OSPM недостаточно) устройства могут иметь и локальные аварийные средства «пожаротушения». Они также должны выдавать сообщения ACPI (если, конечно, будет кому их принять — при катастрофическом перегреве многие процессоры просто автоматически выключаются).

Еще раз напомним, что требуется для реализации вышеописанных возможностей управления.

- Системная плата, аппаратно поддерживающая ACPI (естественно, в конструктиве ATX, в котором предусмотрено программное включение-отключение питания).
- BIOS с поддержкой ACPI.
- Подключаемые к системной плате устройства, поддерживающие ACPI.
- ОС, поддерживающая ACPI (например, Windows 98/ME/2000).

Системные платы и устройства, поддерживающие PnP и APM, теперь считаются устаревшими.

[1] Примером встроенного контроллера является контроллер традиционной клавиатуры (и мыши PS/2). Здесь микроконтроллер по своей внутренней программе выполняет низкоуровневое обслуживание подключенных к нему устройств (принимает скан-коды и посылает управляющие команды клавиатуре) и сигнализирует хосту (центральному процессору) о событиях, требующих его внимания (прерывания по нажатии-отпускании клавиш). Хост взаимодействует со встроенным контроллером через пару портов (60h и 64h), посылая команды и считывая состояние и данные, а также получает запрос прерывания. Встроенные контроллеры разгружают центральный процессор от рутины по обеспечению ввода-вывода.

Часть II

Ядро компьютера — системная плата, процессор и память

ГЛАВА 6

Системная плата

Системная (system board), или материнская (motherboard), плата персонального компьютера является основой системного блока, определяющей архитектуру и производительность компьютера. На ней устанавливаются следующие обязательные компоненты:

- Процессор(ы), а для процессоров 8086-80386 и сопроцессор.
- Память: постоянная (ROM или Flash BIOS), оперативная (DRAM), а для не самых новых процессоров и кэш (SRAM).
- Обязательные системные средства ввода-вывода: контроллеры клавиатуры, прерываний, DMA, таймеры, CMOS RTC, средства управления динамиком.
- Интерфейсные схемы и разъемы шин расширения.
- Кварцевый генератор синхронизации.
- Схема формирования сброса системы по сигналу PowerGood от блока питания или кнопки Reset.
- Схема управления блоком питания (отсутствует в платах конструктива AT).
- Регуляторы напряжения (Voltage Regulation Module, VRM). Как правило, это управляемые импульсные преобразователи напряжения +5 или +12 В (для ATX12V) в более низкое, требуемое для питания современных низковольтных процессоров, памяти и интерфейсов.
- Средства мониторинга состояния системного блока: измерители скорости вращения вентиляторов и температуры процессора и других «горячих» компонентов; измерители питающих напряжений; сигнализаторы несанкционированного доступа и т. п. Эти средства позволяют программно (через загружаемое ПО или меню CMOS Setup) снимать показания измерителей и датчиков, а также при должной настройке вырабатывать прерывание, сигнализирующее о критических событиях, и даже принимать экстренные меры (вплоть до выключения питания при перегреве). Средства мониторинга присутствуют практически на всех современных системных платах.

Помимо этих сугубо обязательных средств на большинстве современных системных плат устанавливают и контроллеры НГМД, интерфейсы COM- и LPT- портов, 4-6 портов USB, пару каналов ATA и/или 2-4 порта SATA. Этот набор по нынешним меркам является обязательным для «голых» системных плат, часто к нему добавляют и контроллер FireWire (1394), а также адаптер локаль

ной сети (Ethernet 10/100 и даже 1000 Мбит/с). Существуют и системные платы с интегрированными видео-, аудио- и прочими устройствами, обеспечивающие полную функциональность компьютера без всяких карт расширения. При необходимости интегрированные устройства могут быть заменены устройствами, установленными в слоты расширения (правда, иногда не все устройства системной платы можно полностью отключить). Размещение на системной плате контроллеров, требующих интенсивного обмена данными (ATA, SCSI, графический адаптер), позволяет использовать преимущества локального подключения к шине памяти и процессора. Цель размещения других контроллеров на системной плате — сокращение общего числа плат компьютера. Какая плата лучше — «голая» или с интегрированной периферией, — зависит от назначения компьютера. Интегрированные видео- и аудиоустройства, как правило, по своим параметрам являются не выдающимися, но вполне удовлетворяющими запросам многих пользователей. Компьютер на интегрированной системной плате может оказаться дешевле, чем собранный из конструктора «сделай сам». Компьютер на «голой» плате более гибок в плане модернизации, однако при его сборке могут возникнуть проблемы совместимости компонентов, которые на интегрированных платах уже решены их разработчиками и изготовителями.

Системные платы первых ПК, выполненных на процессорах 8088/86, помимо процессора содержали несколько периферийных БИС (контроллеры прерываний, прямого доступа к памяти, контроллер шины) и связующую логику на микросхемах малой и средней степени интеграции. Современные платы исполняются на основе *чипсетов* — наборов из нескольких БИС, реализующих все необходимые функции связи основных компонентов — процессора, памяти и шин расширения. Чипсет определяет возможности применения различных типов процессоров, основной и кэш-памяти, а также ряд других характеристик системы, наиболее важных в плане ее функциональности и перспектив модернизации. Тип чипсета существенно влияет и на производительность — при одинаковых установленных компонентах (процессор, память, графический адаптер и жесткий диск) производительность компьютеров, собранных на разных системных платах (читай: чипсетах), может различаться на 30 %.

6.1. Архитектура системной платы

По мере «взросления» компьютеров постоянно расширяются функции чипсета системной платы и изменяются подходы к его построению. В задачу чипсетов для 80286/386 входили увязка шины процессора с относительно несложным контроллером памяти и подключение к этой связке шины (E)ISA, на которой располагались все устройства. Постепенно стала усложняться подсистема памяти — появился кэш на системной плате, а потом к нему добавился встроенный кэш процессора. Для процессоров класса 486 производительности шины (E)ISA оказалось уже недостаточно, и появились новые шины. Шина VLB, как просто физически оформленная разъемом системная шина процессора класса 486, особых хлопот чипсету не доставляла. Однако появилась шина PCI, для которой пришлось строить мост от системной шины. Поначалу ее называли

«пристроечной» (mezzanine bus), но вскоре она надолго стала центральной шиной, вокруг которой компоновались все остальные элементы. Ее центральное место не оспаривалось, поскольку шина PCI имела высокую производительность — 132 Мбайт/с. Традиционно на схемах шину PCI изображают посередине, как экватор. Процессор и память (вместе с кэш-памятью) изображают выше — «севернее», а шину ISA и все устройства, подключаемые к PCI и ISA, изображают ниже — «южнее экватора». Соответствующие части чипсета получили укоренившиеся названия *северных* (north) и *южных* (south). Созвучное слово «серверный» относится к чипсету, ориентированному на применение в компьютерах-серверах.

Архитектура системной платы прошла путь от шинно-мостовой к хабовой, особняком держится архитектура HyperTransport. Независимо от архитектуры системной платы и физической реализации соединений все современные периферийные устройства (или контроллеры и адаптеры их интерфейсов) представляются *логическими устройствами* (точнее, функциями). Стандартный набор атрибутов PCI (конфигурационным пространством со стандартными заголовками) обеспечивает удобный единый интерфейс конфигурирования устройств (распределения системных ресурсов). Этот интерфейс поддерживается и в PCI-X, и в PCI-E; он учитывается и в HyperTransport. Традиционные (legacy) устройства (PIC 8259A, DMA 8237A, COM- и LPT-порты и другие аксессуары PC) в плане конфигурирования держатся особняком — их конфигурация является статической и не меняется на протяжении более двух десятков лет.

Шинно-мостовая архитектура

В шинно-мостовой архитектуре имеется центральная магистральная шина, к которой остальные компоненты подключаются через мосты. В роли центральной магистрали сначала выступала шина (E)ISA, затем ее сменила шина PCI. Шинно-мостовая архитектура чипсетов просуществовала долгое время и пережила много поколений процессоров (от 2-го до 7-го). Перемещение вторичного кэша с системной платы на процессор (P6 и Pentium 4 у Intel и K7 у AMD) несколько упростило северную часть чипсета — в ней не надо управлять статической кэш-памятью, а остается лишь обеспечивать когерентность процессорного кэша с основной памятью, доступ к которой возможен и со стороны шины PCI.

Шина PCI в роли главной магистрали удержалась недолго: видеокартам с 3D-акселератором ее пропускной способности, разделяемой между всеми устройствами, оказалось недостаточно. Тогда и появился порт AGP как выделенный мощный интерфейс между графическим акселератором и памятью (а также процессором). При этом задачи северного моста усложнились: контроллеру памяти приходится работать уже на три фронта — ему посылают запросы процессор(ы), мастера шины PCI (и ISA, но тоже через PCI) и порт AGP. Пропускная способность AGP в режиме 2x/4x/8x составляет 533/1066/2133 Мбайт/с, так что шина PCI по производительности стала уже второстепенной. Однако в шинно-мостовой архитектуре она сохраняет свою роль магистрали подключения всех периферийных устройств (кроме графических). В качестве мощного представителя шинно-мостовой архитектуры можно рассматривать чипсет AMD-

760 (рис. 6.1). Здесь имеются первичная шина PCI на 64 бит и 66 МГц, являющаяся «экватором», и вторичная шина для подключения рядовой периферии.

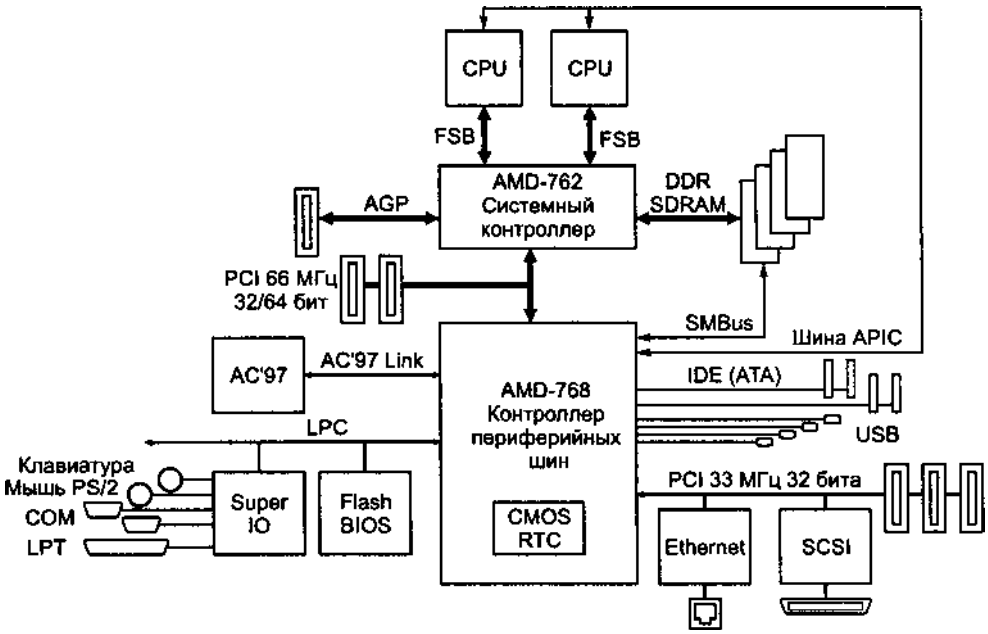


Рис. 6.1. Шинно-мостовая архитектура на примере AMD-760

Шина, к которой подключается множество устройств, является узким местом по ряду причин. Во-первых, из-за большого числа устройств, подключенных (электрически) к шине, не удается поднять тактовую частоту до уровня, достижимого в двухточечных соединениях. Во-вторых, шина, к которой подключается множество разнотипных устройств (особенно расположенных на картах расширения), обременена грузом обратной совместимости со старыми периферийными устройствами. Например, предусмотренные возможности повышения производительности PCI используются не всегда: расширение разрядности до 64 бит обходится слишком дорого (большое число проводников порождает свои проблемы), а повышение частоты до 66 МГц для шины возможно лишь если все ее абоненты поддерживают эту частоту. Достаточно установить одну «простую» карту PCI, и производительность центральной шины падает до начальных 133 Мбайт/с. То же можно сказать и о PCI-X: достаточно подключить к ней одно устаревшее устройство PCI, и все протокольные усовершенствования будут отменены.

Хабовая архитектура

С введением высокоскоростных режимов UltraDMA (ATA/66, ATA/100, а затем и ATA/133) связь двухканального контроллера IDE с памятью через шину PCI стала уже слишком сильно нагружать эту шину. Кроме того, появились вы

сокоскоростные интерфейсы Gigabit Ethernet, FireWire (100/200/400/800 Мбит/с) и USB 2.0 (480 Мбит/с). Ответом на эти изменения в расстановке сил стал переход на *хабовую архитектуру чипсета.* В данном контексте *хабы* — это специализированные микросхемы, обеспечивающие передачу данных между своими внешними интерфейсами. Этими интерфейсами являются «прикладные» интерфейсы подключения процессоров, модулей памяти, шин расширения и периферийные интерфейсы (ATA, SATA, USB, FireWire, Ethernet). Поскольку к одной микросхеме все эти интерфейсы не подключить (слишком сложна структура и много требуется выводов), чипсет строится, как правило, из пары основных хабов (северного и южного), связанных между собой высокопроизводительным каналом.

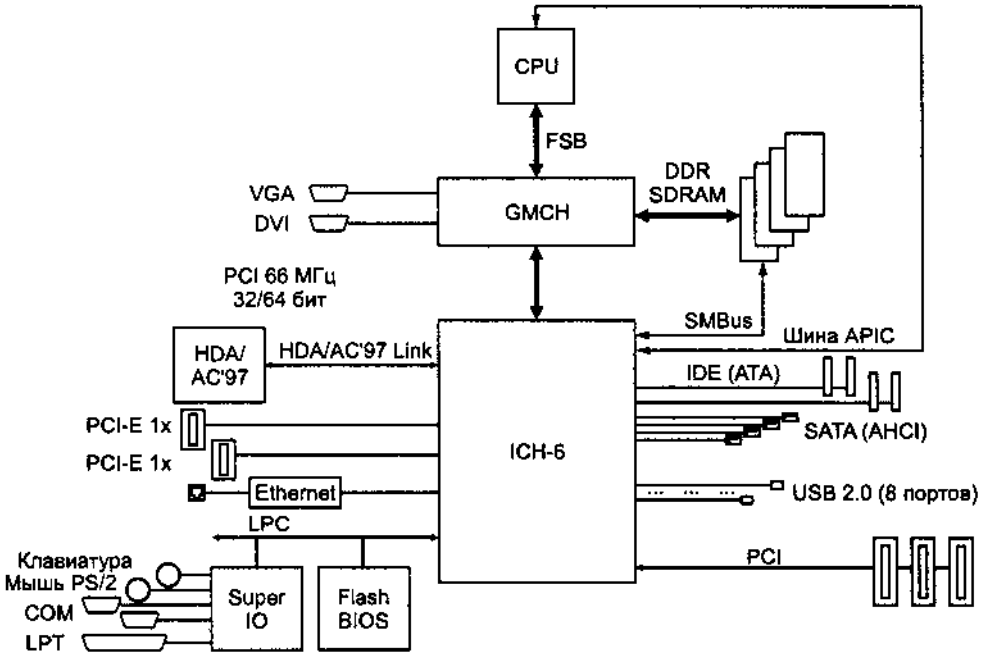


Рис. 6.2. Хабовая архитектура на примере чипсета Intel с ICH-6

Северный хаб чипсета выполняет те же функции, что и северный мост шинно-мостовой архитектуры: он связывает шины процессора, памяти и порта AGP. Однако на южной стороне этого хаба находится уже не шина PCI, а высокопроизводительный интерфейс связи с южным хабом (рис. 6.2). Пропускная способность этого интерфейса составляет 266 Мбайт/с и выше, в зависимости от чипсета. Если чипсет имеет интегрированную графику, то в северный хаб входит и графический контроллер со всеми своими интерфейсами (аналоговыми и цифровыми интерфейсами дисплея, шиной локальной памяти). Чипсеты с интегрированным графическим контроллером могут иметь внешний порт AGP, который становится доступным при отключении встроенного графического контроллера. Есть чипсеты, у которых порт AGP является чисто внутрен

ним средством соединения встроенного контроллера, и внешний графический контроллер к ним может подключаться только по шине PCI.

С появлением PCI-E архитектура не слишком изменилась: северный хаб (мост) вместо порта AGP теперь предлагает высокопроизводительный (8x или 16x) порт, а то и пару портов PCI-E для подключения графического адаптера. Маломощные (1x) порты PCI-E могут предоставляться как северным, так и южным хабами (это решает разработчик чипсета). В последнем случае корневой комплекс PCI-E (см. 14.10) «расползается» по двум микросхемам чипсета, связанным между собой «фирменным» интерфейсом. Использования PCI-E как единой коммуникационной базы внутри чипсета пока не наблюдается.

Архитектура HyperTransport

Технология (архитектура) HyperTransport (HT) задумывалась как альтернатива шинно-мостовой архитектуре системных плат. Технология разработана компаниями AMD, Apple Computers, Broadcom, Cisco Systems, NVIDIA, PMC- Sierra, SGI, SiPackets, Sun Microsystems, Transmeta. Первый релиз вышел в 2001 году, в 2003-м — версия 1.10. Прежнее кодовое название — LDT (Lighting Data Transport).

Основная идея HT — замена шинного соединения компонентов (периферийных устройств) системой двухточечных встречно направленных соединений. При этом достижима более высокая тактовая частота интерфейсов, что обеспечивает их более высокую (по сравнению с шиной) пропускную способность. Структурная схема компьютера архитектуры HT приведена на рис. 6.3. Главный мост (host bridge) обеспечивает связь HT с ядром — процессором и памятью. Периферийные контроллеры, требующие высокой пропускной способности, реализуются в виде HT-туннелей. В архитектуре предусматривается и мостовая связь с шиной PCI.

Архитектура HT обеспечивает все типы транзакций процессоров и устройств PCI, PCI-X и AGP, используемые в PC. Транзакции выполняются в виде серий передач пакетов различных типов. В традиционных транзакциях целевое устройство идентифицируется адресом: чтение и запись в пространстве памяти, ввод-вывод в конфигурационном пространстве, а также считывание вектора прерывания из PIC 8259A и специальные циклы PCI (см. 14.2). Для унификации транзакций все пространства отображаются на единое 40-битное пространство адресов (объем 1 Тбайт), адрес передается в управляющих пакетах. Первые 1012 Гбайт пространства выделены для отображения обычного пространства памяти (для ОЗУ и ввода-вывода, отображенного на память). В оставшейся 12-гигабайтной области размещаются конфигурационное пространство (32 Мбайт), пространство ввода-вывода (32 Мбайт), память SMM, пространства адресов для выдачи векторов и подтверждения прерываний; 54 Мбайт остались в резерве. Транзакции HT обеспечивают программное взаимодействие процессора с устройствами, прямой доступ к памяти и одноранговое взаимодействие устройств с адресацией в описанном комбинированном пространстве. Существует сетевое расширение спецификации, поддерживающее обмен сообщениями (как в сетях), причем возможны и широковещательные сообщения.

Транзакции выполняются расщепленным способом: инициатор посылает пакет-запрос и данные для транзакции записи, целевое устройство посылает пакет- ответ и данные для транзакций чтения. Технология HT обеспечивает упорядоченность выполнения транзакций; есть возможность регулировать качество обслуживания (Quality of Service, QoS), что позволяет организовывать изохронные передачи.

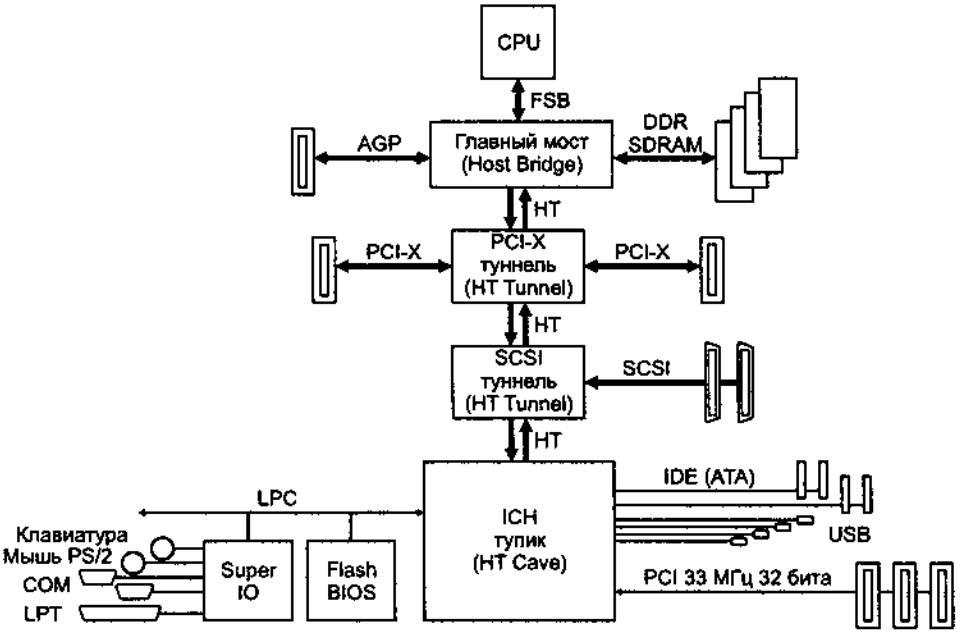


Рис. 6.3. Архитектура HyperTransport

Сигнализация прерываний в HT реализуется тоже пакетами: устройство посылает *сообщение* — выполняет транзакцию записи по адресу, указанному ему при конфигурировании (аналогично MSI на шине PCI). Обработчик прерывания посылает сообщение о завершении обработки прерывания (End Of Interrupt, EOI), делая запись по другому адресу, связанному с данным устройством. Такой механизм сигнализации запросов и подтверждений позволяет преодолеть неэффективность традиционного для PC механизма прерываний с помощью специальных линий IRQ.

Архитектура HT основана на двусторонней пакетной передаче данных между парой устройств. Устройство HT может выступать в роли инициатора или/и целевого устройства транзакций. По топологическим свойствам различают несколько типов устройств HT:

- Туннель (tunnel) — устройство с двумя интерфейсами HT; такие устройства могут собираться в цепочку (daisy chain), образующую логическую шину. Цепочка подключается к хосту (процессору с главным мостом), отвечающему за конфигурирование всех устройств и управляющему работой HT.

- Мост (bridge) — устройство, соединяющее одну логически первичную шину (подключенную к хосту) с одной или несколькими логически вторичными шинами (цепочками). Мост имеет набор регистров, информация которых позволяет управлять распространением транзакций между этими шинами (аналогично мосту PCI).
- Коммутатор (switch) — устройство с несколькими интерфейсами HT, по структуре аналогичное нескольким мостам PCI, подключенным к одной (внутренней) шине.
- Тупик, или пещера (cave) — устройство с одним интерфейсом HT.

Хост (host) — это «хозяин шины», подключающийся к ней через главный мост и выполняющий функции конфигурирования (аналогично и совместимо с PCI). Основной вариант топологии — цепочка устройств-туннелей, подключенная верхним концом к хосту. Каждый интерфейс HT состоит из двух независимых частей: передатчика и приемника. Каждому устройству при конфигурировании выделяются свои области в адресном пространстве. В цепочке устройства-туннели транслируют пакеты сверху вниз (нисходящий трафик) и снизу вверх (восходящий). Если в нисходящем управляющем пакете устройство обнаруживает свой адрес, оно «понимает», что обращаются к нему, и принимает соответствующую информацию (управляющие пакеты и данные). Восходящий трафик туннель транслирует «вслепую». На полученные запросы устройство отвечает посылкой пакетов вверх, включая их в транслируемый восходящий трафик. Таким образом обеспечивается программное взаимодействие процессора с устройствами. Собственные запросы на доступ к памяти устройство посылает тоже вверх, как и запросы (обращения) к другим устройствам (независимо от положения целевого устройства — выше или ниже в цепочке). Доставку пакета адресату обеспечивает главный мост: он разворачивает пакет, принятый из цепочки (адресованный не к ОЗУ), и посылает его вниз — так организуется одноранговое взаимодействие. На пакет, адресованный к ОЗУ, главный мост организует ответ от контроллера памяти, реализуя таким образом прямой доступ к памяти.

Возможны и более сложные топологии, например дерево (с мостами), позволяющее подключать больше тупиковых устройств. Возможна и цепочка с двумя хостами (на обоих концах), которая может использоваться двояко. В первом варианте обеспечиваются избыточность (дублирование функций хоста) и разделяемость узлов (доступность обоим хостам). При этом один главный мост становится ведущим («настоящим», разворачивающим одноранговые запросы и ответы), через него обеспечивается конфигурирование узлов. Другой мост становится ведомым — он является лишь средством связи второго хоста (процессора) с узлами. Программно при конфигурировании (инициализации HT) роли мостов можно поменять. Во втором варианте одно из устройств разбивает шину (перестает работать туннелем), в результате получаются два хоста со своими короткими цепочками собственных (неразделяемых) устройств. С применением коммутаторов можно строить и более сложные, но беспетлевые топологии.

Технология HyperTransport предназначена для соединения компонентов компьютеров и коммуникационной аппаратуры, но только в пределах платы — сло-

ты и карты расширения технологией HT не рассматриваются. Для передачи информации используются два встречных однонаправленных набора высокоскоростных сигналов:

- CAD[n:0] — шина управления (control), адреса (address) и данных (data) разрядностью 2, 4, 8, 16 или 32 бита, причем во встречных направлениях может использоваться различная разрядность. У передатчика сигналы CADOUTx, у приемника — CADINx;
- CTL — сигнал-признак, позволяющий различать передачи пакетов управляющей информации и данных. У передатчика сигнал CTLOUT, у приемника — CTLIN;
- CLK — сигнал синхронизации (по фронту и спаду), для каждого байта CAD используется своя линия CLK (их может быть 1, 2 или 4). У передатчика сигналы CLKOUTx, у приемника — CLKINx.

Сигналы передаются по дифференциальным парам проводов с импедансом 100 Ом, сигналы — LVDS (низковольтные дифференциальные, уровень1,2 В). Частота синхронизации 200, 300, 400, 500, 600, 800 и даже 1000 МГц обеспечивает физическую скорость передачи 400, 600, 800, 1000, 1200, 1600 и 2000 МТ/с (миллионов передач в секунду), что при самых больших разрядности (32 бит) и частоте обеспечивает пиковую скорость передачи данных до 8 Гбайт/с. В первой версии предельная частота была 800 МГц, что давало скорость 6,4 Гбайт/с. Поскольку пакеты могут передаваться одновременно в обоих направлениях, можно говорить о суммарной пропускной способности 12,8 или 16 Гбайт/с.

Помимо сигналов для передачи пакетов, имеются сигналы сброса и инициализации (PWROK — признак стабильности питания и синхронизации, RESET# — сброс цепочки устройств), а также управления энергопотреблением (LDTSTOP# — разрешение/запрет использования соединения при смене состояний системы, LDTREQ# — индикатор активности соединения или его запроса устройством). Эти сигналы «медленные», их формируют передатчики с открытым стоком (open-drain), все одноименные сигналы цепочки объединяются, выполняя функцию «монтажного ИЛИ». Уровни сигналов — LVTTL/CMOS (2,4 В).

По замыслу разработчиков, HT должна стать архитектурой построения PC, однако пока что используется лишь технология HT. В вышеприведенном примере главный мост реализует интерфейс AGP. В 64-битных процессорах AMD, в которых применяется HT, главный мост размещается в самом процессоре. При этом у процессора оказывается два интерфейса: интерфейс памяти (пока что DDR SDRAM) и HT в качестве системной шины. В распространенных чипсетах (от VIA, SiS) к интерфейсу HT подключается только северный хаб, обеспечивающий лишь интерфейс подключения графического адаптера — AGP или PCI-E. Южный хаб соединяется с северным собственным интерфейсом, так что использования HT как универсальной транспортной структуры для множества компонентов пока не наблюдается.

Северные мосты и хабы

Северный хаб (как и мост) определяет основные возможности системной платы:

- Поддерживаемые процессоры — типы, частоты системной шины, возможности мультипроцессорных или избыточных конфигураций. Типы процессо

ров определяются протоколами системной шины, которых в настоящее время несколько:

- шина Pentium процессоров для сокета 7, Super7 (и сокета 5); частоты 50-100 МГц;
- шина P6 процессоров для сокета 8, слотов 1 и 2, сокета-370; частоты 66-133 МГц;
- шина Pentium 4 для сокетов с 423, 478/479, 603/604 и 775 контактами; частота синхронизации 100-266 МГц при 4-кратной «накачке» обеспечивает частоту передачи данных 400-1066 МГц;
- шина EV-6 процессоров Athlon, Duron, Semptron для слота A и сокета A (462 контакта); частоты передачи данных 200-400 МГц (тактовая частота в два раза ниже);
- интерфейс HyperTransport процессоров со встроенным контроллером памяти (Athlon 64, Opteron, мобильные Turion 64 и Semptron) для сокетов с 754 и 939/940 выводами.

♦ Типы памяти и частота работы шины памяти[1]:

- DRAM (FPM, EDO, BEDO) с временем доступа 50-80 нс;
- SDRAM (PC66, PC100, PC133) с частотами 66-133 МГц;
- DDR SDRAM (PC1600, PC2100, PC2700, PC3200) с частотами 100- 200 МГц (частота передачи в два раза выше);
- DDR2 SDRAM (PC2-3200, PC2-4300, PC2-5300, PC2-5300, PC2-6400) с частотами 200-400 МГц (частота передачи в два раза выше);
- RDRAM (PC600, PC700, PC800, PC1066) с частотами 300, 356, 400 и 533 МГц.

♦ Максимальный объем памяти. На него влияет ряд факторов:

- число слотов под модули памяти и поддерживаемые объемы модулей (допустимое число устанавливаемых модулей при работе на самой высокой частоте шины памяти может оказаться меньше, чем число слотов);
- максимальное количество «рядов» микросхем памяти (может ограничивать возможное число устанавливаемых двусторонних модулей).

♦ Число каналов памяти — пока чаще один, но для повышения пропускной способности применяются два канала. Поначалу двухканальность использовалась только для RDRAM (здесь меньше интерфейсных сигналов в канале), теперь есть двухканальные контроллеры DDR SDRAM и DDR2 SDRAM. В оба канала должны быть установлены попарно однотипные модули (как раньше пары SIMM-72 для Pentium).

♦ Возможность и эффективность применения разнородной памяти (например, DRAM вместе с SDRAM в старых платах, SDRAM и DDR SDRAM в более новых) и модулей с разным быстродействием (разная латентность при оди

[1] На системных платах для процессоров со встроенным контроллером памяти характеристики памя ти (тип, число каналов, частоту) задает процессор.

наковой частоте). В ряде случаев разнородная память снижает производительность всей памяти, и не всегда эта потеря окупается получаемым увеличением объема ОЗУ.

- Для старых плат с DRAM — возможность чередования банков (у современных типов памяти чередование банков внутреннее).
- Поддержка контроля достоверности памяти и исправления ошибок (ECC).
- Средства подключения графического акселератора (высокопроизводительное подключение), для которого уже имеется несколько вариантов:
  - порт AGP и его характеристики (режим 2x/4x/8x, внеполосная адресация SBA, быстрая запись Fast Writes); для чипсетов с интегрированной графикой интересна доступность порта при отключении внутреннего графического адаптера;
  - слоты PCI-E 8x или 16x для подключения графического адаптера (1 или 2 порта); слоты PCI-E 1x может обеспечивать как северный, так и южный хаб;
  - графический адаптер с интерфейсом HyperTransport (пока что это теоретический вариант).
- Возможности системы управления энергопотреблением (ACPI или APM) — реализуемые энергосберегающие режимы процессора и памяти, управление производительностью, SMM.

Северный мост плат для сокетов 5, 7 и Super7 определяет также политику записи кэша, применяемые типы и быстродействие микросхем статической памяти, возможный размер кэша и кэшируемой области основной памяти. Для современных плат без кэша все эти параметры определяются процессором, а политику обратной записи поддерживают уже все платы.

Северный мост определяет также поддерживаемые частоты и разрядность шины PCI и PCI-X, возможное количество контроллеров шины PCI (число пар сигналов арбитра PCI), способы буферизации, возможности одновременных обменов. Северный хаб на эти параметры уже не влияет, поскольку шины PCI и PCI-X подключаются к южному хабу.

Южные мосты и хабы

Южный хаб чипсета обеспечивает подключение шин PCI, PCI-X и «маломощных» портов PCI-E, ISA (но уже не всегда), ATA (2 канала), SATA, USB, FireWire, а также «мелких» контроллеров ввода-вывода, памяти CMOS и флэш- памяти с системным модулем BIOS. В южной части располагаются таймер (8254), контроллер прерываний (совместимый с парой 8259 или APIC), контроллер DMA для шины ISA и периферии системной платы. Если в чипсет интегрирован звук, то южный хаб (мост) имеет контроллер интерфейса AC-Link или HDA Link для подключения аудиокодека, а то и сам аудиокодек. Поскольку шина ISA отправляется в отставку, для контроллеров ввода-вывода, ранее подключавшихся к шине X-BUS (это практически та же ISA), ввели новый интерфейс LPC (Low Pin Count). Он, как и следует из названия, имеет малое чис

ло линий [6], что значительно облегчает разработку чипсета и системной платы. Флэш-память для хранения системной памяти BIOS стали помещать в специальный хаб (firmware hub), соединяемый с южным хабом отдельной шиной (аналогичной LPC). Флэш-память может подключаться и прямо к шине LPC. Для подключения энергонезависимой памяти (EEPROM) хаб может иметь дополнительный последовательный интерфейс. Для обслуживания процессоров, имеющих дополнительную сервисную шину SMBus, а также для поддержки слота CNR хаб может иметь последовательный интерфейс I^2C (Inter IC — интерфейс связи микросхем). Этот же интерфейс может использоваться для чтения идентификаторов модулей памяти (I^2C и SMBus — близкие родственники, несколько различающиеся набором команд). В южный хаб интегрированных чипсетов вводят и контроллер локальной сети (как правило, Ethernet).

Логически южный хаб представляется как набор виртуальных мостов и устройств, подключенных к главной шине PCI. Однако обмены данными с широкополосными устройствами (IDE, SATA, USB, FireWire, Ethernet, AC'97 или HDA) на внешнюю шину PCI все-таки не «выплескивают», иначе теряется смысл южного хаба.

Южный хаб (или мост) определяет перечисленные далее параметры системной платы:

- Параметры шины PCI (только для хабов):
  - версия интерфейса и режимы (PCI, PCI-X, PCI-X 2.0);
  - разрядность (32 или 64 бита);
  - частота (33 или 66 МГц для PCI, до 133 МГц для PCI-X);
  - допустимое количество контроллеров шины (число каналов арбитра, которое влияет на число слотов и встроенных устройств PCI).
- Число маломощных (4x) портов PCI-E.
- Параметры интерфейсов ATA:
  - поддерживаемые режимы UltraDMA — ATA/33, ATA/66, ATA/100, ATA/133;
  - независимость каналов — электрическое разделение каналов, возможность одновременной работы двух каналов.
- Параметры интерфейса SATA: тип контроллера (желательно AHCI), число портов, возможность одновременного использования с параллельной шиной.
- Число портов и версия шины USB.
- Наличие интерфейса AC-Link или HDA Link.
- Наличие шины ISA.
- Возможность эмуляции DMA на шине PCI (PC-PCI, DDMA).
- Возможности мониторинга состояния:
  - число каналов измерения питающих напряжений;

- число каналов измерения температуры;
- число каналов измерения частоты вращения вентиляторов.

Контроллеры гибких дисков, интерфейсных портов, клавиатуры, CMOS RTC могут входить в собственно чипсет, а могут быть реализованы и на отдельных «инородных» микросхемах. От них зависят следующие параметры системной платы:

- наличие порта PS/2 Mouse (есть во всех платах ATX);
- режимы параллельного порта (стандартный, двунаправленный, ECP, EPP, поддержка FIFO и DMA);
- режимы последовательных портов (стандартом считается совместимость с 16550A и поддержка FIFO и DMA);
- поддержка IrDA;
- типы поддерживаемых дисководов (2,88 Мбайт поддерживают теперь почти все контроллеры, но эта возможность не востребована дисководами и дискетами).

Синхронизация и потоки данных

Как видно, в чипсете обеспечивается взаимодействие множества шин, большинство которых синхронные. Вопросы синхронизации решаются по-разному. У чипсетов для шины Pentium память всегда работала на частоте системной шины (60-100 МГц), а частота шины PCI (номинал 33 МГц) была к ней привязана с коэффициентом 1 : 2 или 1 : 3. При частоте системной шины, отличной от 66 или 100 МГц, шина PCI оказывалась либо разогнанной, либо приторможенной.

В чипсетах с портом AGP частоту шины памяти стремятся повысить, иначе память становится узким местом: к ней обращаются акселератор с AGP, ведущие устройства PCI и, наконец, сам процессор. При этом у процессора может быть частота шины всего 66 МГц (как, например, у процессоров Celeron). Для любителей разгонов полезно такое свойство чипсетов, как *асинхронность* — возможность относительно произвольного задания частот системной шины, шины памяти, порта AGP, шины PCI. Заметим, что частота шин LPC и шины подключения хаба с BIOS (FWH) совпадает с частотой PCI (33 МГц), и разгон шины PCI влечет за собой разгон и этих шин, однако поведение их абонентов на повышенных частотах может огорчить пользователя невозможностью разгона. Конечно же, здесь асинхронность условна — опорный генератор все-таки один, но коэффициенты для каждого домена синхронизации (группы тесно связанных узлов) задаются раздельно. Таким образом, можно из всех компонентов «выжать» максимум производительности. Однако при определенных соотношениях частот компонентов (как правило, не равных степени двойки) из-за промежуточной буферизации данных наблюдается снижение суммарной производительности системы.

На рис. 6.4 обозначены основные компоненты системной платы, связывающие их интерфейсы и основные потоки транзакций. Кружком на линиях отмечен

инициатор транзакций, стрелки указывают на целевое устройство, направление передачи данных может быть любым. Здесь же указана пиковая пропускная способность интерфейсов этих компонентов при разных частотах. На рисунке изображен однопроцессорный вариант. Подключение второго (и следующих) процессоров Pentium 4, P6 и Pentium (если это возможно) не повышает общую производительность шины процессоров (они разделяют общую шину). Для процессоров Athlon ситуация иная: поскольку процессоры подключаются к хабу выделенными каналами, суммарная потребность в пропускной способности для двухпроцессорной шины удваивается. В мультипроцессорных системах на процессорах с системной шиной HyperTransport вопросы пропускной способности ставятся иначе (см. 7.8).

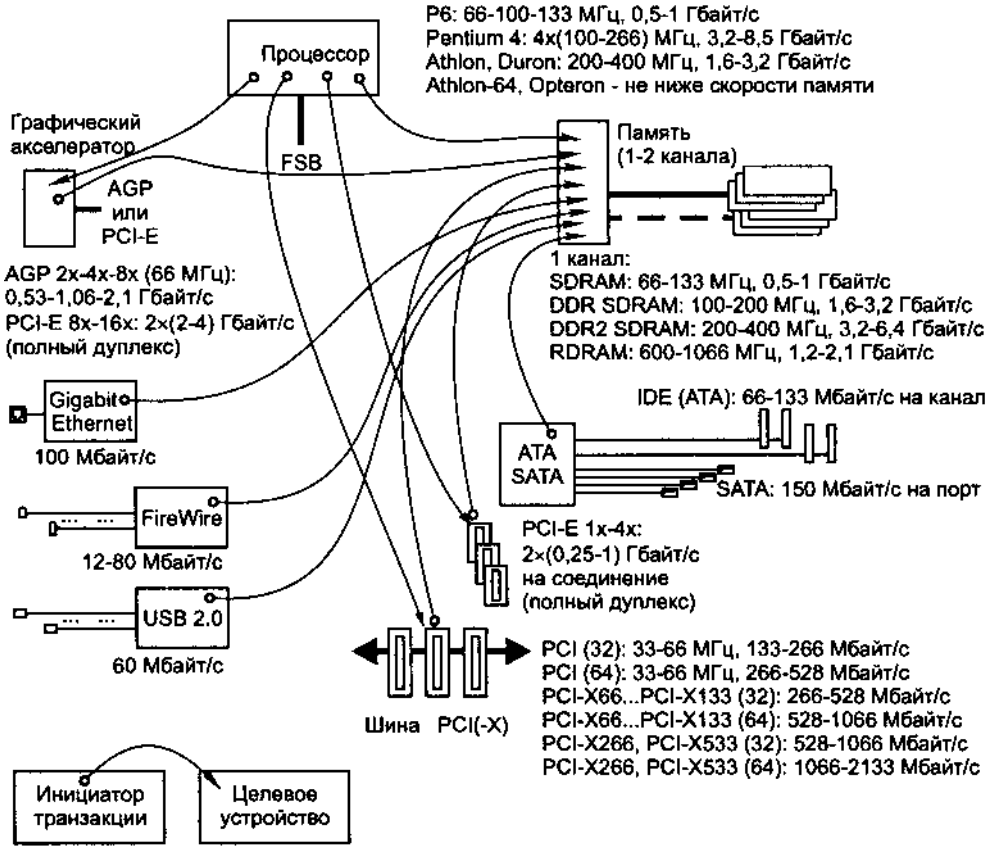


Рис. 6.4. Компоненты и потоки данных на системной плате

Из рисунка видно, что в плане потоков данных центральным компонентом компьютера является память. Вполне очевидно, что компьютер должен быть сбалансирован по пропускной способности интерфейсов. Бессмысленно делать пропускную способность любого интерфейса выше пропускной способности

памяти — выигрыша в производительности системы это не даст, поскольку скорость в каждом потоке определяется наименьшей из скоростей его источника и приемника. Для повышения пропускной способности памяти допускается применение второго канала, что делается уже во многих чипсетах для DDR SDRAM и RDRAM. Для SDRAM и DDR RDRAM такое решение технически обходится дороже — слишком много линий интерфейса, однако эти трудности уже преодолены.

Интерфейс подключения графического адаптера (AGP или PCI-E), как правило, является самым мощным потребителем пропускной способности обращений к памяти со стороны периферии. Следующими по «уровню притязаний» являются интерфейсы устройств хранения (ATA, SATA, SCSI), адаптера Gigabit Ethernet, шин FireWire и USB 2.0. Интерфейс устройств хранения (естественно, и сами устройства) определяет общую производительность компьютера, поскольку через них проходит поток свопинга для работы виртуальной памяти. Пропускная способность остальных интерфейсов влияет лишь на скорость работы с соответствующей периферией.

Чипсеты и платы

Хотя чипсеты в значительной степени определяют свойства системных плат, выполненных на их основе, у разработчика плат всегда остаются возможности упростить плату и «испортить хорошую вещь». Так что системные платы, выполненные на одном и том же чипсете, могут иметь разные характеристики по производительности и разный диапазон поддерживаемых устанавливаемых компонентов (процессоров, памяти, интерфейса). И конечно же, существенную роль в реализации всех полезных свойств чипсета играют BIOS и применяемые версии системных драйверов. Чипсеты ориентируются на разные применения системных плат, и функции, необходимые для сервера, могут оказаться лишними для офисного компьютера, а за излишества всегда приходится платить. Поэтому нельзя чипсеты выстроить по порядку от худшего к лучшему, они позиционируются в многомерном пространстве противоречивых требований. Сравнивать интегрированные чипсеты нужно не только по общим параметрам, но и по характеристикам графики, звука, адаптера локальной сети. Основные параметры распространенных системных плат (и чипсетов) приведены в литературе [10]. Результаты тестирования и сравнения системных плат регулярно публикуются в периодических изданиях и в Сети, например на сайте iXBT.com.

Микросхемы чипсета при инициализации во время теста POST программируются по многим параметрам, часть из которых (константы) хранится в BIOS, а часть — в энергонезависимой памяти конфигурации, включающей ячейки CMOS и ESCD системы PnP. Таким образом, имеются программные способы как оптимальной настройки, так и вывода платы из строя записью определенных значений в энергонезависимую память. Эту запись производит утилита CMOS Setup, а также такие «экспансивные» операционные системы, как Windows.

6.2. Установка и конфигурирование компонентов

Современные системные платы имеют ряд сменных или добавляемых компонентов. В процессе модернизации (upgrade) системной платы возможны замена процессора, наращивание объема и повышение быстродействия ОЗУ, замена версии BIOS. Эти действия обычно связаны с изменениями аппаратных и программных параметров, о которых и пойдет речь.

Процессоры

Процессоры, установленные в компьютерах XT, AT-286 и AT-386, обычно заменять не требовалось: выходят из строя они сами по себе крайне редко — скорее отказывают другие компоненты системной платы, а замена процессора более мощным в упомянутых моделях, как правило, не предусматривалась. Начиная с процессоров класса 486, модернизация, заключающаяся в замене процессора более мощным, стала традиционной. Системные платы стали выпускать с расчетом на различные модификации и тактовые частоты процессоров — получился своеобразный конструктор «собери сам». Процессоры стали устанавливать в стандартизованные сокеты ZIF (Zero Insertion Force — нулевое усилие вставки), а затем и в слоты — щелевые двухрядные разъемы. Назначение выводов разъемов поначалу определялось процессорами-первопроходцами от фирмы Intel, а другие фирмы в своих процессорах выдерживали совместимость с этими сокетами. Начиная с процессоров K7, фирма AMD повела свою линию сокетов и слотов. Унификация расположения выводов процессоров одного класса и наличие конфигурационных переключателей на системных платах позволяют пользователю (даже не слишком подготовленному) легко заменять старые процессоры более мощными.

Платы для симметричных мультипроцессорных систем имеют пару сокетов (слотов). В них устанавливают процессоры, пригодные для таких конфигураций. Долгое время в мультипроцессорных системах применялись только процессоры фирмы Intel — конкурирующие фирмы (AMD, Cyrix и IBM) мультипроцессированием не занимались. Эту «традицию» нарушила фирма AMD своим процессором Athlon и последующими. Мультипроцессорные (как и мультиядерные) системы рассматриваются в 7.5. Следует помнить, что в симметричных мультипроцессорных системах внутренние частоты всех процессоров должны совпадать (внешняя частота у них одна, поскольку исходит от общего генератора синхронизации). Для этих целей лучше брать все процессоры с одним степпингом и одинаково устанавливать для них конфигурационные джамперы.

В «добрые старые времена» процессоров 4-5-го поколений их разработчики стремились к взаимной совместимости своих продуктов, причем не только программной, но и аппаратной. В любую системную плату можно было установить процессор из широкого спектра возможных — от Intel, AMD, Cyrix и пр. В последние годы ситуация изменилась, сокеты и слоты быстро сменяют друг друга,

и модернизацией за счет смены процессора занимаются нечасто. Новый процессор, радикально отличающийся от установленного год назад, скорее всего, потребует и новой системной платы. Здесь кратко рассмотрены характеристики современных сокетов для процессоров Intel и AMD начиная с 6-го поколения; информацию по старым сокетам и слотам (с шиной Pentium и P6) можно найти в [2], [3]. Полоса слотовых процессоров, похоже, прошла. Современные сокеты называются по числу контактов, «древние» сокеты имели порядковые номера. Основные данные по ныне существующим сокетам и слотам приведены в табл. 6.1. Выбирая системную плату, приходится сначала определяться с типом процессора.

Таблица 6.1. Сокеты и слоты

Сокет (слот)	Питание	Поддерживаемые процессоры
Сокет 370[1]	1,3–2,05 В	Celeron, Pentium III, VIA Cyrix III
Сокет 423	1,1–1,6 В	Pentium 4: 0,18 мкм 1,3–2 ГГц
Сокет 478	1,1–1,8 В	Pentium 4: 0,18 мкм 1,4–2 ГГц;
	1,1–1,8 В	0,13 мкм 2,0A–3,4 ГГц;
	0,84–1,6 В	90 нм 2,8 A/E–3,4 E ГГц
Сокет 775	0,84–1,6 В	Pentium 4: 0,18 мкм 1,4–2 ГГц;
		0,13 мкм 2,0A–3,4 ГГц;
		90 нм 2,8 A/E–3,4 E ГГц
Сокет 603	1,1–1,8 В	Xeon 1,4–2 ГГц;
Сокет 604	1,1–1,8 В	Xeon с частотой FSB 133/533 и выше
	0,84–1,6 В	
Сокет A (462)	1,1–1,85 В	Athlon, Athlon XP, Duron, Sempron (2200+...3100+)
Сокет 754	0,8–1,55 В	Athlon 64 (2800+...3700+), Sempron (3100+, 3300+)
Сокет 939	0,8–1,55 В	Athlon 64 (3500+...4000+), Athlon 64FX
Сокет 940	0,8–1,55 В	Athlon 64FX, Opteron

[1] Назначение выводов AM2, AH4, X4, E27 и S35 может быть различным (см. далее).

К сожалению, полной совместимости между всеми процессорами, устанавливаемыми в сокет (слот) одного типа, нет. Возможный тип устанавливаемого
процессора определяется следующими свойствами системной платы:

- типом сокета (слота);
- наличием двух раздельных источников питания, если того требует процессор;
- возможностью установки требуемых напряжений питания процессора и допустимой мощности регуляторов напряжения;
- возможностью установки требуемой частоты синхронизации и коэффициента ее умножения;
- поддержкой процессора конкретной версией BIOS;
- указанием на применимость данного процессора, сделанным разработчиком системной платы в ее описании (или указанием конкретного типа системной платы в списке совместимости, публикуемом разработчиком процессора).

Если первые четыре пункта определяются однозначно, то для последних, как говорится, возможны варианты. Версию BIOS (особенно если применяется флэш-память) можно обновить и обеспечить тем самым поддержку устанавливаемого процессора. Что касается списков совместимости, то они условны. Разработчик платы может заранее заявить о совместимости с будущим процессором, но будут ли они «жить дружно» — вопрос. Напротив, разработчик процессоров может не включить конкретную системную плату в свой список совместимости, но они смогут нормально работать в паре. Типов системных плат гораздо больше, чем типов процессоров, и если производитель платы не позаботился о доставке образцов своих изделий для тестирования с конкретным процессором, такая плата может не попасть в список изготовителя процессоров. Существуют и «черные списки» — списки несовместимости, заполняемые сборщиками компьютеров.

BIOS определяет тип установленного процессора (начиная с 5-го поколения Intel и 4-го AMD) в начале теста POST по инструкции CPUID (см. 7.6), по которой процессор сообщает идентификатор производителя (разработчика), семейство, модель и степпинг. По этим данным BIOS формирует имя процессора (например, «Intel Pentium 4»), которое POST выводит на экран (и сообщает в CMOS Setup). Текстовые названия известных процессоров прописаны в теле BIOS, так что неверное название в сообщении свидетельствует о слишком старой (для данного процессора) версии BIOS. Процессоры AMD имя процессора сообщают по инструкции CPUID, так что тесту POST не требуется искать имена по таблицам. В процессорах Intel этот механизм реализован только начиная с Pentium 4. Текущую тактовую частоту ядра POST определяет с помощью системного таймера, либо выполняя определенный цикл инструкций и подсчитывая число проходов за известный интервал, либо снимая показания счетчика меток реального времени (TSC) в начале и в конце интервала измерения. Последний способ точнее, но он работает только на процессорах, имеющих этот счетчик (Pentium и выше). Чтением определенных модельно-специфических регистров процессора POST может определить установленный коэффициент умножения частоты.

Идентифицировав процессор, POST может выполнить загрузку «заплаток» микрокода процессора (download microcode), если в BIOS имеются блоки, подходящие для установленного процессора, и загрузка не запрещена. Отсутствие блока для устанавливаемого процессора является поводом для перепрошивки BIOS, хотя «заплатки» можно загружать и из файла в начале загрузки ОС.

Для установленного процессора требуется задать напряжение питания и частоту ядра, которая определяется частотой системной шины (FSB frequency) и коэффициентом умножения. Эти параметры задаются вручную или с различной степенью автоматизации, для чего используются джамперы или параметры CMOS Setup (Soft Menu). В сокетах 1-7 пользователь может задавать все эти параметры сравнительно произвольно, руководствуясь своими знаниями об устанавливаемом процессоре и, возможно, идеями разгона. В сокетах и слотах для процессоров 6-8-го поколений предусмотрен интерфейс, с помощью которого процессор сообщает требуемое напряжение питания, автоматически управляя

регулятором напряжения, а также заказывает частоту FSB. Интерфейс прост: выделяется несколько сигнальных линий, часть из которых заземляется внутри корпуса (картриджа) процессора. В новых процессорах идентификатор питания может динамически изменяться процессором (см. 7.4). Генератор синхронизации и регулятор напряжения работают под управлением вышеуказанных сигналов, отрабатывая задания в соответствии со своими возможностями. Если регулятор не может выдать требуемый номинал, он вообще не должен подавать напряжение.

Коэффициент умножения у большинства современных процессоров теперь фиксирован, что совместно с автоматизацией выбора частоты и напряжения связывает руки оверклокерам («разгонщикам»). Но, к их радости, не все производители системных плат придерживаются строгих правил, предписываемых изготовителями процессоров, и предлагают выбор между автоматическим и ручным (произвольным) выбором параметров. Но и на «строгих» платах в слоте можно «обмануть» процессор, заклеивая или заземляя требуемые контакты на краевом разъеме картриджа. Для сокета этот «обман» осложняется. В переходниках «сокет — слот» перекоммутация линий технически проще (да и потери от возможного повреждения переходника меньше, чем для системной платы); во многих переходниках имеются специальные джамперы, позволяющие вручную устанавливать параметры даже для самых «строгих» автоматических плат. Однако ручное задание параметров небезопасно.

ВНИМАНИЕ

Установка завышенных частоты ядра и напряжения питания опасна для процессоров, особенно без принятия должных мер по охлаждению.

Рассмотрим интерфейсы конфигурирования процессоров в различных сокетах и слотах.

Сокеты для процессоров Intel

Для процессоров P6 применялось несколько типов сокетов и слотов [2], [3], из которых здесь рассмотрим только *сокет 370* (рис. 6.5), переживший две редакции. Первоначально он был введен для процессоров Celeron (упрощенных процессоров Pentium II), позже с некоторыми изменениями сокет 370 приспособили и к Pentium III с ядром Coppermine, и к соответствующему ему процессору Celeron с поддержкой SSE (Streaming SIMD Extensions — потоковые SIMD- расширения), названному Celeron-2. Казалось бы, логична полная совместимость процессоров Celeron и Pentium III с сокетом (без учета неиспользуемых сигналов), но все не так просто. Есть пять выводов с изменившимся назначением, из-за которых процессор, вставленный в сокет, может отказаться работать:

- Сигнал аппаратного сброса RESET# у процессоров Celeron (включая процессоры с поддержкой SSE) выведен на контакт X4, у Pentium III — на AH4. У универсальных системных плат сигнал сброса подается на оба вывода (при необходимости такую перемычку легко напаять на плату-переходник, на системной плате паять страшнее). Без аппаратного сброса процессор запускаться не будет.

- Контакт AM2 на системных платах под Celeron соединяли с шиной GND. Для процессоров Pentium III и Celeron с поддержкой SSE этот вывод значится зарезервированным, причем требуется, чтобы он оставался свободным. Освободить на печатной плате контакт, связанный с шиной, — дело почти невозможное. Однако если не жалко процессора и не волнует вопрос гарантий, то этот (и только этот!) вывод можно «откусить» прямо на процессоре. Другой вариант — снять крышку сокета и выпаять (выкусить) контактный лепесток.
- У процессоров Pentium III и Celeron с поддержкой SSE введены новые сигналы RTTCTRL и SLEWCTRL Их контакты (S35 и E27) рекомендуется соединять с шиной GND через резисторы 300 Ом, но, похоже, их можно оставлять и неподключенными.

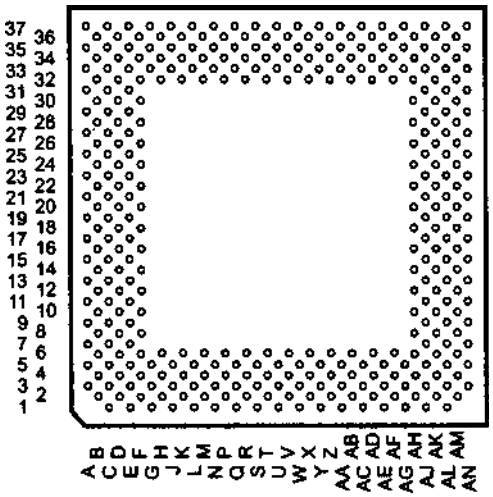


Рис. 6.5. Сокет 370

В сокете 370 линии идентификации VID[3:0] расположены на контактах AJ37, AL37, AM36 и AL35, напряжения соответствуют табл. 6.2. Напряжение питания ядра можно проверить на контактах B10, B14, B18... Сигналы выбора частоты BSEL0 и BSEL1 (табл. 6.3) расположены на контактах AJ33 и AJ31 соответственно. У процессоров Celeron и Pentium III коэффициенты умножения частоты фиксированы в зависимости от спецификации.

Таблица 6.2. Идентификация питания в сокете 370

VID[3:0]	Напряжение, В	VID[3:0]	Напряжение, В
1111	1,30	0111	1,70
1110	1,35	0110	1,75
1101	1,40	0101	1,80
1100	1,45	0100	1,85
1011	1,50	0011	1,90

продолжение

Таблица 6.2 (*продолжение)*

VID[3:0]	Напряжение, В	VID[3:0]	Напряжение, В
1010	1,55	0010	1,95
1001	1,60	0001	2,00
1000	1,65	0000	2,05

ПРИМЕЧАНИЕ

Старые процессоры Pentium III могут иметь питание ядра 1,8-2,05 В, вторичный кэш питается от 3,3 В; более новые питаются напряжением 1,6-1,65 В; старые процессоры Celeron питаются от 2 В, Celeron II — от 1,5 В.

Таблица 6.3. Выбор частоты синхронизации системной шины P6

BSEL1 (SELFSB1)	BSEL0 (SELFSB0)	Частота, МГц
0	0	66
0	1	100
1	0	Резерв
1	1	133

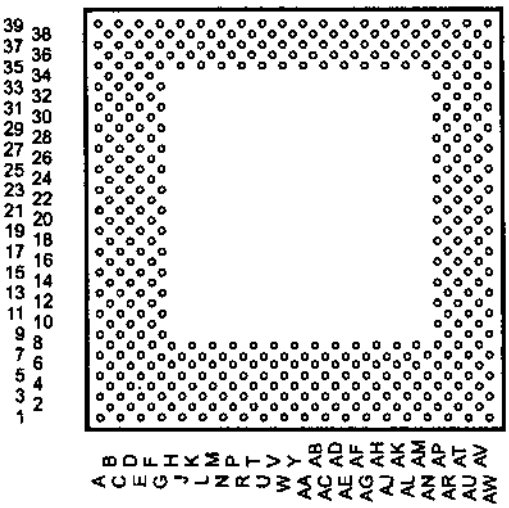


Рис. 6.6. Сокет 423

Для *процессора Pentium 4* был введен *сокет 423* (рис. 6.6), не совместимый ни с какими прежними (шина Pentium 4 сильно отличается от шины P6). Для этого процессора требуется мощный блок питания — процессор на 1,5 ГГц потребляет 70 Вт. В этот сокет устанавливаются процессоры с частотами 1,3-2 ГГц (0,18 мкм), частота системной шины фиксирована — 100 МГц (частота передачи данных — 400 МГц). В сокете 111 контактов используются для подачи питания, 112 — для «земли». Напряжение питания ядра задается 5-битным идентификатором VID[4:0] с дискретностью 25 мВ (табл. 6.4), формально спецификация на сокет определяет напряжение в диапазоне от 1,100 до 1,850 В. Фактически, про-

цессоры используют напряжение около 1,6 В. С конца 2004 года фирма Intel прекратила поддержку этого типа сокета (процессоры для него перестали выпускать раньше).

Таблица 6.4. 5-битная идентификация питания в сокетах 423, 478 и 603

VID[4:0]	VCC	VID[4:0]	VCC
11111	Отключено	01111	1,475
11110	1,100	01110	1,500
11101	1,125	01101	1,525
11100	1,150	01100	1,550
11011	1,175	01011	1,575
11010	1,200	01010	1,600
11001	1,225	01001	1,625
11000	1,250	01000	1,650
10111	1,275	00111	1,675
10110	1,300	00110	1,700
10101	1,325	00101	1,725
10100	1,350	00100	1,750
10011	1,375	00011	1,775
10010	1,400	00010	1,800
10001	1,425	00001	1,825
10000	1,450	00000	1,850

Следующим для Pentium 4 был *сокет 478* (рис. 6.7), в котором дополнительные выводы используются в основном для увеличения числа «земляных» выводов (181, 180 и 179 в трех редакциях сокетов). Число питающих выводов уменьшено до 85.

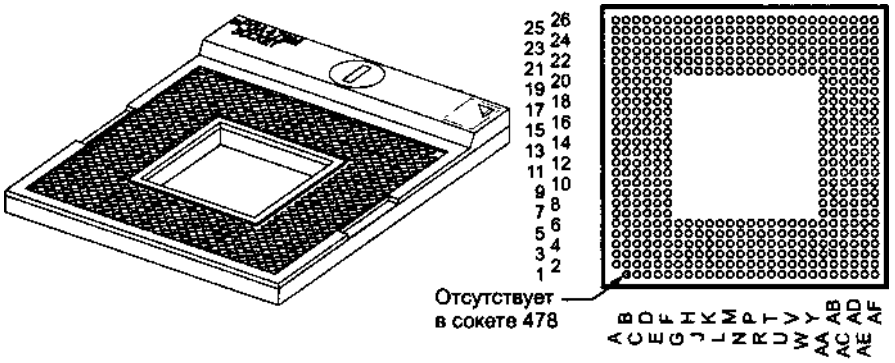


Рис. 6.7. Сокеты 478 и 479

Первоначально для сокета 478 предназначались процессоры, выполненные по технологии 0,18 мкм (2002 г., частоты — 1,4-2,0 ГГц); для них значения VID[4:0] совпадают с принятым для сокета 423, а напряжение питания — 1,6 В. Далее появились процессоры, изготовленные по технологии 0,13 мкм (2004 г., часто-

ты ядра — 2,0A[1]-3,4 ГГц, частота шины — 100/400 МГц, Гиперпотоковые), у которых используется напряжение питания от 1,475 до 1,6 В (чем выше частота, тем выше напряжение). Затем (2005 г.) для этого же сокета стали выпускать процессоры, выполненные по технологии 90 нм (0,09 мкм) с частотами 2,8A/E, 3E-3,4E ГГц; частота системной шины — 200/800 МГц, расширение — SSE3, Гиперпотоковые. Теперь для идентификации напряжения питания используется 6-битный идентификатор VID[5:0] (табл. 6.5), позволяющий задавать напряжение с дискретностью 12,5 мВ в диапазоне 0,835-1,600 В. Заметим, что биты VID[4:0] кодируют напряжения в диапазоне 1,100-1,600 В, так же как и в предыдущих версиях данного сокета, а коды, отводившиеся ранее для напряжений 1,625-1,850 В, теперь кодируют напряжения 0,8375-1,0875 В. Бит VID5, формально старший, на самом деле имеет самый маленький вес (12,5 мВ). В сокете процессор заказывает частоту системной шины сигналами BSEL[1:0] (табл. 6.6). Варианты сокетов 478 различаются не только спецификациями основного питания, но и назначением нескольких выводов:

- Контакт AD1 в первых двух редакциях был «землей» (VSS); для процессоров, изготовленных по технологии 90 нм, он считается входом Bootselect и должен быть свободен (если заземлен, процессор работать не будет).
- Контакт AE26 в первой редакции был «землей» (VSS); для процессоров 0,13 мкм он считается входом ImpSel, управляющим импедансом шинных буферов (заземлен — 50 Ом, свободен — 60 Ом); для процессоров, изготовленных по технологии 90 нм, он считается входом Optirnized/Compat# и должен быть свободен (как признак поддержки данного процессора).
- Контакт AF3 в первых двух редакциях был резервным; для процессоров, изготовленных по технологии 90 нм, он является входом для подачи напряжения питания VCCVIDLB (1,2 В) на схемы идентификации напряжения.
- Сигнал ProcHot# (контакт C3) в первой редакции был только выходом процессора (сообщение о срабатывании цепей термоконтроля), в последующих редакциях с его помощью чипсет может принудительно включать цепи термоконтроля (у процессоров, начиная с CPUID 0xF27).
- Контакт AD3 в первых двух редакциях был резервным; для процессоров, изготовленных по технологии 90 нм, это выход VID5.

Таблица 6.5. 6-битная идентификация питания в сокетах 478, 604 и 775

VID[5:0]	VCC	VID[5:0]	VCC
001010	0,8375	011010	1,2125
101001	0,8500	111001	1,2250
001001	0,8625	011001	1,2375
101000	0,8750	111000	1,2500
001000	0,8875	011000	1,2625
100111	0,9000	110111	1,2750

[1] Здесь буквы A и E, следующие за значением частоты ядра, позволяют отличать новые процессоры (A) от старых с той же частотой (E).

VID[5:0]	VCC	VID[5:0]	VCC
000111	0,9125	010111	1,2875
100110	0,9250	110110	1,3000
000110	0,9375	010110	1,3125
100101	0,9500	110101	1,3250
000101	0,9625	010101	1,3375
100100	0,9750	110100	1,3500
000100	0,9875	010100	1,3625
100011	1,0000	110011	1,3750
000011	1,0125	010011	1,3875
100010	1,0250	110010	1,4000
000010	1,0375	010010	1,4125
100001	1,0500	110001	1,4250
000001	1,0625	010001	1,4375
100000	1,0750	110000	1,4500
000000	1,0875	010000	1,4625
111111	Отключен	101111	1,4750
011111	Отключен	001111	1,4875
111110	1,1000	101110	1,5000
011110	1,1125	001110	1,5125
111101	1,1250	101101	1,5250
011101	1,1375	001101	1,5375
111100	1,1500	101100	1,5500
011100	1,1625	001100	1,5625
111011	1,1750	101011	1,5750
011011	1,1875	001011	1,5875
111010	1,2000	101010	1,6000

Таблица 6.6. Задание частоты системной шины процессоров Pentium 4

Сокет-478, 603, 604 BSEL[1:0]	Сокет-775 BSEL[2:0]	Частота, МГц
00	000	100/400 или 266/1064
01	001	133/533
10	010	200/800
11	011	166/667
	100	Резерв
	101	Резерв
	110	Резерв
	111	Резерв

Следующий *сокет 775* (рис. 6.8) значительно отличается от предшественников: теперь сокет вместо гнезда стал «ежиком», «колючки» которого упираются в контактные площадки процессора в корпусе LGA. Дополнительные контакты здесь тоже пошли на нужды питания: 226 контактов питания (VCC), 24 контакта

VTT и 273 контакта «земли» (VSS). Идентификация питания сигналами VID[5:0] совпадает с последней редакцией сокета 478, диапазон напряжений — **0,** 8375-1,600 В (см. табл. 6.5). Частота шины задается уже тремя сигналами BSEL[2:0] (см. табл. 6.4). Сокет комплектуется 4-контактным разъемом для подключения вентилятора с ШИМ-управлением (рис. 6.8).

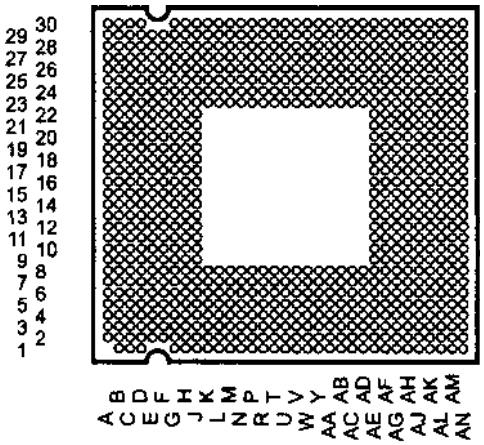


Рис. 6.8. Сокет 775

Для процессоров Xeon предназначены *сокеты 603* и *604* (рис. 6.9).

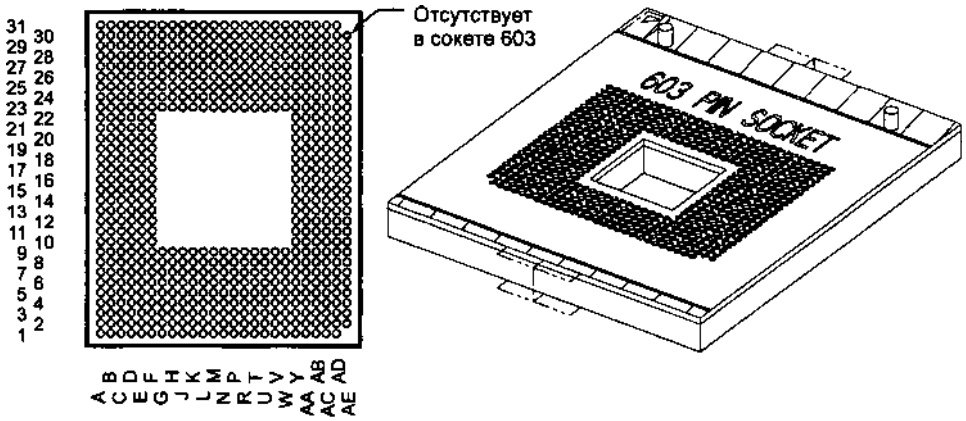


Рис. 6.9. Сокеты 604 и 603

В отличие от обычных процессоров, в интерфейс Xeon входит дополнительная шина SMBus (I_2C) для доступа к термодатчику и памяти идентификации процессора. Правда, эта шина присутствует не на всех моделях процессоров. Кроме того, имеется сигнал для отключения внутренних терминаторов: в мультипроцессорной системе терминаторы должны включаться только на последнем про

цессоре, подключенном к шине. Частоту FSB заказывает процессор сигналами BSEL[1:0] (контакты AB3 и AA3, см. табл. 6.6), однако процессоры для сокета 603 использовали только частоту 100 МГц. У процессоров Xeon с частотами ядра 0,8-2 ГГц коэффициент умножения частоты FSB задается состоянием сигналов в момент окончания сброса (табл. 6.7). Для последующих процессоров в информационном листке таблица задания коэффициентов не приводится — очевидно, коэффициент умножения фиксирован. У 64-битных процессоров XeonMP коэффициент умножения задается иными сигналами (табл. 6.8).

Таблица 6.7. Задание коэффициента умножения частоты процессоров Xeon 0,8-2,4 ГГц

Fcore/FSB	Fcore, ГГц	LINT[1]/ NMI	A20M#	IGNNE#	LINT[0]/ INTR
8	0,8	H	H	H	H
9 или 23	0,9 или 2,3	H	H	H	L
10	1,0	H	H	L	H
11	1,1	H	H	L	L
12	1,2	H	L	H	H
13	1,3	H	L	H	L
14	1,4	H	L	L	H
15	1,5	H	L	L	L
16	1,6	L	H	H	H
17	1,7	L	H	H	L
18	1,8	L	H	L	H
19	1,9	L	H	L	L
20	2,0	L	L	H	H
21	2,1	L	L	H	L
22	2,2	L	L	L	H
24	2,4	L	L	L	L

Таблица 6.8. Задание коэффициента умножения частоты 64-битных процессоров XeonMP

Fcore/FSB	Fcore, ГГц	A21#	A20#	A19#	A18#	A17#	A16#
16	2,66	L	H	L	L	L	L
17	2,83	L	H	L	L	L	H
18	3,0	L	H	L	L	H	L
19	3,16	L	H	L	L	H	H
20	3,33	L	H	L	H	L	L
22	3,66	L	H	L	H	H	L

В *сокете 603* для «земли» и питания первоначально (2001 г.) использовалось по 155 контактов. Затем число контактов VCC и VSS увеличили до 190 и 189 соответственно. Напряжение питания задается 5-битным идентификатором VID[4:0] так же, как и в сокете 423 (см. табл. 6.4).

Сокет 604 был введен для процессоров с частотой FSB 133/533 МГц, причем дополнительный контакт (AE30) служит исключительно механическим клю-

чом, предотвращающим установку новых процессоров в старые системные платы. В первой редакции в плане питания сокет 604 был идентичен сокету 603 (190 и 189 контактов для питания и «земли»). Однако для процессоров с частотой системной шины 200/800 МГц появился вариант сокета 604 с 6-битной идентификацией питания (см. табл. 6.5), в нем для питания и «земли» используются 181 и 185 контактов. Чтобы новый процессор (с более низким питанием) не запускался при его ошибочной установке в старые системные платы с сокетом 604, применяют контакт G7. В первой редакции сокета он был «землей» (VSS), для новых процессоров он считается входом BootSelect и должен быть свободен (если его заземлить, процессор работать не будет). Также появился FORCEPR# (контакт A15), позволяющий принудительно активировать цепи термоконтроля (TCC). Кроме того, вывод E1 стал ключом, управляющим подачей питания VTT (сигнал VTTEN): на процессоре он свободен, на системной плате должен подтягиваться к питанию. У старых процессоров он был заземлен.

Далее появилась редакция сокета для 64-битных процессоров с кэшем L3 (2005 год), в которых для питания ядра (VCC) используется 130 контактов, а 31 контакт выделен под питание кэша L3 (Vcache). При этом идентификация основного питания осталась 6-битной, а напряжение питания кэша задается специальными сигналами CVID[3:0] (табл. 6.9). Для кэша может использоваться такой же регулятор, как и для основного питания, при этом его старшие биты CVID[5:4] должны быть подключены к «1». Для CVID[3:0] задействуются выводы, ранее применявшиеся для подачи питания VCC. Очевидно, что установка процессора с нераздельным питанием в плату с раздельным питанием приведет к отключению регулятора питания кэша и на часть выводов питание VCC подано не будет (что плохо). Обратное несоответствие тоже не сулит ничего хорошего: процессор может пытаться подавать низкий уровень на выводы CVID, на которые основной регулятор напряжения будет подавать напряжения питания (с большим током). Упоминаний о механических ключах, препятствующих установке несоответствующего процессора, нет. Контакт C1 у некоторых процессоров, выполненных по технологии 90 нм, считается входом Optimized/Compat# (у других он резервный), на системной плате он должен быть свободен (как признак поддержки данного процессора). Этот контакт используется у процессоров как с раздельным, так и с совместным питанием.

Таблица 6.9. Идентификация питания кэша в сокете 604

CVID[3:0]	VCC	CVID[3:0]	VCC
1111	Отключено	0111	1,275
1110	1,100	0110	1,300
1101	1,125	0101	1,325
1100	1,150	0100	1,350
1011	1,175	0011	1,375
1010	1,200	0010	1,400
1001	1,225	0001	1,425
1000	1,250	0000	1,450

Сокеты для процессоров AMD

Для процессоров 6-го поколения (вплоть до K6-III) фирма AMD применяла сокеты 7 и Super 7 с интерфейсом «классического» процессора Pentium.

Для *процессора Athlon* фирма AMD ввела *слот* Д механически (и только!) совместимый со слотом 1 (SC242). Более дешевые *процессоры Duron*, а также Athlon, начиная с модели 4 (кэш 256 Кбайт), Athlon XP, Athlon MP и первые модели Sempron (сменившие Duron) используют *сокет А*, он же *сокет 462* (рис. 6.10). И в слоте, и в сокете задействована шина Athlon System Bus (основанная на EV-6), которая не применяется в процессорах x86 иных фирм.

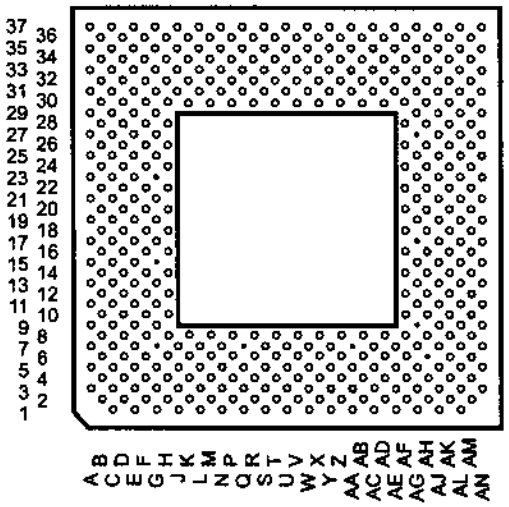


Рис. 6.10. Сокет 462 (сокет А)

Для задания напряжения питания используются сигналы VID[4:0] (контакты J7, L7, L5, L3 и L1, табл. 6.10); напряжение питания ядра можно проверить на контактах B4, B8, B12.

Таблица 6.10. Идентификация питания в сокете-А (462)

VID[4:0]	VCC_CORE	VID[4:0]	VCC_CORE
00000	1,850	10000	1,450
00001	1,825	10001	1,425
00010	1,800	10010	1,400
00011	1,775	10011	1,375
00100	1,750	10100	1,350
00101	1,725	10101	1,325
00110	1,700	10110	1,300
00111	1,675	10111	1,275
01000	1,650	11000	1,250
01001	1,625	11001	1,225

продолжение

Таблица 6.10 *(продолжение)*

VID[4:0]	VCC_CORE	VID[4:0]	VCC_CORE
01010	1,600	11010	1,200
01011	1,575	11011	1,175
01100	1,550	11100	1,150
01101	1,525	11101	1,125
01110	1,500	11110	1,100
01111	1,475	11111	NoCPU

Для задания тактовой частоты системной шины (частота передачи данных вдвое выше) процессоры имеют выходные сигналы FSB_Sense [1:0] (табл. 6.11), первоначально сигналы не использовались (частота была фиксирована — 100 МГц). Частота ядра определяется тактовой частотой и коэффициентом умножения. Процессоры сообщают чипсету коэффициент умножения частоты по линиям FID[3:0] (контакты Y3, Y1, W3 и W1, табл. 6.12). В зависимости от принятого коэффициента чипсет формирует пакет инициализации SIP, который передается процессору по специальному последовательному интерфейсу. Коэффициент умножения, напряжение питания и быстродействие кэша задаются перемычками на корпусе процессора, которые на отдельных моделях можно нарисовать обычным карандашом.

Таблица 6.11. Задание частоты системной шины в сокете 462

Сокет-462 FSB_Sense [1:0]	Тактовая частота, МГц
00	200
01	166
10	Резерв
11	133

Таблица 6.12. Кодирование коэффициента умножения процессоров Athlon и Duron

FID[3:0]	Коэффициент
0000	11
0001	11,5
0010	12
0011	≥12,5'
0100	5
0101	5,5
0110	6
0111	6,5
1000	7
1001	7,5
1010	8
1011	8,5
1100	9
1101	9,5

FID[3:0]	Коэффициент
1110	10
1111	10,5

[1] Процессорам с любыми коэффициентами, превышающими 12,5, требуются одинаковые данные SIP.

Для процессоров Athlon 64 и Sempron с 64-битным расширением предназначены сокеты 754, 939 и 940; в этих процессорах установлен встроенный северный мост с интерфейсами DDR SDRAM и HyperTransport. Сокеты предназначены для процессоров в корпусах mPGA размером в плане 40 x 40 мм, шаг выводов 1,27 мм. У сокета 754 матрица выводов размером 29 x 29 мм, у сокетов 939 и 940 — 31 x 31 мм (рис. 6.11). Питание идентифицируется в соответствии с табл. 6.13. Тактовая частота — 200 МГц. Контроллер памяти обеспечивает ECC- контроль; при обнаружении исправимой ошибки он записывает в память исправленное значение (чтобы ошибки не накапливались). Поддерживаются модули памяти DDR200, DDR266, DDR333 и DDR400. Реальная частота шины памяти зависит и от частоты ядра (множителя тактовой частоты 200 МГц). При этом память DDR200 и DDR400 работает на максимальной частоте (DDR400 в процессорах с частотой ядра 800 МГц работает на частоте 160 МГц, а не 200). Память DDR266 и DDR333 при определенных коэффициентах умножения работает на частотах, меньших номинальных; худшие случаи: при коэффициенте умножения 5 — 125 МГц вместо 133, при коэффициенте 6 — 150 Мгц вместо 166.

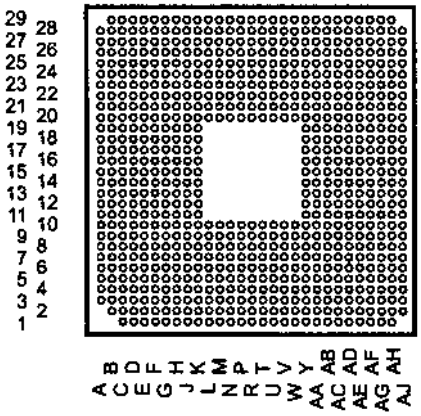


Рис. 6.11. Сокет 754

Таблица 6.13. Идентификация питания в сокетах 754, 939 и 940

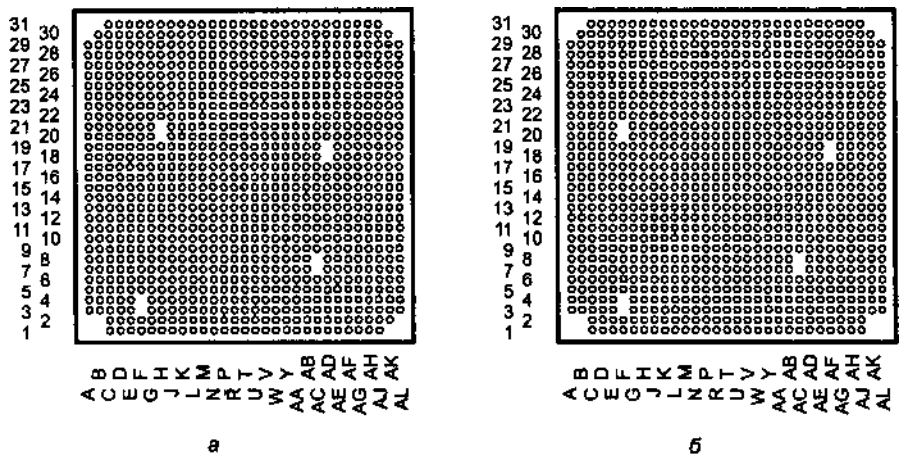
VID[4:0]	VCC_CORE	VID[4:0]	VCC_CORE
00000	1,550	10000	1,150
00001	1,525	10001	1,125
00010	1,500	10010	1,100
00011	1,475	10011	1,075
00100	1,450	10100	1,050

продолжение

Таблица 6.13 (*продолжение)*

VID[4:0]	VCC_CORE	VID[4:0]	VCC_CORE
00101	1,425	10101	1,025
00110	1,400	10110	1,000
00111	1,375	10111	0,975
01000	1,350	11000	0,950
01001	1,325	11001	0,925
01010	1,300	11010	0,900
01011	1,275	11011	0,875
01100	1,250	11100	0,850
01101	1,225	11101	0,825
01110	1,200	11110	0,800
01111	1,175	11111	NoCPU

В *сокете 754* имеется один 16-битный интерфейс HyperTransport, который может работать со скоростями передач 400, 800, 1200 или 1600 MT/s (миллионов передач в секунду); скорость устанавливается при инициализации интерфейса. Разрядность входных и выходных данных составляет 16 бит, так что достижима скорость передачи от 0,8 до 3,2 Гбайт/с в каждом направлении. Поскольку возможен полнодуплексный обмен, интерфейс может обеспечивать пропускную способность до 6,4 Гбайт/с. Интерфейс памяти 64-битный, возможно подключение до трех небуферированных модулей DIMM.



а *б*

Рис. 6.12. Сокеты: а - 939, *б* - 940

В *сокетах 939* и *940* (рис. 6.12) возможно использование до трех 16-битных интерфейсов HyperTransport. Число и характеристики интерфейсов зависят от типа PC: для настольных компьютеров интерфейс один, для серверных и рабочих станций — три. Некоторые процессоры обеспечивают скорость передач до 2000 MT/s. Когерентность памяти поддерживается только у процессоров, пред-

назначенных для мультипроцессорных систем. При этом процессоры для двухпроцессорных систем поддерживают когерентность только на одном интерфейсе (любом), а процессоры для мультипроцессорных систем — на всех трех. Поддержание когерентности оборачивается некоторым повышением задержек обращения к памяти. В сокетах 939 и 940 шина данных интерфейса DDR SDRAM 128-битная, что позволяет использовать как двухканальную, так и одноканальную память. Процессоры для сокета 939 поддерживают небуферированную память до четырех модулей (DIMM), для сокета 940 — до восьми модулей с регистрами (registered DIMM).

Оперативная память (DRAM)

Наиболее частые изменения конфигурации PC связаны именно с оперативной памятью — обычно стремятся к увеличению ее объема и повышению производительности. Вся оперативная память современных PC располагается на системной плате. Первые модели (XT, AT-286) позволяли наращивать оперативную память путем установки в слот ISA специальных карт расширения. Однако быстродействие памяти, подключенной через шину расширения, оставляет желать лучшего. Кроме того, появились компактные модули SIMM, SIPP, а позднее и DIMM, корпуса микросхем памяти стали более емкими, и проблема занимаемой площади стала менее острой. По этим причинам многие модели AT-286 и большинство моделей AT-386 и выше в качестве оперативной уже не воспринимали память на модулях расширения, устанавливаемых в слоты шин расширения. Отметим, что были модели AT-286, у которых модуль памяти устанавливался в специальный слот системной шины, а у некоторых серверных платформ ОЗУ — на отдельных платах или платах процессоров, но это уже не унифицированные рядовые компьютеры.

В качестве оперативной памяти используют микросхемы динамической памяти (DRAM) различных типов, подробно рассмотренных в главе 8. На системную плату устанавливают модули DIMM-168 (DRAM или SDRAM), DIMM-184 (DDR SDRAM), DIMM-240 (DDR2 SDRAM) или RIMM (RDRAM). Платы с гнездами для микросхем в корпусах DIP и ZIP, а также для модулей SIPP, SIMM («короткие» 30-контактные или «длинные» 72-контактные) с памятью DRAM уже вышли из обращения. Бывает, что базовый объем (например, 4 или 16 Мбайт) запаивается на системную плату непосредственно, а в зависимости от потребностей пользователя дополнительные модули устанавливаются в гнезда. Допустимый объем, возможные типы, организация и быстродействие памяти определяются чипсетом, количеством и типом разъемов (SIMM, DIMM, RIMM) для установки памяти и версией BIOS.

Модули памяти устанавливаются *банками* (теперь их иногда называют рядами). Банк работоспособен, только если он заполнен, причем однотипными микросхемами (модулями). Для процессоров класса Pentium и выше разрядность банка составляет 8 байтов, для работы достаточно установить один модуль DIMM или RIMM. Лучше брать один модуль требуемого объема, а при необходимости добавлять дополнительные. Для высокой частоты работы памяти число устанавливаемых модулей может быть меньшим, чем для низкой.

На современных системных платах объем корректно установленной памяти определяется автоматически (в отличие от первых машин, где его необходимо было задавать переключателями или джамперами). Некоторые версии BIOS при обнаружении факта изменения объема памяти просят подтвердить новое значение — войти в меню стандартного конфигурирования CMOS Setup и выйти с сохранением значений в CMOS.
Для конфигурирования системной платы важно знать *спецификацию быстродействия* применяемой памяти. Для модулей синхронной памяти указывают максимальную частоту синхронизации и латентность (CAS latency), более быстрая память при той же частоте имеет меньшее значение латентности. От спецификации быстродействия зависит эффективность (и даже возможность) применения памяти в конкретной системной плате на заданной частоте системной шины. Использование более медленной памяти (даже соседней спецификации, отличающейся, казалось бы, незначительно) может привести к появлению дополнительных тактов ожидания при операциях с ОЗУ, что заметно снижает производительность компьютера. Если же попытаться задать временную диаграмму памяти неоправданно быстрой, работа компьютера, скорее всего, будет неустойчивой. Для каждого типа памяти и каждой тактовой частоты имеется оптимальная спецификация памяти: менее быстродействующая память приведет к лишним (для данного типа памяти) тактам ожидания, более быстродействующая не даст преимуществ, но будет дороже. На временные диаграммы памяти влияет много факторов — задержки сигналов зависят от чипсета, наличия промежуточных буферов, длины проводников платы, количества устанавливаемых модулей и микросхем на них и т. п. Поэтому в каждой модели системной платы оптимальные спецификации для используемых тактовых частот свои. Требуемая спецификация быстродействия обычно указывается в документации на системную плату.

ВНИМАНИЕ ----------

Нестабильно работающая память является самой распространенной причиной «зависаний» и внезапных перезагрузок компьютера.

Чипсет (его контроллер памяти) может настраиваться на установленную память вручную (через CMOS Setup) или автоматически, считыванием идентификаторов установленных модулей или использованием собственных методов «исследования» свойств памяти. Современные чипсеты позволяют во время теста POST выполнять автоматическую идентификацию типов (и быстродействия) установленных модулей памяти и задавать оптимальные временные диаграммы в зависимости от установленной частоты системной шины, хотя реализация этой возможности зависит и от применяемой версии BIOS. Конечно, самый точный метод — считывание идентификаторов из EEPROM модуля (см. 8.3), но микросхемы EEPROM отсутствуют на дешевых модулях, не все чипсеты умеют считывать из них данные и не все версии BIOS это делают.
При ручном конфигурировании настройкой CMOS Setup задают требуемые параметры (спецификации быстродействия, а иногда и подробный список ряда временных параметров), которые должны соответствовать установленным мо

дулям. Параметры могут указываться для каждого слота памяти индивидуально, в более простых версиях Setup указываются параметры для всей памяти. В последнем случае, если используются модули с разным быстродействием, то указывают спецификацию самого медленного из них. Если от компьютера требуется стабильная работа, не следует «разгонять» память относительно рекомендованных (обычно автоматически определяемых) диаграмм — «разогнанная» память «имеет право» сбоить. Успешное прохождение тестов (POST, Checkit и пр.) — еще не гарантия стабильной работы.

В чипсете может быть предусмотрен контроль достоверности хранения данных, для чего должны быть установлены соответствующие модули памяти (и контроль должен быть включен настройкой CMOS Setup). По способу *контроля ошибок* различают следующие модули:

- None Parity — без четности (к сожалению, наиболее распространенные);
- Parity — с битами четности каждого байта, при поддержке чипсетом контроля четности позволяют обнаруживать ошибки;
- ECC — контроль всего слова с избыточным CRC-кодом, позволяющим выявлять и исправлять ошибки;
- EOS — модули, у которых механизм ECC «спрятан» в структуру модуля с контролем паритета;
- PG — модули с генератором четности (фикция для «ублажения» системных плат, требующих присутствия битов четности).

Кроме того, модули могут быть симметричными и асимметричными, иметь разные номиналы питающего напряжения, различаться параметрами регенерации и т. п. Здесь свойства элементов динамической памяти только перечислены, подробнее о великом множестве их нюансов см. главу 8.

ВНИМАНИЕ ----------

Установку и замену модулей памяти можно выполнять только при обесточенной системной плате. В платах (и блоках питания) ATX для этого требуется выключить питание механическим выключателем (или отсоединив шнур питания). У плат ATX в дежурном режиме (standby) на модули памяти может подаваться питание +3,3 В (на его присутствие могут указывать светодиоды, расположенные около гнезд памяти).

Для извлечения модулей DIMM следует развести в стороны рычаги экстракторов, расположенных по краям слота, — они вытолкнут модуль из слота (пытаться выдергивать модуль самому опасно). При установке модуля DIMM экстракторы должны защелкнуть модуль с обеих сторон — если этого не происходит, следует проверить соответствие модуля слоту (по ключам) и правильность ориентации.

Иногда при неполадках в памяти достаточно вынуть, продуть и поставить обратно модули памяти — неисправность может быть вызвана загрязнением контактов слота или модуля памяти.

Слоты расширения

Слоты расширения предназначены для установки карт различного назначения, расширяющих функциональные возможности компьютера. На слоты выводятся стандартные шины расширения ввода-вывода, а также промежуточные интерфейсы наподобие AMR и CNR. Стандартизованные шины расширения ввода-вывода обеспечивают основу функциональной расширяемости PC-сов- местимого персонального компьютера, который с самого рождения не замыкался на выполнении сугубо вычислительных задач. Хотя многие компоненты, ранее размещавшиеся на платах расширения, постепенно «переселяются» на системную плату, для настольных компьютеров набор шин расширения ввода-вывода имеет важное значение.

Ниже перечислены ныне используемые шины расширения ввода-вывода, реализованные в виде слотов на системной плате:

- PCI — самая распространенная шина, применяемая в компьютерах на процессорах класса 486 и выше. На системной плате 3-4 слота PCI могут сосуществовать со слотами шины ISA (на более старых платах с EISA и MCA). На платах, где это единственная шина расширения, число слотов увеличивают до 5-8. Шина и карты расширения существуют для разных напряжений питания интерфейсных схем (3,3, 5 В и универсальные); с частотой 33 (PCI 2.0) и 33/66 МГц (PCI 2.1); разрядностью 32 и 64 бита.
- PCI-X — высокопроизводительная модификация PCI, механически и электрически совместимая с PCI. Существует в версиях 1.0 и 2.0, различающихся доступными режимами и скоростями обмена.
- PCI-E (PCI Express) — слоты выделенных последовательных интерфейсов подключения периферийных устройств. Различаются числом линий (1x, 4x, 8x, 16x), слоты PCI-E 8x и 16x используются для подключения графической карты (акселератора). Периферийные устройства других типов с интерфейсом PCI-E пока не распространены.
- AGP — выделенный порт (единственный слот) для подключения графического акселератора, логически являющийся и слотом PCI. Слот (и графические карты) AGP 1.0 поддерживают напряжение питания интерфейсных схем 3,3 В и режим передачи 2x; AGP 2.0 поддерживает напряжение 1,5 В и режим 4x; AGP 3.0 — напряжение 0,8 В, режим 8x. Универсальные слоты и карты AGP поддерживают два номинала питания. Набор предоставляемых возможностей (режимы 2x/4x/8x, SBA, Fast Write) зависит от реализации порта и настройки CMOS Setup.
- ISA-8 и ISA-16 — традиционные универсальные слоты подключения периферийных адаптеров, не требующих высоких скоростей обмена. Раньше шина ISA была единственной шиной расширения, и для нее выпускалось (и выпускается) великое множество разнообразных карт расширения. Сейчас ISA изживается из системных плат.
- Слот PC Card, он же PCMCIA — слот расширения блокнотных компьютеров, который может присутствовать и в компьютерах настольного исполне

ния (слот устанавливается во внешний трехдюймовый отсек, к шине PCI или ISA подключается через карту-мост). Универсальный слот может работать и в режиме Card Bus — шины, являющейся упрощенным вариантом PCI.

На старых системных платах встречаются следующие шины:

- EISA — дорогая (по стоимости и системной платы, и плат расширения) 32-битная шина средней производительности, применявшаяся в основном для подключения контроллеров дисков и адаптеров локальных сетей в серверах. В настоящее время вытеснена шиной PCI. В слоты EISA можно устанавливать карты ISA (но не наоборот).
- MCA — шина компьютеров PS/2, применявшаяся и в некоторых серверных платформах. Производительность средняя. Адаптеры для шины MCA не слишком распространены, слоты MCA не совместимы ни с одним из типов карт расширения.
- VLB — локальная шина процессора (486), использовалась в паре со слотом ISA/EISA для подключения контроллеров дисков, графических адаптеров и контроллеров локальных сетей. С процессорами пятого поколения и выше не применяется.

Конфигурирование шин расширения предполагает в основном настройку их временных параметров (в CMOS Setup):

- Для шины PCI/PCI-X задаются частота синхронизации, режимы PCI-X (Mode 1, Mode 2) и другие параметры (конкурентные обращения, слежение за палитрами и т. д.).
- Для порта AGP задаются частота (номинал 66 МГц), поддерживаемые режимы, а также апертура AGP.
- Для шин ISA и PCI иногда настройкой CMOS Setup приходится распределять системные ресурсы (главным образом, линии запросов прерываний, см. 4.4).
- Для шины ISA помимо частоты (которая должна быть порядка 8 МГц) задают время восстановления для 8- и 16-битных обращений к памяти и вводу-выводу. Неустойчивая работа адаптеров может потребовать замедления шины ISA, но в настоящее время понижение ее производительности не слишком отражается на производительности компьютера в целом.

Количество и состав слотов шин расширения на различных платах варьируются. Типы слотов легко определить визуально, в этом поможет рис. 6.13. На этом рисунке, конечно, присутствие всех типов шин показано условно — реально на системных платах может находиться не более двух-трех типов слотов. Распространённые сочетания: AGP плюс PCI, AGP плюс PCI плюс ISA, PCI плюс ISA, PCI-E плюс PCI, PCI-E плюс PCI плюс ISA. При этом, естественно, слот AGP только один, а слоты PCI все с одним номиналом напряжения (но часть из них может быть 64-битными).

У карт PCI, PCI-E и AGP, в отличие от ISA/EISA и VLB, компоненты расположены на левой стороне печатной платы. Для экономии площади печатной платы часто используют так называемый *разделяемый слот* (shared slot). На самом

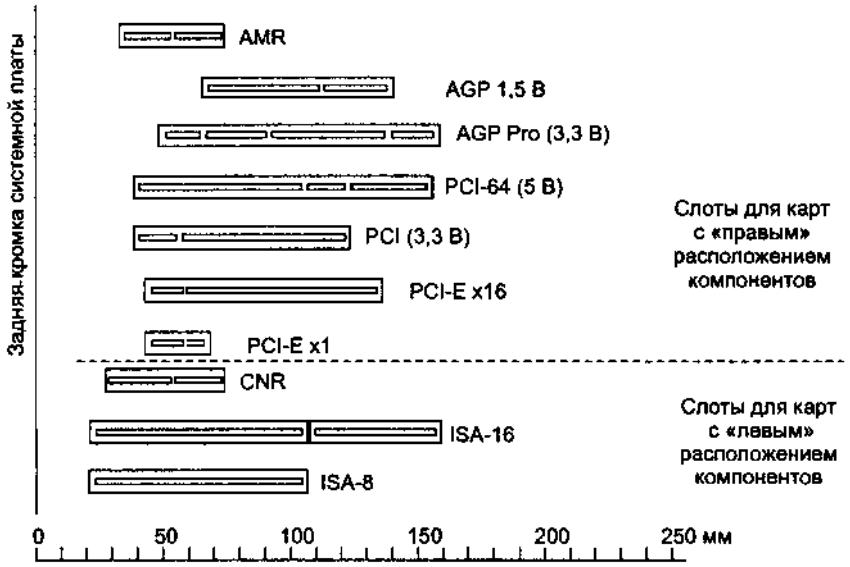


Рис. 6.13. Вид и положение слотов шин расширения

деле это разделяемое окно на задней стенке корпуса, которое может использоваться либо картой ISA, либо картой PCI. Таким образом, максимальное суммарное количество установленных адаптеров ISA и PCI оказывается на единицу меньшим, чем видимое количество слотов на системной плате.

Помимо слотов шин расширения на системной плате могут присутствовать слоты для расширения аудио- и коммуникационных возможностей системной платы — AMR и CNR, введенные фирмой Intel, или ACR. Эти слоты нельзя рассматривать как стандартные слоты расширения ввода-вывода, с помощью которых конечный пользователь может модернизировать компьютер; они адресованы производителям компьютеров (OEM) для предоставления дополнительных возможностей по комплектации. На этих слотах могут присутствовать не все интерфейсы, о чем, естественно, знают производители, но для пользователей это может оказаться неожиданностью, приводящей к сюрпризам при попытках модернизации. Установленная карта может потребовать поддержки в системной микросхеме BIOS, которую конечному пользователю ввести затруднительно. Как показывает опыт, время жизни этих слотов по сравнению с ISA и PCI слишком коротко — слот CNR пришел на смену AMR (без обратной совместимости) всего через несколько лет. Применение AMR или CNR для подключения аудиокодека позволяет вынести чувствительные к наводкам аналоговые схемы подальше от весьма «шумных» компонентов системной платы.

На системных платах со встроенным графическим контроллером может присутствовать слот расширения возможностей этого контроллера. Например, слот ADIMM для системных плат на базе чипсета SiS630, выглядящий как слот AGP (но иначе расположенный), позволяет установить либо дополнительную буферную память графического адаптера со 128-битной шиной данных, что

значительно повышает производительность акселератора, либо мост, позволяющий подключать к графическому контроллеру дополнительные интерфейсы — телевизионные интерфейсы (аналоговые и цифровые), интерфейс плоской панели и дополнительный независимый VGA-интерфейс. Для системных плат на чипсетах Intel может присутствовать слот AIMM, он же GPA, для подключения модулей видеопамяти.

Синхронизация и разгон

Основной тактовый генератор системной платы вырабатывает высокостабильные импульсы опорной частоты, используемой для синхронизации процессора, памяти и шин ввода-вывода. Поскольку быстродействие этих подсистем существенно различается, каждая из них может синхронизироваться со своей частотой. Помимо этих тактовых частот на системной плате присутствуют и другие — для синхронизации COM-портов, системного и CMOS-таймеров, НГМД и других периферийных адаптеров, но они не привлекают к себе внимания (нет поводов для их изменения). Когда появились компьютеры с тактовой частотой, обеспечивающей производительность выше стандартной модели XT (4,77 МГц) или AT (8 МГц), для совместимости с программами, у которых какие-либо задержки формировались путем подсчета циклов процессора, ввели режим и переключатель Turbo. В режиме Turbo процессор работает на максимальной скорости, а в «нормальном» — на пониженной, обеспечивающей «эталонную» производительность. Со временем производительность компьютера даже на пониженной от начального «эталона» скорости ушла далеко вперед, и большого смысла переключение режима уже не имеет. Когда говорят о производительности компьютера, обычно подразумевают, что он работает в режиме Turbo, так что этот режим и следовало бы называть нормальным. При наличии переключателя Turbo в машинах с процессором 8088/286/386 обычно переключали частоту синхронизации. В компьютерах на процессорах класса 486 и выше частоту оперативно переключать нельзя по разным причинам (например, потому что собьется умножитель частоты процессора). В них переключатель Turbo, если он имеется, может, например, отключать вторичное кэширование или включать режим прерывистой синхронизации (см. 7.4).

Ниже перечислены присутствующие на системной плате частоты:

- Host Bus Clock, она же FSB Clock, — частота системной шины (внешняя частота шины процессора). Эта частота является опорной для многих других, раньше устанавливалась перемычками (джамперами), теперь устанавливается через CMOS Setup. Современные процессоры используют частоты генераторов от 100 до 266 МГц, при этом частота передачи данных может быть удвоенной (у процессоров K7) и учетверенной (Pentium 4).
- CPU Clock, или Core Speed, — внутренняя частота процессора, на которой работает его вычислительное ядро. Эта частота на системной плате не присутствует, но на плате могут быть средства задания *коэффициента умножения.* Коэффициент умножения большинства современных процессоров фиксирован изготовителем; раньше его задавали перемычками на системной плате, заземляющими определенные выводы процессора. Новые процессоры

способны динамически изменять коэффициент умножения (для управления потреблением). Заметим, что не все модели процессоров воспринимают все сигналы управления коэффициентом умножения. Кроме того, одному и тому же положению джамперов могут соответствовать разные значения коэффициентов — трактовка управляющих сигналов зависит от марки и модели процессора.

- Memory Bus Clock — частота синхронизации памяти SDRAM, DDR(2) SDRAM или RDRAM должна соответствовать спецификации применяемых модулей. Повышение этой частоты позволяет повысить производительность памяти, что особенно важно для систем с портами AGP и PCI-E.
- AGP Clock — частота порта AGP, номинал — 66,6 МГц.
- PCI Bus Clock — частота шины PCI. Слишком низкая частота шины PCI тормозит обмен данными, что особенно заметно на графических адаптерах, SCSI-контроллерах и адаптерах скоростных локальных сетей, установленных в слоты PCI. Слишком высокая частота может привести к неустойчивости работы адаптеров. Согласно спецификации PCI 2.0, частота должна быть 25-33,3 МГц. Спецификация PCI 2.1 допускает применение частоты шины 66,6 МГц по согласию всех абонентов шины, в PCI-X частота (согласованная) может достигать 133 МГц.
- ISA Bus Clock — частота шины ISA, которая должна быть близка к 8 МГц. Она обычно задается в BIOS Setup через коэффициент деления системной частоты. Гнаться за ее оптимизацией в современных компьютерах смысла не имеет.

Вышеперечисленные частоты синхронизации в той или иной степени взаимосвязаны между собой. В синхронных чипсетах частоты соседних подсистем связаны жестко, как правило, простыми отношениями вроде 1 : 2, 1 : 3, 2 : 3. В асинхронных чипсетах частоты относительно независимы, что открывает больше перспектив для оптимизации производительности и «разгона» (overclocking) компьютера. Разгоняют все что можно: процессор, память, PCI, AGP и графический контроллер. Мечта «оверклокера» — плата с асинхронным чипсетом и большим количеством значений частоты для каждой подсистемы, что позволяет близко подходить к физическому пределу быстродействия. «Породистые» платы (например, от Intel) таких возможностей не предоставляют. Если штатные средства (джамперы или параметры CMOS Setup) не позволяют выставить желаемые частоты, для разгона иногда заменяют кварцевый резонатор другим, обеспечивающим их при штатных коэффициентах деления (умножения). Однако перепайка резонатора (как и других компонентов) наверняка приведет к потере гарантии на системную плату.

Наиболее частый объект разгона — центральный процессор, теории и практике его разгона посвящено много статей и сайтов Сети (например, iXBT.com). Изложенные далее взаимосвязи быстродействия компонентов, частоты, напряжения питания и рассеиваемой мощности применимы и к остальным подсистемам — памяти, шинам расширения, порту AGP и самому графическому акселератору. Правда, в этих подсистемах нет возможности «играть» напряжением питания

(кроме некоторых моделей акселераторов). Для памяти и шин расширения повышение частоты может приводить к уменьшению возможного числа устанавливаемых модулей и карт расширения. Это вполне понятно, поскольку каждый модуль (карта расширения) вносит свою нагрузку на шину (активную и реактивную), что приводит к затягиванию процессов переключения.

Вполне очевидно, что производительность конкретного процессора зависит от тактовой частоты ядра и частоты системной шины. Первая составляющая определяет темп обработки, а вторая — скорость доставки инструкций и данных. Максимально допустимая тактовая частота определяется как задержками между различными сигналами, так и предельной рассеиваемой мощностью. К примеру, для ячейки памяти после подачи адреса невозможно достоверно считать данные ранее, чем через характерное для нее время доступа. Мощность, рассеиваемая цифровыми схемами, возрастает с ростом частоты их переключений. Когда мощность, выделяемая процессором, начинает превышать мощность, отводимую радиатором, процессор перегревается и начинает сбоить, а позже необратимо выходит из строя (сгорает). Заметим, что чем меньше размеры элементов, тем меньше энергии необходимо затратить на их переключение. Этим объясняется, что процессоры, выполненные по более тонкой технологии (современный уровень — 90 нм), работают на более высоких частотах и потребляют меньшую мощность, чем их предшественники (например, изготовленные по технологии 0,18 мкм). Мощность, рассеиваемая процессором, снижается и при понижении напряжения питания. Но при этом замедляются переходные процессы и, следовательно, снижается допустимая частота синхронизации (а снижение частоты, в свою очередь, уменьшает потребляемый ток). Повышение тактовой частоты само по себе увеличивает потребляемую мощность, а для обеспечения стабильности работы (повышения быстродействия внутренней логики) может потребоваться некоторое повышение напряжения питания, что дает дополнительное увеличение потребляемой мощности. Таковы, в общих чертах, причины взаимного влияния частоты, напряжения питания и выделяемой мощности. В современных процессорах частота ядра определяется частотой внешней шины и коэффициентом умножения. Возможность независимого выбора внешней частоты синхронизации и внутреннего коэффициента умножения как раз и обеспечивает возможность «разгона» процессоров, а наличие управляемого регулятора напряжения позволяет подобрать подходящее питание.

Если вы хотите поэкспериментировать с «разгоном», соблюдайте осторожность. Не пытайтесь удвоить скорость, есть «разумные» пределы. Не повышайте сразу напряжение питания, проверьте степень нагрева процессора и радиатора стабилизатора напряжения после повышения частоты. Кстати, перегреть можно не только сам процессор — может расплавиться пластмассовый сокет. Если работа нестабильна, проверьте параметры временных характеристик ОЗУ и кэша в CMOS Setup. Повышать напряжение имеет смысл, только если температура приемлема, а работа неустойчива. При этом следует помнить, что повышение напряжения может и не привести к устойчивой работе.

Разогнанный процессор может нормально тестироваться программами типа Checklt, PCCheck, но устойчивая работа компьютера возможна далеко не всегда.

Проверкой работоспособности системы может служить длительная активная работа с каким-либо «тяжелым» приложением, например, в среде Windows, лучше в многозадачном режиме. Поскольку разгоном чаще всего пользуются для ускорения игр, признаком работоспособности может быть и длительное устойчивое функционирование игры в демонстрационном режиме. Причиной неустойчивой работы может являться недостаточное быстродействие динамической памяти и вторичного кэша. Эффект от «разгона» процессора может нивелироваться относительно медленной памятью, поскольку при переходе на более высокую частоту системной шины BIOS совместно с чипсетом увеличивает число тактов ожидания в циклах памяти. Изменение частоты системной шины может приводить и к снижению частоты шины PCI, что также снижает производительность компьютера в целом.

При тщательном подборе всех компонентов возможна ситуация, когда разогнанный компьютер будет устойчиво работать в реальных приложениях. Если это не так, то при сбоях, «зависаниях» и «вылетах» не ругайте команду Билла Гейтса (для этого есть масса других поводов), а восстановите «статус-кво». Стоит ли рисковать из-за 10 % прироста производительности в ответственных применениях, решайте сами. Следует иметь в виду, что фирма-производитель ставит маркировку частоты, исходя из обоснованных критериев качества и надежности.

BIOS

Базовая система ввода-вывода *(BIOS)* является ключевым элементом системной платы, без которого все ее замечательные компоненты представляют собой лишь набор дорогих «железок». BIOS, пользуясь средствами, предоставляемыми чипсетом, управляет всеми компонентами и ресурсами системной платы. Из этого следует, что используемая версия BIOS в значительной степени привязана к чипсету, и, кроме того, она должна «знать» особенности применяемых компонентов (процессор, память, интегрированные контроллеры). Код BIOS хранится в микросхеме энергонезависимой постоянной памяти *(ROM BIOS)* или флэш-памяти *(Flash BIOS).* С точки зрения регулярной работы тип носителя BIOS принципиального значения не имеет. С точки зрения модифицируемости флэш-память имеет явное преимущество (иногда, правда, оборачивающееся недостатком) — возможность модернизации прямо в компьютере. Определить, какой носитель BIOS используется на данной системной плате, можно, сняв наклейку с микросхемы (на ней обычно напечатаны выходные данные BIOS) и прочитав обозначение:

- 28Fxxx — флэш-память 12 В;
- 29Cxxx — флэш-память 5 В;
- 29LVxxx — флэш-память 3 В (редкий вариант);
- 28Cxxx — EEPROM, близкая по свойствам к флэш-памяти;
- 27Cxxx — память EPROM, записываемая на программаторе и стираемая ультрафиолетом (если есть стеклянное окошко);

- PH29EE010 — ROM фирмы SST, перезаписывается аналогично флэш-памяти;
- 29EE011 — флэш-память 5 В фирмы Winbond;
- 29C010 — флэш-память 5 В фирмы Atmel.

Причин взяться за модернизацию BIOS может быть несколько, некоторые из них перечислены ниже:

- Некорректная работа в некоторых режимах (например, самопроизвольный переход в энергосберегающий режим, выражающийся в остановках винчестера, гашении экрана или внезапном резком снижении производительности вроде бы нормально функционирующего компьютера). По мере выявления ошибок производитель выпускает новые версии BIOS (возможно, и с новыми ошибками).
- Несогласованность драйверов BIOS с требованиями новых версий ОС.
- Получение новых функциональных возможностей, повышение производительности.
- Желание иметь самую свежую версию (для любителей экспериментировать на себе).
- Желание стереть конфигурационную информацию в NVRAM (включая и ESCD), если для этой цели нет переключателя или параметра в CMOS Setup. Утилита перепрограммирования флэш-памяти выполняет это действие автоматически или предлагает его выполнить из своего меню.

Обновление флэш-BIOS предполагает программирование микросхем в целевом устройстве без дополнительной аппаратуры, с использованием собственного процессора PC, что «по-научному» называется In-System Write (ISW). Для этого необходима возможность загрузки утилиты программирования и собственно обновленного кода, для чего обычно используют накопители на гибких дисках. При неудачной модификации BIOS возможность загрузки с дискеты может оказаться утерянной, и если системная плата не предусматривает режима восстановления (boot recovery), придется задействовать внешний программатор.

Перед обновлением BIOS оцените свои возможности для отступления. Если системная плата и применяемый тип микросхемы не поддерживают режим восстановления, в случае неудачи есть шанс столкнуться с проблемой поиска программатора флэш-памяти. А если микросхема запаяна в плату, а не установлена в «кроватку», проблема поиска осложнится тем, что понадобится программатор с *адаптером для OBP* (On-Board Programming) *на данной системной плате.* Этот адаптер должен обеспечивать доступность линий адреса, данных, управления и питания флэш-памяти при неработающем процессоре системной платы. Такими адаптерами обладают далеко не все программаторы, поддерживающие требуемый тип флэш-памяти, и их подключение предусматривается далеко не всеми системными платами.

Новую версию BIOS лучше всего получать от изготовителя системной платы. Фирмы-разработчики BIOS (например, AMI, Award) новые версии BIOS для конечных пользователей не поставляют. Свои новые продукты с инструмен

тальными средствами они поставляют разработчику системной платы, производящему окончательную подгонку BIOS под конкретную модель платы, особенности которой он знает лучше всех. В первом приближении BIOS различных системных плат с одинаковыми или близкими чипсетами могут оказаться (или показаться) совместимыми — по крайней мере, при включении выводится заставка, проходит тест POST и даже загрузка. Однако при более тщательном тестировании может оказаться, например, что невозможно обратиться к дискам (гибким или жестким), не работают порты, доступна не вся память и т. п. Хорошо, если при этом удастся загрузить утилиту перепрограммирования BIOS, чтобы вернуться к старой *(предварительно сохраненной!)* версии.

Утилиты перезаписи флэш-памяти привязаны к поддерживаемым типам микросхем энергонезависимой памяти, системным платам (чипсетам) и производителям (иногда и версиям) BIOS. Обычно не удается штатным способом (в компьютере) переписать BIOS со сменой производителя (Award, AMI, Phoenix). Как вариант возможна замена (хотя бы временная) микросхемы BIOS на снятую с аналогичной системной платы, но если микросхема припаяна, а не установлена «в кроватку», процедура замены сильно осложняется. Смело заниматься перепрограммированием BIOS можно только когда вы имеете доступ к программатору, и микросхема BIOS установлена в «кроватке».

Если новая версия BIOS не позволяет загрузить компьютер, ряд системных плат позволяют включить *режим восстановления.* Для этого на плате должен быть специальный переключатель или джампер. В режиме восстановления работает только дисковод, в который необходимо установить специальную дискету с файлом-образом ROM BIOS. При этом «сообщения» пользователю могут сводиться к подмигиванию индикатора дисковода и гудкам динамика. Язык этих сообщений должен приводиться в описании системной платы. Иногда режим восстановления включается автоматически (если блок начальной загрузки получает управление в начале теста POST, он всегда может оценить корректность содержимого основного блока ПЗУ и при необходимости включить режим восстановления).

Если же после неудачного перепрограммирования режим восстановления не спасает (или отсутствует), а доступного программатора нет, то есть хотя и рискованный, но возможный вариант «горячей замены» ROM BIOS. Для этого из аналогичной работоспособной системной платы извлекают микросхему BIOS, устанавливают ее вместо испорченной, включают и загружают компьютер как для режима перезаписи BIOS. При этом в Setup должно быть разрешено применение теневой памяти для области системного модуля BIOS. Далее, не выключая питания (опасно, но в безвыходном положении можно рискнуть), заменяют микросхему на неверно записанную и выполняют процедуру перезаписи. Компьютер продолжает работать, поскольку код BIOS исполняется из теневой области ОЗУ. Для перезаписи может быть использован файл-образ, полученный как копия «спасительной» микросхемы, сделанная той же программирующей утилитой.

Если говорить о недостатках флэш-BIOS, имеется в виду не только опасность потери работоспособности системной платы из-за неосмотрительных действий

пользователя, модернизирующего BIOS, но и дополнительное «поле деятельности» для вирусов. Стереть BIOS, зная работу чипсета и конкретной микросхемы памяти, можно даже отладчиком DEBUG (как — на всякий случай не скажу). Парольная (программная) защита перезаписи может быть взломана, а надежная аппаратная защита (необходимостью подачи высокого напряжения для стирания и программирования, а также сигнала защиты записи) имеется далеко не у всех микросхем энергонезависимой памяти и системных плат.

Решившись на обновление BIOS, необходимо придерживаться следующих рекомендаций:

- Убедитесь в том, что системная плата поддерживает программирование флэш-памяти (ISW).
- Убедитесь, что установленная микросхема BIOS не относится к EPROM. У этих микросхем имеется окошко, которое можно прощупать через наклейку или увидеть, сняв ее. Однако отсутствие окошка — еще не явный признак флэш-памяти: имеются микросхемы EPROM 27xxx без окошка.
- Установите джамперы в режим программирования флэш-памяти.
- Компьютер желательно подключить к источнику бесперебойного питания — сбой питания во время программирования при отсутствии режима восстановления (переключателя Boot Recovery) может привести к потере возможности программирования в режиме ISW.
- В CMOS Setup необходимо отключить применение теневой памяти (Shadow ROM) к области BIOS и запретить функции энергосбережения (Power Management — Disable).
- ОС для запуска утилиты программирования должна загружаться в реальном режиме и без драйверов верхней памяти (HIMEM.SYS, EMM386.EXE, QEMM386.SYS и т. п.). Этого можно достичь загрузкой с системной дискеты, не содержащей ссылок на драйверы в файле CONFIG.SYS (или самого этого файла). В MS-DOS 6.x можно шунтировать стартовые файлы нажатием клавиши F5 в начале загрузки. При использовании Windows в меню, появляющемся при нажатии клавиши F8 в начале загрузки, выбирают команду Safe mode command prompt only.
- Загрузив утилиту программирования, прежде всего сделайте файл резервной копии текущей версии BIOS — она может вскоре пригодиться.
- Утилита обычно определяет тип установленной флэш-памяти. Если определить тип ей не удается («unknown»), программирование выполнять нельзя — требуется подыскать подходящую утилиту.
- Если во время программирования появляются сообщения об ошибках — не выключайте питание, *не нажимайте кнопку Reset или клавиши перезагрузки*. Попытка перезагрузки в этом случае может привести к «зависанию» компьютера навсегда или до восстановления. Не выходя из утилиты, попытайтесь восстановить прежнюю версию BIOS с ранее сделанной копии.
- После успешного завершения обновления перезагрузите компьютер и поработайте с новой версией BIOS. Старую версию желательно сохранить (на

дискете она занимает не так уж много места) — возможные проблемы новой версии могут проявиться значительно позже.

- Если модификация была безуспешной и привела к невозможности загрузки компьютера, воспользуйтесь переключателем (джампером) Boot recovery и восстановите прежнюю версию BIOS, после чего верните переключатель в исходное состояние.
- Пользоваться возможностью перепрограммирования блока начальной загрузки без веских на то причин не стоит — версия его кода на нормальную работу PC обычно не влияет. Перепрограммировать блок начальной загрузки можно только при нормальной работе основного блока BIOS, в противном случае сбой программирования блока начальной загрузки загонит пользователя в капкан.
- Некоторые утилиты позволяют очищать блоки параметров — память ESCD. Эта очистка приведет к потере информации об установленных устройствах PnP, что потребует их повторного конфигурирования. В некоторых случаях такая чистка даже полезна, поскольку система PnP пока еще далека от совершенства.

Иногда перепрограммировать флэш-BIOS приходится и для того, чтобы проинициализировать (или сбросить) некоторые параметры в энергонезависимых ячейках памяти чипсета, которые для обычных утилит (CMOS Setup) недоступны, но могут быть неудачно настроены, например при установке ОС Windows.

Помимо обновления версии перепрограммирование BIOS используют и для других целей. Можно, например, изменить (вставить) логотип, появляющийся во время теста POST, на произвольную растровую картинку определенного формата. Возможно также изменение параметров, принимаемых в CMOS Setup по умолчанию (BIOS Defaults, Power-On Defaults, см. 6.6). Для этого существуют специальные утилиты, ориентированные на определенные версии BIOS.

Память CMOS — питание и обнуление

Память CMOS, совмещенная с часами-таймером, является энергонезависимой памятью конфигурации компьютера. Помимо ячеек стандартного назначения в CMOS имеются ячейки, которые используются по усмотрению разработчика BIOS для хранения текущих параметров чипсета, задаваемых встроенной утилитой CMOS Setup. Для питания этой памяти на системной плате устанавливается литиевая батарейка (аккумуляторы применяются редко). Она имеет нормальный срок жизни несколько лет. О необходимости ее замены говорит сообщение «CMOS Battery State Low» или «CMOS Checksum Error» во время теста POST, обычно появляющееся после длительного (несколько дней) перерыва в работе машины. Первым признаком необходимости ее замены может быть и остановка внутренних часов-календаря при выключении машины (они превращаются в счетчик «моточасов»). Иногда параметры Setup из-за разряда батареи теряются и без диагностических сообщений.

Память CMOS является важным узлом компьютера, и правильность ее питания может существенно влиять на «здоровье» компьютера в целом. Случай из практики: «села» батарейка, терялись время и параметры, но поменять батарейку не торопились еще и из-за «капризности» системной платы. После включения компьютера плата долгое время «не заводилась» — POST нажатием кнопки Reset удавалось запустить лишь после длительного прогрева. Проверка и даже замена блока питания результатов не дала. Однако после ОТКЛЮЧЕНИЯ батарейки плата стала работать нормально (естественно, теряя содержимое CMOS при выключении, но запускаясь без проблем). Установка свежей батарейки полностью восстановила работоспособность платы.

На старых платах батарейка представляла собой обычно синий бочонок, припаянный к плате. Сейчас пришла пора их массового выхода из строя на системных платах машин 286 и 386. При этом теряется информация CMOS, но что гораздо хуже — электролит может растечься и вызвать появление паразитных контактов и разъедание элементов системной платы. Протекшую батарейку надо обязательно извлечь, а плату отчистить щеточкой и промыть. Найти новую батарейку такого же размера бывает сложно, но ее можно заменить любой другой с аналогичным напряжением (обычно 3-4,5 В). Новую батарейку можно подключить к контактам разъема внешней батареи (Ext. Bat.), имеющимся на большинстве системных плат (рис. 6.14), сняв перемычку питания от внутренней батареи. Существуют внешние батарейки для PC, заключенные в пласт-массовые корпуса с проводами подключения. Этот корпус с помощью «липучки» закрепляют в удобном месте. Возможно применение простого и надежного самодельного агрегата многоразового использования: в деревянную бельевую прищепку вкалываются две канцелярские кнопки с припаянными проводами, и ими «закусывается» батарейка-таблетка (например, типа 2732). Закрепить такую конструкцию в корпусе не составит особого труда. На современных системных платах чаще применяется батарейка-таблетка в специальном держателе, которую легко заменить.

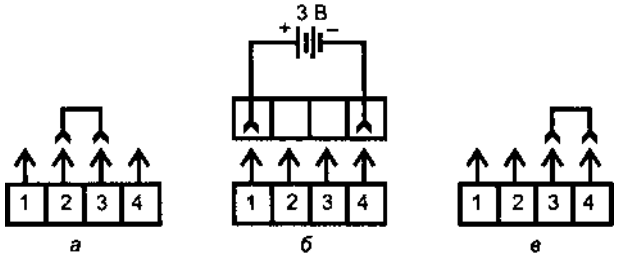


Рис. 6.14. Подключение внешней батарейки и обнуление CMOS: а — работа от внутренней батарейки, б — работа от от внешней батарейки, *в* — обнуление CMOS

Разъем подключения внешней батарейки используется и для обнуления CMOS (на старых платах). Такая необходимость может возникнуть, например, при утере входного пароля, заданного в CMOS Setup (или при необходимости его «взлома»). Теоретически, для этого достаточно при выключенном компьютере

на несколько минут переставить перемычку в положение, показанное на рис. 6.14, *в*.

На случай утери пароля во многих старых версиях BIOS от Award имелся «черный ход» — заводские пароли, например, ?award, AWARD_SW, KDD, j262, j256, j322, Syxz, HLT. Если паролем закрыт только вход в CMOS Setup, то пароль (для Award и AMI) может быть определен с помощью специальных утилит (если паролем закрыта и загрузка, утилиты, естественно, не запустить).

Иногда для сброса пароля предназначен отдельный джампер или переключатель (применяется, если пароль хранится не в CMOS, а в NVRAM). В этом случае, переключив джампер, необходимо включить компьютер — только тогда пароль будет сброшен, после чего вернуть джампер в исходное состояние.

ВНИМАНИЕ ----------

Обнулением CMOS следует пользоваться с осторожностью. Информация о конфигурации, которая там хранится, восстанавливается относительно легко — проблемы могут возникнуть только с нестандартными параметрами жестких дисков, заданными вручную, но ими уже давно не пользуются. Если ранее параметры определялись автоматически, то после обнуления такое определение легко повторить (могут остаться лишь сомнения в выборе режима LBA или Large для дисков большого объема). Помимо информации Setup в CMOS может быть записан ключ привязки какого-либо прикладного ПО к конкретной системной плате, и при обнулении CMOS ключ будет утерян. Так, например, легко было «убить» легальную копию пакета «1С:Бухгалтерия». Для страховки от таких неприятностей после установки подобных защищенных продуктов следует сохранить в файле содержимое CMOS. Это позволяет сделать, например, тестовый пакет QAPIus.

Однако бывают случаи, когда штатными способами пароль не сбросить. Тогда есть еще один способ — закоротить выводы микросхемы CMOS *при отключенном питании и отключенной батарейке*. Для этого кусочек фольги (годится от шоколадной конфеты) прикладывается сверху к микросхеме и аккуратно приглаживается ногтем к выводам по периметру корпуса. Чтобы не утруждать себя идентификацией микросхемы CMOS, эту операцию можно проделать со всеми «подозрительными» многовыводными микросхемами — их не так уж и много.

Обнуление CMOS (сброс всех параметров к значениям, заданным по умолчанию) может быть выполнено чисто программно, записью определенного кода в одну из ячеек CMOS. Для BIOS Award и AMI требуется записать код 17h в ячейку 17, для BIOS Phoenix — в ячейку FFh. Это можно сделать даже с помощью отладчика (DEBUG), запущенного в «чистой» среде DOS (не в окне Windows), следующими командами:

- `-0 70 17` — установка адреса ячейки CMOS (для Phoenix BIOS — команда `-O 70 FF`);
- `-0 71 17` — запись данных;
- `Q` — выход.

Периодическое разрушение информации CMOS при включении питания может быть вызвано вовсе не батарейкой, а недостаточной задержкой сигнала

PowerGood относительно момента установления питающего напряжения или, наоборот, излишней задержкой этого сигнала после выключения источника (см. главу 3). Определить причину довольно просто. Если перед включением питания удержать нажатой кнопку Reset и отпустить ее только через несколько секунд, в большинстве случаев это имитирует увеличение задержки сигнала PowerGood. Если при таком способе включения данные CMOS сохраняются, дело в *малой задержке при включении*. Если данные CMOS все равно теряются, нужно проверить версию задержки при отключении. Для этого кнопку Reset следует нажимать перед выключением питания и удерживать еще несколько секунд — этим имитируется ускорение снятия сигнала PowerGood. Если при таком способе выключения данные CMOS сохраняются, дело в *большой задержке при выключении*. В обоих случаях требуется замена или ремонт (подстройка) блока питания.

6.3. Конструктивы и установка плат

Унификация и стандартизация компонентов PC распространяется на системные платы, предназначенные для установки в корпуса обычного исполнения. Некоторые «фирменные» платы имеют специфические габаритные и присоединительные размеры, и их можно устанавливать только в «родные» корпуса. Таким специфическим конструктивом отличаются, например, платы и корпуса компьютеров IBM PS/2, Acer, Compaq, Digital, Packard Bell и ряд других. К ним, естественно, некоторые последующие тезисы неприменимы. Здесь будут рассмотрены конструктивы системных плат, предназначенных для установки в корпуса машин конструктива ATX. Системные платы AT интереса уже не представляют. Установка плат ATX в традиционный корпус AT с блоком питания AT (как, впрочем, и обратная комбинация) весьма проблематична.

Стандарт ATX на конструктив системной платы и корпуса PC определяет размеры плат: полный формат 305 x 244 мм, Mini-ATX — 284 x 208 мм, Micro- ATX — 244 x 244 мм, Flex ATX - 229 x 191 мм. Здесь задается длина (размер по задней кромке 305, 284, 244 или 229 мм), а ширина может быть и меньшей. По сравнению с предшествующим конструктивом AT стандарт ATX существенно упрощает соединения, задавая достаточно удобное местоположение ключевых компонентов системной платы. Представление о расположении компонентов платы ATX и размеров ее различных вариантов дает рис. 6.15. Основные особенности компоновки ATX перечислены ниже:

- Все внешние разъемы (клавиатуры, мыши и встроенной периферии) располагаются в два этажа и сгруппированы у правого края платы. Для них в корпусе ATX предусмотрено одно большое прямоугольное окно.
- Процессор может располагаться под блоком питания, и тогда его радиатор может обдуваться потоком воздуха внутреннего вентилятора блока питания или дополнительного вентилятора, устанавливаемого снаружи блока питания. Расстояние по высоте до блока питания позволяет менять процессор, не снимая системной платы.

- Разъемы адаптеров НГМД и IDE располагаются у правого переднего края платы. Это позволяет хорошо разместить кабели в корпусе и сократить их длину, что немаловажно для режимов PIO Mode 4 и UltraDMA порта IDE.
- Модули памяти устанавливаются в легкодоступном месте.
- В дополнение к традиционному набору питающих напряжений введен источник питания 3,3 В, позволяющий упразднить один из регуляторов VRM на системной плате.
- Для блока питания определен сигнал программно-управляемого отключения питания, что является эффективной защитой от преждевременного выключения питания при незакрытых приложениях. Полное отключение питания обеспечивается выключателем блока питания, который занял старое положение на задней панели корпуса.
- Блок питания для ATX имеет «дежурный» маломощный источник +5V Standby для питания цепей управления потреблением и устройств, активных и в спящем режиме (например, факс-модема, способного по звонку «разбудить» машину).

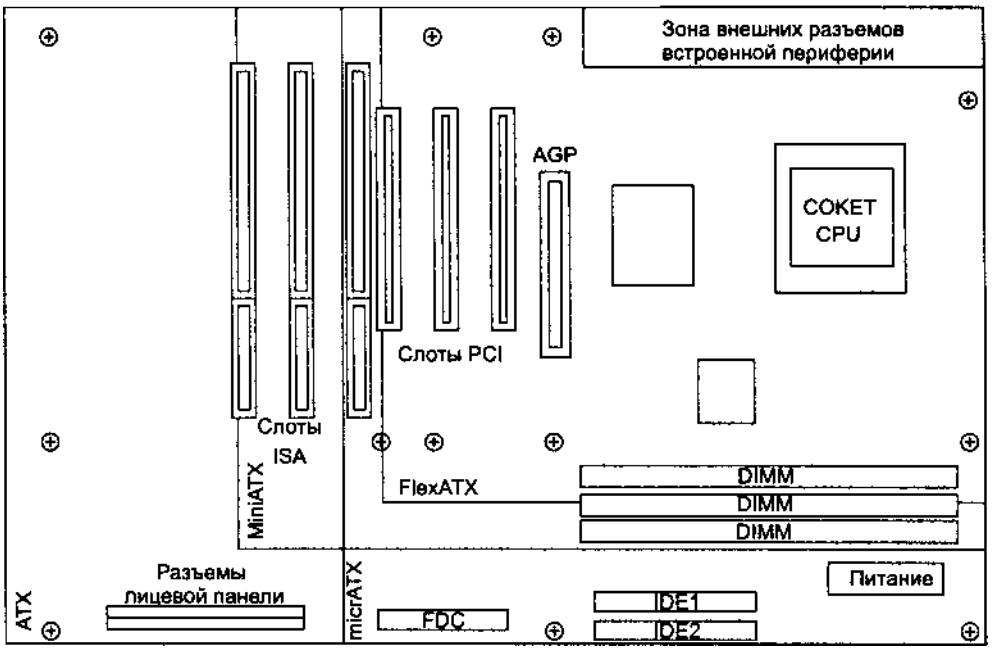


Рис. 6.15. Системная плата ATX

Платы имеют фиксированное относительно задней кромки расположение слотов и окна под разъемы, а также унифицированную систему крепежных отверстий, что облегчает ремонт и модернизацию системных блоков.

Платы устанавливаются с помощью пластмассовых вставок, входящих в прорези шасси. Эти вставки обеспечивают вертикальную и продольную (вдоль оси слотов расширения) фиксацию платы. Они позволяют выставить плату в правильное положение относительно задней стенки корпуса, которое уточняется при установке в слоты плат расширения. В требуемом положении плата фиксируется одним или несколькими винтами, завинчиваемыми в предварительно установленные в шасси резьбовые втулки. Эти же винты обеспечивают теоретически единственную точку соединения заземленного (через блок питания) корпуса компьютера с общим проводом источника питания.

Для того чтобы снять системную плату, из нее необходимо извлечь все карты расширения и отсоединить кабели подключения (по крайней мере, короткие). В корпусах типа Mini Tower необходимо снять (или, в некоторых корпусах, выдвинуть) шасси, на котором закреплена системная плата. Отвинтив крепежные винты, плату немного сдвигают влево, после чего ее можно снять с шасси. Установка платы производится в обратном порядке, фиксирующие винты затягиваются после установки платы в корпус и установки какой-либо платы расширения.

Ниже перечислены наиболее распространенные ошибки при установке платы:

- Недостаточное количество точек крепления. Шасси и плата обычно имеют избыточное количество возможных точек установки вставок и резьбовых втулок, из которых не все могут совпадать друг с другом. Используемые точки крепления обязательно должны окружать зону слотов расширения со всех четырех сторон (углов). В противном случае при установке плат расширения висящий край системной платы отогнется, что может привести к ненадежному контакту и даже скрытому обрыву печатных проводников системной платы.
- Неправильное использование крепежных винтов. Диаметр крепежных отверстий на плате позволяет вставлять в них как пластмассовые вставки, так и металлические крепежные винты. Отверстия, предназначенные для винтов, обычно с обеих сторон платы имеют ободок печатной шины «земли», или, наоборот, их окружает зона, свободная от печатных проводников. Около отверстий, предназначенных для пластмассового крепежа, близко к краю могут проходить тонкие печатные проводники. Если эти отверстия использовать для металлических винтов, возможно короткое замыкание проводников на корпус или даже их обрыв во время затягивания винтов. Если отверстие с близко расположенными проводниками все-таки приходится задействовать для винтов, на них следует установить (приклеить) изолирующие шайбы.
- Использование слишком длинных винтов. При этом винт не удается затянуть до фиксации платы. Это чревато ненадежностью заземления общего провода, что может приводить к случайным сбоям в работе. Если винт не удается затянуть, а винта покороче нет, можно подложить шайбу (если заземляющий ободок есть на нижней стороне платы, шайба может быть и изоляционной).

6.3. Подключение системной платы

Системная плата имеет множество интерфейсных разъемов, часть которых выводится на заднюю панель. Помимо них имеются внутренние разъемы для подключения питания, компонентов лицевой панели корпуса, интерфейсов накопителей на гибких и жестких дисках, портов ввода-вывода.

Питание к платам ATX подается через один 20- или 24-штырьковый разъем. Новые платы ATX12V получают питание (+12 В для преобразователей напряжения) и от дополнительного 4-контактного разъема. Надежные ключи не позволяют по ошибке перевернуть разъемы питания или соединить их со смещением (что было возможно в AT). Некоторые платы «переходного периода» имели дополнительно пару разъемов для питания от блоков питания AT, что, конечно, вело к потере возможности программного отключения питания.

Одна из точек крепления платы обеспечивает соединение общего провода GND с металлическим шасси системного блока, заземленного через сетевой шнур питания.

К *компонентам лицевой панели* относятся:

- кнопки RESET, POWER, TURBO (устарела);
- ключ блокировки клавиатуры;
- индикаторы включения, режимов энергопотребления, обращения к жесткому диску;
- динамик (на современных платах имеется встроенный динамик);
- интерфейсы — инфракрасный приемопередатчик, разъемы USB, FireWire, микрофонного входа и выхода на наушники.

Состав этих компонентов может меняться — есть, например, компьютеры с жидкокристаллическими дисплейными панелями, отображающими состояние системы. На некоторых (так называемых мультимедийных) корпусах установлены стереодинамики. Существуют и другие экзотические варианты.

Все компоненты лицевой панели обычно присоединяются отдельными парами или тройками проводов. Их разъемы подключаются к штырьковым разъемам, которые чаще всего располагаются вдоль передней кромки системной платы ближе к левому краю. Провода имеют запас длины, которого должно быть достаточно, чтобы дотянуться до любого возможного места расположения разъема. По укладке этих проводов можно судить о квалификации и аккуратности сборщика — хорошим тоном является подвязка проводов к шасси без их натяга и излишних свободных петель. Здесь дело не только в эстетике — непривязанные провода норовят лечь на лопасти вентилятора процессора. Излишний шум, возникающий от этого ненужного трения, позволяет только гадать, что произойдет раньше — перетрутся провода или сгорит заторможенный вентилятор, а потом и охлаждаемый им процессор (второй вариант явно хуже плохого первого).

У «породистых» корпусов, ориентированных на «родные» системные платы, все органы лицевой панели подключаются одним шлейфом и разъемом, что гораздо изящнее, но менее универсально.

Штырьковые разъемы подключения обычно имеют маркировку на системной плате. Сами разъемы на проводах маркировку имеют не всегда, но разноцветные провода легко проследить до точек подключения. Схему органов управления и индикации, а также распространенные варианты их маркировки иллюстрирует рис. 6.16.

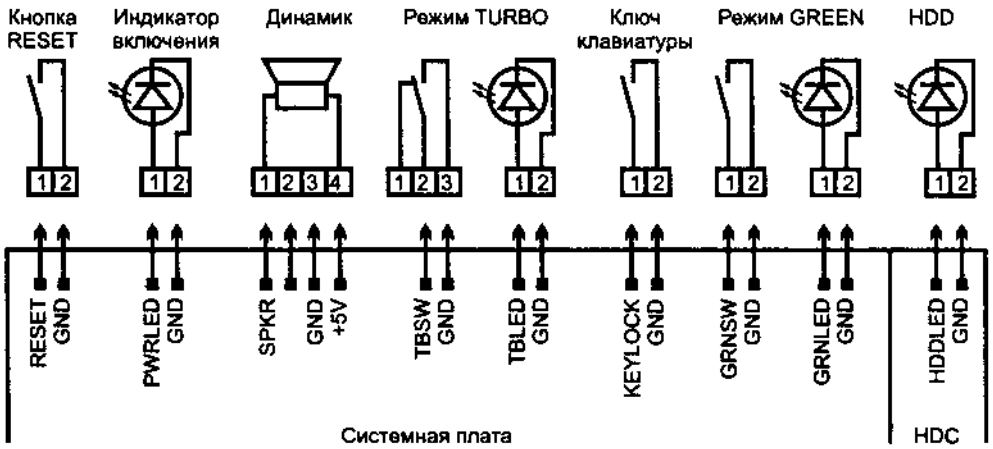


Рис. 6.16. Органы управления и индикации

Если маркировка на системной плате и описание платы отсутствуют, разобраться с подключением можно экспериментально, причем при соблюдении несложных правил довольно быстро и без особого риска. Далее мы приведем основные приметы разъемов подключения.

Разъем динамика практически всегда четырехштырьковый, причем динамик подключается к крайним выводам.

Для подключения ключа клавиатуры Keylock и индикатора включения питания на старых корпусах и платах использовался пятиштырьковый разъем. Иногда на него же выводился и сигнал от кнопки Reset (рис. 6.17). Сейчас эти элементы чаще подключаются отдельными двухштырьковыми разъемами.

Кнопки и индикаторы одним из выводов подключаются к шине GND. Подключение кнопки вместо индикатора и наоборот безобидно. Ток на индикатор ограничен резистором, установленным на системной плате, и короткое замыкание контактов его разъема безопасно. Подключение индикатора к разъему кнопки может привести разве что к его тусклому свечению, что наглядно свидетельствует об ошибке подключения. Для светодиодного индикатора важна полярность подключения — при ошибочном подключении он светиться не будет, для двухцветных индикаторов цвета окажутся перепутанными. Полярность подключения кнопок Reset и Keylock, естественно, безразлична. Разъем переключателя Turbo имеет три вывода (общий всегда посередине), но подключается к двухштырьковому разъему. От того, какая пара контактов будет подключена, зависит лишь то, какому положению (нажатому или отжатому) соответствует режим Turbo.

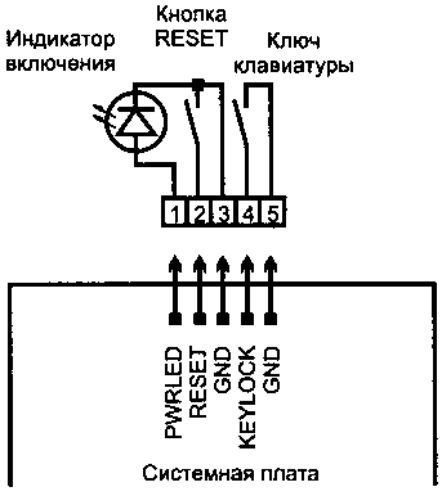


Рис. 6.17. Старый вариант подключения кнопки Reset, ключа и индикатора

Следует заметить, что компьютер может нормально работать с неподключенными органами лицевой панели (кроме кнопки питания в ATX).

Исходя из вышеизложенного, можно предложить следующую методику «слепого» подключения:

1. Подключите к системной плате разъемы питания, динамика, клавиатуру и графический адаптер с монитором. Эти подключения не вызывают вопросов, хотя разъем динамика по виду можно спутать с разъемом внешней батарейки. Но разъем батарейки обычно находится недалеко от нее самой.
2. Включив компьютер, по экрану монитора и щелчкам динамика при тесте памяти убедитесь в запуске теста POST.
3. Если имеется пятиштырьковый разъем, подключите к нему соответствующие органы управления и индикации. Если такого нет, определите разъем для кнопки Reset и подключите ее. Определить разъем легко, закорачивая

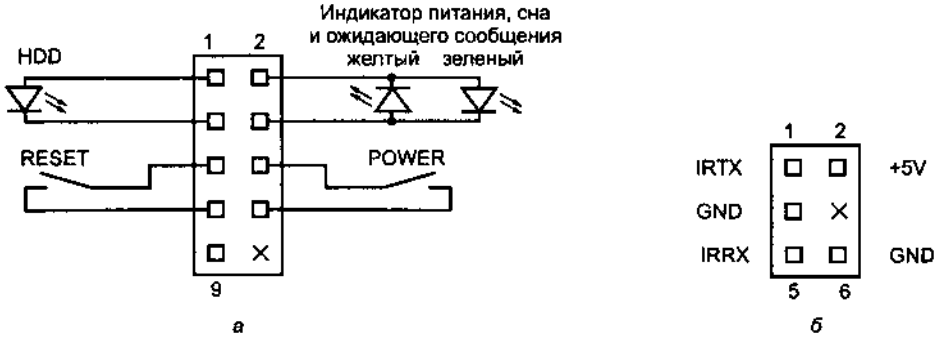


Рис. 6.18. Разъемы кнопок, индикаторов и инфракрасного приемопередатчика

контакты двухштырьковых разъемов, — по сигналу Reset снова начнется тест POST, что будет видно на мониторе и слышно через динамик.

4. Определите разъем индикатора включения питания — подключенный к нему индикатор должен всегда светиться при включенном питании.
5. Разъем блокировки клавиатуры определяется по такому признаку: при закорачивании его контактов в конце выполнения теста POST появляется сообщение об ошибке клавиатуры.
6. Индикатор обращения к жесткому диску может подключаться и к дополнительной плате контроллера IDE или SCSI, если не используется контроллер, расположенный на системной плате. Правильно подключить разъем можно во время загрузки ОС с жесткого диска — при этом индикатор должен мигать.

С начала нового века компания Intel определила стандартные штырьковые разъемы (шаг 2,54 мм) для подключения компонентов лицевой панели. Здесь ключами являются пропущенные штырьки (на плате) и отсутствующие отвер-

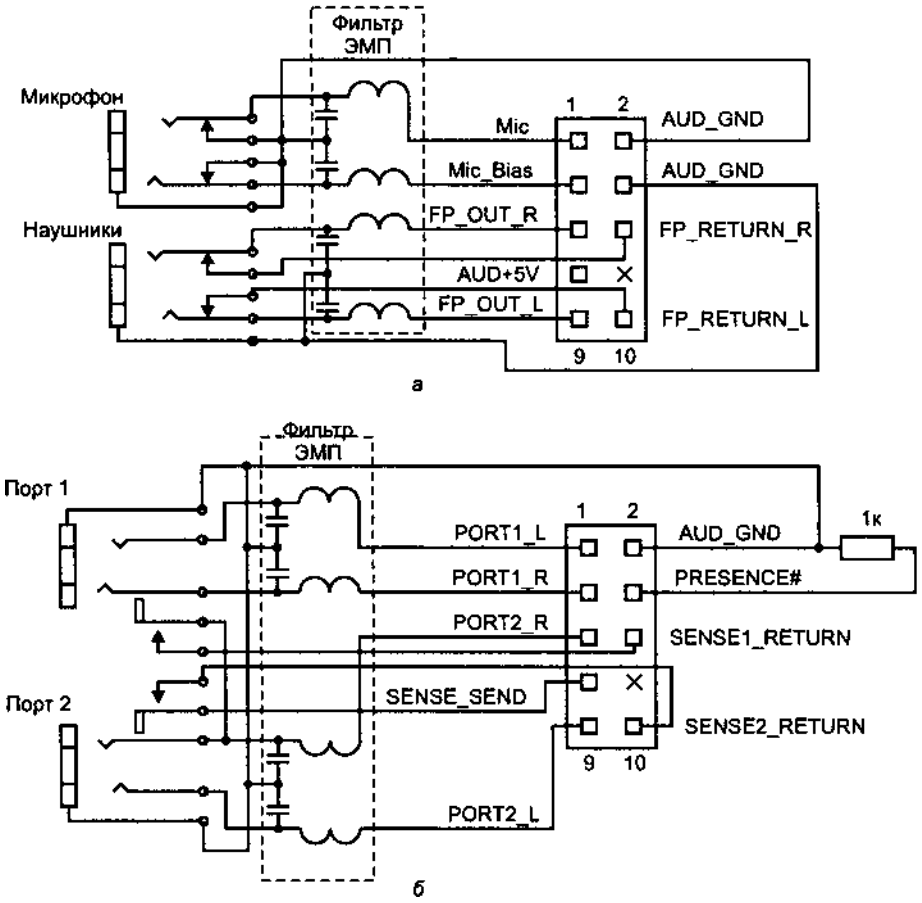


Рис. 6.19. Разъемы аудиоинтерфейсов: а — для AC'97, б — для HDA

стия (на ответной части, установленной на кабеле). Разъем для подключения кнопок сброса (Reset) и включения/выключения питания (Power), а также индикатора активности винчестера (HDD LED) и режима потребления (и ожидающего сообщения) показан на рис. 6.18, *а.* Разъем подключения инфракрасного приемопередатчика изображен на рис. 6.18, *б.*

Для подключения микрофона и наушников со стороны лицевой панели предназначены разъемы, изображенные на рис. 6.19. Для аудиокодека AC'97 применяется разъем, показанный на рис. 6.19, *а*. Если аудиоразъемы на лицевой панели не используются, то должны быть установлены джамперы (перемычки) между контактами 1-2, 3-4, 5-6 и 9-10. Без перемычек линейный выход на задней панели может оказаться неподключенным, а на микрофонный вход будут наводиться помехи. Для аудиокодека HDA схема подключения несколько сложнее (рис. 6.19, *б):* здесь задействованы контакты, с помощью которых кодек определяет факт подключения-отключения разъемов (см. 12.2). Если аудиоразъемы не подключены, то никаких джамперов устанавливать не надо — факт подключения определяется с помощью цепи Presence#, соединяемой с «землей» через резистор 1 кОм на блоке разъемов.

Для шины USB используются разъемы, приведенные на рис. 6.20 *а и б;* для шины FireWire (IEEE 1394) — на рис. 6.20, в. Эти разъемы могут иметь бандажи с ключевым вырезом, который не позволяет перепутать кабели подключения USB и FireWire. Разъем для портов IEEE 1394a голубого цвета, для IEEE 1394b — красного. Разъемы, подключаемые к портам FireWire, должны соответствовать типу порта.

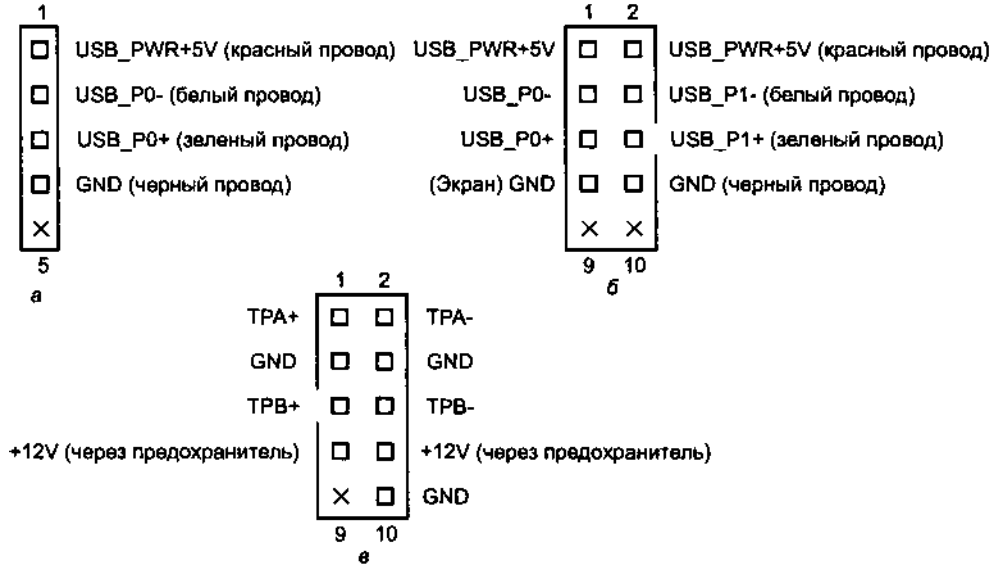


Рис. 6.20. Разъемы последовательных шин: а, б — USB, *в* — FireWire

Периферийные интерфейсы внешних устройств (клавиатуры, мыши, последовательных, параллельных и игровых портов, а для интегрированных плат еще и аудиовидеоинтерфейсы) на платах ATX выводятся прямо на разъемы, выходящие в окно задней стенки компьютера. Остальные внешние разъемы, монтируемые на самой задней стенке или на скобках-заглушках, подключаются к соответствующим штырьковым разъемам системной платы кабелями-шлейфами. Здесь существенно соблюдение правильности подключения — первый провод шлейфа маркируется цветной краской, а первый контакт разъема подписывается и имеет квадратную контактную площадку на печатной плате (ее видно с нижней стороны платы). Самыми распространенными ошибками подключения шлейфов являются переворот разъема на 180° и боковое или продольное смещение контактов. От этих ошибок спасает пластмассовый ободок, окружающий штырьки некоторых разъемов, и его ключевая прорезь. Однако с ключами случаются и конфузы: бывает, что разъем шлейфа, имеющий ответный ключевой выступ, на шлейф наколот неправильно. Тогда для правильного соединения этот выступ приходится срезать (это проще, чем качественно «переколоть» разъем). Шлейфы, как и провода лицевой панели, следует подвязывать во избежание их соприкосновения с вентилятором процессора.

Разъемы интерфейсов внешних устройств на плате легко распознаются по числу штырьков (размеру): 10 — COM-порты, 16 — GAME-порт, 26 — LPT- порт, 4 (5) — PS/2 Mouse. Комплект «выкидышей» — внешних разъемов с кабелями-шлейфами — должен поставляться в комплекте с системной платой. Проблемы могут возникнуть с разными вариантами распайки 10-штырьковых внутренних разъемов COM-портов (см. 16.1).

Интерфейс клавиатуры — разъем DIN или MiniDIN (см. 11.1) — в основном используют по прямому назначению (для подключения клавиатуры), но также он годится для получения питания +5 В при подключении некоторых внешних устройств. Питание +5 В выводится на разъем через плавкий предохранитель, припаянный на плате вблизи разъема. При подключении внешнего устройства (не клавиатуры) из-за перегрузки предохранитель может перегореть, и клавиатурный интерфейс окажется неработоспособным. Обнаруживается эта неисправность легко — проверкой напряжения +5 В на выходе разъема, «лечится» — перепайкой предохранителя (но не «жука»).

Интерфейс мыши PS/2 — аналогичный интерфейс - может быть запрещен настройкой CMOS Setup. В стандарте BTX эти разъемы уже отсутствуют.

Интерфейс IrDA — 5 штырьков, расположенных в один ряд (см. 16.2), — подключают к приемопередатчику, расположенному на лицевой панели, или к шлейфу переходного разъема (обычно mini-DIN), установленного на задней панели. Для работы в режиме IrDA обычно требуется выполнить соответствующую настройку CMOS Setup (как правило, для порта COM2).

Периферийные интерфейсы внутренних устройств выводятся на двухрядные штырьковые разъемы, к которым подключаются соответствующие кабели- шлейфы.

Накопители на гибком диске подключаются шлейфом с 34-контактным разъемом. Ошибочное подключение (поворот на 180°) заметно по постоянному свечению индикатора на дисководе.
Устройства IDE (ATA) подключаются шлейфом с 40-контактным разъемом, но для режима UltraDMA/66 и более быстрых требуется 80-проводной шлейф. При ошибочном подключении (поворот на 180°) системная плата после включения не подает признаков жизни и не реагирует на кнопку Reset. Если перепутаны каналы IDE, поведение компьютера зависит от «сообразительности» BIOS и параметров Setup. Новые версии BIOS позволяют для всех четырех возможных устройств IDE задать режим идентификации Auto и загружаться с первого обнаруженного устройства. Правильность подключения шлейфов и накопителей можно проверить установкой параметра IDE Autodetect в CMOS Setup — все подключенные накопители на жестких дисках должны распознаваться. Подробнее о разрешении проблем с IDE см. главу 19.
Устройства SATA подключаются к соответствующим разъемам, ошибочное подсоединение невозможно благодаря надежному ключу.
Шина SCSI, если ее контроллер присутствует на плате, может выводиться на различные разъемы, позволяя как подключать внутренние устройства, так и устанавливать внешние разъемы. Здесь имеются нюансы, связанные с вариантами топологии и терминаторами, подробнее они описаны в 20.4.

1.5. «Оживление» системной платы

Что делать, если компьютер не хочет работать? Во-первых, определить свою позицию с учетом гарантийных обязательств (и перспектив их исполнения поставщиком). Если вы решились залезть внутрь компьютера, не имея достаточного опыта работы с «железом», советую принять к сведению следующие рекомендации, относящиеся к самому началу общего процесса поиска и устранения неисправностей. Советы по более детальному тестированию отдельных подсистем приводятся в посвященных им разделах.
Модульная конструкция PC-совместимых компьютеров существенно облегчает процесс поиска и устранения неисправностей. Подозрительные узлы снимают и временно заменяют функционально аналогичными. Неисправные модули (системные платы, платы адаптеров, блоки питания, клавиатуры, мыши, мониторы и т. п.) в принципе ремонтопригодны, но заниматься их восстановлением в единичных количествах, не имея специального оборудования и комплектующих, экономически вряд ли целесообразно. Для этого существует масса мелких и крупных специализированных фирм, в которых концентрируются знания, опыт, оборудование и запасные части (каналы их получения). Так что удел большинства пользователей и технического персонала — выявление отказавшего модуля и его замена исправным, что тоже неплохо, поскольку приводит к конечному результату — восстановлению работоспособности PC. Вооружившись лишь дискетой с DOS и несложной диагностической программой типа PCCheck, Checklt или QAPlus, тестером и крестообразной отверткой, можно устранить большинство неисправностей, встречающихся на практике.

ИМАНИЕ----------

Установка и изъятие всех компонентов системной платы, подключение и отключение разъемов питания должны производиться только ПРИ ОТКЛЮЧЕННОМ ПИТАНИИ. Для плат ATX требуется выключение питания механическим выключателем (а не кнопкой на лицевой панели) или отсоединение шнура питания.

Если при включении компьютер не подает признаков жизни, первым делом проверьте напряжение питания на разъеме, идущем от блока питания к системной плате (расположение контактов и цветовую маркировку см. в главе 3). Проверьте и сигнал PowerGood на том же разъеме — он должен иметь уровень логической единицы ТТЛ (выше 2,4 В): при его низком уровне системная плата пребывает в состоянии сброса. Если питания нет, проверьте его наличие на разъеме блока питания, отключив питание от системной платы и накопителей. Если питание пропадает при подключенной системной плате, ищите короткое замыкание. Распространенные причины — неправильно установленные крепежные винты (иногда под них необходимо подкладывать изолирующие шайбы) или проводящий мусор в слоте (это легче всего проверить и устранить продувкой перевернутой системной платы). Возможно, для поиска замыкания придется извлечь из слотов все платы расширения и отключить все накопители.
Если питание в норме, то порядок диагностики может быть следующим:

1. Извлеките все периферийные адаптеры, видеоадаптер и память. Отключите от системной платы шлейф интерфейса IDE. Оставьте подключенными только динамик и кнопку Reset. В таком виде при включении «компьютера» должно раздаваться три гудка — процессор не обнаруживает память. Если этого не происходит, значит, процессор не может начать выполнение теста POST. Причин может быть множество:
   - процессор неверно установлен в сокет;
   - неверно заданы тип процессора, тактовая частота и коэффициент умножения — проверьте установку джамперов;
   - отсутствуют какие-либо компоненты (CPU, ROM BIOS, кварцевый резонатор);
   - стерта информация в ROM BIOS — если есть переключатель, включающий режим восстановления (Boot Recovery), попробуйте им воспользоваться;
   - вышла из строя системная плата — причиной может быть несоблюдение правил подключения внешних устройств, установка или изъятие плат из слотов при включенном питании (что редко сходит с рук).
2. Если три гудка по включении (и аппаратном сбросе) исполняются, можно устанавливать память (для начала — минимальный комплект). На появление исправной памяти компьютер должен отозваться одним длинным и двумя короткими гудками — признаком отсутствия графического адаптера (если, конечно, он не интегрирован в системную плату). Общепринятое назначе

ние звуковых сигналов диагностики, когда компьютер еще не способен вывести сообщение на экран, приведено в 5.2.

3. После того как системная плата «признала» память, пора установить графический адаптер и подключить к нему монитор. При исправности адаптера (и его совместимости с данной системной платой, с чем тоже бывают проблемы) на мониторе сначала появится заставка BIOS графического адаптера, а после нее — основная заставка BIOS с тестом памяти и предложением входа в Setup.
4. Иногда причиной неработоспособности системной платы становится некорректное задание параметров CMOS Setup. В этом случае помогает обнуление Setup, для чего на некоторых системных платах имеется соответствующий переключатель или джампер. Однако некорректно заданные параметры, хранящиеся в NVRAM, стереть, возможно, удастся лишь утилитой перепрограммирования флэш-памяти, а вот удастся ли ее загрузить — зависит от глубины повреждения конфигурации, предусмотрительности разработчика BIOS и системной платы, а также изворотливости специалиста по ремонту.
5. Когда тест POST успешно проходит инициализацию графического адаптера, задача диагностики упрощается, поскольку теперь на экране могут появиться и диагностические сообщения. Они чаще всего касаются клавиатуры и дисководов. Расшифровка часто встречающихся сообщений приведена в 5.2. Ошибка клавиатуры может возникать по нескольким причинам:
   - действительно неисправная клавиатура;
   - «залипла» (или случайно прижата) одна или несколько клавиш;
   - не подключен кабель клавиатуры (или плохой контакт в разъеме);
   - клавиатура заблокирована (повернут ключ замка или случайно закорочены контакты разъема KeyLock);
   - перегорел предохранитель питания клавиатуры.

   Сообщение об ошибке жесткого диска или его контроллера в случае применения дисков IDE, скорее всего, указывает на неправильное подключение интерфейсных или питающих кабелей накопителей, отсутствие (или отключение) контроллера или неверную установку джамперов Master/Slave на накопителях (подробнее см. в 19.2).
6. При сообщении об ошибке гибкого диска (FDD Failure или Seek Error) обратите внимание на индикатор накопителя — его постоянное свечение указывает на ошибку подключения интерфейсного кабеля. В конце теста POST индикатор должен мигнуть — если этого не происходит, проверьте подключение питания и наличие (разрешение работы) контроллера. Подробнее о проблемах с дисководами см. в 9.7.
7. После успешного выполнения теста POST система BIOS делает попытку загрузки ОС с гибкого или жесткого диска. Если компьютер не загружается с жесткого диска (или CD-ROM, LS120), попытайтесь загрузить его с системной дискеты, убедившись в правильности конфигурирования дисководов в CMOS Setup (параметр Swap Floppy). Загрузка с дискеты может быть

запрещена через CMOS Setup настройкой порядка загрузки — при установке параметра Boot Sequence A: C: первая попытка загрузки делается именно с дискеты, а при установке параметра Boot Sequence C: A: сначала делается попытка загрузки с жесткого диска. Кроме того, загрузка с дискеты может быть запрещена параметрами безопасности (Security) или антивирусной защиты.

8. Если компьютер не загружается и с дискеты, проверьте интерфейсный кабель и питание дисковода, сам дисковод, а также попробуйте заменить адаптер FDC (что по нынешним временам сделать нелегко, поскольку он чаще всего расположен на системной плате). Если загрузка с дискеты начинается (судя по позиционированию головок дисковода), но «зависает» без диагностических сообщений, велика вероятность неисправности памяти, которую программа POST тестирует весьма условно.
9. Если компьютер не загружается только с жесткого диска, проверьте интерфейсный кабель, питание и контроллер винчестера, а также его параметры в CMOS Setup. Для современных дисков чаще всего используют тип Auto (или тип 47 с параметрами, автоматически определенными с помощью IDE Autodetect). Однако старые винчестеры, для которых ранее использовали тип 1-46, в случае автоматического определения параметров могут быть сконфигурированы на другую логическую геометрию (см. главу 19). При этом ОС может перестать загружаться, хотя при загрузке с дискеты, возможно, удастся прочитать каталог жесткого диска. В таком случае следует перепробовать варианты задания типа 1-46, имеющие значения емкости диска, близкие к указанному на накопителе. В современных дисках с автоматическим определением типа может быть связан неправильный выбор режима работы с большими дисками — LBA или Large. Поскольку здесь только два варианта, подобрать подходящий легче.
10. Когда наконец удалось загрузить ОС, полезно запустить какую-либо диагностическую программу и протестировать системную плату и память. Для тестирования памяти утилитами типа Checklt лучше всего загружать DOS (можно и с дискеты), причем не используя драйверы HIMEM.SYS и EMM386.EXE или им подобные. Можно их исключить из файла CONFIG.SYS (или переименовать сам файл) или шунтировать исполнение конфигурационных файлов, нажав клавишу F5 или F8 при появлении сообщения Starting MS-DOS (для версий 6.x). Достаточно придирчивый тест расширенной памяти по умолчанию выполняется при загрузке драйвера HIMEM.SYS — он, например, может обнаружить ошибки стыковки кэш-памяти и процессора, которые не выявляются тестовыми утилитами.
11. Теперь, когда компьютер «ожил», можно последовательно подключать ранее удаленные компоненты (адаптеры) и, наблюдая за поведением компьютера при загрузке ОС и тестировании, выявить неисправный узел. На самом деле жизнь, конечно, сложнее и многообразнее, но цель данной книги не в выдаче рецептов на все случаи жизни, а в описании принципов работы и взаимодействия подсистем PC.

1.6. Конфигурирование компьютера — CMOS Setup

Компьютеры класса AT могут иметь различный и изменяемый состав аппаратных средств, и многие их элементы требуют программного конфигурирования. Первые модели AT-286, а также ряд компьютеров PS/2 и других «экзотических» моделей использовали внешнюю утилиту конфигурирования, загружаемую с диска. Параметры конфигурирования, установленные с помощью утилиты Setup, запоминаются в энергонезависимой памяти. Часть из них всегда хранится в традиционной памяти CMOS Memory, объединенной и с часами-календарем RTC (Real Time Clock). Другая часть волей разработчика может помещаться и в энергонезависимую (например, флэш) память (NVRAM). Помимо этой части статически определяемых параметров имеется область энергонезависимой памяти ESCD для поддержки динамического конфигурирования системы (PnP), которая может автоматически обновляться при каждой перезагрузке компьютера. Этот процесс динамического конфигурирования и является причиной «задумчивости» при перезагрузке даже мощных компьютеров, имеющих средства PnP, а также не всегда предсказуемого поведения программного обеспечения, вызванного изменением распределения ресурсов по инициативе той же системы PnP. Компьютеры с шиной EISA для конфигурирования EISA-адаптеров используют специальную внешнюю утилиту *ECU* (EISA Configuration Utility).

Утилита Setup встроена в ROM BIOS всех современных компьютеров. Утилита *CMOS Setup* имеет интерфейс в виде меню, иногда даже оконный с поддержкой мыши. Оконный интерфейс в данном случае раздражает, поскольку вместо быстрого входа в текстовое меню компьютер долго ищет подключенную мышь, после чего выводит окна в режиме графики низкого разрешения (дань совместимости). При этом никаких принципиально новых возможностей (по сравнению с текстовым режимом и управлением от клавиатуры) не предоставляется.

Меню утилиты Setup, способы перемещения по пунктам и выбора параметров зависят от наклонностей производителя и версии BIOS, но они понятны из краткого пояснения на экране. Нажатие клавиш F1 или Alt+H вызывает краткую контекстную справку, обычно связанную с навигацией. Смысловых пояснений значения параметров она не дает. Состав управляемых параметров, детальность и гибкость управления варьируются от предельно подробных, в которых может запутаться и опытный пользователь, до предельно кратких. Что лучше — дело вкуса. Ниже приведем краткий обзор распространенных вариантов настройки. В конкретной версии они представлены, конечно же, лишь выборочно. Некоторые параметры могут называться и не совсем так, как указано в описании, но быть созвучными (в английском варианте). За более чем двадцатилетний период развития PC некоторые термины получили новое значение — если раньше под типом микросхем памяти (DRAM Type) подразумевали объем микросхем (64K, 25K, 1M), то теперь это действительно тип (FPM, EDO, BEDO, SDRAM, DDR(2) SDRAM, RDRAM). В связи с этим возможно двоякое толкование не

которых параметров, но нельзя объять необъятное и перечислить все существующие на сей день параметры.

Группа параметров, задающих «тонкие» варианты настройки (режимы и временные диаграммы), требует знания функционирования подсистем компьютера. Общие принципы настройки таковы: чем выше частоты, меньше коэффициенты деления и количество тактов ожидания (Wait States), тем выше производительность затрагиваемой подсистемы и компьютера в целом, если подсистема используется интенсивно. Пределы ускорения определяются быстродействием и количеством применяемых компонентов и могут быть выявлены эмпирически. Однако возможны побочные эффекты, когда «разгон» одной подсистемы приводит к неработоспособности другой, на первый взгляд, с нею не связанной. Многие группы параметров имеют общую настройку *автоконфигурирования* (automatic configuration). Разрешение автоконфигурирования — типичное для таких параметров, как коэффициенты делителей частоты, количество тактов ожидания и т. п., — позволяет установить если и не оптимальную, то в большинстве случаев вполне нормально работающую конфигурацию. При запрете автоконфигурирования эти параметры придется задавать вручную (пользователь получает дополнительную возможность ошибиться).

Вход, выход и сохранение параметров Setup

Для *входа в Setup* во время выполнения теста POST появляется предложение нажать клавишу Del. Иногда вместо Del предлагаются клавиши Ctrl+Alt+Esc, Esc или Ctrl+Esc. Бывают и экзотические варианты (нажать клавишу F12 в те секунды, когда в правом верхнем углу экрана виден прямоугольник). Некоторые версии BIOS позволяют войти в Setup по комбинации Ctrl+Alt+Esc в любой момент работы компьютера. Предложение (и способ — нажатие клавиши F1 или F2) входа в Setup появляется, если тест POST обнаружит ошибку оборудования, которая может быть устранена средствами Setup. Удержание клавиши Ins во время теста POST в ряде версий BIOS позволяет задать настройки по умолчанию, отменяя все «ускорители». Это помогает восстановить работоспособность после излишне агрессивных попыток «разогнать» компьютер.

После входа в Setup пользователю в первую очередь предоставляется доступ к общим настройкам (Main Menu, Standard CMOS Setup). Тонкие настройки служат для конфигурирования чипсета системной платы, они могут скрываться за названиями Advanced Settings, Advanced CMOS Setup, Soft Menu, Chipset Configurations и им подобными.

Заданные параметры сохраняются при выходе из Setup (пункт Write to CMOS and Exit, Save and Exit) и начинают действовать с момента начала следующего теста POST. Таким образом, если нет уверенности в правильности заданных параметров, можно выйти из Setup без сохранения новых значений (пункт Do Not Write to CMOS and Exit, Exit without Saving). Для установки нескольких типовых вариантов конфигураций в Setup обычно предусматривается несколько команд:

- Auto Configuration with BIOS Defaults — установка нормальных параметров, исходная точка для оптимизации настройки (на которой можно и остановиться).

- Optimal (значок зайца) — установка оптимальных параметров, обеспечивающих штатную производительность всех компонентов.
- Auto Configuration with Power-on Defaults, Fail-Safe (значок черепахи) — установка консервативных параметров, используется для «отката» после попыток задания более эффективных конфигураций. Если системная плата не работает и с такими параметрами, необходимо проверить ее аппаратное конфигурирование — установку джамперов, съемных элементов (процессор, память, кэш и т. п.). Если в Setup не войти, те же значения параметров можно получить, удерживая клавишу Del (иногда Ins) во время включения компьютера, или для этих целей имеется специальный переключатель (джампер) на системной плате. Способ спасения зависит от версии BIOS и модели системной платы.

Выбранные значения параметров рекомендуется написать на бумаге. К сожалению, функция печати экрана по нажатии клавиши Printscreen из Setup работает не всегда (до инициализации во время загрузки LPT-порт может удерживать низкий уровень на выходе Init#, что не позволяет принтеру печатать). Но затраты времени, даже если записывать придется вручную, помогут впоследствии сэкономить время, силы и нервы в критической ситуации, поскольку некоторые значения параметров могут приводить к полной потере работоспособности компьютера (к счастью, временной — до исправления этих значений). Неудачные параметры конфигурации (или забытый пароль) при невозможности входа в Setup можно сбросить отключением питания CMOS (замыканием контактов 3, 4 разъема аккумулятора). В некоторых системах пароль сбрасывается только специальной перемычкой. Однако в CMOS хранятся отнюдь не все параметры — часть их содержится в памяти NVRAM, которую отключением батарейки изменить (очистить) невозможно по определению (это действительно энергонезависимая память). Хорошо, если на системной плате есть специальный переключатель для обнуления NVRAM (или хотя бы ESCD). Такой переключатель срабатывает, если в положении «очистка» на системную плату будет подано питание (вместе с сигналом аппаратного сброса). После обнуления плату включают с нормальным положением переключателя. Изредка в Setup встречается параметр, предназначенный для обнуления NVRAM. Если явных способов очистки нет, то при необходимости остается воспользоваться утилитой перепрограммирования флэш-BIOS (NVRAM обычно является областью микросхемы-носителя флэш-BIOS). Однако для этого необходимо иметь файл-образ BIOS и утилиту программирования. Записью некорректной информации в NVRAM иногда грешит Windows 95 при установке. Это может привести к потере работоспособности отдельных узлов и даже платы в целом, не устранимой средствами Setup.

Общее конфигурирование

Общее конфигурирование доступно через пункт меню, который может называться Main Menu, Standard CMOS Setup или подобным образом. В этом меню обычно сообщается размер установленного ОЗУ и задаются следующие параметры:

- Дата и время (System Time, System Date) — установка часов в CMOS RTC.
- Разрешение переключения на летнее/зимнее время в последнее воскресенье октября и апреля (Daylight Saving).
- Параметры дисководов (FDD, Legacy Diskette), определяющие режим работы контроллера НГМД в соответствии с типом привода.
- Типы устройств ATA (Master и Slave для обоих каналов). Возможные варианты: Auto — автоопределение (подходит в большинстве случаев), None — отказ от использования даже подключенного устройства, User Type HDD — винчестер с параметрами, заданными вручную (для старых или дефектных винчестеров), CD-ROM, LS-120, ZIP-100, MO (магнитооптический диск), Other ATAPI Device (устройство ATAPI другого типа).
- Параметры клавиатуры (Keyboard Features) — состояние NumLock после инициализации, задержка (Typematic Delay) и скорость (Typematic Rate) автоповтора.
- Остановка POST при обнаружении ошибок (Halt On): по любым ошибкам, никогда не останавливаться, не останавливаться при ошибке клавиатуры (позволяет загружать компьютер без клавиатуры), не останавливаться при ошибке диска.
- Выбор первичного (используемого через BIOS) графического адаптера (Primary display, Init Display First) и его режим. Позволяет выбрать графический адаптер (Onboard, AGP, PCI, ISA), применяемый при загрузке.
- Установка паролей (User Password — для рядового пользователя, Supervisor Password — для пользователя, которому разрешается вход в Setup).
- Разрешение вывода приглашения к входу в Setup (Hit Del Message Display), запрет позволяет утаить способ входа от неискушенного пользователя.

Встроенная антивирусная защита может блокировать попытки записи в главный загрузочный сектор диска (и выдавать соответствующее предупреждение). Возможен и контроль за изменением размера доступной памяти.

Управление процессором

Установка тактовых частот шины (FSB Freq.) и ядра процессора (CPU Internal Freq., Core Freq.) может быть автоматической или ручной. В последнем случае могут предоставляться возможности задания частот FSB и шины памяти (независимо или с выбором возможных пар). Коэффициент умножения частоты (Frequency Multiple, CPU Ratio, Multiplier Factor) для большинства процессоров только наблюдаем, но не управляем.

Напряжение питания ядра процессора (CPU Core Voltage, Vcore) также может определяться автоматически или задаваться вручную. Как правило, повышать напряжение ядра приходится при разгоне процессора (относительно его штатной частоты).

Управление параметрами кэширования включает разрешение/запрет кэша L1, L2 и L3 (если есть) и разрешение/запрет ECC-контроля для кэша L2 (и L3). Политика кэширования (WB или WT) в современных системах не задается

(политика WB давно освоена). Параметр In-Order Que Depth задает глубину очереди упорядоченных запросов ядра процессора к памяти.
Для процессоров, поддерживающих выдачу серийного номера (Pentium III), предусмотрена команда запрета такой выдачи.
Для процессоров с поддержкой многопоточности имеется команда, запрещающая эту возможность (CPU Hyper Threading). Аналогичная команда предназначена и для мультиядерных процессоров.
Приоритет процессора (CPU Priority) определяет очередность предоставления ему доступа к системной шине при наличии запросов от графического адаптера и других мастеров шин.

Управление памятью

Настройка временных диаграмм работы для современных типов памяти выражается в выборе тактовой частоты шины памяти и значения латентности. Эта настройка может выполняться автоматически на основыве информации, получаемой из EEPROM на модулях памяти (By SPD). Ручная настройка должна соответствовать спецификациям быстродействия (тактовая частота и время доступа) установленных модулей памяти. Для плат с двухканальной памятью возможны одноканальный и двухканальный режимы; BIOS обычно определяет режим автоматически, в зависимости от физически установленных модулей (они должны быть попарно идентичны).
Команды управления регенерацией памяти позволяют оптимизировать распределение времени между «полезными» обращениями к ОЗУ и накладными расходами на регенерацию. К этим командам относятся приведенные ниже, а также их параметры (например, число запросов регенерации), которые могут находиться в очереди (Refresh Queue Depth):

- Hidden Refresh — скрытая регенерация, позволяет несколько повысить производительность памяти. В большинстве случаев противопоказаний не имеет.
- Slow Refresh — снижение частоты регенерации, повышает производительность, снижает потребляемую памятью мощность, но действительно стабильно работает лишь при применении всех модулей (микросхем) памяти с расширенным периодом регенерации. Активизировать эту возможность надо с осторожностью.
- Concurrent Refresh — конкурирующая регенерация (одновременное обращение к памяти со стороны процессора и контроллера регенерации). Повышает производительность.
- Burst Refresh — пакетная регенерация, при которой запросы регенерации собираются в пачки (размером вплоть до полного количества строк). Повышает производительность, но приводит к регулярным довольно длительным захватам шины памяти, что не всегда допустимо.
- DRAM Burst at 4 Refresh — пакетная регенерация по четыре строки в пакете. Повышает производительность, но не занимает шину слишком долго, как предыдущий вариант.

- Staggered Refresh — «шахматная» регенерация: при наличии нескольких банков они регенерируются поочередно. Снижает пиковые броски тока потребления памяти, что полезно с точки зрения снижения помех.
- Decoupled Refresh Option — раздельная регенерация памяти шины ISA и основной памяти. Повышает производительность, поскольку операции на шине ISA выполняются медленнее.

Настройка времени удержания открытых страниц DRAM после обращения (Precharge Closing Policy) позволяет оптимизировать политику управления страницами памяти.

Для выбора способа управления вентилем линии A20, разрешающим доступ процессора ко всему пространству памяти, служит команда Gate A20. Стандартный способ (через контроллер клавиатуры 8042) работает медленно, ускоренный (Fast) иногда вызывает проблемы совместимости ПО и системной платы, поскольку его реализация зависит от чипсета.

Для чипсетов и памяти с контролем достоверности можно отключить контроль или выбрать тип контроля (DRAM Data Integrity Mode, Data Integrity). Контроль четности (Parity Check) позволяет только обнаруживать некоторые ошибки, для ECC-контроля можно выбирать лишь между обнаружением и исправлением однократных ошибок. Выбор исправления несколько снижает производительность памяти.

В пространстве физических адресов может быть выделено окно (Memory Hole) под границей 16 Мбайт, которое не используется для ОЗУ. Окно требуется для некоторых контроллеров, устанавливаемых в слоты ISA.

Часть областей адресов верхней памяти (UMA) может быть «затенена» оперативной памятью (см. 4.2). Для управления теневой памятью служат команды с названиями Shadow Memory, в которых указываются диапазоны затеняемых адресов или/и назначение затеняемых областей (System ROM Shadow, Video ROM Shadow, Adaptor ROM Shadow). Возможно и разрешение кэширования теневой памяти (Shadow Memory Cacheable) — второй виток ускорения доступа к содержимому ROM. Для современных ОС, работающих в обход BIOS, теневая память не дает заметных преимуществ в работе.

В системных платах с интегрированной графикой часть основной памяти может выделяться для нужд графического адаптера. Эта память становится разделяемой (shared memory), соответствующими параметрами задают ее размер (то есть объем «видеопамяти»). На этот размер, естественно, уменьшается объем ОЗУ относительно объема установленной памяти. Для ускорения доступа к этой памяти предназначена опция разрешения USWC (Uncached Speculative Write Combining), объявляющая эту область некэшируемой, допускающей спекулятивное чтение и комбинирование записей. Короткие записи процессора в память с атрибутом USWC собираются (комбинируются) в специальном буфере, физическая запись в память происходит пачками (пакетами).

Конфигурирование шин ISA, PCI и порта AGP

Первичным параметром для конфигурирования этих интерфейсов является, тактовая частота, которая может задаваться независимо или через отношение

с другими частотами. Кроме того, может варьироваться число тактов ожидания (WS, Wait States) для различных операций (чем меньше значение, тем выше скорость, но возможна нестабильность работы).

Для *шины ISA* номинальная тактовая частота составляет 8,33 МГц; она может быть несколько повышена, но при этом может потребоваться введение дополнительных тактов ожидания. Длительность циклов обращений определяется параметрами AT Cycle Wait State, 8/16-bit Memory, I/O Wait State, I/O Recovery Time, Fast AT Cycle Enable/Disable.

Для *шины PCI* номинальной является частота 33 МГц, при разрешении параметров PCI 2.1 возможна частота 66 МГц (если ее допускает чипсет). Для PCI-X тактовая частота может достигать 133 МГц, при этом может быть разрешена быстрая запись в память с частотой 266 (2x) и 533 МГц (4x). Устройства и мосты PCI и PCI-X автоматически настраивают частоту и протокольные расширения под слабейшее устройство, однако в Setup могут присутствовать опции для принудительного ограничения возможностей. Для ускорения записи в память устройств PCI может иметься опция разрешения отправленных записей (Post Write), завершающихся для процессора до выполнения физической записи. Для объединения одиночных записей от процессора в пакетную транзакцию PCI возможно использование буферов записи, их работу разрешают опциями PCI Burst Mode, PCI Bursting.

На шине PCI может присутствовать несколько устройств-мастеров, их совместная работа регулируется механизмом арбитража, которым управляет ряд параметров. Возможно управление приоритетами обращения к шине (PCI Bus Arbitration, Arbitration Priority) — фиксированное предпочтение процессору, устройствам PCI или ротация приоритетов. Ротация приоритетов возможна и среди мастеров шины PCI (Master Priority Rotation). В режиме конкурентных обращений у активного мастера арбитр может отобрать право управления шиной до окончания транзакции; разрешение этого режима (PCI Concurrency, Peer Concurrency) может влиять на производительность системы (в обе стороны). Число тактов, в течение которых мастер имеет право не отдавать шину при лишении права управления, регулирует параметр PCI Latency Timer (управляет устройствами PCI). Параметр PCI Preemption Timer (управляет арбитром) задает время, в течение которого арбитр не будет отбирать право управления у мастера при поступлении запроса от его конкурента. Параметр PCI Initial Latency Timer к арбитражу прямого отношения не имеет, он определяет максимально допустимое время от начала транзакции до начала передачи данных (разрешение излишней «задумчивости» снижает эффективность использования шины).

Параметр PCI VGA Palette Snooping разрешает широковещательную запись в регистр палитр VGA, необходимую, когда видеооверлейная плата и графический адаптер установлены на разных шинах.

Для шин ISA и PCI требуется скоммутировать *линии запросов прерываний.* При наличии шины ISA обычно для каждой линии входа контроллера прерываний (IRQx) можно выбрать вариант ее использования. Параметр PCI/PnP определяет ее в пул ресурсов, распределяемых системой PnP для шины PCI (устройств

системной платы и устанавливаемых в слоты PCI), параметр ISA/EISA отдает ее шинам ISA и EISA.

Для шины PCI определяют номера линий IRQx, соответствующих линиям INTA#, INTB#, INTC# и INTD# шины PCI (команда может называться 1st, 2nd, 3rd, 4th Available IRQ).

Для *порта AGP* номинальная тактовая частота — 66 МГц, а пропускная способность определяется поддерживаемым режимом (AGP 1x/2x/4x/8x Mode), который может быть принудительно ограничен. Для скоростных режимов (4x, 8x) возможно ручное управление мощностью приемопередатчиков (AGP Drive Strength), причем раздельное для их p- и n-транзисторов. Эти параметры влияют на надежность передачи данных и потребляемую мощность (порт и карта AGP способны согласовать их автоматически). Быстрая программная запись в память графической карты разрешается параметром AGP Fast Write; возможность предвыборки при чтении этой памяти — параметром AGP Prefetch. Параметр AGP to DRAM Prefetch разрешает предвыборку при чтении акселератором системной памяти. Для снижения радиопомех может использоваться расширение спектра (AGP Spread Spectrum) — модуляция тактовой частоты. Параметр AGP Aperture задает размер области ОЗУ, к которой акселератор обращается через таблицу переопределения адресов (GART).

Встроенная периферия

Периферия, встроенная в системную плату, конфигурируется через меню Peripheral Setup или Integrated Peripheral. Каждый компонент, как правило, может быть разрешен (включен) или запрещен, для разрешенных компонентов возможна настройка занимаемых ресурсов, а также их параметров.

Для контроллера НГМД (FDD) может разрешаться (запрещаться) невостребованный режим (FDC Mode 3) работы с дискетами 2,88 Мбайт. Параметр Floppy Drive Swap включает взаимную замену имен дисков A и B (физический диск B, подключенный к «прямому» разъему, получает логическое имя A и с него становится возможной загрузка, физический диск A, становится логическим диском B). Параметры дисководов, заданные в Standard Setup, обычно относятся к физическим именам.

Для контроллера шины ATA (PCI IDE Bus Master) задаются разрешенные режимы передачи (PIO Mode, IDE DMA Transfer Mode, Ultra DMA Mode). Из-за несовместимости со старыми устройствами эти режимы иногда приходится занижать. Параметр IDE Multiple Sector Mode разрешает многосекторные команды (очень старые устройства их не поддерживали). Параметр IDE 32-bit Transfer разрешает 32-битные обращения к 16-битному регистру данных IDE, что ускоряет обмен в режиме PIO, но может быть источником проблем при некорректных драйверах.

Для контроллеров SATA указывается вариант использования портов. Для эмуляции ATA/IDE каждому порту (SATA Port i) назначается номер канала (Primary, Secondary) и устройства (Master, Slave), что обеспечивает совместимость с ОС, не

поддерживающими интерфейс SATA. Указание номера порта (Port 0, Port 1...) означает естественный для SATA режим, реализующий преимущества SATA.

Для последовательных портов (Onboard Serial Port) базовые адреса определяют номер COM-порта. Второй порт, как правило, может быть сконфигурирован на использование для инфракрасной связи (IrDA) с указанием выбранного режима. Возможно применение порта и для интерфейса MIDI (может присутствовать и выделенный интерфейс MIDI).

Для параллельного порта (Onboard Parallel Port) задается список разрешенных режимов работы: традиционный (Standard, Normal, Std, SPP, Compatible), Fast Centronics, Bidirectional, EPP, ECP, ECP+EPP, 1284 Compliance. Для режимов ECP и Fast Centronics может быть разрешен доступ DMA и выбран номер канала.

Для контроллеров USB параметр USB Legacy Support разрешает эмуляцию доступа к клавиатуре и мыши USB через доступ к регистрам контроллера клавиатуры. Для стандартных обращений к этим устройствам (через сервисы BIOS) эмуляция не требуется. При наличии контроллера USB 2.0 он включается отдельно от обычного (1.0); его использование может быть невозможным из-за проблем с драйверами ОС.

В традиционном контроллере клавиатуры и мыши PS/2 может быть отключена поддержка мыши, при этом прерывание IRQ12 освобождается (без этого прерывания мышь PS/2 работать не будет).

Управление загрузкой

В Setup имеются параметры, управляющие режимом выполнения теста POST, а также загрузкой ОС. Для ускорения процедуры POST (обхода некоторых проверок) можно выбрать команду Quick Boot, Quick Power On Self Test. Раньше тест памяти сопровождался щелчками, по мере ускорения переходящими в писк динамика; этот звук отключается командой Memory Test Tick Sound. Ускорению начала загрузки способствует и запрет проверки позиционирования головок НГМД (Floppy Drive Seek at Boot).

Остановку POST и ожидание нажатия клавиши F1 (в том числе и при неподключенной клавиатуре) можно отменить командой Wait for F1 If Any Error. Здесь может быть задан список ошибок или, наоборот, исключения ошибок, требующих нажатия клавиши. Запрет останова по ошибке клавиатуры требуется для серверов, работающих с отсоединенной клавиатурой.

Для загрузки компьютера можно потребовать введения пароля — в Password Checking Option, Security Option выбрать команду Always (Всегда). Параметр Setup требует пароля только при входе в Setup, при выборе параметра None пароль не проверяется. Если пароль не задан пользователем явно, то для AMI BIOS пароль по умолчанию — «AM1», для AWARD BIOS — «BIOSTAR» или «AWARD_ SW».

Последовательность опроса устройств при загрузке задается параметром Sys- temBoot Sequence или параметрами First Boot Device, Second Boot Device, Third Boot Device (1-, 2-, 3-е устройства загрузки). Традиционно BIOS при готовности дисковода A начинает загрузку с дискеты, а при неготовности гибкого диска —

с жесткого диска (последовательность A, C). Изменение последовательности является одним из средств защиты от несанкционированного доступа к компьютеру и предохраняет от случайной загрузки с дискеты, оставленной в приводе (именно забытые дискеты почему-то чаще несут на себе вирус в загрузочном секторе). Кроме того, загрузка с измененной последовательностью проходит быстрее, особенно если отключить параметр Floppy Drive Seek at Boot. Современные версии BIOS позволяют составлять списки загрузочных устройств, включая в них дискеты (A, Legacy Floppy), жесткие диски ATA и SATA (указывая номер канала и устройства или имена C, D, E, F), CD-ROM и иные устройства (ATAPI, LS-120, ZIP-100, USB Floppy, USB HDD, USB Flash, сетевые карты с BootROM, у которых расширение BIOS способно перехватить вектор прерывания INT 18h). Готовность устройств проверяется по порядку списка, попытка загрузки выполняется с первого готового устройства. Если загрузка с него не удается (из-за некорректности загрузчика или ошибки устройства), процесс загрузки останавливается с соответствующим сообщением.

Управление энергосбережением и питанием

Группа параметров меню Power Management управляет системой снижения энергопотребления. В плане энергосбережения определены следующие режимы работы компьютера:

- *Full On Mode* — режим полной мощности;
- *Doze Mode* — снижение активности на 80 % (умеренное понижение частоты процессора);
- *Standby Mode* — снижение активности на 92 % (понижение частоты процессора до минимума);
- *Suspend Mode* — снижение активности на 99 % (процессор остановлен и прерывания не отрабатывает, из этого состояния компьютер выходит довольно долго, за единицы секунд).

Поведение монитора и жесткого диска в различных режимах может задаваться относительно произвольно. Режимы снижения активности (и потребления) включаются через заданный интервал неактивности пользователя (клавиатура, мышь) или соответствующей подсистемы (отсутствие обращений к жесткому диску). В нормальный режим компьютер переходит по определенным заданным событиям. Некорректная настройка и ошибки в BIOS могут приводить к неожиданному резкому снижению производительности. Простейшим выходом из такой ситуации является запрет режимов снижения потребления, однако для компьютеров с автономным питанием энергосбережение весьма существенно. Переход в режим пониженного потребления позволяет также уменьшать шум от работы вентиляторов (при автоматическом управлении скоростью вращения).

В современных системных платах имеется развитая система управления питанием на основе интерфейса ACPI; в более старых компьютерах использовалась система APM (Advanced Power Management). Разрешение автоматического

управления энергопотреблением определяется параметром PM Control By ACPI или PM Control By APM.

Для управления питанием в конструктивах ATX (и BTX) используются кнопка-выключатель на передней панели, а также кнопки (клавиши) клавиатуры и даже мыши. Для выключенного компьютера любое нажатие кнопки Power на лицевой панели вызывает *включение питания* (если оно подано и не отключено механическим выключателем на задней стенке). Можно запрограммировать включение питания (Power Up Control) по двойному щелчку мыши (Power Up By Mouse), от клавиатуры (Power Up By Keyboard), по сигналу от модема (Power Up By Modem), по пробуждению от локальной сети (Wake On LAN), шины USB (Power Up By USB), а также автоматически по будильнику (Alarm, Automatic Power Up). С клавиатуры питание может включаться либо нажатием специальной клавиши (KB98), либо набором кодового слова (Password, пароль из 1-5 символов). Для будильника можно установить день (число, дни недели или все дни) и время включения.

Если питание компьютера внезапно пропадает, то можно выбрать варианты поведения по его появлению (Power back, State After Power Failure, Power Lost Resume State): выключать компьютер (Always Off), включать (Always On) или переводить в состояние, предшествующее исчезновению питания (Auto).

Выключение ПК может быть запрограммировано (Off by Power Button, Power Button < 4 Sec, Soft Power Off) либо по любому нажатию кнопки (Instant Off), либо только по длительному нажатию (Delay 4 Sec Off). В последнем случае короткое нажатие кнопки переведет компьютер в энергосберегающее состояние *Suspend*, тип которого определяется параметром ACPI Suspend Type. Нормально выключение выполняется по команде ОС в конце завершения работы (shutdown), но эту возможность можно запретить (Soft Power Off).

Возможные варианты энергосберегающего состояния (Suspend Type):

- Disabled — не используется (короткое нажатие кнопки игнорируется);
- S1 (POS) — состояние Power On Suspend, в котором все компоненты по возможности снижают потребление (и производительность), но хранят свои контексты (текущее состояние) сами;
- S3 (STR) — состояние Suspend To RAM, в котором все контексты (в том числе и процессора) сохраняются в ОЗУ, переводимое в режим автономной регенерации. При этом потребление минимально (процессор остановлен). У портативных компьютеров имеется еще состояние STD (Suspend To Disk) — энергонезависимое сохранение контекста, при котором пропадание питания не страшно (в состоянии STR потеря питания приводит к потере контекста).

Выход (Resume) из состояния *Suspend* возможен по событиям, определенным в Setup (Power Down & Resume Events, Wake Up Events). К этим событиям относятся прерывание PME (от устройств PCI, модема, адаптера локальной сети); можно назначить возобновление и по будильнику (Resume by Alarm). Будильник программируется: можно указывать день (или все дни) и время подачи сигнала возобновления.

Управление потреблением (Power Management), если оно разрешено, может быть сконфигурировано на фиксированные установки (максимального или минимального сбережения) либо подробно конфигурироваться пользователем (User Defined Mode). В последнем случае пользователь определяет критерии переходов (время неактивности Doze Timer, Standby Timer, Suspend Timer) и характеристики состояний.
Для процессора может определяться степень снижения тактовой частоты для определенных состояний.
Для дисплея выбираются состояния системы (обычно Suspend), в которых он выключается (Video Off Option), и метод его выключения (Video Off Method) — сигналами DPMS (см. 10.4) или пустой экран (DPMS Standby, DPMS Suspend, DPMS Off, Blank Screen). Для перевода в энергосберегающее состояние только дисплея может быть задан список событий (Monitor Event In Full On Mode), которые считаются признаками активности (LPT port Activity, COM port Activity, ISA/ PCI Master Activity, IDE Activity, Floppy Activity, VGA Activity, Keyboard Activity, Mouse Activity). Любое из отслеживаемых событий сбрасывает таймер переключения режима (переключение происходит, если соответствующий таймер успевает досчитать до заданного значения).
Для жестких дисков автономный переход в энергосберегающее состояние определяется по таймеру, отсчитывающему время паузы обращений. Режим, в который переводится HDD по таймеру, определяется параметрами Hard Disk Power Down Mode: Disabled, Standby, Suspend.

Мониторинг состояния

Современные системные платы оборудованы средствами мониторинга состояния питания и охлаждения, которые доступны в Setup (PC Health Status, Hardware Monitor). Эти средства позволяют наблюдать измеренные значения питающих напряжений (выходов блока питания и напряжение питания процессора), температуры и скорости вращения вентиляторов. В рабочем режиме мониторинг указанных параметров может выполняться с помощью специальных утилит.
В компьютере может измеряться температура процессора (CPU Temperature), системной платы (MB Temperature), а также любого компонента, на котором установлен выносной термодатчик. Для процессора могут быть установлены пороги температуры, при которых включается режим снижения потребления (CPU Critical Temperature), выполняется аварийный останов (CPU Shutdown Temperature) или просто выдается предупреждение (CPU Warning Temperature). Для режима пониженного потребления указывается степень понижения скорости (CPU Slow Clock Ratio).
Измерение скорости возможно для вентиляторов процессора (CPU Fan Speed), корпуса (Chassis Fan Speed) и блока питания (Power Fan Speed), если они снабжены датчиками вращения. Возможно включение предупреждения при остановке вентилятора. Возможно также управление включением и скоростью вращения вентиляторов (Fan Control) в зависимости от состояния потребления (например,

CPU Fan OFF in Suspend) или температуры. Это позволяет минимизировать шум от компьютера.
В инструментах мониторинга может присутствовать средство контроля за вскрытием системного блока: факт вскрытия (по сигналам от контактных датчиков) регистрируется в энергонезависимой памяти. Монитор позволяет определить, было ли вскрытие после последнего сброса признака.

1.7. Выбор системной платы PC

Рассмотрев устройство системной платы PC, кратко перечислим основные характеристики, на которые следует обращать внимание при ее выборе:

- Конструктивное исполнение определяет предполагаемый тип корпуса или, наоборот, определяется его типом (Mini Tower, Midi Tower, Big Tower, Desktop, Slim Line) традиционного стандарта или ATX (NLX).
- Чипсет — набор микросхем, определяющий архитектуру и производительность платы.
- BIOS — производитель и версия, определяющие функциональные возможности, поддержку PnP, ACPI (или APM).
- Возможность перезаписи, блокировки и восстановления Flash BIOS.
- Поддерживаемые типы и количество процессоров, определяемые типом сокетов и слотов, возможностями конфигурирования чипсета и BIOS.
- Возможности выбора питающего напряжения для процессора и максимальный ток его питания, определяемые типом применяемого регулятора (VRM), возможность раздельного питания ядра и кэша L3 (для некоторых моделей Xeon).
- Поддерживаемые частоты синхронизации процессора и шин.
- Максимальный объем ОЗУ; количество и тип применяемых модулей памяти — SDRAM, DDR(2) SDRAM, RDRAM; количество каналов, возможность использования модулей с различными спецификациями быстродействия; применимость памяти с исправлением ошибок (ECC Memory).
- Тип интерфейса для графического адаптера: порт AGP (и его возможности) или PCI-E (8x или 16x и число слотов).
- Количество и ассортимент слотов шин ввода-вывода (ISA, PCI(-X), PCI-E).
- Количество каналов IDE-интерфейса и поддерживаемые режимы Ultra DMA 66/100/133).
- Количество портов SATA и тип контроллера (желателен AHCI).
- Наличие и параметры контроллера SCSI.
- Наличие COM-портов с FIFO-буферами, возможность подключения инфракрасного приемопередатчика, использования COM-порта в качестве порта MIDI.
- Наличие LPT-порта, поддерживаемые режимы (SPP, EPP, ECP), наличие FIFO-буферов для ECP.

- Наличие разъема PS/2-Mouse.
- Наличие и число каналов USB, наличие внешних разъемов.
- Наличие и параметры графического адаптера.
- Наличие и параметры звукового контроллера (цифрового аудиоканала и синтезаторов).

Понятно, что для каждого класса применений существенны свои параметры: для файл-сервера, например, не нужен аудиоканал, но память с ECC — вовсе не лишняя. Для домашнего компьютера может быть справедливо обратное утверждение. Ассортимент существующих системных плат широк, так же как и диапазон их цен. Идеальной платы для всех случаев жизни, конечно, не существует. Помимо технических и экономических характеристик приходится учитывать и сервис, предлагаемый поставщиком. При покупке платы сразу обычно можно выявить только грубые неисправности. Более тонкие проблемы возникают уже при детальном тестировании или только при эксплуатации в рабочих условиях на реальных приложениях. Хороший поставщик с пониманием отнесется к этим проблемам, и вам не придется отчаянно доказывать свою возможную правоту в претензиях по качеству.

Платы известных производителей привлекательны поддержкой в Интернете — на веб-сервере фирмы можно найти новые версии BIOS и драйверов для различных ОС, что помогает разрешать многие вопросы. С поддержкой «безымянных» плат могут возникнуть проблемы, тем более что на них иногда устанавливают систему BIOS, подходящую весьма условно.

ГЛАВА 7

Процессоры

Процессор является основным «мозговым» узлом, в функции которого входит исполнение находящегося в памяти программного кода. В настоящее время под словом «процессор» подразумевают микропроцессор — микросхему, которая, помимо собственно процессора, может содержать и другие узлы — например, кэш-память. В IBM-совместимых ПК применяются процессоры *семейства x86*. В оригинальной машине IBM PC использовался процессор 8088 с 16-разрядны- ми регистрами. Все следующие модели процессоров, в том числе 32-разрядные (386, 486, Pentium, Pentium Pro, Pentium II/III, Celeron, Pentium 4 от Intel, K5, K6 и K7 (Athlon и Duron) от AMD, MI, MII и MIII от Cyrix/VIA), с расширениями MMX, SSE, SSE2, SSE3 и 3DNow! включают в себя системы команд и архитектуры предыдущих моделей, обеспечивая совместимость с ранее написанным ПО. Новые процессоры с 64-битными расширениями также поддерживают все команды и режимы своих предшественников.

В этой главе описываются общие свойства семейства x86 и наиболее распространенные процессоры 6-8-го поколений. Ввиду ограниченности объема книги здесь не рассматриваются детали внутреннего устройства, а также не приводятся описания команд. За этой информацией можно обратиться к [3], где описаны архитектура, система команд и функционирование 32-битных процессоров в реальном и защищенном режимах с объяснением многих системных функций, которые не рассматриваются в большинстве распространенных литературных источников. Все более «древние» процессоры рассмотрены в [2]. Информация, представленная в этих изданиях и в данной главе, собрана из фирменных каталогов и информационных материалов, публикуемых производителями процессоров на своих веб-серверах.

ПРИМЕЧАНИЕ----------

Далее в тексте знак + после условного названия процессора означает, что описание верно как для указанного процессора, так и для следующих за ним процессоров. При этом 286 означает процессор 80286, 386 — 80386, ..., P5 означает Pentium, P6 — ряд процессоров 6-го поколения, включая Pentium Pro, Pentium II/III и Celeron. Например, P5+ означает применимость к процессорам Pentium, Pentium MMX, Pentium Pro, Pentium II, Celeron, Pentium III и Pentium 4.

7.1. Исполнение программного кода

Программный код — это последовательность команд, или *инструкций*, каждая из которых определенным образом закодирована и расположена в целом числе смежных байтов памяти. Каждая инструкция обязательно имеет *операционную часть*, несущую процессору информацию о требуемых действиях. *Операндная часть,* указывающая процессору, где находится его «предмет труда» — операнды, — может присутствовать в явном или неявном виде и даже отсутствовать. Операндная часть может описывать от нуля до двух операндов, участвующих в выполнении данной инструкции (есть инструкции, в которых, помимо двух операндов, задается еще и параметр инструкции). Здесь могут быть сами значения операндов (непосредственные операнды); явные или неявные указания на регистры процессора, в которых находятся операнды; адрес (или его составная часть) ячейки памяти или порта ввода-вывода; регистры процессора, участвующие в формировании адреса, и разные комбинации этих компонентов. Длина инструкции 32-битного процессора семейства x86 может быть от 1 до 12 байтов (а с префиксами — и до 17 байтов) и определяется типом инструкции. Исторически сложившийся формат инструкций x86 довольно сложен, и «понять», сколько байтов занимает конкретная инструкция, процессор может, лишь декодировав ее первые 1-3 байта. Инструкции могут предшествовать префиксы (к счастью, всегда однобайтные, но их может быть несколько), указывающие на изменение способа адресации, размера операнда или/и необходимость многократного (по счетчику и условию) повторения для данной инструкции. Адрес (логический) текущей исполняемой инструкции хранится в специальном регистре — *указателе инструкций* (Instruction Pointer, IP), который соответствует счетчику команд фон-неймановской машины. После исполнения так называемой *линейной инструкции* этот указатель увеличивает свое значение на ее длину, то есть указывает на начало следующей инструкции. Линейная инструкция не нарушает порядок выполнения инструкций, определяемый последовательностью их расположения в памяти (по нарастанию адреса). Помимо линейных инструкций существуют *инструкции передачи управления*, среди которых различают инструкции *переходов* и *вызовов* процедур. Эти инструкции в явном или неявном виде содержат информацию об адресе следующей выполняемой инструкции, который может указывать на относительно произвольную ячейку памяти. Инструкции переходов и вызовов могут быть *безусловными* (ни от чего не зависящими) и *условными*. Произойдет ли условный переход (вызов) или нет, зависит от состояния флагов (признаков) на момент исполнения данной инструкции. Если переход (вызов) не состоится, то исполняется инструкция, расположенная в памяти вслед за текущей. *Вызов* процедуры характерен тем, что перед ним процессор сохраняет в стеке (стек — это область ОЗУ) адрес следующей инструкции, и на этот адрес передается управление после завершения исполнения процедуры (этот адрес извлекается из стека при выполнении инструкции возврата). *Переход* выполняется безвозвратно.

Последовательность исполнения инструкций, предписанная программным кодом, может быть нарушена под воздействием внутренних или внешних (относительно процессора) причин. К внутренним причинам относятся *исключения*

(exceptions) — особые ситуации, возникающие при выполнении инструкций. Наглядным примером исключения является попытка деления на ноль. При возникновении условия исключения процессор автоматически выполняет вызов процедуры обработки исключения, после которой он может вернуться к повторному исполнению инструкции, породившей исключение, или следующей за ней. Вариант поведения зависит от типа произошедшего исключения. Исключения широко используются современными операционными системами. На основе обработки исключений строится система виртуальной памяти и реализуются многие функции многозадачных операционных систем. Внешними причинами изменения нормальной последовательности инструкций являются *аппаратные прерывания* — вызовы процедур под воздействием электрических сигналов на специальные выводы процессора или по получении сообщения по специальному интерфейсу контроллера прерываний (см. 4.4). Эти сигналы могут подаваться совершенно неожиданно для исполняемой программы; правда, у программиста есть возможность заставить процессор (компьютер) игнорировать все прерывания или их часть. Злоупотреблять этой возможностью нельзя (да и не всегда она есть), поскольку на аппаратных прерываниях строится, например, отсчет времени и другие системные и прикладные функции компьютера. Источниками аппаратных прерываний являются контроллеры и адаптеры периферийных устройств, генераторы меток времени, системы управления питанием и другие подсистемы. Есть еще так называемые *программные прерывания*, но они отнюдь не нарушают последовательность инструкций, предписанную программистом. Поэтому прерываниями они по сути не являются — это всего лишь особый способ вызова системных сервисов BIOS (см. 5.2) и операционной системы. И наконец, последовательность инструкций может изменяться по сигналу аппаратного сброса или инициализации процессора. С этого, собственно, и начинается функционирование компьютера: процессор переводится в исходное состояние и «запускается». При этом указатель инструкций совместно с другими регистрами, участвующими в формировании адреса инструкции, генерирует адрес, на 15 байт меньший максимального физического адреса. По этому адресу должна располагаться инструкция, с которой начинается инициализация компьютера.

В компьютере обязательно должен присутствовать центральный процессор (Central Processing Unit, CPU), который исполняет основную программу. В многопроцессорной системе функции центрального процессора распределяются между несколькими обычно идентичными процессорами для повышения общей производительности системы, а один из них назначается главным. В помощь центральному процессору в компьютер часто вводят *сопроцессоры*, ориентированные на эффективное исполнение каких-либо специфических функций. Широко распространены *математические сопроцессоры*[1], обрабатывающие числовые данные в формате с плавающей точкой; *графические сопроцессоры*, выполняющие геометрические построения и обрабатывающие графические изображения; *сопроцессоры ввода-вывода,* разгружающие центральный процессор от несложных, но многочисленных операций взаимодействия с устройствами. Бы

[1] Теперь эти действия выполняет блок FPU, «вмонтированный» во все современные процессоры.

вают и другие сопроцессоры, однако все они несамостоятельны — исполнение основного вычислительного процесса осуществляется центральным процессором, который в соответствии с программой выдает «задания» сопроцессорам на исполнение их «партий». Процессор фон-неймановской машины, фактически, может выполнять только один процесс, передавая управление от инструкции к инструкции согласно исполняемой программе. При этом могут исполняться переходы, ветвления и вызовы процедур, но вся эта цепочка запрограммирована разработчиком программы. Для реакции на события, асинхронные по отношению к исполняемому в данный момент процессу, используют аппаратные прерывания. Особенности обработки прерываний в PC рассмотрены в 4.4, а их процессорная сторона — в 7.2.

Переключение задач и виртуальные машины

Прерывания используют и для *переключения задач* в многозадачных системах. Пусть, например, имеются два *потока инструкций* (например, две прикладные программы), которые должны выполняться как бы одновременно (по-настоящему одновременно один фон-неймановский процессор их выполнить не может). Можно запустить один поток, а через некоторое время его работы при аппаратном прерывании (от таймера) сохранить в памяти образ его текущего состояния (все регистры, программно-доступные этому процессу) и запустить другой поток. Через некоторое время при следующем прерывании выполнить обратное переключение: сохранить состояние второго потока (в другом месте памяти), загрузить в регистры процессора образ состояния первого потока и продолжить его выполнение. Эти *переключения задач* следует выполнять в течение исполнения обоих программ с частотой, создающей у пользователя иллюзию непрерывности и одновременности. Понятно, что ресурсы процессора (производительность) в этом случае делятся между задачами пропорционально выделяемым им квантам времени. Чтобы пользователя такая производительность процессов удовлетворяла (а еще учтем накладные расходы на сохранение и восстановление образов при переключениях), у процессора должна быть достаточная мощность. Процессоры семейства x86, начиная со второго и особенно с третьего (386) поколения, имеют встроенные средства поддержки многозадачности (число задач почти не ограничено), работающие в защищенном режиме. Переключение задач производится по сигналу прерывания от таймера совершенно «прозрачно» для процессов, работающих псевдопараллельно. Благодаря этой прозрачности программисту, разрабатывающему прикладную программу, в большинстве случаев не надо заботиться об обеспечении многозадачности. В распоряжение его программы предоставляется *виртуальная машина* (тоже Фон-неймановская), в которой управление передается последовательно этой программой, как будто она — единственная. Конечно, поддержка виртуальных машин требует определенных усилий со стороны многозадачной операционной системы, которой приходится распределять не только процессорное время, но и память, устройства хранения, ввода-вывода и коммуникационные устройства — то есть все ресурсы реального компьютера. В этом ей помогают

специальные средства, введенные в процессоры x86 2-3-го поколений и постоянно развиваемые в следующих поколениях.

Защищенный режим и виртуальная память

Для того чтобы потоки (задачи) не мешали друг другу (по недосмотру или умышленно), требуются меры принудительной защиты критических ресурсов. Современные операционные системы используют *защищенный режим* процессора, в котором эти меры реализуются на аппаратном уровне. Поскольку программа может взаимодействовать с подсистемами компьютера только через пространства памяти и портов ввода-вывода, а также аппаратные прерывания, то защищать нужно эти три типа ресурсов. Для того чтобы потоки могли исполняться независимо друг от друга, каждому из них (точнее, каждой виртуальной машине) предоставляется *виртуальная память*, которой он может распоряжаться по своему усмотрению.

В защищенном режиме существует разделение *привилегий* между пользовательскими процессами и операционной системой. ОС предоставляются неограниченные права управления всеми процессами, их виртуальной памятью, прерываниями и вводом-выводом; у пользовательских процессов возможности скромнее. Для этого разделения определен ряд инструкций процессора, которые можно исполнять только с определенным уровнем привилегий процесса. В процессорах x86 была заложена 4-уровневая система привилегий; реально из них используются только два крайних уровня: *supervisor* (0-й уровень) — неограниченные возможности, *user* (3-й уровень) — самые жесткие ограничения. Такая двухуровневая защита характерна и для многих других процессоров.

Для защиты и виртуализации памяти в процессорах x86 предусмотрены два основных механизма: сегментация и страничная трансляция адресов.

При *сегментации* операционная система выделяет каждому процессу *сегменты* — области памяти различного назначения, с разными правами доступа и разного размера. Из одних сегментов можно только читать данные, в другие возможна и запись. Для программного кода выделяются специальные сегменты, инструкции могут выбираться и исполняться только из них. По отношению к принципу хранимости программы это является искусственным ограничением для фон-неймановской машины, но целесообразность такого ограничения очевидна. Процессору «безразлично» содержимое ячейки памяти, на которую передалось управление, — он всегда пытается трактовать ее как код инструкции (или префикс). Если ошибочно передать управление на область данных, то дальнейшее поведение процессора непредсказуемо — это так называемый «вылет». Защита не позволяет передать управление на сегмент данных: сработает исключение защиты, которое обрабатывается операционной системой, и ошибочный процесс принудительно завершится. Таким образом, вероятность вылета снижается. Чтобы выдержать принцип хранимости программы, на время ее загрузки в память или программной модификации ту же область объявляют сегментом данных, в который разрешена запись. Система защиты может полностью контролировать распределение памяти, генерируя исключения в случаях различных нарушений.

При *страничной трансляции адресов* (paging) виртуальная логическая память (для каждой виртуальной машины) делится на страницы одинакового (фиксированного) размера. Любая страница виртуальной логической памяти (адресуемой программой в пределах выделенных ей сегментов) может отображаться на любую область физической памяти (реально установленной оперативной). Отображение поддерживается с помощью специальных таблиц страничной трансляции адресов, в которых помимо записей, описывающих связи адресов, есть указания на присутствие/отсутствие страницы в физической памяти. Благодаря этому страница памяти, не нужная процессору в данный момент времени, может быть выгружена на устройство хранения (диск). На ее место можно загрузить нужную страницу. Заявку на загрузку нужной страницы делает сам процессор без каких-либо усилий со стороны выполняемой программы: если программе потребовалась ячейка виртуальной памяти из страницы, образа которой нет в физической памяти, вырабатывается специальное исключение. Обработчик этого исключения (это часть ОС) находит свободную физическую страницу (возможно, выгрузив на диск ту, которая, по его мнению, пока не нужна), «подкачивает» на нее с диска требуемую информацию и возвращает управление процессу, прерванному исключением. Этот процесс ничего «не заметит» (кроме некоторой задержки выполнения инструкций). Таким образом, в распоряжение всех процессов, исполняемых на компьютере псевдопараллельно, предоставляется виртуальная оперативная память, размер которой ограничен суммой объема физической оперативной памяти и области дисковой памяти, выделенной для подкачки страниц.

В большинстве современных ОС механизм сегментации не используется из-за чрезмерной сложности. В них применяется *плоская* модель памяти (формально для всех целей определен один и тот же сегмент), а для организации виртуальной памяти и ее защиты (но не такой строгой) задействован механизм страничной трансляции адресов. В исходном варианте управления страницами не было функционального разделения на страницы кода и данных. Этой брешью в защите пользуются вирусы: они ухитряются запустить вредоносный код, размещенный среди данных. В новых процессорах появилась возможность запрета исполнения кода из страниц, предназначенных для данных. Фирма Intel называет эту возможность *Execute Disable Bit,* AMD — *Enhanced Virus Protection*, за обоими названиями стоит один и тот же бит в дескрипторе страницы (или каталога страниц), который можно задействовать при использовании расширения физического адреса, а также в 64-битных режимах адресации (см. далее).

Конечно же, эффективность защиты (устойчивость компьютера к ошибкам и вирусам) в значительной мере определяется предусмотрительностью разработчиков операционной системы. Вышеописанная защита от исполнения в семействе
OS Windows реализована только в SP2 для ОС Windows XP. Средства виртуализации и защиты памяти, ввода-вывода и прерываний подробно описаны в [3].

Архитектура и микроархитектура процессоров

Под *архитектурой* процессора понимается его программная модель, то есть программно-видимые свойства. В этой книге рассматриваются только процес

соры x86 с архитектурой IA-32 (Intel Architecture 32 bit) 6-8-го поколений со всеми их мультимедийными, потоковыми и 64-битными расширениями (AMD x86-64 и Intel EM64T). Архитектура IA-64 (процессоры Itanium) в персональных компьютерах не используется и здесь не рассматривается.

Основные программно-видимые свойства процессора — это набор его регистров, система команд (определяющая также работу с памятью) и механизм обработки прерываний. Процессоры x86 являются явно выраженными представителями CISC-архитектуры: по сложности системы команд им нет равных, при этом базовых архитектурных регистров довольно мало. По мере развития семейства в процессоры вводят все более мощные команды, позволяющие сокращать число инструкций, требуемое для решения одних и тех же задач. Однако эти команды все сложнее исполнять. Количество архитектурных регистров увеличивается: появились блоки MMX, XMM, а в 64-битных расширениях еще и 8 дополнительных общих регистров.

Под *микроархитектурой* понимается внутренняя реализация программной модели. Для одной и той же архитектуры IA-32 разными фирмами и в разных поколениях применяются существенно различающиеся микроархитектурные реализации: при этом, естественно, стремятся к максимальному повышению производительности (скорости исполнения программ). Начиная с процессоров P6 (и AMD K5), в микроархитектуре применяется RISC-ядро, исполняющее микрооперации (uOps), на которые раскладываются сложные инструкции x86. В результате производительность процессора (по скорости выполнения инструкций x86) зависит от способа разложения и скорости исполнения микроинструкций. При этом повышать производительность можно различными способами: ускорять выполнение микроопераций (за счет повышения тактовой частоты), по возможности распараллеливать выполнение микроопераций, сокращать число микроопераций, требуемых для исполнения одной инструкции x86. У лидеров «процессоростроения» — Intel и AMD — подходы к оптимизации различаются: при сопоставимой производительности процессоры AMD работают на более низких тактовых частотах. Однако заметим, что повышение производительности процессоров x86 обходится слишком дорого (по сравнению с «чистыми» RISC-архитектурами) — требует очень сложных управляющих устройств, на которые и уходит значительная часть транзисторов процессора и которые, к тому же, выделяют значительную мощность. Компьютеры на «чистых» RISC-процессорах (например, Power MAC) обеспечивают ту же прикладную производительность, что и IBM PC на Pentium 4, но при этом тактовая частота RISC-процессоров в несколько раз ниже частоты CISC-процессо- ров Pentium 4.

Поясним основные понятия, относящиеся к конвейеризации и распараллеливанию выполнения инструкций (точнее, микроопераций).

Конвейеризация (pipelining) предполагает, что каждая инструкция обрабатывается за несколько этапов, причем каждый этап выполняется на своей ступени конвейера процессора. При выполнении инструкция продвигается по конвейеру по мере освобождения последующих ступеней. Таким образом, на конвейере одновременно может обрабатываться несколько последовательных инструкций.

В современных процессорах параллельно могут работать несколько конвейеров, так что производительность процессора можно оценивать темпом выхода выполненных инструкций со всех его конвейеров. Для достижения максимальной производительности процессора — обеспечения полной загрузки конвейеров — программа должна составляться с учетом микроархитектурных особенностей процессора. Конечно, и код, сгенерированный обычным способом, будет исполняться на более новых процессорах достаточно быстро (за счет более высокой тактовой частоты). Однако ряд программ на процессорах Pentium III работает быстрее, чем на Pentium 4, при одинаковых тактовых частотах: сверхдлинный конвейер Pentium 4 на «поворотах» непредсказуемо ветвящихся программ «заносит», что приводит к его простоям. Конвейер «классического» процессора Pentium имеет пять ступеней. Конвейеры процессоров P6 с *суперконвейерной архитектурой* (superpipelined) имеют большее число ступеней (10-12), что позволяет упростить каждую из них и, следовательно, сократить время пребывания в них инструкций. *Гиперконвейер* Pentium 4 имеет уже 20 ступеней для повторно выполняемых участков программного кода (из кэша трасс), а если считать его полную длину (начиная с декодирования), то наберется около 30 ступеней.

Скалярным называют процессор с единственным конвейером, к этому типу относятся все процессоры Intel до класса 486 включительно. *Суперскалярный* (superscalar) процессор имеет более одного конвейера (Pentium — два); эти конвейеры способны обрабатывать инструкции параллельно.

Инструкции переходов и особенно ветвлений нарушают непрерывность работы начальных ступеней конвейера, поскольку они должны начинать выборку и декодирование инструкций с нового, заранее неизвестного адреса. *Предсказание переходов* (branch prediction) позволяет продолжать выборку и декодирование потока инструкций после выборки инструкции ветвления (условного перехода), не дожидаясь проверки самого условия. В процессорах прежних поколений инструкция перехода приостанавливала конвейер (выборку инструкций) до исполнения собственно перехода, на чем, естественно, терялась производительность. Предсказание переходов направляет поток выборки и декодирования по одной из ветвей, при этом используется ряд методов предсказания:

- При *статическом предсказании* (схема заложена в процессор) переходы по одним условиям, вероятнее всего, произойдут, а по другим — нет. Переходы назад скорее произойдут (это типичный цикл), вперед — нет (типично для обработки ошибок).
- *Динамическое предсказание* опирается на предысторию вычислительного процесса — для каждой конкретной команды перехода (ее адреса в памяти) накапливается статистика поведения, на основе которой предсказывается переход. Для динамического предсказания в процессор вводят таблицу *BTB* (Branch Table Buffer — буфер таблицы переходов), напоминающую кэш с ассоциативным поиском.
- *Программные «намеки»* (hints) — новые префиксы инструкций (появились в P4), перекрывающие статическое предсказание. Намеки закладываются в программный код на этапе компиляции.

Конвейеризация в процессорах x86 осложняется архитектурными особенностями: большим разбросом инструкций по сложности и, соответственно, по времени выполнения. Для повышения производительности те ступени конвейера, на которых производится наиболее сложная работа, стараются распараллеливать. При этом в процессор вводится большое количество исполнительных элементов, которые могут одновременно обрабатывать разные инструкции (или разные части одной сложной инструкции, разбитой на несколько более простых).

Для того чтобы процессор мог параллельно обрабатывать несколько инструкций, программный код должен быть написан (скомпилирован) так, чтобы в «поле зрения» процессора (то есть на его конвейере) оказывалось побольше фактически независимых друг от друга инструкций. В процессорах RISC-архи- тектур, как правило, много универсальных регистров, что располагает к написанию кода, удобного для параллельного исполнения. Малое число и неравноправность основных регистров x86 не располагают к программированию с учетом параллельности исполнения. Однако если несколько инструкций, обращающихся к одному и тому же регистру, не имеют фактических зависимостей по данным (не используют результаты друг друга), их можно исполнять одновременно в разных *физических* регистрах процессора (архитектурно невидимых), которых может быть много. Это делается путем *переименования регистров* (register renaming) — временного сопоставления физических регистров логическим с отслеживанием правильной последовательности смены состояний архитектурных (логических) регистров.

Продвижение данных (data forwarding) подразумевает начало исполнения инструкции до готовности всех операндов. При этом выполняются все возможные действия, и декодированная инструкция с одним операндом помещается в исполнительное устройство, где дожидается готовности второго операнда, выходящего с другого исполнительного устройства.

При *исполнении по предположению*, называемом также *спекулятивным* (speculative execution), используется результат предсказаний переходов: инструкции по предсказанной ветви перехода не только декодируются, но и по возможности исполняются до проверки условия перехода. Если предсказание сбывается, то труд оказывается не напрасным; если не сбывается, приходится выполнять откат — в этом случае конвейер оказывается недогруженным и простаивает несколько тактов (как минимум столько, сколько ступеней у конвейера).

Исполнение с изменением последовательности инструкций (out-of-order execution), свойственное RISC-архитектуре, теперь реализуется и для процессоров x86. При этом изменяется порядок внутренних манипуляций данными, а внешние (шинные) операции ввода-вывода и записи в память выполняются, конечно же, в порядке, предписанном программным кодом. Однако эта способность процессора в наибольшей степени может блокироваться несовершенством программного кода (особенно 16-битных приложений), если он генерируется без учета возможности изменения порядка исполнения инструкций.

Описанные термины и технологии вкладываются в общее понятие «динамического исполнения» (dynamic execution), введенное фирмой Intel с появлением процессоров P6. Улучшенное динамическое исполнение (enhanced dynamic exe

cution) отличается улучшениями различных сторон, в частности, улучшением предсказаний переходов.

Благодаря усложнению микроархитектуры от поколения к поколению возрастает производительность процессоров, причем этот рост обеспечивается двумя факторами. Во-первых, растет тактовая частота ядра. Во-вторых, увеличивается относительный темп выполнения инструкций. Так, в среднем процессоры 1-, 2-, 3-, 4-, 5-, 6- и 7-го поколений завершают очередную инструкцию с интервалом

12, 5, 4, 2, 1, 1/2 и 1/3 тактов. Полтакта и треть такта на инструкцию — звучит, конечно, странно. Но если вспомнить о 8-байтной шине данных, позволяющей за один такт загрузить фрагмент кода, содержащего несколько команд, и о нескольких исполнительных устройствах, одновременно приступающих к их выполнению, то вопросы рассеиваются. Конечно же, время, требующееся для полного прохождения инструкции от ее выборки до исполнения, измеряется десятками тактов (и растет с удлинением конвейера).

7.2. Программная модель современных процессоров x86

Современные представители семейства x86 являются 32-битными процессорами; в новых моделях появилось 64-битное расширение. История 32-битных процессоров Intel (архитектуры IA-32) началась с процессора 80386. Он вобрал в себя все черты своих 16-битных предшественников 8086/88 и 80286 для обеспечения совместимости с громадным объемом ПО, существовавшего на момент его появления.

Существуют понятия *разрядности адреса* и *разрядности данных. Разрядность адреса* определяет, сколько битов (16, 32 или 64) используется в регистрах, формирующих адрес данных или инструкций, расположенных в памяти. *Разрядность данных* определяет, сколько битов используется в инструкциях, оперирующих словами. Каждому режиму работы процессоров соответствуют своя разрядность, применяемая по умолчанию. При необходимости для каждой исполняемой инструкции разрядность адреса или/и операнда может изменяться с помощью специальных *префиксов* (байтов перед кодом инструкции).

32-битные регистры процессоров позволяют непосредственно адресовать до 4 Гбайт памяти, что во времена появления процессора 80386 можно было считать «почти бесконечностью». Встроенный блок управления памятью поддерживает механизмы *сегментации* и *страничной трансляции адресов.*

Расширения x86-64 и EM64T в первую очередь предназначены для радикального увеличения объема адресуемой памяти: 64-битные регистры позволяют адресовать до $2^{64} = 18{,}4 \times 10^{18}$ байт. Это число и является пределом объема виртуальной памяти 64-битного процессора, но пока используют только младшие 48 битов адреса.

Процессоры предоставляют четырехуровневую *систему привилегий* для защиты памяти, ввода-вывода и прерываний, а также механизм *переключения задач* для многозадачных ОС. *Система команд* процессоров постоянно расширяется при

сохранении всех команд предшествующих процессоров x86. С расширением системы команд расширяется и набор архитектурных регистров (MMX, XMM, новые общие 64-битные регистры).

Процессоры могут работать в различных *режимах*, определяющих возможности адресации памяти и защиты: в реальном (16-разрядном) режиме процессора 8086, в режиме виртуального процессора 8086 (V86), в защищенном 32-разряд- ном (и защищенном 16-разрядном) режиме. Режим работы процессора задается операционной системой с учетом режима работы приложений (задач). У процессоров с 64-битным расширением появляются новые режимы, среди которых есть и режимы, обеспечивающие совместимость с 32-разрядными операционными системами и приложениями. Новые режимы используются только в 64-бит- ных ОС, а полностью их преимущества доступны только 64-битным приложениям.

Режимы работы процессоров

32-битные процессоры могут работать в одном из следующих режимов:

- *Режим реальной адресации* (real address mode), или просто *реальный режим* (real mode), полностью совместим с 8086. В этом режиме возможна адресация до 1 Мбайт физической памяти (на самом деле, как и у 80286, почти на 64 Кбайт больше).
- *Защищенный режим виртуальной адресации* (protected virtual address mode), или просто *защищенный режим* (protected mode). В этом режиме у процессора включаются механизмы сегментации и страничной трансляции. Механизм сегментации позволяет поддерживать виртуальную память объемом до 64 Тбайт. На практике используется только страничная трансляция, благодаря которой каждой задаче предоставляется до 4 Гбайт виртуального адресного пространства. По умолчанию и адреса, и операнды имеют разрядность 32 бита[1]. В защищенном режиме процессор может выполнять дополнительные инструкции, недоступные в реальном режиме; ряд инструкций, связанных с передачей управления, обработкой прерываний, и некоторые другие выполняются иначе, чем в реальном режиме.
- *Режим виртуального процессора 8086* (Virtual 8086 Mode, V86) является особым состоянием задачи защищенного режима, в котором процессор функционирует как 8086 (16-битные адрес и данные). На одном процессоре в таком режиме могут параллельно исполняться несколько задач с изолированными друг от друга ресурсами. При этом использование физического адресного пространства памяти управляется механизмами сегментации и трансляции страниц. Попытки выполнения недопустимых команд, выхода за рамки отведенного пространства памяти и разрешенной области ввода-вывода контролируются системой защиты. Более эффективен *расширенный режим вир*

[1] Есть возможность организации 16-разрядного защищенного режима в стиле процессора 80286, но этот режим не представляет интереса.

туального процессора 8086 (Enhanced Virtual 8086 Mode, EV86), в котором оптимизирована виртуализация прерываний.

- «*Нереальный» режим* (unreal mode, он же *big real mode)* — это «неофициальный» режим, который поддерживают все 32-битные процессоры. Он позволяет адресоваться к 4-гигабайтному пространству памяти. В этом режиме инструкции исполняются так же, как и в реальном режиме, но с помощью дополнительных сегментных регистров FS и GS программы получают непосредственный доступ к данным во всей физической памяти.
- В *режиме системного управления* (System Management Mode, SMM) процессор выходит в иное, изолированное от остальных режимов пространство памяти. Этот режим используется в служебных и отладочных целях. С его помощью, например, скрытно выполняются функции управления энергопотреблением, эмулируются обращения к несуществующим аппаратным средствам (эмуляция клавиатуры и мыши PS/2 для USB).

Для процессоров x86-64 вышеперечисленные режимы объединены понятием *legacy mode*; кроме того, появился новый режим *long mode* с двумя подрежимами:

- *64-битный режим* (64-bit mode) — это режим полной поддержки 64-битной виртуальной адресации и 64-битных расширений регистров. В этом режиме используется только плоская модель памяти (общий сегмент для кода, данных и стека). По умолчанию разрядность адреса составляет 64 бита, а операндов (для большинства инструкций) — 32 бита, однако префиксом (REX) можно заказать 64-битные операнды. Имеется новый способ адресации данных — относительно указателя инструкций. Режим предназначен для использования 64-битными ОС при запуске 64-битных приложений — он включается операционной системой для сегмента кода конкретной задачи;
- *режим совместимости* (compatibility mode) позволяет 64-битным ОС работать с 32- и 16-битными приложениями. Для приложений процессор выглядит как обычный 32-битный со всеми атрибутами защищенного режима, сегментацией и страничной трансляцией. 64-битные свойства используются только операционной системой, что отражается в процедурах трансляции адресов, обработки исключений и прерываний. Режим включается операционной системой для сегмента кода конкретной задачи.

32-битные ОС используют процессоры x86-64 только в режиме *legacy mode* (как обычный процессор IA-32).

Архитектурные регистры и типы данных

Процессоры могут оперировать с операндами разнообразных типов и размеров:

- целыми числами (со знаком и без знака) размером в байт, слово (16 бит), двойное слово (DWord, 32 бита), учетверенное слово (QWord, 64 бит) и двойное учетверенное слово (DQWord, 128 бит);
- строками байтов, слов, двойных и учетверенных слов;
- битами, битовыми полями и строками битов;

- числами в формате с плавающей точкой (FP) размером в 32, 64 и 80 бит.

Возможность работы с длинными операндами (64 и 128 бит) появилась в процессорах с расширениями MMX и XMM, базовая архитектура IA-32 таких возможностей не предоставляла. Операнды инструкций могут находиться в регистрах процессора, памяти (или в порте ввода-вывода), а также в самой инструкции (непосредственный операнд). Наиболее эффективно процессор работает с операндами, расположенными в его регистрах. Состав регистров, с которыми работают прикладные программы, приведен на рис. 7.1.

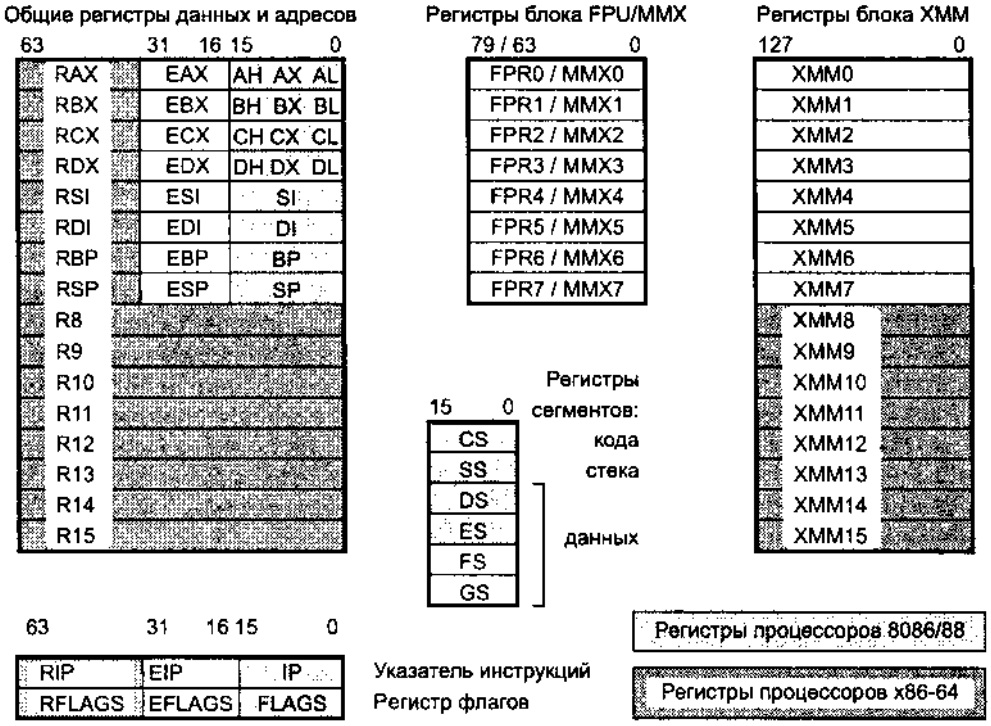


Рис. 7.1. Прикладные регистры процессоров x86

Для обращений к памяти у процессоров x86 используются 16-битные *сегмент*ные *регистры*, с помощью которых формируется логический адрес (см. 7.3):

- CS (Code Segment) — для адресации выбираемых инструкций;
- SS (Stack Segment) — для работы со стеком;
- DS, ES, FS и GS — для обращения к данным.

Названия 32-битных регистров начинаются с буквы E, названия 64-битных — с буквы R. Блоки FPU (математический сопроцессор), MMX и XMM в архитектуре процессоров IA-32 x86 держатся особняком. Они присутствуют не во всех процессорах и являются пристройками к центральному процессору с его набором обычных целочисленных регистров. Эти блоки предназначены для ускоре

ния вычислений с данными различных форматов. При отсутствии математического сопроцессора (он стал неотъемлемой частью процессора только начиная с процессоров 486DX и Pentium) прикладная программа все-таки может использовать инструкции FPU, но для этого операционная система должна поддерживать *эмуляцию сопроцессора*. Эмулятор сопроцессора — это программа- обработчик прерывания или исключения от сопроцессора, которая должна «выловить» код операции сопроцессора, определить местонахождение данных и выполнить требуемые вычисления, опираясь на целочисленную арифметику центрального процессора. Понятно, что эмуляция будет выполняться во много раз медленнее, чем те же действия, выполняемые настоящим сопроцессором. Эмуляция для блоков MMX и XMM не предусматривается — эти блоки предназначены для ускорения вычислений в приложениях реального времени, и выполнять их с крайне низкой скоростью эмуляции было бы просто бессмысленно.

Помимо регистров общего назначения, предназначенных для использования прикладными программами, процессоры имеют ряд *регистров системного назначения* (на рисунке не показаны). Эти регистры прикладными программами обычно не используются. К ним относятся системные адресные регистры, управляющие регистры, регистры отладки и тестирования. Ряд этих регистров являются *модельно-специфическими* (Model-Specific Registers, MSR), они предназначены для управления расширениями отладки, мониторингом производительности, машинным контролем, кэшированием областей физической памяти и другими функциями. Их назначение привязывается к микроархитектуре конкретного процессора, состав меняется от модели к модели, доступ привилегирован. Доступность регистров различных групп зависит от режима работы процессора и уровня привилегий задачи.

Регистры общего назначения

Все процессоры имеют целочисленное арифметико-логическое устройство (АЛУ), с которым связаны *регистры общего назначения.* Это их название для классического процессора x86 условно, поскольку регистры неравноправны и у каждого есть свое «амплуа»: одни регистры предназначены для арифметико-логических операций, другие — для вычисления адреса и т. п. В 32-битных процессорах были только 8 общих регистров, к которым можно было обращаться и как к 16-, и как к 32-битным (например, AX и EAX). Четыре регистра допускали и побайтное обращение к младшему или старшему байту (например, AL и AH). В 64-битном расширении добавили еще 8 общих регистров (R8...R15), и обращения к ним унифицировали: к любому из 16 общих регистров можно обращаться как к 64-, 32-, 16- или 8-битному регистру (всегда используются младшие биты).

Регистр флагов EFLAGS служит для хранения признаков результатов выполнения инструкций (знак, переполнение, ноль, и т. п.), в нем же расположен и флаг разрешения прерываний.

Регистр-указатель инструкции EIP соответствует «счетчику команд» фон-неймановской машины, он содержит логический адрес текущей инструкции.

В 64-битном варианте RIP может использоваться и для относительной адресации данных.

Блок FPU

Блок FPU содержит стек из восьми 80-битных регистров и несколько вспомогательных регистров. FPU присутствует во всех современных процессорах; его не было в 486SX, а в предыдущих процессорах он как *математический сопроцессор* устанавливался дополнительно. Блок FPU предназначен для расширения вычислительных возможностей центрального процессора — выполнения арифметических операций, вычисления основных математических функций (тригонометрических, экспоненты, логарифма) и т. д. В разных поколениях процессоров он назывался по-разному — FPU (Floating Point Unit — блок чисел с плавающей точкой) или NPX (Numeric Processor extension — числовое расширение процессора). Сопроцессор поддерживает семь типов данных: 16-, 32-, 64-битные целые числа; 32-, 64-, 80-битные числа с плавающей точкой (FP- форматы) и 18-значные числа в двоично-десятичном (BCD) формате. Применение сопроцессора повышает производительность вычислений в сотни раз.

Блок MMX и расширение 3DNow!

Технология MMX ориентирована на приложения мультимедиа, 2D/3D-графику и коммуникации. Основная идея MMX заключается в одновременной обработке нескольких элементов данных за одну инструкцию — так называемая технология *SIMD* (Single Instruction — Multiple Data). Расширение MMX использует новые типы целочисленных данных, упакованных в 64-битных регистрах: восемь байтов, четыре слова, два двойных слова или одно учетверенное слово. Эти типы данных обрабатываются в 64-битных *регистрах* MMX0-MMX7, представляющих собой младшие 64 бита 80-битных регистров FPU. Блок MMX присутствует практически на всех современных процессорах.

Каждая инструкция MMX выполняет действие сразу над всем комплектом операндов (8, 4, 2 или 1), размещенных в адресуемых регистрах. Еще одна особенность технологии MMX — поддержка *арифметики с насыщением* (saturating arithmetic). Ее отличие от обычной арифметики с циклическим переполнением (wraparound mode arithmetic) заключается в том, что при возникновении переполнения в результате фиксируется максимальное возможное значение для данного типа данных, а перенос игнорируется. В случае переполнения снизу в результате фиксируется минимальное возможное значение. Граничные значения определяются типом (знаковый или беззнаковый) и разрядностью переменных. Такой режим вычислений удобен для обработки сигналов: для аудиосигнала насыщение не так сильно меняет его звучание, как эффект от циклического переполнения. Те же предпочтения имеют место и при определении цветов изображения. Совпадение регистров MMX и FPU накладывает ограничения на чередование кодов FPU и MMX — забота об этом лежит на программисте приложений с MMX.

Блок MMX (и поддержка соответствующих инструкций) имеется на всех современных процессорах, начиная с 6-го поколения. В расширении SSE набор целочисленных инструкций для блока MMX расширен.

Технология 3DNow! (21 инструкция), введенная фирмой AMD в процессорах K6-2, расширяет возможности блока MMX. Эта технология работает с упакованными данными в FP-формате одинарной точности (два 32-битных числа), а также упакованными (8 байт, 4 слова, 2 двойных слова) и 64-битными целыми числами. В процессорах Athlon набор инструкций 3DNow! был расширен — появились еще 24 инструкции. Расширенный набор в данной книге обозначается как 3DNow!E. Новые инструкции предназначены для сигнальных процессоров (12 инструкций DSP), работающих с упакованными FP-числами; расширен набор инструкций MMX (12 новых целочисленных), расширено управление кэшированием (7 инструкций для ускорения передачи данных). Часть инструкций совпадают с одноименными инструкциями SSE. Набор инструкций *3DNow! Professional* (72 инструкции), введенный в процессорах Sempron, полностью совместим с SSE. В книге он обозначен как 3DNow!P.

Расширение 3DNow! дает заметный результат при обработке графики, хотя и не претендует на вытеснение графических ускорителей, а призвано служить их мощным дополнением. Инструкции сигнальных процессоров (DSP) позволяют повысить производительность таких приложений, как программные модемы (включая ADSL), MP3 и процессоры объемного звучания (Dolby Digital surround sound).

В процессорах Intel расширение 3DNow! не используется.

Блок XMM и расширение SSE

Расширение SSE (Streaming SIMD Extensions — потоковые SIMD-расширения) предназначено для ускорения обработки больших потоков данных в формате с плавающей точкой. Потоковое расширение реализуется на аппаратном *блоке XMM*, в котором первоначально было восемь 128-битных регистров XMM0...XMM7. В 64-битном расширении число регистров XMM увеличили до 16. Регистры XMM позволяют работать с различными операндами:

- четырьмя вещественными числами одинарной точности (32-битные);
- парой вещественных чисел двойной точности (64-битные, только в SSE2);
- упакованными целыми числами: 16 байт, 8 слов, 4 двойных слова или пара учетверенных (по 64 бита) слов (только в SSE2).

Блок позволяет выполнять векторные (они же пакетные) и скалярные инструкции. *Векторные инструкции* реализуют операции сразу над всеми комплектами операндов. *Скалярные инструкции* работают с одним комплектом операндов — младшим словом. Помимо инструкций с новым блоком XMM в расширение SSE входят и дополнительные целочисленные инструкции с регистрами MMX, а также инструкции управления кэшированием. В процессоре Pentium 4 набор инструкций получил очередные расширения — *SSE2*, а позже и *SSE3*, касающиеся добавления новых типов 128-битных операндов для блока XMM. Использование блока XMM позволяет смешивать выполнение в режиме SIMD инструкций с целочисленными данными и с плавающей точкой.

В процессорах Pentium III появилось расширение SSE, а в Pentium 4 — расширение SSE2, предназначенное для 3D-графики, кодирования/декодирования

видео, а также шифрования данных. Позже появилось и расширение SSE3. Блок XMM и инструкции SSE, SSE2 и SSE3 используются и в современных процессорах AMD.

Набор инструкций (система команд)

Набор инструкций современных процессоров x86 вбирает в себя инструкции всех предыдущих поколений. Инструкции можно разделить на прикладные, используемые «полезными» приложениями, и системные, используемые операционной системой для создание среды, в которой работают приложения. На рис. 7.2 изображены группы инструкций процессоров x86 в порядке их появления в разных поколениях и моделях процессоров. Присутствие подмножеств инструкций, заключенных в овалы, можно определить по флагам архитектурных расширений, сообщаемым по инструкции CPUID.

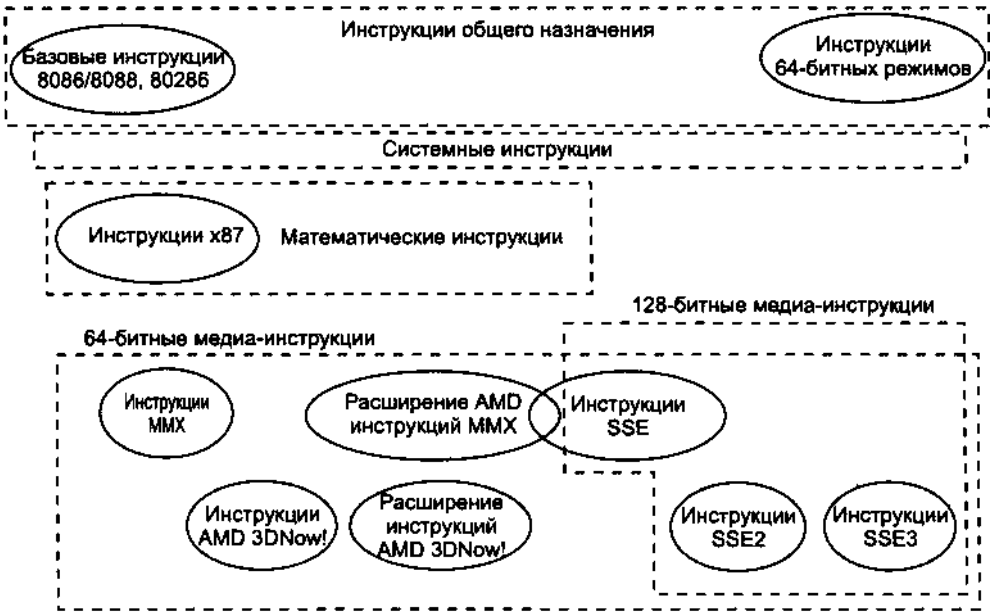


Рис. 7.2. Набор инструкций современных процессоров x86

Прикладные инструкции можно разделить на несколько групп (на рисунке группы объединены пунктирными линиями).

Инструкции общего назначения — основные целочисленные инструкции x86, используемые практически всеми программами. Эти инструкции загружают, сохраняют и обрабатывают данные, расположенные в регистрах общего назначения и памяти. Часть инструкций служит для изменения последовательности исполнения — это инструкции условных и безусловных переходов, вызовы процедур. *Базовые инструкции* общего назначения реализованы во всех процессорах x86 (за исключением некоторых инструкций, наличие которых определяет

ся через инструкцию CPUID). *Инструкции 64-битных режимов* (long mode instructions) появились только в процессорах с 64-битными расширениями и доступны лишь в соответствующих режимах.

Инструкции с плавающей точкой x87 работают с FPU, они используются в старых приложениях, требующих точных вычислений. Заметим, что эти инструкции поддерживают разные форматы данных с плавающей точкой (FP-чисел): 80 бит — расширенная точность, 64 бита — двойная точность, 32 бита — одинарная точность. Кроме того, поддерживаются двоично-десятичные форматы и выполняется преобразование форматов. 80-битный FP-формат обеспечивает максимальную точность и максимальный диапазон охватываемых чисел. Набор инструкций x87 постепенно расширялся, в P6 появились инструкции условных пересылок, сокращающих число ветвлений программы.

64-битные медиа-инструкции расширений оперируют с данными, расположенными в 64-битных регистрах MMX. Они выполняют целочисленные операции и вычисления с плавающей точкой в скалярном и векторном вариантах и предназначены для медиа-приложений, работающих с блоками данных. *Векторные целочисленные инструкции* появились в исходном наборе *MMX*, их состав расширялся инструкциями *AMD Extension to MMX* и частью инструкций *SSE* (введенных Intel). *Векторные инструкции с плавающей точкой* появились в *3DNow*!, их набор расширился в *AMD Extensions to 3DNow! Instructions.*

128-битные медиа-инструкции оперируют данными, расположенными в 128-бит - ных регистрах XMM. Они выполняют целочисленные операции и вычисления с плавающей точкой в скалярном (с одним комплектом операндов) и векторном (с несколькими комплектами, принцип SIMD) вариантах. Эти инструкции предназначены для высокопроизводительных медиа- и научных приложений, работающих с блоками данных. В потоковом расширении *SSE* определены инструкции с плавающей точкой одинарной точности (32 бит), в *SSE2* введены инструкции двойной точности (64 бит), в *SSE3* введено 13 дополнительных инструкций (к 144 существующим в SSE и SSE2), включая SIMD-инструкции с FP-числами двойной точности, целыми числами, а также инструкции управления памятью и кэшированием.

Ряд инструкций представляют собой мосты между вышеприведенными группами. Они обеспечивают пересылку данных между блоками MMX, XMM и общими регистрами. Ряд инструкций (как целочисленных, так и с плавающей точкой) можно исполнять как в блоках MMX (и FPU), так и в блоках XMM.

События — прерывания и исключения

Прерывания (interrupt) и исключения (exception), обобщенно называемые *событиями* (event), нарушают нормальный ход выполнения программы для обработки внешних событий или сигнализации о возникновении особых условий или ошибок. По возникновении события процессор сохраняет в стеке регистр (E)FLAGS и указатель CS:(E)IP на ту инструкцию, которую он должен будет выполнить после обработки события. Этой инструкцией будет следующая за той, во время исполнения которой произошло событие, или та же самая. В защи

щенном режиме при возникновении ряда исключений в стеке сохраняется еще и код ошибки. После сохранения этих значений процессор переходит к исполнению кода обработчика данного события, определяя точку входа в него через номер (0-255) по таблице прерываний. Номер элемента в таблице прерываний называется *вектором прерывания*, он определяется источником события. Обработчик события должен заканчиваться специальной инструкцией возврата IRET, по которой из стека восстанавливаются указатель CS:(E)IP и прежнее значение флагов. Для исключений, в которых сохраняется и код ошибки, обработчик до выполнения инструкции IRET должен извлечь из стека код ошибки.

Различают четыре типа событий:

- Исключения (внутренние прерывания) процессора и сопроцессора, вектор определяется типом произошедшего события.
- Немаскируемые внешние прерывания по входу NMI, вектор фиксирован (01).
- Маскируемые внешние прерывания по входу INT (или по шине APIC), вектор передается контроллером прерываний. Реакция процессора может быть запрещена (точнее, отложена) сбросом флага IF (процессор отреагирует на ожидающее прерывание, когда флаг IF будет установлен).
- Программно вызываемые прерывания, вектор определяется в команде.

Последние события из этого списка в прямом смысле прерываниями не являются, поскольку представляют собой лишь специфический механизм вызова процедур — не по адресу, а по его номеру в таблице, причем независимо от состояния флага IF. Программные прерывания широко используются для вызовов сервисов BIOS и ОС — это компактно и удобно. Программное прерывание INT 3 (однобайтная инструкция, в отличие от остальных — двухбайтных) применяется для расстановки точек останова в отладчиках (встроенные средства отладки вырабатывают исключение с вектором 01).

Исключения (внутренние прерывания) процессора генерируются при возникновении особых условий выполнения текущей инструкции. В большинстве своем они не столько асинхронны, сколько неожиданны для программного кода. Номер вектора определяется процессором в зависимости от происхождения исключения. Исключения подразделяются на три типа.

- *Отказ* (fault) — исключение, возникающее до исполнения инструкции, когда процессор обнаруживает невозможность ее исполнения; состояние машины при этом не меняется. К отказам относится, например, попытка обращения к отсутствующей страницы памяти (используется для организации виртуальной памяти).
- *Ловушка* (trap) — исключение, возникшее в результате исполнения инструкции (например, деления на ноль), при этом состояние машины под действием инструкции оказывается изменившимся.
- *Авария* (abort) — исключение, для которого невозможно точно определить породившую его инструкцию (и невозможны корректные действия для продолжения работы). К этим исключениям относятся двойной отказ (исключе

ние при обработке исключения) и исключения от средств машинного контроля (machine-check).

Маскируемые внешние прерывания обрабатываются процессором по сигналу на входе INT только при установленном флаге разрешения прерываний IF.

Немаскируемые прерывания обрабатываются процессором независимо от состояния флага разрешения прерывания IF. К ним относятся прерывания, приходящие по линии NMI (Non-Maskable Interrupt), а для процессоров, поддерживающих режим системного управления, еще и по линии SMI# (System Management Interrupt).

Каждому номеру (0-255) прерывания или исключения соответствует элемент в *таблице дескрипторов прерываний* (Interrupt Descriptor Table, IDT), который трактуется в зависимости от режима процессора.

В *реальном режиме* таблица IDT содержит *дальние адреса* (двойные слова) обслуживающих процедур и после сброса располагается начиная с нулевых адресов.

В *защищенном режиме* таблица IDT содержит 8-байтные *дескрипторы прерываний*, может хранить от 32 до 256 дескрипторов и располагаться в любом месте физической памяти. Дескрипторы могут быть *шлюзами*, или *вентилями* (gates), прерываний, ловушек или задач. Шлюзы прерываний и ловушек служат для вызова процедур обработки, расположенных в сегментах, описанных их дескрипторами. Эти шлюзы обеспечивают защиту при передаче управления: процессор передает управление обработчику только при определенных соотношениях привилегий прерываемого кода, кода обработчика и привилегии дескриптора. Код обработчика прерываний должен быть не менее привилегированным, чем код прерываемой задачи (иначе сработает исключение защиты). Если прерывающая процедура выполняется на более высоком уровне привилегий, то процессор автоматически переключает стек (это дополнительный расход времени на обращение к памяти). По прерыванию (в том числе и аппаратному) возможно и переключение задач, для этого дескриптор прерывания должен быть шлюзом задачи.

В *64-битном режиме* (long mode) дескрипторы имеют новую 16-байтную структуру, и здесь иные правила переключения стека. Кроме того, в этом режиме не выполняется аппаратное переключение задач.

Под исключения (внутренние прерывания) в процессорах Intel резервируются векторы 0-31 в таблице IDT, однако в PC часть из них перекрывается системными прерываниями — сервисами BIOS и DOS, а также аппаратными прерываниями. Эти перекрытия особенно актуальны для защищенного режима; они усложняют процедуры обработки прерываний.

В начале обработки любого (в том числе и программного) прерывания процессор автоматически сбрасывает флаг разрешения прерываний IF. Процедура обработки завершается инструкцией IRET, по которой из стека восстанавливаются автоматически сохраненные регистры (в восстановленном регистре флагов прерывания разрешены), и процессор начинает выполнение инструкции, следующей за той, после которой исполнялось прерывание. Конечно, программно во время обслуживания прерывания возможно умышленное или случайное изме

нение указателя или содержимого стека, и тогда инструкция IRET «отправляет» процессор по другому адресу, в результате чего компьютер может зависнуть. Если на время обработки требуется реакция и на другие прерывания, обработчик должен установить флаг IF. Прерывания, обслуживаемые до завершения обработки предыдущего, называются *вложенными*. Вложенные прерывания чреваты опасностью переполнения стека, поскольку каждое «вложение» будет задействовать его для своих целей. Переполнение стека может также являться причиной зависаний. Длинные процедуры обработки со сброшенным флагом IF могут привести к потере системного времени, поскольку «часы» операционной системы используют аппаратные прерывания от таймера. Процедура обслуживания для каждого источника аппаратных прерываний должна быть написана весьма осмотрительно и учитывать нюансы работы остальных подсистем. Общее правило — процедура должна выполнять минимальные действия, а если требуется программно сложная реакция, то процедура должна послать сообщение о событии, которое будет обрабатываться позже (не в обработчике прерывания).

Для обработки *аппаратных прерываний* в многопроцессорных системах традиционные аппаратные средства становятся непригодными, поскольку прежняя схема подачи запроса INTR и передачи вектора в цикле INTA# явно ориентирована на единственность процессора. Для решения этой задачи в процессоры, начиная со второго поколения Pentium, введен усовершенствованный программируемый контроллер прерываний (Advanced Programmable Interruption Controller, APIC). Этот контроллер имеет внешние сигналы локальных прерываний LINT[1:0] и интерфейсную шину, по трем проводам которой (PICD[1:0] и PICCLK) процессоры связываются с контроллером APIC системной платы. Для локальных запросов прерываний процессоры имеют линии LINT0, LINT1. Локальные прерывания обслуживаются только тем процессором, на выводы которого поступают сигналы их запросов. Общие (разделяемые) прерывания (в том числе и SMI) приходят к процессорам в виде сообщений по интерфейсу APIC. При этом контроллеры предварительно программируются — тем самым определяются функции каждого из процессоров в случае возникновения того или иного аппаратного прерывания. Контроллеры APIC каждого из процессоров и контроллер системной платы, связанные интерфейсом APIC, выполняют маршрутизацию прерываний (interrupt routing), причем как статическую, так и динамическую. Внешне программный интерфейс обработки прерываний остается совместимым с управлением контроллером 8259A, что обеспечивает прозрачность присутствия APIC для прикладного программного обеспечения. Режим обработки прерываний посредством APIC разрешается сигналом APICEN по аппаратному сбросу, впоследствии он может быть запрещен программно.

7.3. Организация памяти

В первых процессорах семейства память предоставлялась в виде сегментов размером по 64 Кбайт, а суммарный объем программно адресуемой памяти не превышал 1 Мбайт. Архитектура PC ограничивала размер оперативной памяти

объемом 640 Кбайт, начиная с нулевых адресов. Эта область называется *стандартной памятью* (conventional memory), и для прикладных программ из нее остается доступной область порядка 400-550 Кбайт (остальное «съедает» операционная система вместе с разными драйверами). Потребности решаемых задач довольно быстро переросли эти ограничения, и в процессоры ввели средства организации *виртуальной памяти.* Впервые (для x86) они появились в процессоре 80286, но удобный для употребления вид приняли только в 32-битных процессорах (80386 и выше). Во-первых, было снято ограничение на 64-кило- байтный размер сегмента — теперь любой сегмент может иметь почти произвольный (до 4 Гбайт) размер. Во-вторых, был введен механизм страничной трансляции адресов. С его помощью (без сегментации) реализуется виртуальная память размером до 4 Гбайт, а также появляется возможность формирования физического адреса с разрядностью 36 бит (64 Гбайт адресуемой физической памяти). С появлением 64-битных расширений размер виртуальной памяти увеличился до 2^{64} байт. Правда, в первых реализациях процессора принято ограничение виртуального адреса до 48 бит, при этом физический адрес имеет длину 52 бит (возможно и меньше).

Пространство памяти (memory space) предназначено для хранения кодов инструкций и данных, для доступа к которым имеется богатый выбор способов адресации (24 режима). Память может логически организовываться в виде одного или множества сегментов произвольной длины (в реальном режиме — фиксированной). Помимо сегментации в защищенном режиме возможно (при страничной трансляции адресов) разбиение логической памяти на страницы размером 4 Кбайт, каждая из которых может отображаться на любую область физической памяти. Начиная с 5-го поколения появилась возможность увеличения размера страницы до 4 Мбайт. Сегментация и страничная трансляция адресов могут применяться совместно и по отдельности. Сегментация является средством организации логической памяти на прикладном уровне. Страничная трансляция адресов применяется на системном уровне для управления физической памятью. Сегменты и страницы могут выгружаться из физической оперативной памяти на диск и по мере необходимости подкачиваться с него обратно в физическую память. Таким образом реализуется виртуальная память.

Эффективный адрес

При обращении к памяти (к данным), как и при формировании адреса перехода, процессор строит *эффективный адрес*, который может включать до трех компонентов (рис. 7.3). Такой сложный способ задуман для облегчения доступа к элементу массива: компонент *BASE* — базовый адрес массива, *INDEX* — номер элемента, *DISPLACEMENT* — смещение внутри элемента. Массив может состоять из байтов, слов, двойных и учетверенных слов — это учитывается масштабным коэффициентом *SCALE* (1, 2, 4 или 8). Компоненты эффективного адреса могут быть константами (в инструкции), находиться в регистрах и даже в памяти. Такая универсальность оборачивается значительными микроархитектурными издержками.

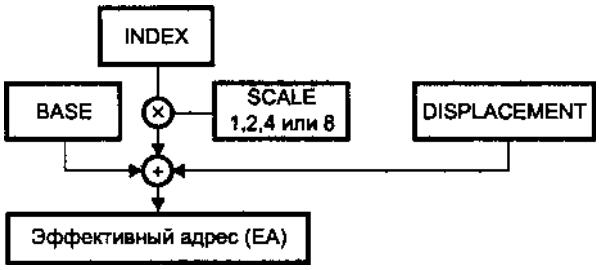


Рис. 7.3. Формирование эффективного адреса

В процессорах с 64-битным расширением появилась возможность адресации данных относительно текущего значения указателя инструкций.

Преобразование адресов

Применительно к памяти различают три адресных пространства: логическое, линейное и физическое. По сочетанию сегментации и страничной трансляции различают две модели памяти:

- В сегментной модели памяти приложение использует несколько сегментов памяти (для кода, данных, стека) и может переключать используемые сегменты. В этой модели приложение оперирует логическими адресами.
- В плоской модели памяти приложению для всех целей выделяется единственный сегмент. В этой модели приложение оперирует линейными адресами. Плоская модель гораздо проще и удобнее в обращении и используется в современных ОС.

Логический адрес состоит из селектора сегмента Seg и эффективного адреса, называемого также смещением (offset). Логический адрес обозначается в форме Seg:Offset. *Селектор сегмента* хранится в старших 14 битах сегментного регистра (CS, DS, ES, SS, FS или GS), участвующего в адресации конкретного элемента памяти. По значению селектора из специальных *таблиц дескрипторов сегментов*, хранящихся в памяти, извлекается начальный адрес сегмента. Поскольку каждая задача может иметь до 16K селекторов (2^{14}), а смещение, ограниченное размером сегмента, - достигать 4 Гбайт, логическое адресное пространство для каждой задачи может равняться 64 Тбайт. Это максимальное пространство виртуальной памяти, доступное программисту при использовании сегментной модели. Операционная система может ограничить число доступных сегментов и их конкретные размеры.

Преобразование логического адреса в физический для 32-битных процессоров иллюстрирует рис. 7.4. Блок сегментации транслирует логическое адресное пространство в 32-битное пространство линейных адресов. *Линейный адрес* образуется сложением базового адреса сегмента с эффективным адресом. Базовый адрес сегмента в реальном режиме образуется умножением содержимого применяемого сегментного регистра на 16 (как и в 8086). В защищенном режиме базовый адрес загружается из дескриптора, хранящегося в таблице, по селектору, загруженному в используемый сегментный регистр.

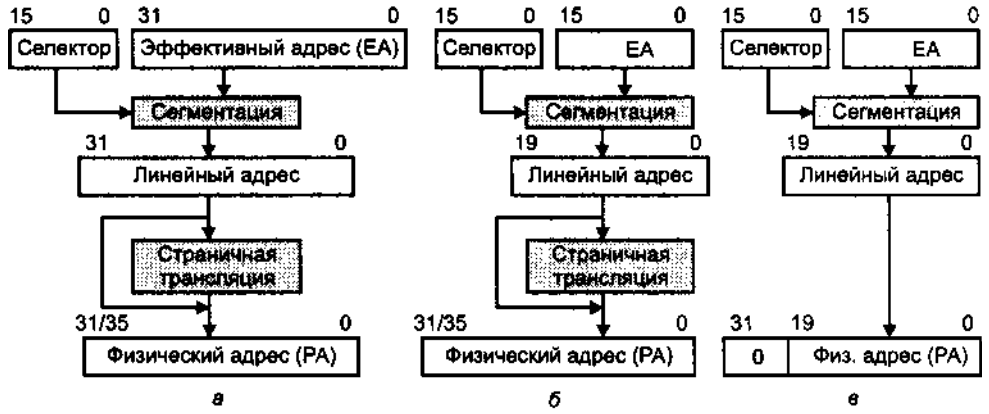


Рис. 7.4. Формирование адреса памяти в 32-битных процессорах: а — в защищенном режиме, *б* — в режиме V86, в — в реальном режиме

Физический адрес памяти образуется после преобразования линейного адреса *блоком страничной трансляции адресов*. Он выводится на внешнюю шину адреса процессора. В простейшем случае (при отключенном блоке страничной трансляции) физический адрес совпадает с линейным. Включенный блок страничной трансляции адресов осуществляет трансляцию линейного адреса в физический страницами размером 4 Кбайт (для последних поколений процессоров также возможны страницы размером 2 или 4 Мбайт). Блок трансляции может включаться только в защищенном режиме.

В 64-битном режиме сегментация упразднена (рис. 7.5, *а):* приложения оперируют только линейными виртуальными адресами. В процессорах с 64-битными расширениями механизм сегментации оставлен только для режима совместимости (рис. 7.5, *б)*. Эти адреса должны соответствовать канонической форме: их старшие биты, выходящие за предел поддерживаемой разрядности, должны быть нулевыми. В противном случае сработает исключение защиты. Из сегментных регистров процессор использует только регистры CS, FS и GS. В дескрипторе сегмента, на который указывает CS, задействуются лишь атрибуты: признак 64-битного режима, размер операнда по умолчанию и уровень привилегий. Регистры FS и GS требуются для нового режима адресации: в дескрипторе сегмента, на который они ссылаются, базовый адрес может применяться как смещение при вычислении адреса (эффективного, виртуального и линейного — теперь это одно и то же).

Блок страничной трансляции адресов позволяет использовать разрядность физического адреса, отличную от разрядности линейного адреса. В процессорах различных моделей соотношения разрядностей менялись:

- В 386SX при 32-битном линейном адресе физический был 24-битным (до 16 Мбайт физически адресуемой памяти).
- В большинстве 32-битных процессоров до 6-го поколения использовался 32-битный физический адрес (до 4 Гбайт физически адресуемой памяти).

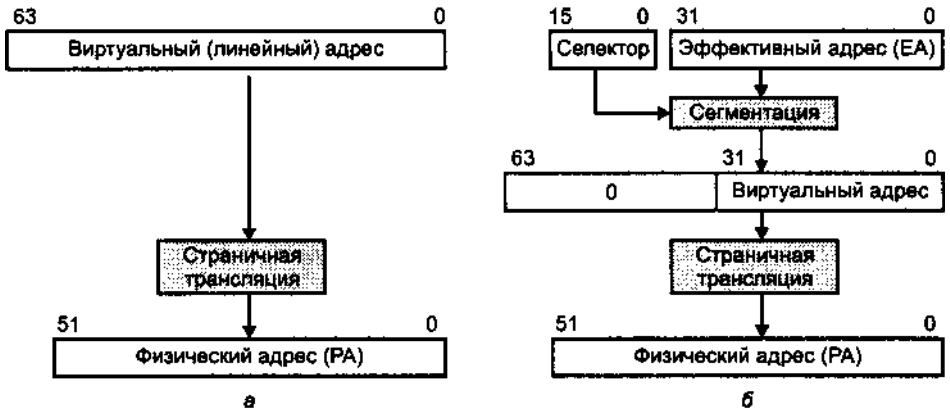


Рис. 7.5. Формирование адреса памяти процессоров с 64-битным расширением: а — в 64-битном режиме, *б* — в режиме совместимости

- В современных процессорах 6-8-го поколений используется расширение физического адреса *(PAE):* 32-битный линейный адрес транслируется в 36-бит- ный физический адрес (до 64 Гбайт физически адресуемой памяти).
- В процессорах с 64-битным расширением на текущем этапе линейный адрес ограничен 48-битным пределом, а физический адрес может иметь разрядность до 52 бит. Использование «полноразмерного» (64-битного) линейного адреса потребовало бы слишком громоздких структур таблиц дескрипторов страниц, на эти жертвы пока не идут.

Страничная трансляция адресов и виртуальная память

Страничное управление памятью — это общепринятый (для разных семейств процессора) механизм организации виртуальной памяти с подкачкой страниц по запросу. *Страничная трансляция адресов* выполняется блоком управления памятью (Memory Management Unit, MMU), расположенным в процессоре, с использованием *каталогов* и *таблиц дескрипторов страниц* — структур данных в физической (оперативной) памяти. Страничная трансляция адресов приводит к тому, что непрерывная область линейных (и эффективных) адресов может отобразиться в виде разбросанных страниц физической памяти. Для того чтобы различные подсистемы компьютера могли программно общаться на «общем языке» линейных адресов, применяют локальные конструкции наподобие MMU. Для графических адаптеров используют таблицу GART (например, в порте AGP, см. 14.9), для контроллеров шин (например, USB и FireWire) строятся специальные конструкции дескрипторов передач.

Блок MMU делит линейный адрес на *виртуальные страницы* фиксированного размера (4K, 4M, 2M). На такие же страницы делится и адресное пространство физических адресов. Часть страниц физического пространства занята ОЗУ,

часть отображается на области памяти, назначенные периферийным устройствам (ввод-вывод, отображенный на память).

Каждая виртуальная страница может присутствовать в физической памяти (ОЗУ или в области памяти ПУ) или нет. Текущее описание страницы хранятся в *дескрипторе страницы* — структуре данных в ОЗУ. Дескрипторы страниц организуются в иерархии каталогов и таблиц, пример такой иерархии для «классических» страниц размером 4К приведен на рис. 7.6. На «корень» этой иерархии указывает управляющий регистр CR3 (доступ к нему привилегирован). Каждый уровень иерархии таблиц использует свой фрагмент линейного адреса. Самый младший фрагмент адреса (offset) является смещением внутри и виртуальной, и физической страниц (у них размеры обязательно совпадают). Элементы каталогов и таблиц страниц содержат физический *базовый адрес* (если таблица или страница, на которую они ссылаются, присутствует в физической памяти) или указание на местоположение элемента в файле подкачки. *Атрибуты* определяют присутствие в памяти, разрешение записи (R/W) и привилегии доступа (User/Supervisor, U/S), а также управляют кэшированием страниц. При использовании расширения физического адреса (Physical Address Extension, PAE), а также 64-битных режимов адресации появляется возможность *защиты страниц с* данными от ошибочного исполнения программного кода — бит NX (No Execute) в дескрипторах таблиц или страниц. Эта возможность присутствует не во всех моделях процессоров и может быть отключена. ОС может использовать данный атрибут для защиты от определенного класса вирусных атак (основанных на переполнении буфера).

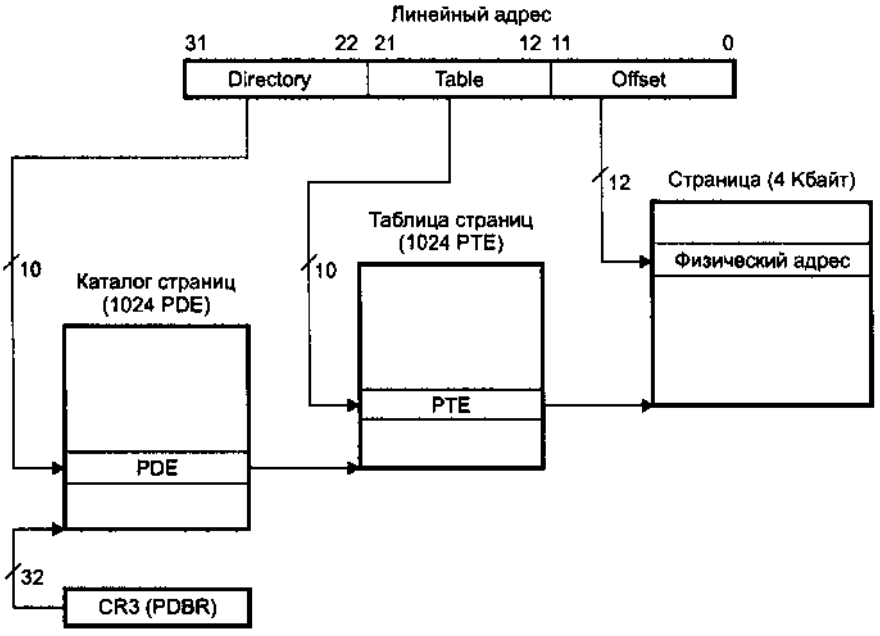


Рис. 7.6. Страничное преобразование адреса (для страниц 4К без PAE)

Размер страницы может принимать значение 4К, 2М и 4 Мбайт. Чем меньше размер страницы и больше объем адресуемой памяти, тем больше приходится использовать дескрипторов страниц и тем сложнее структура каталогов и таблиц и больше работы по их обслуживанию. Для процессоров x86 характерны следующие варианты разбиения на страницы:

- «классические» страницы по 4К с 3-ступенчатым адресом (как на рис. 7.6);
- страницы по 4М с 2-ступенчатым адресом PSE (Page Size Extension — расширение размера страницы);
- страницы по 4К с 4-ступенчатым адресом и расширенным (36-битным) физическим адресом (PAE);
- страницы по 2М с 2-ступенчатым адресом и 36-битным физическим адресом (PAE);
- страницы по 4К с 5-ступенчатым 48-битным линейным адресом и 52-бит- ным физическим адресом, в 64-битном режиме (long mode) процессоров с 64-битным расширением;
- страницы по 2М с 4-ступенчатым 48-битным линейным адресом и 52-бит- ным физическим адресом, в 64-битном режиме (long mode) процессоров с 64-битным расширением.

Процессор выполняет трансляцию адресов автоматически в процессе исполнения инструкции, обращающейся к памяти (и, естественно, при выборке очередной инструкции). Если самой страницы, к которой выполняется обращение, в физической памяти нет, то возникает исключение «отказ страницы», обработчик которого и обеспечивает ее появление в физической памяти. Для этого, возможно, придется «подкачать» ее образ с устройства хранения. Если в ОЗУ не осталось нераспределенных страниц, то обработчику придется освободить какую-то страницу, выгрузив (если надо) ее содержимое в файл подкачки. Если с прошлой подкачки из файла данная страница не менялась (а этот факт отражается атрибутом в ее дескрипторе), то выгрузка не требуется (и экономится время).

Таблицы каталогов и страниц, относящиеся к линейному адресу, по которому процессор выполняет обращение к памяти, должны присутствовать в ОЗУ. В противном случае возникнет тупиковая ситуация — двойной отказ с соответствующим исключением.

Дескрипторы страниц и каталогов находятся в памяти и занимают учетверенное слово (64 бит). Эти дескрипторы процессору приходится автоматически считывать, чтобы добраться до целевой ячейки памяти. Чтение нескольких слов (в самом тяжелом случае — четырех дескрипторов) на пути к искомому элементу — слишком дорогая расплата за виртуализацию памяти. Для сокращения затрат времени введено кэширование описателей страниц — *буфер ассоциативной трансляции* (Translate Look-aside Buffer, TLB). В буфере TLB, расположенном в процессоре, хранятся последние использованные описатели страниц и старшая часть виртуального адреса (все поля, левее поля смещения). Описатели в буфере ищутся ассоциативно: старшая часть линейного адреса сравнивается с адресами во всех элементах TLB. Если обнаруживается совпадение — по

везло, и обращение к памяти будет быстрым. От объема и эффективности буфера TLB существенно зависит производительность процессора. В современных процессорах TLB для инструкций и данных разделяют, количество элементов увеличивают и ускоряют ассоциативный поиск. Как и для всякой системы кэширования, существенна политика замещения элементов — это еще одно поле для оптимизации на микроархитектурном уровне. В процессорах x86 заполнение элементов TLB для программ прозрачно и неуправляемо, есть только специальные инструкции очистки элементов TLB. Заметим, что в процессорах иных архитектур элементы в TLB заполняются программно и заблаговременно, а не автоматически и только по факту обращения к памяти (как в x86).

Стек

Стек представляет собой непрерывную область памяти, адресуемую регистрами ESP (указатель стека) и SS (селектор сегмента стека). Особенность стека заключается в том, что данные в него помещаются и из него извлекаются по принципу «первым вошел — последним вышел». Данные помещаются в стек с помощью инструкции PUSH (заталкивание), а извлекаются по инструкции POP (вытаскивание). Помимо явного доступа к стеку с помощью инструкций PUSH и POP стек автоматически используется процессором при выполнении инструкций вызова, возвратов, входа и выхода из процедур, а также при обработке прерываний.
Стек используют для разных целей:

- организации прерываний, вызовов и возвратов;
- временного хранения данных, когда под них нет смысла выделять фиксированные места в памяти;
- передачи и возвращения параметров при вызовах процедур.

До использования стека он должен быть инициализирован так, чтобы регистры SS:ESP указывали на область реальной оперативной памяти (стек в ПЗУ, естественно, работать не может). Прикладные программы получают, как правило, от операционной системы готовый к употреблению стек. В защищенном режиме сегмент состояния задачи содержит четыре селектора сегментов стека (для разных уровней привилегий), но в каждый момент задействуется, естественно, только один стек. Если для стека определен слишком маленький сегмент, то возможно *переполнение стека* (stack overflow). Переполнение в ОС защищенного режима вызывает срабатывание защиты, в ОС реального режима приводит к «загадочным» вылетам и зависаниям. Переполнение может происходить при интенсивных прерываниях, когда до завершения обработки одного прерывания возникает и обрабатывается другое, более приоритетное (вложенные прерывания).

Кэширование памяти

В современных процессорах организация взаимодействия с памятью играет важную роль, поскольку основная память по своей природе гораздо медлитель-

нее ее «клиентов» — исполнительных устройств процессора. Концептуальная структура связки современного процессора и памяти приведена на рис. 7.7.

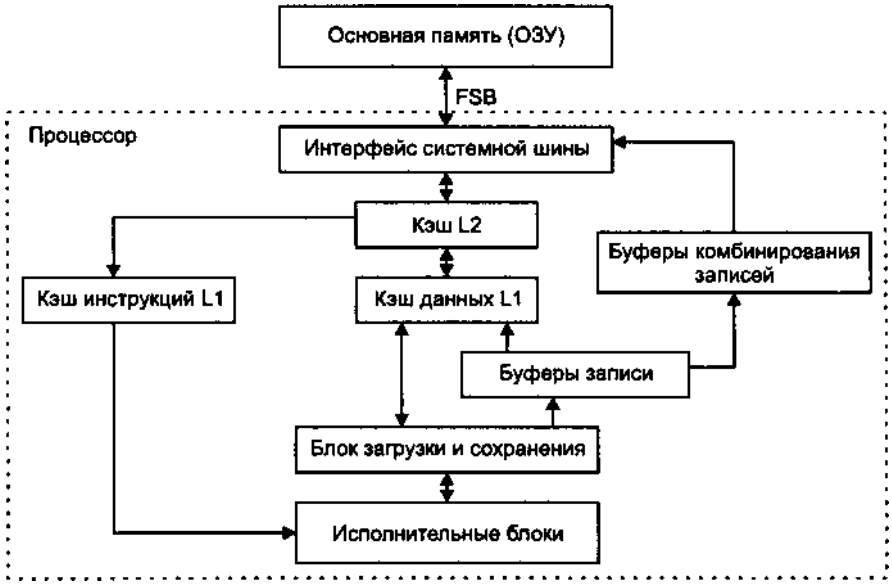


Рис. 7.7. Процессор, память и кэш

Первичный кэш данных (L1 data cache) хранит данные, к которым процессор недавно обращался (читал или записывал) при исполнении программы. *Первичный кэш инструкций* (L1 instruction cache) содержит недавно выполнявшиеся инструкции, которые, возможно, в скорости будут исполняться повторно, а также следующие за ними/упреждающие считывания. Отличие этого кэша — отсутствие необходимости отслеживания записей (инструкции только считываются), что упрощает реализацию кэша. Объем первичного кэша невелик (8-128 Кбайт). Для современных процессоров характерно применение раздельного первичного кэша (так называемая гарвардская архитектура), хотя возможен и единый первичный кэш для данных и инструкций (принстонская архитектура). В процессорах Pentium 4 первичный кэш устроен несколько иначе (см. 7.7).

Вторичный кэш (L2 cache) обычно общий, его размер в несколько раз больше, но быстродействие, как правило, ниже, чем первичного. Вторичный кэш может быть эксклюзивным — в нем не будут храниться данные и инструкции, находящиеся в первичном кэше. С точки зрения эффективности использования кэш-памяти это выгодно. Возможен и другой вариант — инклюзивный, когда во вторичном кэше имеется копия информации первичного.

Кэш третьего уровня (L3 cache) применяется нечасто, поскольку его реализация слишком дорога (его объем должен быть больше, чем у вторичного). Встроенный кэш L3 имеется у процессоров Xeon MP, его размер уже достигает 8 Мбайт.

Буфер отложенной записи служит для временного хранения данных, предназначенных для записи, когда кэш или память заняты выполнением других обращений.

Буфер комбинирования записи предназначен для объединения разрозненных записей в некэшируемой памяти. В случае, когда физическая очередность записей несущественна, данный буфер позволяет выполнять записи более эффективно за счет сокращения числа транзакций (при их удлинении).

Микроархитектурные решения конкретных процессоров могут отличаться от приведенной схемы, перечисленные блоки могут присутствовать и не в полном составе, но общая идея сохраняется. В процессорах до 6-го поколения вторичный кэш отсутствовал — он располагался на системной плате и обслуживался внешним кэш-контроллером. Кэширование может быть и трехуровневым. Кэшем третьего уровня оказывается кэш, установленный на системной плате с сокетом 7, когда в него устанавливают процессор AMD K6-3, обладающий встроенным двухуровневым кэшем. В процессорах 1-3-го поколений (кроме некоторых моделей класса 386) встроенный кэш отсутствовал, а внешний кэш применялся в особо «продвинутых» системных платах.

Кэш-контроллеры должны обеспечивать *когерентность* (coherency) — согласованность данных кэш-памяти всех уровней с данными основной памяти при том условии, что обращение к этим данным может производиться и мастерами шин (PCI со всеми «родственниками», ISA и т. д.). Следует также учесть, что процессоров может быть несколько, и у каждого может быть свой внутренний кэш.

Организация кэш-памяти

Контроллер кэша оперирует *строками* (cache line) фиксированной длины. Строка может хранить копию блока основной памяти, размер которого, естественно, совпадает с длиной строки. С каждой строкой кэша связана информация
об адресе скопированного в нее блока основной памяти и ее состоянии. Строка может быть *действительной* (valid) — это означает, что в текущий момент времени она достоверно отражает соответствующий блок основной памяти, — или *недействительной*. Информация о том, какой именно блок занимает данную строку (то есть старшая часть адреса или номер страницы), и о ее состоянии называется *тегом* (tag) и хранится в связанной с данной строкой ячейке специальной *памяти тегов* (tag RAM). В операциях обмена с основной памятью строка обычно участвует целиком (не разделенный на секторы кэш), для процессоров класса 486 и выше длина строки совпадает с объемом данных, передаваемых за один пакетный цикл (для 486-го это 4 x 4 = 16 байт, для Pentium —
4 x 8 = 32 байт, в процессорах 6-8-го поколений используются и 64-байтные строки). Возможен и вариант разделенного на секторы (sectored) кэша, при котором одна строка содержит несколько смежных ячеек — *секторов*, размер которых соответствует минимальной порции обмена данных кэша с основной памятью. При этом в записи каталога, соответствующей каждой строке, должны храниться биты действительности для каждого сектора данной строки. Разделение на секторы позволяет экономить память, необходимую для хранения ка

талога при увеличении объема кэша, поскольку большее количество битов каталога отводится под тег, и выгоднее использовать дополнительные биты действительности, чем увеличивать глубину индекса (количество элементов) каталога.

Строки кэша под отображение блока памяти выделяются при промахах операций чтения, в P6 строки заполняются и при записи. Запись блока, не имеющего копии в кэше, производится в основную память (для повышения быстродействия запись может производиться через буфер отложенной записи). Поведение кэш-контроллера при записи в память, когда копия затребованной области находится в некоторой строке кэша, определяется его алгоритмом, или *политикой записи* (write policy). Существуют две основных политики записи данных из кэша в основную память: *сквозная запись* (Write Through, WT) и *обратная запись* (Write Back, WB).

Политика WT предусматривает одновременное выполнение каждой операции записи (даже однобайтной), попадающей в кэшированный блок, в строку кэша и в основную память. При этом процессору при каждой операции записи приходится выполнять относительно длительную запись в основную память. Алгоритм довольно прост в реализации и легко обеспечивает целостность данных за счет постоянного совпадения копий данных в кэше и основной памяти. Для него не нужно хранить признаки присутствия и модификации — вполне достаточно информации тега (при этом считается, что любая строка всегда отражает какой-либо блок, а какой именно — указывает тег). Но эта простота оборачивается низкой эффективностью записи. Существуют варианты этого алгоритма с отложенной буферизованной записью, при которой данные в основную память переписываются через FIFO-буфер во время свободных тактов шины.

Политика WB позволяет уменьшить количество операций записи на шине основной памяти. Если блок памяти, в который должна производиться запись, отображен в кэше, то физическая запись сначала производится в эту действительную строку кэша, которая отмечается как *грязная* (dirty), или модифицированная, то есть требующая выгрузки в основную память. Только после этой выгрузки (записи в основную память) строка станет *чистой* (clean) и ее можно будет использовать для кэширования других блоков без потери целостности данных. В основную память данные переписываются только целой строкой. Эта выгрузка контроллером может откладываться до наступления крайней необходимости (обращение к кэшированной памяти другим абонентом, замещение в кэше новыми данными) или выполняться в свободное время после модификации всей строки. Данный алгоритм сложнее в реализации, но существенно эффективнее, чем WT. Поддержка системной платой кэширования с обратной записью требует обработки дополнительных интерфейсных сигналов для выгрузки модифицированных строк в основную память, если к этой области производится обращение со стороны таких контроллеров шины, как другие процессоры, графические адаптеры, контроллеры дисков, сетевые адаптеры и т. п.

В зависимости от способа определения взаимного соответствия строки кэша и области основной памяти различают три архитектуры кэш-памяти: кэш прямого отображения (direct-mapped cache), полностью ассоциативный кэш (fully

associative cache) и их комбинацию — наборно-ассоциативный кэш (set-asso- ciative cache). Подробно эти архитектуры рассмотрены в [2], здесь ограничимся краткими характеристиками.

В кэш-памяти *прямого отображения* адрес памяти, по которому происходит обращение, однозначно определяет строку кэша, в которой может находиться требуемый блок. Поскольку объем основной памяти много больше объема кэша, на каждую строку кэша может претендовать множество блоков памяти с одинаковой младшей частью адреса (смещением внутри страницы). Одна строка в определенный момент может, естественно, содержать копию только одного из этих блоков, и информация о том, какой именно блок занимает данную строку, хранится в памяти тегов. Количество ячеек в памяти тегов должно равняться количеству строк кэша, а разрядность памяти тегов должна быть достаточной, чтобы вместить старшие биты адреса кэшируемой памяти, не попавшие на шину адреса кэш-памяти. Помимо адресной части тега с каждой строкой кэша связаны биты признаков действительности и модификации данных. Независимо от объема затребованных данных в кэш из основной памяти строка переписывается целиком. Если контроллер кэша реализует *упреждающее считывание* (read ahead), то в последующие свободные циклы шины обновится также и следующая строка (если она была *чистой).* Чтение «про запас» позволяет при необходимости осуществлять пакетный цикл чтения из кэша через границу строки.

Такой кэш имеет самую простую аппаратную реализацию и применяется во вторичном кэше большинства системных плат. Однако ему присущ серьезный недостаток: если в процессе выполнения программы процессору поочередно будут требоваться блоки памяти, смещенные относительно друг друга на величину, кратную размеру страницы, то кэш будет работать интенсивно, но вхолостую (cache trashing). Увеличение размера кэша при сохранении архитектуры прямого отображения даст не очень существенный эффект, поскольку разные задачи будут претендовать на одни и те же строки кэша. Объем кэшируемой памяти (Mcached) при архитектуре прямого отображения определяется объемом кэш-памяти (Vcache) и разрядностью памяти тегов (*N):* $M_{CACHED} = V_{CACHE} \times 2^N$. Так, для кэша размером 256 Кбайт и 8-битной памяти тегов (типичный вариант для системных плат с сокетами 5 и 7) объем кэшируемой памяти составит Mcached = 256 Кбайт x 2^8 = 64 Мбайт.

Наборно-ассоциативная архитектура кэша позволяет каждому блоку кэшируемой памяти претендовать на *одну из нескольких* строк кэша, объединенных в *набор* (set). Можно считать, что в этой архитектуре есть несколько параллельно и согласованно работающих каналов (банков) прямого отображения, где контроллеру кэша приходится принимать решение о том, в какую из строк набора помещать очередной блок данных. В простейшем случае каждый блок памяти может помещаться в одну из двух строк (two way set-associative cache — двухканальный наборно-ассоциативный кэш). Наборно-ассоциативная архитектура широко применяется для первичного кэша современных процессоров. Объем кэшируемой памяти определяется так же, как и в предыдущем варианте, но здесь фигурируют объем одного банка (а не всего кэша) и разрядность относящихся к нему ячеек тега.

В отличие от предыдущих у *полностью ассоциативного* кэша *любая строка* может отображать *любой блок* памяти, что существенно повышает эффективность использования его ограниченного объема. Реализация полностью ассоциативного кэша является сложной аппаратной задачей, которая пока решена только для небольших объемов первичного кэша в некоторых процессорах. Применение полностью ассоциативной архитектуры во вторичном кэше пока не предвидится.

Обеспечение когерентности

Кэш-память процессоров строится с учетом возможности обращений к памяти со стороны внешних абонентов — других процессоров или иных контроллеров шины. Процессоры имеют механизмы внешнего слежения за состоянием собственного кэша с соответствующими аппаратными интерфейсами. Для поддержания согласованности данных кэша и основной памяти процессор отрабатывает *циклы слежения* (snoop cycle, или inquire cycle), инициированные внешней (Для него) системой. В этих циклах, происходящих при обращении к памяти со стороны внешнего абонента, процессор определяет присутствие затребованной области в своем кэше. Если область отображается в кэше, то действия процессора зависят от состояния соответствующей строки кэша и типа внешнего обращения. Обращение по записи аннулирует данную строку. Обращение по чтению к области, соответствующей модифицированной («грязной») строке, вызовет выгрузку ее содержимого в основную память, прежде чем внешний абонент выполнит реальное считывание. В процессорах 6-8-го поколений обращение к «грязной» строке со стороны другого процессора может вызывать выгрузку ее содержимого непосредственно в обращающийся процессор, что экономит время. Выгрузка этой строки в основную память производится позже согласно алгоритму обратной записи.

Кэш современных процессоров поддерживает протокол обеспечения когерентности *MOESI*, названный по определяемым им состояниям *M* (Modified), O (Owned), *E* (Exclusive), 5 (Shared) и *I* (Invalid). Процессоры Pentium поддерживали протокол только в части «MESI». Первичный кэш инструкций реализует протокол лишь в части «SI», поскольку он не допускает записи. Состояния строк для каждого процессора определяются следующим образом:

- M-состояние — строка модифицирована и присутствует в кэше только этого процессора, в основной памяти и кэше других процессоров копии недействительны;
- O-состояние — строка присутствует в кэше этого процессора и потенциально может присутствовать в кэшах других процессоров, копия в памяти недействительна;
- E-состояние — строка присутствует в кэше только этого процессора, но не модифицирована (ее копия в основной памяти действительна); запись переведет ее в *M*-состояние, не вызывая внешнего цикла обращения;
- 5-состояние — строка присутствует в кэше этого процессора и потенциально может присутствовать в кэшах других процессоров, копия в памяти действи

тельна; запись в нее должна сопровождаться сквозной записью в основную память, что повлечет аннулирование соответствующих строк в других кэшах;

- I-состояние — строка отсутствует в кэше, действительные данные можно прочитать из памяти или из кэша другого процессора.

Для работоспособности самомодифицирующегося кода процессор контролирует операции записи в память на попадание в область, представленную в кэше инструкций. Контроль выполняется на уровне физических адресов, в случае попадания строка аннулируется.

Типы памяти

В пространстве памяти компьютера имеются области, для которых кэширование принципиально недопустимо (например, разделяемая память адаптеров) или непригодна политика обратной записи. Кроме того, кэширование иногда полезно отключать при выполнении однократно исполняемых участков программы (например, инициализации) с тем, чтобы из кэша не вытеснялись более полезные фрагменты.

В процессорах 6-го поколения в связи с их «беспорядочностью» и «спекулятивностью» обращения к памяти могут производиться с различными методами повышения эффективности. Эта беспорядочность также не всегда допустима.

По возможностям кэширования и требованиям к упорядоченности память классифицируется следующим образом:

- Некэшируемая (UnCacheable, *UC)* память. Все обращения процессора по чтению и записи выполняются строго в порядке, предписанном программным кодом, и выходят на системную шину. Никакие спекулятивные чтения и предварительные выборки не используются. Разновидность некэшируемой памяти — память *CD* (Cache Disable), для которой кэширование запрещено дескриптором страницы. Некэшируемая память требуется для ввода-вывода, отображенного на память. Работа процессора в этом режиме с обычным ОЗУ приведет к значительному снижению производительности.
- Память с комбинируемой записью (Write Combining, *WC*). Некэшируемая память, когерентность которой протоколом шины не поддерживается. Спекулятивное чтение допустимо, записи могут комбинироваться и откладываться до любого события, вызывающего сериализацию (инструкция CPUID, обращение к некэшируемой памяти, прерывание...). Такой тип может применяться, например, для видеопамяти графического адаптера (порядок записей не важен).
- Память со сквозной записью (Write Through, WT). Кэшируемая память, все операции записи отражаются в кэше и выходят на системную шину. Операции чтения по возможности выполняются из кэша, кэш-промахи вызывают заполнение строк кэша. Спекулятивное чтение и комбинирование записей разрешено. Данный тип применим, например, для буферов кадров, а также для памяти, к которой могут обращаться устройства, подключенные к шине и не поддерживающие протоколов обеспечения когерентности.
- Память с обратной записью (Write Back, *WB).* Кэшируемая память, все операции чтения и записи по возможности выполняются только с кэш-памятью.

Запись на системную шину выходит только при необходимости освобождения строк или по требованию других абонентов шины, что уменьшает необязательный трафик шины. Спекулятивное чтение и комбинирование записи разрешено. Этот тип самый производительный, но требует поддержки протокола обеспечения когерентности от всех абонентов шины, обращающихся к данной области памяти.

- Память с защищенной записью (Write Protected, *WP).* Кэшируемая память, операции чтения по возможности выполняются из кэша, промахи вызывают заполнение строк. Записи выходят на системную шину, аннулируя строки в кэшах всех остальных абонентов шины (процессоров). Данный тип применяют для теневой памяти (shadow ROM), изменение содержимого которой сразу должно стать видимым всем возможным «читателям».

Доступные методы кэширования зависят от возможностей процессора. Базовые методы (сквозная и обратная запись или отмена кэширования) управляются атрибутами системы страничной трансляции адресов, более совершенные методы программируются только через регистры MTRR или PAT (см. далее), если таковые имеются в процессоре.

Управление кэшированием и обращениями к памяти

Все механизмы кэширования в основном прозрачны для прикладных программ и после разрешения кэширования пропускают через себя потоки инструкций и данных без требования явного программного управления. Однако знание особенностей механизмов кэширования помогает в оптимизации кода. Так, например, можно определить оптимальные размеры одновременно обрабатываемых структур данных, при которых кэш не «буксует» (cache thrashing). Процессоры разных моделей имеют различные характеристики отдельных элементов кэша. Определить характеристики элементов кэша процессоров 6-8-го поколений позволяет вызов инструкции CPUID(2). Заметим, что не все модели процессоров способны кэшировать весь объем физически адресуемой памяти (см. описания конкретных процессоров).

Механизм управления кэшированием включает в себя как программные флаги, так и аппаратные средства, позволяющие разрешать и ограничивать возможности кэширования. Программные средства управления включают флаги управляющих регистров и биты элементов каталога и таблиц страниц, а также специальные инструкции. Аппаратные средства включают входные сигналы разрешения кэширования и управления политикой записи и очистки кэша, а также выходные сигналы управления вторичным кэшем. В P6+ имеются также регистры MTRR, определяющие возможности кэширования на уровне областей физической памяти. Эти механизмы имеют не одинаковые области воздействия. Если различные механизмы определяют возможности кэширования конкретной области памяти по-разному, реализуется самое жесткое из ограничений: запрет кэширования имеет приоритет над разрешением, а политика WT отменяет политику WB.

Кэшированием управляют на этапе заполнения строк (в соответствии с объявленным типом памяти), последующие кэш-попадания операций чтения памяти будут обслуживаться только из кэша. Кэш можно программно очистить: инструкция CFLUSH по указанному адресу проверяет его присутствие в строке кэша и аннулирует ее (с предварительной выгрузкой, если она модифицирована). Инструкция WBINVD очищает весь кэш (с предварительной выгрузкой модифицированных строк).

В процессорах с расширением SSE и 3DNow! появились новые инструкции, связанные с кэшированием. Инструкциями PREFETCH можно «намекнуть» процессору на грядущее обращение к памяти по указанному адресу — процессор загрузит группу байтов памяти в строку кэша заданного уровня. Когда данные потребуются для исполнения, они уже будут близко — это значительно сокращает простои конвейера процессора из-за ожидания данных.

Появились инструкции, позволяющие выгружать данные из регистров MMX и XMM в оперативную память, минуя кэш, — это позволяет сохранить в кэше ранее занесенные данные.

Появились и инструкции, упорядочивающие физические обращения к памяти: инструкция SFENSE заставляет процессор выполнить все предыдущие записи, LFENSE — все предыдущие чтения, а MFENSE — все предыдущие обращения к памяти до того, как начнут выполняться последующие обращения. Общее программное управление кэшированием осуществляется посредством битов управляющего регистра CR0: CD (Cache Disable) и NW (No Write Through). При использовании страничной трансляции адресов в управлении кэшированием принимают участие биты регистра CR3 и элементов каталога и таблиц страниц.

В архитектуру процессоров 6-8-го поколений введены регистры MTRR (Memory Type Range Registers), которые реализуют вышеописанные функции аппаратного управления кэшированием, а также изменение порядка записи для определенных областей памяти. С помощью этих регистров в физической памяти может быть определено до 96 областей адресов с одинаковым типом кэширования. Такое распределение позволяет оптимизировать операции с ОЗУ, ПЗУ, видеобуферами и адаптерами ввода-вывода, отображенными на пространство памяти. По аппаратному сбросу регистры MTRR устанавливаются в такое состояние, которое ведет к объявлению некэшируемой всей физической памяти. Дальнейшая инициализация, выполняемая обычно во время теста POST системы BIOS, программирует регистры в соответствии с реальным распределением памяти. Типы памяти и их характеристики приведены в табл. 7.1.

Таблица 7.1. Типы памяти, определяемые регистрами MTRR и PAT

Мнемоника	Код в MTRR[1]	Кэширование	Политика WB	Спекулятивное[2] чтение	Порядок операций
UC	0	Нет	Нет	Да	Строгий
WC	1	Нет	Нет	Да	Слабо упорядоченный (weak ordering)
WT	4	Да	Нет	Да	Спекулятивный (speculative processor ordering)

продолжение

Таблица 7.1 (*продолжение)*

Мнемоника	Код в MTRR[1]	Кэширование	Политика WB	Спекулятивное[2] чтение	Порядок операций
WP	5	Чтение — да, запись — нет	Нет	Да	Спекулятивный
WB	6	Да	Да	Да	Спекулятивный
UC-	7	Нет	Нет	Да	Строгий

[1] Коды 2,3, 8-255 зарезервированы, их применение в P6 вызывает исключение защиты. Код 7 применим только для задания свойств через PAT, через MTRR он может быть изменен на WC.
[2] Чтение, результат которого может не требоваться программе.

Регистры MTRR входят в число модельно-специфических регистров (MSR). Они определяют свойства фиксированных зон для 1-го мегабайта физической памяти и до 8 зон произвольного размера для памяти в любом диапазоне адресов. Фиксированные зоны для адресов 00000-7FFFFh имеют размер по 64 К, для адресов 8000-BFFFFh — по 16 К и для адресов C0000-FFFFFh — по 4 К.

Регистры MTRR позволяют управлять определенными зонами физической памяти статически. Эта задача ложится, как правило, на BIOS компьютера. Начиная с процессора Pentium III введен новый способ управления свойствами памяти на уровне страниц, который позволяет операционной системе и приложениям динамически выбирать оптимальные (с точки зрения повышения производительности и корректности работы) свойства каждой страницы памяти. В дескрипторе страницы имеются биты PCD и PWT, изначально предназначенные для управления кэшированием, и новый бит PAT. При использовании *таблицы атрибутов страниц* (Page Attribute Table, PAT) эти биты задают тип памяти косвенно. Таблица атрибутов содержится в 64-битном *регистре PAT*, в котором каждый байт (вернее, его 3 младших бита) определяет тип памяти для каждого возможного трехбитного кода, собранного из битов PAT, PCD и PWT дескриптора страницы, к которой происходит обращение. Использование регистров MTRR и PAT для определения свойств памяти разрешается соответствующими управляющими битами процессора.

7.4. Особые режимы работы процессора

Здесь рассмотрим начальный запуск процессора, загрузку «заплат» его микропрограмм, работу в режиме SMM, снижение производительности ради экономии энергии и совместную работу нескольких процессоров. Эти режимы практически незаметны прикладному программисту и пользователю, но представление о них дает более полную картину «жизни» процессора в компьютере.

Запуск и инициализация процессоров

Аппаратный сброс (hardware reset) выполняется процессором при включении питания и по сигналу RESET#. При низком уровне сигнала RESET# процессор прекращает выполнение инструкций и перестает управлять системной шиной. В момент окончания сигнала аппаратного сброса процессор воспринимает уровни

сигналов на некоторых линиях интерфейса, что определяет его интерфейсные свойства. Процессору устанавливаются коэффициент умножения тактовой частоты, режим (WB/WT) работы кэша, роль процессора в многопроцессорных системах, способ подачи сигналов прерываний (для процессоров с контроллером APIC) и некоторые другие параметры. Эти уровни задаются чипсетом системной платы в соответствии с ролью процессора и установками джамперов и CMOS Setup. Сброс переводит процессор в реальный режим и устанавливает ряд регистров в определенное состояние.

Аппаратный сброс аннулирует строки кэш-памяти, буферов трансляции (TLB) и таблиц переходов (BTB). После сброса процессор начинает выполнение инструкции, считанной по физическому адресу FFFFFFF0h. Исполняемый программный код должен обеспечить инициализацию системы (регистров процессора, структур данных в памяти). Из этого следует, что, по крайней мере, на начальный период времени после сигнала RESET# компьютер должен иметь образ BIOS в адресах FFFFFFF0-FFFFFFFFh, в то время как в PC на процессорах 8086/88 память ROM BIOS располагалась под границей первого мегабайта (FFFFFh), а компьютеры AT-286 имели ее образ и под границей 16-го мегабайта (FFFFFFh).

Процессоры P6 допускают смену положения вектора начального запуска на 0FFFF0h, правда этой возможностью, похоже, никто не пользуется (в Pentium 4 этой возможности уже нет).

Процессоры Pentium+ имеют дополнительный вход INIT, по которому выполняются примерно те же действия, что и по RESET#, но не очищается внутренняя кэш-память, не изменяется состояние FPU и регистров MSR. Этот сигнал может использоваться для перевода процессора в реальный режим (в стиле 80286) с сохранением данных в кэше. Такой же «мягкий» сброс возможен и по сообщению, получаемому процессором по шине APIC.

Переключение между реальным и защищенным режимами

Переключение процессора в защищенный режим из реального осуществляется загрузкой в системный регистр CR0 слова с единичным значением бита PE (Protect Enable). До переключения в памяти должны быть проинициализированы необходимые таблицы дескрипторов IDT и GDT. *Переключение процессора из защищенного режима в реальный* возможно не только через аппаратный сброс, как это было у 80286, но и сбросом бита PE в CR0. До этого переключения также необходимо загрузить в сегментные регистры селекторы дескрипторов, описывающие свойства сегментов стандартного реального режима. Однако вместо этого можно создать и «нереальный» режим (big real mode), отличающийся от реального возможностью доступа к сегментам большого (до 4 Гбайт) размера. Правда, у процессоров 80286 и 80386 лимит кодового сегмента принудительно ограничивается размером 64 Кбайт, но у более новых процессоров большой размер допустим для всех сегментов. «Нереальный» режим часто используется

менеджерами памяти для DOS и игровыми программами, требующими большого объема памяти.

В режим с 64-битной адресацией процессор можно перевести из обычного защищенного режима. Для этого в пределах первых 4 Гбайт должны быть сформированы необходимые структуры данных (после активации 64-битного режима они могут быть перемещены):

- таблицы IDT с 64-битными дескрипторами (и сами обработчики прерываний и исключений для 64-битного режима);
- таблица GDT с соответствующими дескрипторами;
- 64-битный сегмент состояния задачи (TSS);
- четырехуровневые таблицы страничной трансляции.

64-битный режим активируется установкой в регистре EFER бита LMA и последующим включением страничной трансляции (перед активацией страничная трансляция должна быть отключена, поскольку в 32- и 64-битном режимах механизмы трансляции различны).

Обновление микрокода

Фирма Intel (и не только она) постоянно модернизирует свои процессоры, и даже в пределах одной модели процессоры разного времени выпуска различаются степпингом (см. 7.6). Для процессоров каждого степпинга известны свои ошибки (errata) и методы их исправления. Микроархитектура процессоров начиная с 6-го поколения, позволяет исправлять эти ошибки путем загрузки в процессор блока «заплат», являющегося, очевидно, набором фрагментов микропрограмм. Обновление микрокода (microcode update) должно выполняться во время инициализации процессора после аппаратного сброса, загруженный микрокод действует только до следующего аппаратного сброса (инициализация сигналом INIT# на загруженное обновление не влияет). Фирма Intel отвечает за корректность работы своих процессоров только при загруженных «заплатах», и для каждого степпинга выпускает специальный блок данных (в виде файла). Таким образом, процессор определенного степпинга рассматривается как комплект из собственно процессора и «заплат». Заплаты фирма помещает на своем сайте; правда, доступ к ним закрыт паролями. Пароли сообщаются официальным дилерам, так что за свежими «заплатами» следует обращаться именно к ним. Если дилер не способен предоставить «заплаты» (или сообщить пароль), то можно усомниться в его легальности. Загрузка актуальных «заплат» в процесср организуется в два этапа: требуемый образ «зашивается» в BIOS изготовителем компьютера или пользователем; на этапе инициализации компьютера BIOS организует загрузку микрокода в процессор. Если BIOS не поддерживает процессор требуемого степпинга, следует обновить либо всю систему BIOS (см. 6.2), либо только область с микрокодами. Для обновления в BIOS области с микрокодом могут быть использованы функции DMI. Более полно процедура обновления описана в [2], [3]. Если требуемое обновление микрокода в BIOS встроить не удается, можно использовать загружаемые утилиты обновления,

запускаемые на платформах DOS/Windows из файла AUTOEXEC.BAT при каждой загрузке ОС.
«Заплаты» поставляются в виде блоков данных размером 2048 байт и никакого исполняемого кода не содержат. Блок состоит из 48-байтного заголовка (номер версии заголовка указывается в самом его начале) и собственно данных обновления (2000 байт). Целостность всего блока проверяется контрольной суммой. Поле версии обновления позволяет определить, загружено ли данное обновление в процессор. Блок обновления приемлем только для процессора, имеющего тот же идентификатор (тип, семейство, модель, степпинг), который указан в заголовке блока. Обновление должно загружаться загрузчиком, версия которого соответствует указанной в заголовке. Загрузчик версии 1 (другие пока не описаны) просто записывает линейный адрес данных (адрес блока +48) в MSR 79h. Обновление может выполняться многократно без каких-либо побочных эффектов. Успешность и версию произведенного обновления можно проверить программно, не изменяя состояния обновления. Попытка загрузить обновление с идентификатором, не соответствующим данному процессору, не удастся (процессор проигнорирует эту попытку). В мультипроцессорных системах обновляемые данные для каждого процессора должны соответствовать его типу, модели, степпингу и идентификатору платформы.

Режим системного управления

Современные модели 32-битных процессоров (начиная с некоторых модификаций 486 и 386SL) помимо обычных режимов — реального, защищенного и V86 — имеют дополнительный режим системного управления (System Management Mode, SMM). Этот режим предназначен для выполнения ряда действий с полной изоляцией их от прикладного программного обеспечения и даже от операционной системы. Главным образом этот режим служит для реализации системы управления энергопотреблением, хотя может использоваться и в иных целях.
В режим SMM процессор может войти только по сигналу на входе SMI# (System Management Interrupt), процессоры P5+ могут войти в режим SMM и по приему соответствующего сообщения по шине APIC. Сигнал SMI# для процессора является немаскируемым прерыванием с наивысшим приоритетом. Обнаружив активный сигнал SMI# (низкий уровень), процессор по завершении текущей инструкции и выгрузке буферов записи переключается в режим SMM, о чем сигнализирует своим интерфейсом. Сразу при входе в SMM процессор сохраняет свой контекст — почти все регистры — в специальной памяти *SMRAM* (не в стеке!). Она представляет собой выделенную область физической памяти, доступ к которой разрешается внешними (по отношению к процессору) схемами в шинных циклах обращения к памяти только при сигнализации процессором о нахождении в режиме SMM. После сохранения контекста процессор переходит к *обработке SMI* — соответствующий обработчик расположен в той же памяти SMRAM. Обработчик представляет собой последовательность обычных инструкций, исполняемых процессором в режиме, напоминающем реальный. При входе в режим SMM автоматически запрещаются аппаратные прерывания

(включая немаскируемые) и не генерируются исключения, так что действия процессора однозначно определяются программой обработчика SMI. Процедура обработки завершается инструкцией RSM, по которой процессор восстанавливает свой контекст из образа, хранившегося в SMRAM, и возвращается в обычный режим работы.

При возврате из режима SMM возможны некоторые варианты в зависимости от действий обработчика: он может программно внести изменения в образ контекста процессора, а также запросить рестарт (повторное исполнение) инструкции, предшествующей появлению сигнала SMI#.

Возможность рестарта инструкции ввода-вывода расширяет режим SMM, позволяя организовать на уровне BIOS управление энергопотреблением совершенно прозрачно для прикладного ПО и ОС. Прозрачность SMM обеспечивается следующими свойствами этого режима:

- возможностью только аппаратного входа в SMM;
- исполнением кода SMM в отдельном адресном пространстве;
- полным сохранением состояния прерванной программы в области SMRAM;
- запретом обычных прерываний;
- восстановлением состояния прерванной задачи по выходу из режима SMM.

Память SMRAM должна быть физически или логически выделенной областью размером от 32 Кбайт (минимальные потребности SMM) до 4 Гбайт. В ней выделяются область сохранения контекста и точка входа в обработчик SMI; кроме того, имеется свободная область.

Карта сохранения контекста имеет «официальную» часть, в которой находятся образы архитектурных (видимых) регистров, и непубликуемую часть (зарезервированные поля). Неосторожная модификация полей, запрещенных для записи, может привести к непредсказуемым (для непосвященных) результатам. В зарезервированных полях скрывается автоматически сохраняемый управляющий регистр CR4 и программно невидимые (скрытые) регистры дескрипторов для всех сегментов, но местоположение и формат их образа зависят от модели процессора.

Если режим SMM используется для отключения питания процессора с возможностью быстрого «пробуждения», память SMRAM, хранящая контекст процессора, должна быть энергонезависимой и схемотехнически защищенной от доступа прикладных программ. Если память SMRAM не является энергонезависимой, то системная логика должна обеспечивать возможность ее инициализации (записи программного кода обработчика) процессором из обычного режима работы до появления сигнала SMI#.

Управление энергопотреблением и производительностью

Современные процессоры потребляют значительную мощность (десятки ватт), которая, естественно, выделяется в виде тепла. Мощность (и тепловыделение)

растут с повышением тактовой частоты. Если от процессора не требуется максимально возможная производительность, его можно «притормозить» для снижения потребления. Минимальную мощность процессор потребляет при остановленном тактовом генераторе, при этом он не выполняет никаких функций, а последующая подача синхронизации должна сопровождаться сигналом аппаратного сброса RESET. Схемы внутреннего умножения требуют стабильности внешней частоты во время работы процессора, так что для временного снижения потребления приходится использовать специальные механизмы.

Синхронизация

Синхронизация процессоров осуществляется внешним сигналом, определяющим *частоту системной шины* (FSB clock). Ядро процессора синхронизируется с помощью умножителя частоты, выполненного по схеме генератора с фазовой автоподстройкой частоты. Схема фазовой автоподстройки поддерживает заданное соотношение частоты внутреннего генератора и внешнего сигнала BCLK. *Умножение частоты* в процессоре выполняется с помощью внутреннего управляемого генератора, включенного в контур системы фазовой автоподстройки частоты (ФАПЧ). Структурная схема умножителя приведена на рис. 7.8. Контур фазовой автоподстройки (Phase Lock Loop, PLL) в установившемся режиме обеспечивает нулевой фазовый сдвиг (а следовательно, и совпадение частот) на входах фазового детектора. Эти сигналы образуются путем деления частот F_{BUS} (частота шины) и F_{CORE} (частота ядра) на соответствующие целочисленные коэффициенты, следовательно, в стационарном режиме выполняется условие:

$$F_{CORE}/k_1 = F_{BUS}/k_2 \text{ или } F_{CORE} = F_{BUS} \times k_1/k_2$$

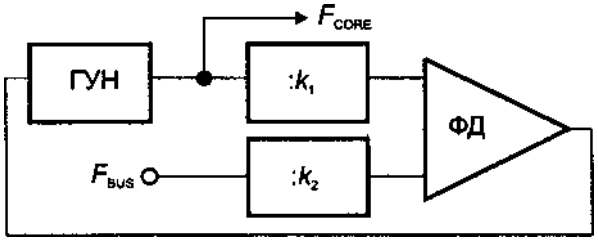


Рис. 7.8. Схема умножения частоты (ГУН — генератор, управляемый напряжением; ФД — фазовый детектор)

Целочисленное деление частоты обеспечивается триггерами или управляемыми счетчиками. Нулевой сдвиг фаз, необходимый для точной привязки внутренних тактов к синхронизации внешней шины, обеспечивается системой автоматического регулирования с астатизмом. Это подразумевает некоторую инерционность — при изменении частоты F_{BUS} на время переходного процесса условия синхронизации выполняться не будут. Точно так же переходный процесс возникает при смене коэффициентов. Поскольку поведение процессора, у которого внутреннее тактирование не привязано к синхронизации внешней шины, непредсказуемо, смена внешней частоты и коэффициентов деления допустима только во время аппаратного сброса. После установки параметров син-

хронизации аппаратный сброс должен удерживаться еще некоторое время, за которое переходный процесс завершится и установится нулевой сдвиг фаз.
Коэффициент умножения частоты может быть фиксирован (на конечном этапе изготовления процессора), может задаваться уровнями сигналов на определенных входах процессора во время действия сигнала RESET#, а может изменяться динамически (для управления энергопотреблением) посредством записи в специальные модельно-специфические регистры. Диапазон частот системной шины, при котором обеспечивается захват в системе ФАПЧ, ограничен и задается в спецификациях на процессоры. Частота системной шины задается выходными сигналами BSEL[1:0] процессора или иным путем.

Энергопотребление

Энергопотребление процессора зависит от напряжения питания, тактовой частоты и режима работы (состояния потребления) процессора.
Напряжение питания ядра у современных процессоров задается сигналами VID[6:0], с помощью которых процессор управляет регулятором напряжения (непосредственно или с участием BIOS и чипсета). Эти сигналы задают 4- 6-битный идентификатор уровня напряжения (VID), трактовка которого (соответствие напряжения коду) зависит от типа сокета (слота) для установки процессора. Ряд процессоров имеют отдельные идентификаторы питания кэша L2 (*CVID*).
На рис. 7.9 приведены состояния, в которых могут находиться современные процессоры (с точки зрения потребления). Названия состояний соответствуют терминологии Intel; в процессорах AMD состояния те же, но они несколько иначе называются.

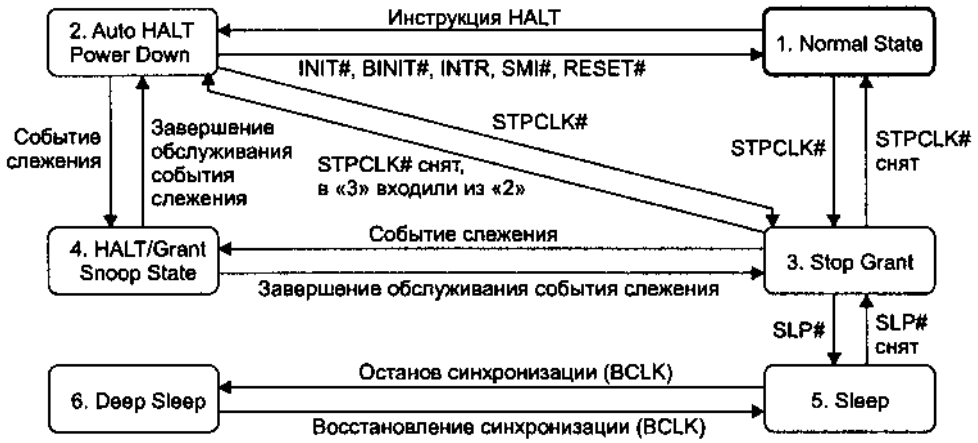


Рис. 7.9. Диаграмма переходов режимов пониженного энергопотребления

В нормальном состоянии *(normal state)* процессор выполняет все свои функции; потребление максимальное.

При исполнении инструкции HALT (останов) процессор переходит в состояние пониженного энергопотребления *Auto HALT PowerDown*. В этом состоянии процессор реагирует на все внешние прерывания (включая сигналы SMI#, INIT#), переходя по ним в нормальное состояние. По выходе из прерывания SMI процессор может остаться в нормальном состоянии или вернуться в состояние *Auto HALT PowerDown.* В данном состоянии процессор наблюдает транзакции на системной шине, отслеживая кэш-попадания для обеспечения когерентности кэшей и памяти.

По сигналу STPCLK# процессор выгружает буферы записи и входит в режим *Stop Grant*, в котором прекращается тактирование большинства узлов процессора, что вызывает снижение энергопотребления примерно в 10 раз. В этом состоянии он прекращает исполнение инструкций и не обслуживает прерывания (а только запоминает их), однако продолжает слежение за шиной, отслеживая кэш-попадания. Запомненные прерывания он обработает позже, когда выйдет в нормальный режим. Из состояния *Stop Grant* процессор выходит по снятии сигнала STPCLK#. Управление сигналом STPCLK# совместно с использованием режима SMM реализует механизм расширенного управления энергопотреблением (Advanced Power Management, APM). При отсутствии активности внешняя схема (чипсет) по команде, исполненной в режиме SMM, устанавливает данный сигнал. По пробуждающему событию внешняя схема (без участия процессора, который «спит») снимает сигнал, и процессор продолжает работу. Кроме того, с помощью сигнала STPCLK# возможно *замедление процессора* (с пропорциональным снижением потребляемой мощности), если на этот вход подавать периодический импульсный сигнал. Скважность импульсов будет определять коэффициент простоя процессора и, следовательно, его производительность (это как бы эквивалентно снижению условной тактовой частоты). В современных процессорах можно заказывать уровень производительности (и потребления) с помощью регистра IA__32__CLOCK__MODULATION MSR, который управляет внутренней приостановкой синхронизации. С его помощью осуществляется «заказ» *(On-Demand Mode*) коэффициента простоя в диапазоне от 12,5 до 87,5 % с шагом 12,5 %.

В состояние *HALT/Grant Snoop* процессор временно попадает из состояния *Stop-Grant* или *AutoHALT Power-Down* по одному из двух событий, обнаруженных на системной шине. По *транзакции сигнализации прерывания* (см. IOA-PlCx в 4.4) процессор переходит в состояние *HALT/Grant Snoop* и возвращается из него, когда «защелкнет» сообщение о прерывании. По *событию слежения* (транзакции к памяти, требующей действий по обеспечению когерентности кэша) процессор переходит в состояние *HALT/Grant Snoop* и возвращается из него, когда событие слежения будет обслужено (неважно чьими силами — данного процессора или сторонними).

В состояние *Sleep* (спящий режим) процессор можно переводить только из состояния *Stop-Grant* (сигналом SLP#). В состоянии *Sleep* процессор не тактирует свои внутренние узлы (кроме схемы умножителя частоты), но сохраняет состояние контекста (всех регистров). Прерывания и циклы слежения не воспринимаются. Процессор может реагировать только на сигналы SLP#, STPCLK#

и RESET#. По снятии сигнала SLP# процессор возвращается в состояние *Stop Grant* и возобновляет тактирование своего блока управления шиной и APIC.

В состоянии «глубокого сна» *Deep Sleep* у процессора останавливается тактирование, он не выполняет никаких функций. При этом он способен сохранять контекст, но внешние схемы не имеют права изменять состояние никаких входных сигналов процессора (результат смены непредсказуем). В состояние *Deep Sleep* процессор переходит при остановке тактового сигнала на входе BCLK; у новых процессоров есть специальные входы DPSLP# и CPLLSTP#, по котором выключаются внутренний генератор и ФАПЧ. Возврат из данного состояния в *Sleep* занимает насколько десятков микросекунд — это время, необходимое для установления режима цепей ФАПЧ внутреннего генератора.

Состояние «глубочайшего сна» *Deeper Sleep* функционально аналогично глубокому сну (поэтому на рисунке не показано), но дополнительное снижение потребления обеспечивается понижением напряжения питания ядра. Сигнал к снижению напряжения обеспечивает не процессор, а платформа (системная плата).

Мобильные процессоры Pentium M сигнализируют о переходе в состояние сна сигналом PSI# (Power Status Indicator), использование этого сигнала повышает эффективность работы (и потери) регулятора напряжения.

В современных процессорах применяются различные технологии энергосбережения, особо актуальные для мобильных компьютеров.

Технология *Intel SpeedStep* позволяет динамически управлять потребляемой мощностью (и производительностью) за счет снижения тактовой частоты ядра с одновременным снижением напряжения питания. Для процессора определяются две точки: нормальной (максимальной) производительности при номинальном питании и пониженной производительности. Переключение осуществляется по входному сигналу GHI#, по которому процессор изменяет коэффициент умножения частоты.

Улучшенная технология *Enhanced Intel SpeedStep* (EIST) развивает эту идею: вместо двух крайних точек появляется возможность использования нескольких градаций производительности (и потребления). Напряжением питания и частотой ядра можно управлять программно записью в специальные регистры (MSR) — это снимает зависимость от чипсета. Согласованность изменения коэффициента умножения и идентификатора напряжения питания (VID), подаваемого на соответствующие выводы, поддерживает процессор. Программное изменение режима воспринимается в любой момент времени, а если к этому моменту предыдущий переход не завершен, то новое изменение на время откладывается. При изменении частоты процессорное ядро (и кэш L2) временно (до 10 мкс) оказываются недоступными. Для того чтобы не потерять когерентность памяти, на время переходного процесса слежение блокируется (сигналом BNR#). Во время перехода не требуется запрета работы арбитра шины, а также выгрузки кэша процессора. Если внутренний термодатчик обнаруживает слишком высокую температуру процессора, снижение частоты и напряжения выполняется автоматически (thermal monitor 2, см. далее).

Для процессоров мобильных компьютеров применяется ряд дополнительных мер по снижению потребления: динамическое управление питанием системной шины, динамическое отключение терминаторов (они потребляют значительную мощность), пониженное напряжение питания терминаторов.

Средства управления энергопотреблением у фирмы Intel появились с процессорами Pentium 2-го поколения, где реализованы состояния 1-4 (см. рис. 7.9). Процессоры Pentium II/III и Celeron имеют дополнительные состояния 5 и 6, состояние 7 (Deeper Sleep) появилось позже (в процессоре Pentium 4 и его вариантах). Технология EIST, особенно актуальная для мобильных применений, в самых высокопроизводительных процессорах не используется (чтобы не терять драгоценного времени).

Фирма AMD для своих процессоров применяет аналогичные методы энергосбережения. Некоторую специфику вносят интерфейсы процессора: системная шина EV6 или HyperTransport в комплекте с интерфейсом памяти DDR SDRAM, но общие идеи те же. Технология AMD PowerNow! обеспечивает «производительность по потребности» — то же управление частотой и напряжением питания, которое используется у Intel в IST и EIST.

Термоконтроль

Современные процессоры имеют встроенные средства термоконтроля, предохраняющие их от выхода из строя в случае перегрева. В процессорах Intel термоконтроль реализуется несколькими способами, разными в разных моделях процессоров.

Термомонитор (thermal monitor) — это встроенные в процессор средства TCC (Thermal Control Circuit), которые при достижении критической температуры включают модуляцию (прерывание) тактирования ядра. Тактирование включается на 30-50 % времени каждого цикла, длительность выключенного состояния тактирования не превышает 3 мкс. Трафик шины отслеживается, прерывания «защелкиваются», но обслуживание откладывается до ближайшего включения тактирования. Когда температура снижается до нижнего порога, восстанавливается нормальный режим. Термомонитор (если присутствует) может быть программно разрешен или запрещен; пороги температур пользователем не конфигурируемы и не наблюдаемы. Те же средства TCC могут быть активированы программно (On-Demand mode, см. ранее).

ВНИМАНИЕ ----------

Автоматический термомонитор включается и выключается через BIOS. Отключение термомонитора может быть чревато перегревом и выходом процессора из строя.

Термомонитор 2 (thermal monitor 2) — средство аналогичного назначения, но для снижения потребления процессор снижает коэффициент умножения частоты и меняет VID. Для процессора определено две точки — нормальных и пониженных частоты и напряжения, переход выполняется быстро (в пределах

4 мкс), но в процессе смены частоты ядро и кэш для внешних транзакций недоступны.
Об активации цепей термоконтроля TCC, вызванной термомонитором, процессор сообщает сигналом PROCHOT# (возможна и генерация прерывания по этому событию).
Сигналом FORCEPR# платформа (системная плата и BIOS) может принудительно активировать средства TCC процессора (на все время активности сигнала), если средства термоконтроля разрешены. Это может потребоваться для защиты иных компонентов (например, по перегреву регулятора напряжения можно заставить процессор снизить потребление).
Независимо от разрешения термоконтроля процессор автоматически выключится (аварийно), если температура достигнет критической точки. Об этом он сообщит сигналом THERMTRIP#.
Для измерения температуры кристалла в процессоры встраивается *термодиод*. Чипсет измеряет падение напряжения на диоде и по нему определяет темпера- туру. Это определение не очень точное, поскольку у термодиодов характеристики имеют значительный разброс. В дорогих процессорах (Xeon) используется энергонезависимая память (PIROM), в которую заносится калибровочная характеристика термодиода. У этих процессоров температуру можно и «спросить» по шине SMBUS. Термодиод, доступный по внешним выводам процессора, не связан с датчиком внутреннего термоконтроля, так что значения температуры, полученные с его помощью, нельзя использовать для определения состояния TCC.

7.4. Мультипроцессорные и избыточные системы

В современных ПК встречаются варианты установки нескольких (двух или более) процессоров на одной системной шине. При этом возможны конфигурации с *симметричной мультипроцессорной обработкой* (Symmetric Multi-Processing, SMP) и *избыточным контролем функционирования* (Functional Redundancy Checking, FRC).
В конфигурации с избыточным контролем функционирования два процессора (пара Master и Checker) выступают как один логический. Основной процессор (Master) работает в обычном однопроцессорном режиме. Проверочный процессор (Checker) выполняет все те же операции «про себя», не управляя шиной, и сравнивает выходные сигналы основного (проверяемого) процессора с теми сигналами, которые он генерирует сам, выполняя те же операции без выхода на шину. В случае расхождения вырабатывается сигнал ошибки, который может обрабатываться как прерывание. FRC-контроль применяют только в особо ответственных системах. Поддержка FRC появилась, начиная с процессоров Intel Pentium, но не во всех последующих моделях; она имеется и у процессоров фирмы AMD.

Симметричные мультипроцессорные системы

Наиболее популярной целью объединения процессоров является *симметричная мультипроцессорная обработка (*SMP). В системе SMP каждый процессор решает свою задачу, порученную ему операционной системой. В документе Intel «Multiprocessor Specification» (MPS) симметрия рассматривается в двух аспектах:

- симметрия памяти — все процессоры пользуются общей памятью, работают с одной копией ОС;
- симметрия ввода-вывода — все процессоры разделяют общие устройства ввода-вывода и общие контроллеры прерываний.

Система может быть симметричной по памяти, но асимметричной по прерываниям от ввода-вывода, если для обслуживания этих прерываний выделяется собственный процессор. В x86 симметрию по прерываниям обеспечивают контроллеры APIC (см. 4.4). Аппаратная (физическая) реализация SMP может быть различной, здесь уже есть своя история:

- объединение нескольких физических процессоров на одной локальной шине — процессоры Pentium, P6, Pentium 4;
- подключение каждого процессора к системной плате (с общей памятью) выделенными шинами — процессоры Athlon;
- подключение к каждому процессору собственного ОЗУ и их объединение с периферийными устройствами через HyperTransport — процессоры Athlon-64, Opteron;
- размещение на одном кристалле нескольких логических процессоров с разделяемыми операционными блоками — «Гиперпотоковые» (hyperthreading) модели Pentium 4;
- размещение на одном кристалле нескольких независимых процессорных ядер с разделяемым вторичным кэшем — мультиядерные модели Pentium 4.

Применение SMP требует поддержки со стороны BIOS, ОС и приложений (чтобы работать быстрее, они должны быть многопоточными). Поддержку SMP имеют такие ОС, как Novell NetWare, Microsoft Windows NT/2000x/XP и различные диалекты Unix. Цена мультипроцессорных версий ОС, как правило, значительно выше цены соответствующих однопроцессорных версий. Поначалу это становилось препятствием к применению гиперпотоковых (и мультиядерных) процессоров в обычных системных платах. Теперь достигнута договоренность

о том, что число процессоров, на которое лицензируется ОС, соответствует числу физических (отдельно покупаемых и устанавливаемых) процессоров. Это открывает возможности широкого распараллеливания на уровне процессоров (ускорение работы CISC-процессоров x86 за счет микроархитектуры обходится все дороже).

Объединение процессоров на локальной шине

В первых мультипроцессорных системах на базе Pentium и P6 процессоры объединялись с помощью общей локальной («фасадной», FSB) шины, через кото

рую они связывались с чипсетом системной платы. В таких системах, в принципе, могут использоваться процессоры одной модели, но разного степпинга. Частоты ядра у них должны совпадать (внешняя частота у них, естественно, едина). Процессоры, объединенные общей локальной шиной, разделяют ресурсы и этой шины, и компьютера (память и периферийные устройства). В каждый момент времени шиной может управлять только один процессор, по определенным правилам они меняются ролями.

Поскольку каждый из процессоров имеет свой внутренний первичный кэш, интерфейс шины обязан поддерживать *согласованность данных* (когерентность) во всех иерархических ступенях оперативной памяти — в первичном и вторичном кэшах, в основной памяти (в Pentium вторичный кэш у процессоров общий). Эта задача решается с помощью локальных циклов слежения, воспринимаемых процессором, даже не управляющим шиной в данный момент.

Симметричные системы имеют специальные механизмы *арбитража* доступа к локальной шине. Процессор — текущий владелец шины — отдает управление шиной другому процессору по его запросу только по завершении операции.

Интерфейс *Pentium* (начиная со второго поколения) позволяет на одной локальной системной шине устанавливать два процессора, при этом почти все их одноименные выводы просто объединяются. Роль конкретного процессора в системе фиксирована — она определяется внешними сигналами во время спада сигнала RESET. Один из процессоров назначается *первичным* (primary processor), или *загрузочным* (Bootstrap Processor, BSP), другой — *вторичным* (Dual Processor, DP). После сигнала RESET сразу начинает функционировать только первичный процессор (BSP), выполняя программный код инициализации. Вторичный процессор начинает функционирование только после приема соответствующего сообщения по шине APIC, посланного в ходе программы инициализации.

В *процессорах P6* заложены более развитые возможности SMP. Системная шина P6, в отличие от локальной шины Pentium, изначально ориентирована на разделяемое управление множеством симметричных (до четырех на шине) и несимметричных (до восьми) агентов. Сокет 8 (Pentium Pro) и слот 2 (Pentium II Xeon) позволяют объединять до четырех процессоров, слот 1 (Pentium II) допускает объединение не более двух процессоров. Какой из процессоров станет первичным, определяется по загрузочному протоколу — здесь нет жесткой аппаратной привязки роли процессора к его «географическому» адресу. Это позволяет повысить надежность системы SMP, поскольку любой процессор может без механического вмешательства во время инициализации взять на себя роль BSP. Протокол мультипроцессорной инициализации работает на шине APIC, он позволяет управлять инициализацией до 15 процессоров. Процессоры могут пользоваться содержимым «чужого» кэша без его предварительной выгрузки в основную память.

Процессоры Celeron официально предназначены лишь для однопроцессорных конфигураций. Однако реально у процессоров на ядре Mendocino имеется сигнал BR1#, требующийся для поддержания SMP, — правда, не там, где хотелось бы. Если использовать такой процессор в корпусе PGA с переходником в слот 1

(для двухпроцессорной системной платы), то на переходнике достаточно организовать связь контакта B75 краевого разъема с контактом AN15 процессорного сокета. Есть и переходники, подготовленные к дуальному использованию Celeron самим изготовителем — на них имеется специальный джампер для штатного и нештатного назначения контакта B75. На процессорах в картриджах SEPP умельцы ухитряются освобождать контакт B75 и соединять его с выводом сигнала BR1# кристалла ядра. Таким образом удается использовать Celeron и в двухпроцессорных системах, что привлекательно, учитывая относительную дешевизну этого процессора. Однако Celeron на ядре Coppermine симметричную мультипроцессорную обработку уже не поддерживает (как и некоторые модели Pentium III Coppermine).

Гиперпотоковые и мультиядерные процессоры

В процессорах Pentium 4 (начиная с частоты 3,06 ГГц) и Xeon применяется *гиперпотоковая* (hyperthreading) технология: один физический процессор одновременно может выполнять два потока инструкций x86. Для фон-неймановской машины это означает, что физический процессор (микросхема, устанавливаемая в сокет) имеет два комплекта архитектурных (прикладных и системных) регистров. В каждом комплекте имеется, естественно, свой указатель инструкций, «идущий» по своему потоку. Таким образом, речь идет о двух *логических процессорах*, физически расположенных на одном кристалле микросхемы. Эти логические процессоры совместно используют ряд общих микроархитектурных блоков физического процессора (вторичный кэш, исполнительные блоки арифметико-логического устройства). Такое разделение позволяет повысить эффективность функционирования исполнительных блоков: один даже «гиперконвейер» Pentium 4, исполняющий инструкции x86, плотно загрузить их работой не может. Конечно, логические процессоры не являются полностью независимыми — иногда приходится ожидать освобождения ресурса, занятого соседом.

Следующий шаг в этом направлении — *мультиядерные процессоры*, в которых на одном кристалле объединены общей шиной несколько функциональнозаконченных процессоров. Структурная схема мультиядерных процессоров Pentium 4 приведена на рис. 7.10. В них каждое ядро обладает собственным кэшем L2 (и, естественно, кэшем данных L1 и кэшем трасс). Размер кэша L2 каждого ядра может достигать 1-2 Мбайт. Интерфейс системной шины у двух ядер может быть как общим, так и раздельным. Двухъядерный процессор, предназначенный для использования в относительно небольших системах (размером до четырех ядер), фактически, является двумя отдельными процессорами, каждый со своим интерфейсным блоком (рис. 7.10, а). Их интерфейсные сигналы объединяются в общую системную шину на системной плате. Процессоры для более крупных систем содержат общий блок интерфейса системной шины (рис. 7.10, б), поскольку объединение на одной шине интерфейсов более чем двух процессоров для высокоскоростных шин вызывает ряд трудностей.

Заметим, что мультипроцессорные кристаллы в RISC-архитектурах начали применять намного раньше, причем в более мощных вариантах (с интегрированным контроллером памяти).

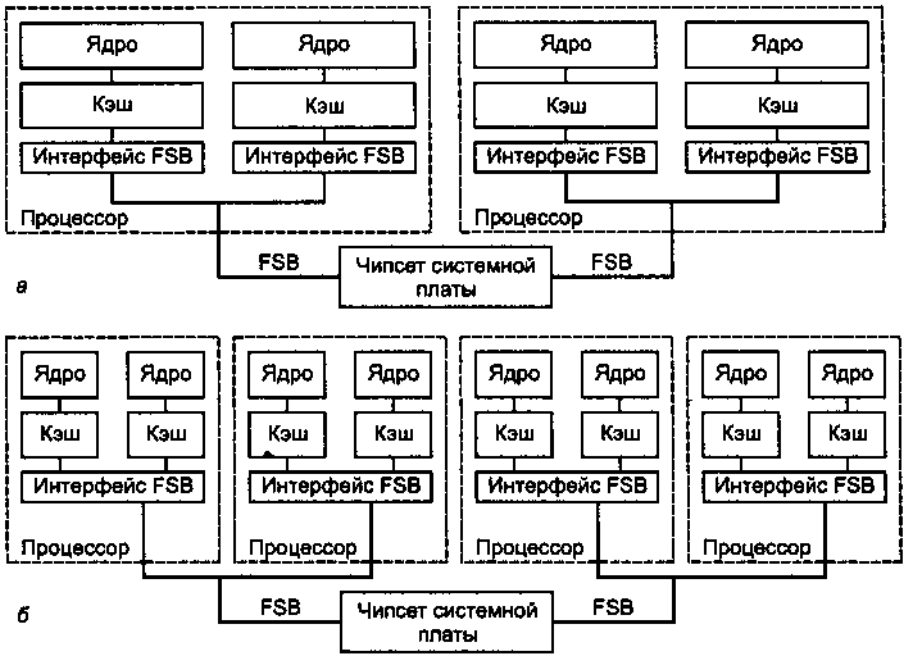


Рис. 7.10. Структура мультиядерных процессоров Pentium 4: а — с раздельными интерфейсами системной шины, *б* — с объединенным интерфейсом

Мультипроцессорные системы Athlon и Opteron

Фирма AMD долгое время для своих процессоров x86 механизм SMP не поддерживала, ограничиваясь поддержкой FRC (избыточный контроль). Первым из процессоров AMD, поддерживающих SMP, стал процессор Athlon. Использованная в нем шина EV6 является двухточечной, и для мультипроцессорных систем чипсет должен каждому процессору предоставлять собственную шину в монопольное владение. Таким образом, средством объединения нескольких процессоров является северный хаб чипсета системной платы. Основным разделяемым ресурсом становится контроллер памяти, расположенный в чипсете.

В процессорах Athlon-64 и Opteron (ядро Hammer) подход к построению системы (даже однопроцессорной) иной: каждый процессор имеет собственный контроллер памяти (DDR SDRAM) с отдельной шиной, к которой непосредственно подключаются модули памяти. У Athlon-64 шина данных памяти 64-битная, у Opteron — 128-битная (двухканальная память). Такое решение позволяет обеспечить минимальные задержки доступа к памяти — они могут быть порядка 45 нс. Для сравнения заметим, что при «классическом» подключении памяти к чипсету задержка доступа со стороны процессора составляет около 100 нс. Для связи с остальными компонентами (периферией и другими процессорами) используется высокопроизводительный интерфейс HyperTransport. У процессора Opteron, предназначенного для серверов (1-8-процессорных систем) и рабочих станций (1-4-процессорных систем), имеется три 16-битных интерфейса

HyperTransport с суммарной пропускной способностью 3 x 6,4 = 19,2 Гбайт/с. HyperTransport обеспечивает «прозрачный» доступ к любым компонентам из любой точки системы транзакциями чтения и записи по адресам памяти или ввода-вывода. Некоторые варианты систем SMP на базе Opteron приведены на рис. 7.11. Здесь каждый процессор входит в систему со своей памятью, ему доступна память и других процессоров. Конечно, время доступа к «чужой» памяти несколько больше, чем к своей; оно зависит от числа *хопов* (hop) через интерфейс HyperTransport. Так, для 4-процессорной системы[1] время доступа при локальном обращении (нулевое число хопов) составляет 100 нс, при обращении к памяти ближайшего соседа (один хоп) — 118 нс, к самой дальней памяти (два хопа) — 136 нс. Обращение к локальной памяти по сравнению с однопроцессорным вариантом происходит несколько дольше — очевидно, это издержки, Вносимые протоколом обеспечения когерентности кэш-памяти разных процессоров.

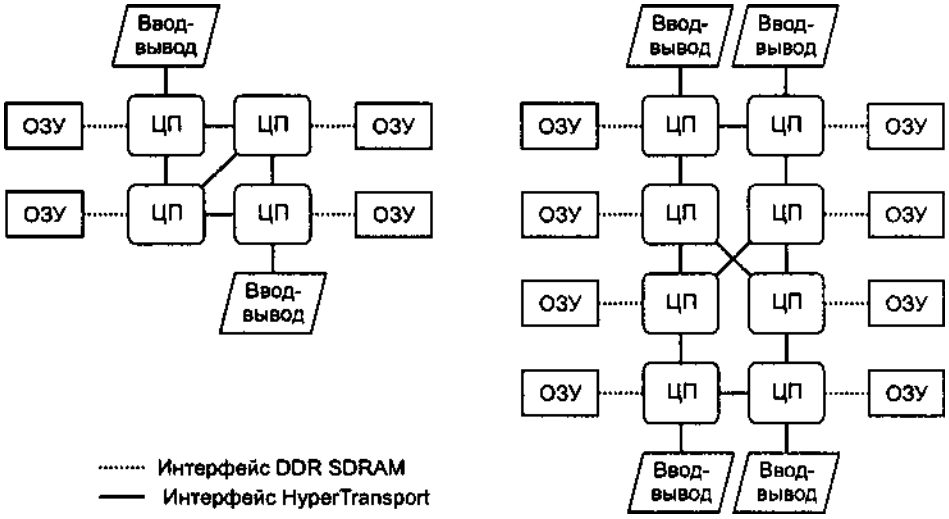


Рис. 7.11. Структуры мультипроцессорных систем Opteron

Варианты использования вышеприведенных топологий соединения могут быть различными. Это может быть и симметричная по памяти система; правда, при распределении памяти и исполняемых потоков по процессорам желательно, чтобы ОС учитывала разницу во времени доступа к «близкой» и «дальней» памяти. Возможны построения и несимметричных (по памяти) систем. Если у объединяемых процессоров организовать раздельное адресное пространство, то получается уже *мультикомпьютерная система*. Каждый компьютер — это процессор, память и средства ввода-вывода, и при совместной параллельной работе они могут обмениваться лишь сообщениями (но не по сети, а через интерфейс HyperTransport).

[1] Opteron 2 ГГц, двухканальная память DDR SDRAM 333 МГц.

7.6. Совместимость и идентификация процессоров

Как уже говорилось, во всех IBM PC-совместимых компьютерах применяются процессоры семейства x86 как от фирмы Intel, так и от других производителей. Семейство PC-совместимых компьютеров живет долго благодаря совместимости программного обеспечения с различными процессорами. Однако программам (и пользователям) отнюдь не все равно, на каком процессоре работать. Далее рассматриваются некоторые вопросы совместимости, способы идентификации, а также сравнительные характеристики распространенных процессоров.

Совместимость процессоров

Для всех процессоров, применяемых в PC, характерна принадлежность к единой архитектурной линии *Intel Architecture* (*IA-32*), в которой процессоры следующих моделей вбирают в себя все свойства и инструкции предыдущих. Правда, некоторые инструкции объявлены не архитектурными, а зависящими от модели, и их наличие и функционирование в следующих моделях не гарантируется. Состав регистров и флагов по мере «взросления» процессоров постоянно расширяется. Для *совместимости программного обеспечения*, написанного для ранних моделей процессоров, со следующими моделями следует осторожно обращаться с неиспользуемыми (зарезервированными) битами и регистрами:

- не изменять значения битов, не используемых в данном процессоре;
- гарантировать нечувствительность программ к значению этих битов;
- при загрузке регистров в зарезервированные биты записывать нули;
- не пытаться использовать эти биты для хранения каких-либо признаков.

Декларируемая обратная совместимость новых моделей с предыдущими означает, что программный код, написанный для процессора 8088, должен таким же образом исполняться и на 80386, и на Pentium 4, и на любых других совместимых процессорах. В большинстве случаев так это и происходит, но программы на более новых процессорах исполняются, естественно, быстрее. Здесь и кроется один из подводных камней совместимости. Дело в том, что большинство программ для PC не только выполняет вычисления, но и управляет различным внутренним и внешним оборудованием. При этом оборудование требует определенной последовательности действий и соблюдения временных характеристик. Устройства с невысоким быстродействием не могут, например, воспринимать последовательные обращения к ним, идущие в соседних тактах системной шины ввода-вывода. Программные способы организации задержек должны опираться на сведения о модели (и если есть возможность, о тактовой частоте) процессора, на котором исполняется код. Иначе возможны перекосы в обе стороны, которые могут приводить к разнообразным неприятным эффектам.

Для введения программных задержек при обращении к портам ввода-вывода в BIOS компьютеров на процессорах 8088, 80286 и 80386 использовалась команда короткого безусловного перехода (JMP SHORT) на следующий адрес. Эта ко

манда сбрасывала конвейер (очередь декодированных инструкций), и процессор был вынужден снова делать выборку кода операции из памяти, а в это время порт «переводил дух» перед следующим обращением. Такой способ задержки применялся не только в BIOS (с привязкой к конкретному поколению процессора), но и в загружаемых программах.

Однако этот способ введения программной задержки для процессоров, имеющих внутренний кэш (то есть 486 и выше, а также некоторые модели 80386), непригоден. Здесь команда JMP, ранее безусловно приводившая к генерации внешнего цикла обращения к памяти, скорее всего, будет обслужена из внутреннего кэша и желаемой задержки не произойдет. Одним из способов введения внешнего цикла шины между циклами вывода является явная операция чтения некэшируемой области памяти. Эта операция в процессорах с упорядоченным обслуживанием инструкций выполняется только после завершения предыдущего цикла вывода, а последующая операция вывода начинается только по завершении этого чтения.

В процессорах шестого поколения возможно изменение порядка выполнения операций, и чтение памяти может обогнать другие операции на внешней шине. Если порядок операций, включая чтение памяти, имеет существенное значение, имеется возможность сериализовать выполнения операций. *Сериализация* означает, что все модификации флагов, регистров и памяти, выполняемые предыдущими инструкциями, должны завершиться до выборки из памяти и исполнения последующей инструкции. При этом очищается очередь предварительно выбранных инструкций. Инструкция CPUID позволяет выполнять сериализацию на любом уровне привилегий.

Помимо быстродействия процессоры отличаются и некоторыми особенностями в обработке инструкций, которые обычно не влияют на выполнение программ, но могут использоваться для идентификации процессоров. Так, например, инструкция PUSH SP на процессоре 8086/88 исполняется иначе, чем на 80286 и более поздних версиях, — различие касается порядка выполнения декремента указателя стека и его сохранения в стеке.

Процессоры последних поколений имеют архитектурные расширения (MMX, SSE, SSE2, SSE3, 3DNow!), полагаться на которые без предварительной идентификации типа процессора весьма рискованно. Определив возможности процессора, программа может эффективно использовать поддерживаемые им расширения архитектуры. Путем программирования регистров MSR можно управлять расширениями архитектуры, естественно, только в сторону отключения имеющихся возможностей.

Идентификация процессоров

Возможность программного определения типа процессора была заложена в архитектуру процессоров x86 с самого начала. В любом процессоре IA-32 сразу после аппаратного сброса в регистре (E)DX можно прочитать *сигнатуру процессора,* дающую общую информацию о процессоре. Начиная с процессоров Pentium (а у AMD — еще с 486-х) появилась инструкция CPUID, по которой любая

программа на любом уровне привилегий в любой момент времени может получить не только сигнатуру, но и дополнительную информацию. Формат инструкции практически безгранично расширяем, с ее помощью процессор может выдать практически весь свой «словесный портрет» (если эту возможность заложат его разработчики). Вызывая инструкцию CPUID, в регистре EAX указывают номер основной (начиная с 0) или расширенной (8000 0000h и выше) функции.

Основные функции инструкции CPUID позволяют получать следующую информацию о процессоре:

- CPUID(O) — производитель процессора (Intel, AMD, Cyrix...) и максимальный номер поддерживаемой основной функции.
- CPUID(l) — сигнатура процессора, включая тип, семейство, модель и степпинг (в EAX); бренд-индекс, начальный идентификатор APIC, размер строки, очищаемой инструкцией CFLUSH (в EBX); набор флагов расширений базовой архитектуры (табл. 7.2), реализованных в данном процессоре (в EDX и ECX). Для процессоров AMD ряд флагов расширений трактуется иначе, их определяют по расширенной инструкции CPUID (8000 0001h).
- CPUID(2) — параметры внутренней системы кэширования: размеры кэш-памяти инструкций, данных и таблиц TLB, их организация, длина строки кэша и т. п. (только для процессоров Intel).
- CPUID(3) — получение *серийного номера процессора* (processor serial number), доступно только в Pentium III некоторых моделей.
- CPUID(4) — дополнительные параметры кэширования.
- CPUID(5) — параметры для инструкций MONITOR/MWAIT.

Таблица 7.2. Расширения базовой архитектуры по CPUID(1)

Бит	Название	Назначение
EDX.0	FPU	Floating Poin Unit – наличие математического сопроцессора
EDX.1	VME	Virtual-8086 Mode Enhancements – расширение режима V86 (виртуализация флага прерываний)
EDX.2	DE	Debuging Extensions – расширение отладки (возможность остановки по обращению к портам)
EDX.3	PSE	Page Size Extension – возможность применения размера страницы в 4 Мбайт
EDX.4	TSC	Time Stamp Counter – наличие счетчика меток реального времени
EDX.5	MSR	Model Specific Register – поддержка модельно-специфических регистров в стиле Pentium (инструкции RDMSR, WRMSR)
EDX.6	PAE	Physical Address Extension – возможность расширения физического адреса до 36 бит
EDX.7	MCE	Machine Check Exception – поддержка исключения машинного контроля #MC
EDX.8	CX8	Поддержка инструкции CMPXCHG8B
EDX.9	APIC	Наличие встроенного программно-доступного контроллера прерываний APIC
EDX.10	–	Зарезервировано

Бит	Название	Назначение
EDX.11	SEP	Sysenter Present — поддержка инструкций быстрых системных вызовов SYSENTER и SYSEXIT
EDX.12	MTRR	Memory Type Range Registers — наличие регистра управления кэшированием MTRRcap
EDX.13	PGE	Page Global Enable — поддержка битов глобальности в элементах каталога и таблиц страниц, а также бита PGE в регистре CR4
EDX.14	MCA	Machine Check Architecture — поддержка архитектуры машинного контроля
EDX.15	CMOV	Conditional Move — поддержка инструкций условной пересылки CMOVcc, а если есть FPU, то и инструкций FCMOVCC и FCOMI
EDX.16	PAT	Page Attribute Table — поддержка таблиц атрибутов страниц (PAT)
EDX.17	PSE-36	36-bit Page Size Extension — возможность использования 36-битной физической адресации для страниц в 4 Мбайт
EDX.18[1]	PSN	Processor Serial Number — поддержка сообщения 96-битного серийного номера по инструкции CPUID(3)
EDX.19	CLFSH	Поддержка инструкции CLFLUSH
EDX.20[1]	–	Зарезервировано
EDX.21[1]	DS	Debug Store — поддержка отладочной записи истории переходов или архитектурных состояний
EDX.22[1]	ACPI	Наличие регистров MSR, позволяющих следить за температурой и программировать модуляцию частоты синхронизации
EDX.23	MMX	Поддержка MMX
EDX.24	FXSR	Fast floating point save and restore — поддержка инструкций быстрого сохранения и восстановления контекста FPU (инструкций FXSAVE и FXRSTOR). Указывает и на доступность индикатора использования этих инструкций операционной системой (CR4.OSFXSR)
EDX.25	SSE	Поддержка инструкций расширения SSE (наличие блока XMM)
EDX.26	SSE2	Поддержка инструкций расширения SSE2
EDX.27[1]	SS	Self Snoop — управление конфликтующими типами памяти путем слежения за собственным кэшем для транзакций, посылаемых на шину
EDX.28[1]	HT	Hyperthreading — поддержка гиперпотоковой технологии
EDX.29[1]	TM	Thermal Monitor — автоматическое понижение производительности при перегреве
EDX.30[1]	–	Зарезервировано
EDX.31[1]	FERR	FERR# Signaling change — изменение назначения сигнала FERR#
ECX.0[2]	SSE3	Поддержка SSE3
ECX.3[2]	Monitor	Инструкции Monitor/Mwait
EXX.4[2]	DS_CPL	Отладочная запись с указанием CPL
ECX.7[2]	EST	Технология Enhanced SpeedStep
ECX.8[2]	TM2	Доступность Thermal Monitor 2
ECX.10[2]	CNXT-ID	Доступность идентификации контекста L1
ECX.13[2]	CMPXCHG16B	Доступность инструкции CMPXCHG16B

[1] Биты для процессоров AMD не используются.
[2] Биты регистра ECX в процессорах AMD не используются.

Фирма AMD из стандартных задействует только функции с номерами 0 и 1, а для получения дополнительной информации применяет перечисленные ниже *расширенные функции CPUID*. Позже расширенными функциями стала пользоваться и фирма Intel, правда, с некоторыми отличиями (см. далее).

- CPUID(8000 0000h) — аналог CPUID(O), но возвращает максимальный номер расширенной инструкции.
- CPUID(8000 0001h) — сигнатура процессора (в EAX) и набор флагов расширений базовой архитектуры, реализованных в данном процессоре. В табл. 7.3 приведены только те флаги расширений, которые трактуются иначе, чем возвращаемые по CPUID (1); остальные совпадают с приведенными в табл. 7.2. Процессоры Intel из этих флагов используют только EDX. 11 (SYSCALL/SYSRET) и EDX. 29 (называя его Intel EM64T).
- CPUID(8000 0002h-8000 0004h) — возвращает 48-символьную ASCIIZ-строку с именем процессора (processor brand string), например «AMD Athlon(tm) processor».
- CPUID(8000 0005h, 8000 0006h) — параметры внутренней системы кэширования. Процессоры Intel CPUID(8000 0005h) не поддерживают, а в CPUID(8000 0006h) формат сообщения отличается от AMD.
- CPUID(8000 0007h) — возможности расширенного управления энергопотреблением (наличие датчика температуры, термозащиты, управления идентификаторами напряжения питания и частоты). Процессоры Intel этот вызов не поддерживают.
- CPUID(8000 0008h) — разрядность виртуального и физического адресов в 64-бит- ном режиме.

Таблица 7.3. Расширения базовой архитектуры по CPUID(8000 0001h)

Бит	Назначение
EDX.11	Поддержка инструкций SYSCALL и SYSRET (отличаются от SYSENTER и SYSEXIT)
EDX.20	Поддержка бита NX (No Execute) – запрет исполнения кода в определенных страницах (или каталогах страниц) памяти
EDX.22	Поддержка инструкций AMD Extensions to MMX (целочисленных 64-битных, включающих некоторые из SSE и SSE2)
EDX.29	Long Mode – поддержка расширения x86-64 (EM-64T)
EDX.30	Поддержка инструкций AMD Extensions to 3DNow!
EDX.31	Поддержка инструкций AMD 3DNow!

Идентификация процессора начинается с определения краткого имени производителя («Genuinelntel» или «AuthenticAMD»). Следующий уровень идентификации — *сигнатура*, в состав которой входят четыре битовых поля.

- *Семейство* (family) определяет номер, привычно называемый *поколением* (3 — 386, 4 — 486, 5 — Pentium, 6 — P6). Процессоры Pentium 4 в поле семейства сообщают число Fh — признак расширенного поля семейства. При этом «эффективный» номер семейства и модели получается сдвигом и суммиро-

ванием расширенного и основного полей. Зачем это понадобилось — непонятно, четырехбитного поля семейства хватило бы еще на много поколений.

- *Модель* — поле, позволяющее различать процессоры внутри семейства, например, Pentium и Pentium MMX; Pentium Pro, Pentium II, Pentium III и их разновидности.
- *Степпинг* (stepping) — номер редакции кристалла процессора, меняющейся в пределах модели. Этот номер определяет наличие тех или иных ошибок, которые, как правило, исправляются соответствующими «заплатами».
- *Тип* — поле, определяющее назначение процессора: обычный, *overdrive* (для замены старых процессоров), с поддержкой мультипроцессорных систем.

Сигнатура должна рассматриваться в совокупности с именем производителя (заметно различающиеся процессоры разных фирм имеют совпадающие сигнатуры). Одному и тому же сочетанию типа, семейства и модели могут соответствовать несколько типов процессоров одной фирмы, например, Pentium II и Pentium II Xeon по ним неразличимы. Эти процессоры различаются элементами системы кэширования, и их можно распознать по дескрипторам, сообщаемым инструкцией CPUID(2).

Бренд-индекс служит для нахождения официального названия процессора в таблице, которая должна храниться в BIOS. Вместо индекса и таблицы может использоваться строка с именем процессора *(processor brand string*), в которой помимо собственно названия может указываться официальная допустимая частота (но текущее значение может отличаться). Например, первые процессоры Intel, в которых появилось сообщение имени, имеют строку «Intel(R) Pentium (R) 4 CPU 1500 MHz».

Уникальный 96-битный идентификатор процессора состоит из 32-битной *сигнатуры* и 64-битного *серийного номера*. По замыслу Intel, этот идентификатор должен стать дополнительным средством аутентификации в Интернете (и других сетях) наряду с именем пользователя и паролем, вводимыми вручную. Однако если имя и пароль можно сменить в любое время, идентификатор присваивается навечно и принудительно. С помощью специально выделенного бита одного из регистров MSR можно запретить процессору сообщать свой идентификатор (доступ к этому регистру привилегирован). По аппаратному сбросу процессора (и только так!) бит обнуляется и полная идентификация разрешается. Декларированное «отключение по умолчанию» возлагается на ОС или BIOS, а для Windows предлагается специальная утилита, опрашивающая значение бита MSR и управляющая его установкой, а также сообщающая прочитанный (по возможности) идентификатор. Похоже, от идеи такой идентификации отказались (в описаниях для процессоров Pentium 4 чтение номера уже не фигурирует).

Для получения исчерпывающей информации об архитектурных возможностях процессора применяются флаги расширений архитектуры (см. табл. 7.2 и 7.3). В зависимости от обнаруженного процессора ОС может задействовать разные варианты реализации своих функций. ОС защищенного режима (а теперь иными практически не пользуются) разрешает или запрещает те или иные архитектурные расширения (в основном записью в регистр CR4). Прикладным програм

мам остается только согласиться с предлагаемым набором свойств процессора или отказаться от работы. Прикладные программы могут получать информацию о процессоре по инструкции CPUID при любом уровне привилегий, а привилегированные программы могут пользоваться и данными о включенных расширениях из регистра CR4. Полученную информацию программа может задействовать, например, для выбора исполняемого кода, оптимального для данного процессора (или отказа исполнения на «недостойном» программы процессоре), а также для настройки констант программных реализаций задержек.

Информация для CPUID «зашивается» в процессор на этапе изготовления кристалла. Заметим, что в числовых параметрах, сообщаемых по CPUID, нигде в явном виде не фигурирует тактовая частота процессора — для каждой модели и степпинга выпускаются процессоры с некоторым диапазоном тактовых частот, а конкретное значение обозначается на корпусе после испытаний и отбраковки. Значение максимальной частоты может присутствовать в строке имени процессора. Эта информация полезна, например, для борьбы с пиратским разгоном (перемаркировкой) процессоров, что, в общем-то, волнует изготовителя.

В процессоре Xeon имеется специальная постоянная (только для чтения) память процессорной информации (Processor Information ROM, *PIROM*), которая хранит такие данные, как электрические спецификации ядра процессора и кэшпамяти (диапазоны частот и питающих напряжений), степпинг и 64-битный серийный номер процессора. Кроме того, имеется энергонезависимая память *Scratch EEPROM*, которая предназначена для занесения системной информации поставщиком процессора (или компьютера с этим процессором) и может быть защищена от последующей записи. Для взаимодействия с PIROM и Scratch EEPROM (а также устройством термоконтроля) процессор имеет дополнительную последовательную шину *SMBus* (System Management Bus) — отдельный электрический интерфейс, с которым работает чипсет системной платы. Обращение к указанным учетным данным получается довольно сложным — это целая процедура, привязанная к реализации чипсета, а не одна инструкция.

Основные характеристики процессоров

В IBM PC применялись процессоры x86 всех поколений, первые пять поколений в этой книге уже практически не рассматриваются (интересующиеся могут обратиться к дополнительной литературе [2], [3]). На сегодняшний день наибольший интерес представляют процессоры 6-8-го поколений. Эти процессоры сравниваются в табл. 7.4, в которой представлены основные характеристики современных процессоров фирм Intel и AMD — лидеров процессоростроения для PC. В таблице приводится *число выводов*, по которому можно определить тип подходящего сокета (и системной платы). Скорость работы процессора характеризуется *частотой ядра*, а также пропускной способностью канала доступа к памяти. Последняя характеристика приводится в таблице как *частота FSB* — частота передачи 8-байтных блоков данных[1]. Для процессоров AMD со встроенным контроллером памяти приводится *частота шины памяти* DDR SDRAM

[1] Для процессоров Intel с интерфейсом шины Pentium 4 частота тактового сигнала шины в 4 раза ниже, см. далее.

(частота передачи блоков в два раза выше); разрядность шины памяти у обычных процессоров — 64 бита (8 байт), у особо мощных процессоров — 128 бит (16 байт). Пиковая пропускная способность обращений к памяти равна произведению разрядности передаваемых блоков и частоты их передачи. Заметим, что процессоры со встроенным контроллером памяти выгодно отличаются и меньшей задержкой доступа к памяти (по сравнению с доступом через системную шину). В параметрах кэша L1 указывается размер кэша инструкций (I) и кэша данных (D); для процессоров Pentium 4 (и его «родственников» с микроархитектурой NetBurst) вместо размера кэша инструкций указан размер кэша трассы. Буква E после размера кэша L2 означает *эксклюзивность*: в кэше уровней 1 и 2 информация не дублируется. Графа «HT, MC» показывает число логических процессоров: HT — признак гиперпотокового ядра, 2C — наличие двух ядер (2C HT означает четыре логических процессора). Графа E64 указывает на поддержку 64-битного расширения (x86-64 у AMD и EM64T у Intel). В графе SIMD представлены поддерживаемые расширения команд для блоков MMX и XMM (MMX не указывается, поскольку поддерживается всеми современными процессорами). Для SSE показан номер последнего из имеющихся расширений (SSE3 означает, что поддерживаются и SSE2, и SSE). В графе «Бит NX» указана поддержка запрета исполнения для страниц памяти.

Таблица 7.4. Сравнительные характеристики процессоров 6-8-го поколений

Процессор	Число выводов (сокет)	Частота ядра ($F_{ядра}$), ГГЦ	Частота FSB (шины памяти), МГц	Объем кэша L1 (I+D), Кбайт	Объем кэша L2 (+L3), Кбайт	HT, MC	E64	SIMD	Бит NX
Процессоры Intel для серверов и рабочих станций									
Xeon 64 MP	604	2,66–3,66	667	12+16	1024 L3[1]: 4096, 8192	HT	+	SSE3	+[1]
Xeon 64	604	2,8–3,6	800	16	2048	HT	+	SSE3	+
Xeon	604	2,8–3,6	400, 533, 800	12+(8–16)	256, 512, 1024 L3[1]: 1024, 2048	HT[1]	+[1]	SSE2, SSE3[1]	+[1]
Xeon MP	603	1,4–3	400	8	256–512 L3: 512–4096	HT	–	SSE2	–
Процессоры Intel для настольных ПК									
Pentium Extreme Edition 840	775	3,2	800	2×16	2×1024	HT, 2C	+	SSE3	+
Pentium 4 Extreme Edition	775	3.2–3,73	800; 1066	12+(8–16)	512; 2048	HT	+[1]	SSE3	+
Pentium 4 Extreme Edition	478	2,0–3,4	533; 800	12+8	512; L3: 2048	HT[1]	–	SSE2	–
Pentium D 820, 830, 840	775	2,80–3,20	800	2×16	2×1024	2C	+	SSE2, SSE3	+

продолжение

Таблица 7.4 (*продолжение*)

Процессор	Число выводов (сокет)	Частота ядра (F_{core}), ГГЦ	Частота FSB (шины памяти), МГц	Объем кэша L1 (I+D), Кбайт	Объем кэша L2 (+L3), Кбайт	HT, MC	E64	SIMD	Бит NX
Pentium 4 и Pentium xxx	775	2,80–3.80	533, 800	12+8	1024, 2048	HT[1]	+[1]	SSE2	+[1]
Pentium 4 (90 нм)	478	2,8A/E–3,8E	533, 800	12+16	1024	HT	–	SSE2, SSE3	–
Pentium 4 (0,13 мкм)	478	2,0–3,4	533, 800	12+8	512 L3[1]: 2048	HT[1]	–	SSE2	–
Pentium 4 (0,18 мкм)	478	1,4–2,0	400	12+8	256	–	–	SSE2	–
Pentium 4 (0,18 мкм)	423	1,3–?	400	12+8	256	–	–	SSE2	–
Celeron D (90 нм)	775 478	2,26–3,2	533	12+8	256	– –	+[1] –	SSE3	+[1] –
Celeron 2,0–2.8 (0,13 мкм)	478	1,7–2,8	400	12+8	128	–	–	SSE2	–
Celeron 1.7–2.8	423	1,7–2,8	400	12+8	128	–	–	SSE2	–
Celeron 850–1.4	370	0,85–1,4	100	16+16	128, 256	–	–	SSE	–
Процессоры Intel для мобильных ПК									
Pentium M 7xx	479	0,6–2,13	400, 533	32+32	2048	–	–	SSE2	+[1]
Pentium M	478, 479	0,9–1,7	400		1024	–	–	SSE2	–
Celeron M 3xx	478, 479	0,9–1,5	400		512, 1024	–	–	SSE2	–
Mobile Celeron (0,13 мкм Pentium III)	478, 479	0,65–1,13	100, 133	16+16	256	–	–	SSE	–
Процессоры AMD для серверов и рабочих станций									
Athlon 64 X2	940					2C			
Opteron	940	1,4–2,4	(100–200) 128 бит	64+64	1024	–	+	SSE2, 3DNow!E	+
Athlon MP (1500+...2800+)	462 (A)	1,33–2,13	266	64+64	256	–	–	3DNow!E	–
Процессоры AMD для настольных ПК									
Athlon 64 FX	939, 940	2,2–2,6	(100–200) 128 бит	64+64	1024	–	+	SSE3, 3DNow!P	+
Athlon 64 (3500+... 4000+)	939	2,2–2,4	(100–200) 128 бит	64+64	512, 1024	–	+	SSE2& 3, 3DNow!E	+
Athlon 64 (2800+... 3700+)	754	1,8–2,4	(100–200)	64+64	512, 1024	–	+	SSE2& 3, 3DNow!E	+

Процессор	Число выводов (сокет)	Частота ядра (F_{core}), ГГц	Частота FSB (шины памяти), МГц	Объем кэша L1 (I+D), Кбайт	Объем кэша L2 (+L3), Кбайт	HT, MC	E64	SIMD	Бит NX
Semptron 3100+, 3300+	754	1,8–2	(100–200)	64+64	128, 256, 512	–	–	SSE2, 3DNow!E	–
Semptron 2200+...3100+	462 (A)	1,5–2,0	266, 333	64+64	256	–	–	SSE, 3DNow!E	–
Athlon XP 1500+...3000+	462 (A)	1,33–2,17	266–400	64+64	256; 512E	–	–	SSE[1], 3DNow!E	–
Duron	462 (A)	1,4–1,8	200–266	64+64	64E	–	–	3DNow!E	–
Процессоры AMD для мобильных ПК									
Turion 64 Mobile	754		(100–200)	64+64	512, 1024	–	+	SSE3, 3DNow!E	+
Mobile Semptron 2600+...3000+	754	1,6–1,8	(100–200)	64+64	128, 256	–	–	SSE2, 3DNow!E	–

[1] Относится не ко всем моделям.

Во встраиваемых PC-совместимых компьютерах (управляющих, инструментальных и т. п.) с успехом применяются (и до сих пор выпускаются) процессоры 4-5-го поколений. Они отличаются от своих более поздних собратьев меньшей ценой, низким энергопотреблением и более легким согласованием с 8- и 16-битной периферией.

Принадлежность процессора к какому-либо из первых шести поколений была вполне очевидной и могла легко определяться по сигнатуре, сообщаемой в инструкции CPUID. С новыми поколениями дело обстоит сложнее: в сигнатуре в поле семейства, которое раньше соответствовало поколению, указывается F — признак «расширенного» семейства. Есть подозрения, что эти сложности возникли от того, что 7-е (K7) и 8-е (x86-64) поколения фирма AMD ввела раньше фирмы Intel. Возможно, от понятия «поколение» придется в будущем отказаться.

7.7. Процессоры фирмы Intel

Названия процессоров фирмы Intel далеко не всегда раскрывают архитектурные (и микроархитектурные) особенности данной модели. По названиям старых процессоров (Pentium, Pentium MMX, Pentium II, Pentium III и первых моделей Pentium 4) можно было однозначно определить наличие основных архитектурных расширений: блока MMX, XMM и соответствующих расширенных наборов инструкций. Вместе с тем название *Celeron* является безликим: под этим именем существуют «облегченные» варианты от Pentium II до Pentium 4. Аналогичная ситуация и с названием *Xeon*: это варианты тех же процессоров, но «утяжеленные» — для серверов. Определить функциональные возможности Celeron по тактовой частоте можно, но не всегда: на «пограничных» частотах существуют версии Celeron на базе Pentium II и Pentium III, а также Pentium III и Pentium 4.

В 2005 году фирма Intel решила отказаться от названия Pentium 4 и указания тактовой частоты, и теперь все новые процессоры для настольных ПК называются «Pentium xxx», где xxx — трехзначный номер. После номера может стоять и буква J, означающая наличие функции *Execute Disable Bit* (поддержки бита NX в таблицах страниц). Трехзначный номер отражает набор архитектурных (расширения системы команд, число логических и физических ядер), микроархитектурных (в основном параметры системы кэширования) возможностей, а также максимальные частоты ядра и системной шины. Простые правила, по которым определяется номер, не публикуются (предлагается утилита, сообщающая параметры процессора по его номеру). Однако среди процессоров «созвучного» ряда номеров (например, 660, 650, 640, 630) более мощные имеют большие номера. Наряду с трехзначными номерами используются и звучные названия, например Pentium Extreme Edition 840; этот процессор не следует путать с процессорами Pentium D 840 и Pentium 4 Extreme Edition (без трехзначного номера).

Процессоры P6

К 6-му поколению процессоров Intel относятся процессор Pentium Pro, все разновидности процессоров Pentium II/III, а также Celeron. Процессоры этого поколения имеют обобщенное название *P6*. С точки зрения принципа организации вычислений главное отличие этого поколения заключается в *динамическом исполнении*, при котором внутри процессора инструкции могут исполняться не в том порядке (out of order), который предполагает программный код. Это решение призвано повысить производительность процессора за счет улучшения микроархитектуры, а не повышения тактовой частоты. При этом требуется большая производительность памяти, что обеспечивается развитием системы кэширования. У процессоров P6 вторичный кэш размещается на самом процессоре (в одной микросхеме или картридже), и для доступа к этому кэшу используется выделенная шина. Фасадная (системная) шина (FSB) применяется для связи с чипсетом — с контроллером памяти и шинами ввода-вывода. У Intel это решение получило звучное название *двойной независимой шины* (Dual Independent Bus, DIB)[1].

Снижение нагрузки на внешнюю шину позволяет эффективно использовать многопроцессорную архитектуру. Системная шина P6 более эффективна для объединения процессоров по симметричной архитектуре, чем шины предыдущих процессоров, оптимизированные для обмена с памятью. Она позволяет без дополнительных схем объединять до четырех процессоров, хотя в обычных процессорах Pentium II/III возможности объединения урезаны до двух.

В ходе эволюции поколения к системе команд Pentium Pro, расширенной относительно Pentium с целью сокращения условных переходов, было добавлено расширение MMX — так появился процессор Pentium II. Затем идею MMX — одновременное исполнение одной инструкции для группы операндов — распро

[1] В настоящее время это решение выглядит не так внушительно по сравнению с выделенной шиной памяти современных процессоров AMD.

странили и на инструкции с плавающей точкой, создав потоковые SIMD-рас- ширения (Streaming SIMD Extensions, SSE) — основной козырь Pentium III. Правда, несколько раньше то же самое (но в меньшем объеме) было сделано фирмой AMD — расширение 3DNow! было реализовано уже в процессорах K6-2 для сокета 7.

Микроархитектура P6 «честно» отработала на ряде моделей процессоров, начиная с Pentium Pro (0,6 мкм, 150 МГц) и до Pentium Ill/Celeron (0,18 и 0,13 мкм,

1,4 ГГц). У предыдущего поколения «живучести» было меньше: первый процессор Pentium имел частоту 60 МГц, последний — только 233 МГц, хотя, возможно, здесь помимо технических аспектов значительное влияние имеют маркетинговые соображения.

Процессоры *Pentium Pro* (1995 г., технология 0,5 мкм, впоследствии 0,35 мкм) выпускались с частотами ядра 150, 166, 180 и 200 МГц и объемом вторичного кэша 256 и 512 Кбайт (1024 Кбайт в специальных моделях). Процессоры могут работать в симметричных мультипроцессорных системах (SMP) — до четырех процессоров на общей шине.

Процессоры *Pentium II* сочетают архитектуру Pentium Pro с технологией MMX. По сравнению с Pentium Pro удвоен размер первичного кэша (1 6 + 1 6 Кбайт), размер вторичного кэша варьируется от 0 до 2 Мбайт. Процессор представляет собой картридж (печатную плату в кожухе или без него) с краевым разъемом (Single Edge Contact Cartridge, SECC), на который выведена системная шина. На плате картриджа установлены микросхема ядра процессора, несколько микросхем, реализующих вторичный кэш, и вспомогательные дискретные элементы (резисторы и конденсаторы). Снятие вторичного кэша с микросхемы процессора позволило решить ряд технологических проблем, существовавших в то время, из-за которых цена «монолитного» процессора Pentium Pro была чрезвычайно высокой.

Первые процессоры Pentium II (1997 г.) выполнялись по технологии 0,35 мкм, питание 2,8 В. Они имели тактовые частоты ядра 233, 266 и 300 МГц при частоте системной шины 66,6 МГц. При этом вторичный кэш работал на половинной частоте ядра и кэшировал только первые 512 Мбайт пространства памяти. Для этих процессоров был разработан *слот 1*, по составу сигналов сильно напоминающий сокет 8 для Pentium Pro. Однако слот 1 позволяет объединять лишь пару процессоров для реализации симметричной мультипроцессорной системы либо системы с избыточным контролем функционирования (FRC). Таким образом, этот процессор представляет собой более быстрый процессор Pentium Pro с поддержкой MMX, но с урезанным мультипроцессированием. Первые модели по инструкции CPUID сообщают идентификатор 063xh.

Следующие модели Pentium II (*Deshutes*, 1998 г.) выполнялись уже по технологии 0,25 мкм, питание 2,0 В. Эта технология позволила поднять тактовую частоту (чем мельче элементы, тем меньше они рассеивают мощность, что особенно критично на высоких частотах). Процессор на 333 МГц имеет частоту системной шины 66,6 МГц, а процессоры на 350, 400 и 450 МГц — уже 100 МГц. Эти процессоры также устанавливаются в слот 1 (опять-таки не более двух

в системе). Начиная с процессоров 350 МГц объем памяти, кэшируемой на L2, увеличили до 4 Гбайт. По инструкции CPUID сообщается идентификатор 065xh.

Процессоры *Pentium III* (*Katmai*, 1999 г.) являются дальнейшим развитием Pentium II: в них появились блок XMM и расширение набора SIMD-инструк- ций — *SSE*, предварительно называвшееся *KNI* (Katmai New Instructions). Появились и новые возможности управления кэшированием. Усовершенствована инструкция CPUID, по которой теперь можно получить уникальный 64-битный идентификатор процессора (тот, что у Xeon можно было прочесть по SMBus). «Простые» процессоры Pentium III в упаковке SECC или SECC2 устанавливаются в слот 1, в FC-PGA — в сокет 370, Pentium III Xeon — в слот 2. По возможностям мультипроцессорных конфигураций эти процессоры аналогичны своим предшественникам Pentium II и Pentium II Xeon. Частота ядра начинается с 500 МГц, частота системной шины — 100 и 133 МГц. Вторичный кэш в первых моделях Pentium III — 512 Кбайт с ECC-контролем — работает на половине частоты ядра, расположен на картридже в виде отдельных микросхем (собственно память и память тегов). Первые модели по CPUID сообщают идентификатор 067xh.

Процессоры с ядром *Coppermine* (иногда сокращенно называют CuMine) тоже называются Pentium III. Несмотря на слово «copper» (или Си — «медь») в названии, медные проводники в них не используются. Технология 0,18 мкм, выпускались в картридже SECC-2 для слота 1 (SC-242) и в корпусе FC-PGA (Flip-Chip PGA) для сокета 370. На кристалле ядра расположен улучшенный вторичный кэш (advanced transfer cache) размером 256 Кбайт с ECC-контролем, работающий на частоте ядра. Вторичный кэш связан с ядром шиной разрядностью 256 бит (у предыдущих P6 с отдельно расположенным кэшем разрядность шины данных кэша составляла 64 бит). По сравнению с Celeron и первыми моделями Pentium III, вторичный кэш CuMine имеет меньшую латентность (задержку от запроса до начала пересылки данных), а пропускная способность его шины выросла в 4 раза. Напряжение питания Для сокета 370 — 1,6 В, для слота 1 — 1,65 В. Частота системной шины — 100 и 133 МГц. Коэффициенты умножения фиксированы изготовителем. Некоторые модели в корпусах FC-PGA не поддерживают SMP. Процессоры в SECC2 имеют то же назначение выводов, что и их предшественники для слота 1. Однако модернизация старых плат может «упереться» в старую версию BIOS и в невозможность получения низкого (1,65 В) напряжения питания. Процессоры в FC-PGA отличаются от семейства Celeron, для которых был введен сокет 370, назначением пяти выводов, и по этой причине совместимости со старыми платами нет. По CPUID процессор сообщает идентификатор 068xh.

Для дешевых настольных компьютеров существуют облегченные варианты процессоров — *Celeron*. «Облегчен» в основном вторичный кэш и ограничена частота системной шины. У первых процессоров Celeron — «младших братьев» Pentium II — вторичного кэша нет (но выпускались такие процессоры в картриджах). Позже (1998 г.) появился процессор Celeron на ядре *Mendocino* с небольшим (128 Кбайт) вторичным кэшем, установленным на кристалле ядра. Этот процессор выпускался и в картридже (по технологии 0,25 мкм), и в корпу

се PPGA (0,22 мкм). Тактовые частоты — от 300 (Celeron 300A) до 533 МГц (буква A позволяет отличить его от предшественников с той же частотой).

Процессоры *Celeron* (их еще называют Celeron II) на ядре *Coppermine* (2000 г., 0,18 мкм, питание 1,5 В) уже имеют блок XXM и поддержку инструкций SSE, то есть это «братья» Pentium III. Тактовые частоты — от 533 (Celeron 533A) до 1100 МГц. Начиная с частоты 800 МГц наконец-то поднята частота шины до 100 МГц. По CPUID сообщается идентификатор 068xh. Упаковка FC-PGA (для сокета 370), по назначению выводов процессор «условно» совместим с платами для Celeron (сигнал RESET# там же, но требуется изоляция вывода A20M от шины GND). Последние модели Celeron поколения P6 выполняются по технологии 0,13 мкм, у них вторичный кэш уже 512 Мбайт, тактовые частоты — от 1000 до 1400 МГц (у моделей с частотами 1 и 1,1 ГГц ставится буква A для отличия от предыдущих).

Помимо широко известных особенностей вторичного кэша (либо его нет, либо есть размером 128 Кбайт), процессор Celeron от Pentium II имеет следующие отличия:

- Разрядность шины адреса сокращена с 36 до 32 бит (адресуемая память — 4 Гбайт).
- Контроль четности шины адреса и шины запроса, ECC-контроль шины данных и контроль неисправимых ошибок шины отсутствует, как и сигнал инициализации шины.
- Процессоры предназначены только для одиночных конфигураций: из сигналов запроса шины официально остался только BR0#, что не позволяет реализовать симметричные двухпроцессорные конфигурации. Реально сигнал BR1# в некоторых моделях присутствует, что позволяет использовать Celeron и в двухпроцессорных системах SMP.
- Коэффициенты умножения частоты фиксированы.

Для мощных компьютеров (серверов) предназначено семейство *Xeon* — «утяжеленные» варианты процессоров Pentium II и Pentium III. Для них ввели новый *слот* 2, который (вместе с интерфейсом нового процессора) позволяет строить как избыточные системы с FRC, так и симметричные 1-, 2-, 4- и даже 8-процес- сорные системы. Объем вторичного кэша — 512 Кбайт, 1 или 2 Мбайт при кэшировании до 64 Гбайт (все адресное пространство при 36-битной адресации). Процессоры Xeon отличаются не только большей мощностью, но и большими размерами — 15,2 x 12,7 x 1,9 см.

Процессоры Xeon имеют новые средства хранения системной информации. Постоянная (только для чтения) память процессорной информации (*PIROM*) хранит такие данные, как электрические спецификации ядра процессора и кэш-памяти (диапазоны частот и питающих напряжений), S-спецификацию и серийный 64-битный номер процессора. Энергонезависимая память *Scratch EEPROM* предназначена для занесения системной информации поставщиком процессора (или компьютера с этим процессором) и может быть защищена от последующей записи. Процессор оборудован термодатчиком (термодиод на кристалле ядра) с программируемым устройством контроля температуры. Это устройство

имеет аналого-цифровой преобразователь, калибруемый по термодиоду конкретного процессора на этапе тестирования картриджа. Константа настройки термометра заносится в PIROM. Устройство термоконтроля программируется — задаются частота преобразований и пороги температуры, по достижении которых вырабатывается сигнал прерывания. Для взаимодействия с PIROM, Scratch EEPROM и устройством термоконтроля процессор имеет дополнительную последовательную шину *SMBus*.

Процессоры *Pentium II Xeon* на ядре Deshutes (0,25 мкм) имеют частоту шины 100 МГц, частота ядра — 400-500 МГц.

Процессоры *Pentium III Xeon* под кодовым названием *Tanner* (0,25 мкм) имеют частоту шины 100 МГц, частота ядра — от 500 МГц. Вторичный кэш — 512 Кбайт,

1 Мбайт или 2 Мбайт, кэш работает на частоте ядра. Этот процессор позиционируется как серверный (поддерживает 4/8 процессорные конфигурации SMP).

Процессоры *Pentium III Xeon* под кодовым названием *Cascades* (0,18 мкм) имеют частоту шины 133 МГц, частота ядра — от 600 МГц. Вторичный кэш — 256 Кбайт, расположен на кристалле ядра, работает на частоте ядра.

Мобильные процессоры семейства P6 предназначены для установки в блокнотные ПК и другие малогабаритные системы с автономным питанием. Эти процессоры выпускаются в нескольких конструктивных исполнениях: миниатюрный корпус BGA1, BGA2 с выводами для припаивания, Micro-PGA2 со штырьковыми выводами, мини-картридж с 240-штырьковым разъемом и модули с коннекторами MMC-1 и MMC-2. В этих исполнениях могут быть процессоры четырех типов: мобильный процессор Pentium III, мобильный процессор Pentium II с внешним вторичным кэшем, мобильный процессор Celeron с кэшем 128 Кбайт и мобильный процессор Pentium II со встроенным кэшем 256 Кбайт. Мобильные процессоры имеют ряд отличий от обычных процессоров Pentium II/III:

- не поддерживаются избыточный контроль функционирования (FRC) и двух-процессорные конфигурации;
- понижено напряжение питания, на некоторых процессорах напряжение питания ядра уже ниже 1 В;
- понижена нагрузочная способность интерфейсных схем;
- введено новое состояние пониженного потребления *Quick Start*, которое отличается от состояния *Stop Grant* тем, что в нем не отслеживаются транзакции симметричных агентов (другого процессора), слежение ведется только за приоритетными агентами шины, поэтому потребление в состоянии *Quick Start* существенно меньше, чем в *Stop Grant.*

Процессоры Pentium 4

Процессор Pentium 4, появившийся в 2001 году, принадлежит к 7-му (по классификации Intel) поколению. С программной точки зрения он представляет собой процессор x86 с очередным расширением системы команд — SSE2, а в

2005 году в Pentium 4 появились и 64-битное расширение (EM64T), и дополнительные команды SSE3. По набору программно-доступных регистров Pentium 4 без EM64T повторяет процессор Pentium III. В процессорах с EM64T разрядность общих регистров расширена до 64 бит, добавлены дополнительные общие регистры и удвоено число регистров XMM. Эти процессоры можно считать представителями восьмого поколения.

С внешней аппаратной точки зрения это процессор с системной шиной нового типа, в которой помимо повышения тактовой частоты применены ставшие уже привычными принципы двойной (2x) и четырехкратной (4x) синхронизации, а также предпринят ряд мер по обеспечению работоспособности на ранее немыслимых высоких частотах. Микроархитектура процессора, получившая название NetBurst, разработана с учетом высоких частот как ядра (от 1,3 ГГц), так и системной шины (от 400 МГц). Название микроархитектуры указывает на сетевую направленность процессора — его мощь требуется для ресурсоемких мультимедийных интернет-приложений. Расширение системы команд ориентировано на задачи, которые становятся посильными для обычных настольных компьютеров:

- потоковые приложения, включая обработку видеоинформации в реальном времени, подразумевающую как декодирование сжатой информации, так и более сложные задачи кодирования;
- редактирование видеоизображений;
- трехмерная визуализация;
- обработка видеосигнала в качестве источника данных;
- связь с телевидением высокой четкости (HDTV);
- распознавание речи;
- интернет-телефония;
- шифрование данных.

Как и процессоры 6-го поколения, Pentium 4, помимо собственно вычислительного ядра, имеет кэш-память двух уровней. *Вторичный кэш*, общий для инструкций и данных, имеет разрядность шины 256 бит (32 байта) и высокую частоту, что обеспечивает высокую пропускную способность (как и в последних процессорах Pentium III). *Первичный кэш данных* имеет такую же высокую пропускную способность, но его объем сократился вдвое (8 Кбайт против 16 в Pentium III)[1]. *Первичный кэш инструкций* в привычном понимании отсутствует, его заменил *кэш трассы* (trace cache). В нем хранятся последовательности микроопераций, в которые декодированы инструкции. В этом кэше могут помещаться до 12К микроинструкций.

В *микроархитектуре NetBurst* применяется очень длинный конвейер (гипер- конвейер, как его называет Intel). Увеличение числа ступеней позволяет упростить (и ускорить, в смысле увеличения тактовой частоты) каждую из них. Как и в P6, здесь присутствует «беспорядочное» исполнительное ядро, работу кото

[1] Позже появились процессоры с первичным кэшем данных 16 и 32 Кбайт.

рому поставляет блок предварительной обработки. Результаты обработки использует упорядоченный блок завершения.

Блок предварительной обработки выбирает инструкции из памяти, декодирует их (транслирует в микрооперации или их последовательности) и обеспечивает доставку микроопераций из кэша трассы. Кэш трассы и транслирующий механизм кооперируются с аппаратурой предсказания ветвлений. В кэше трассы хранятся последовательности микроопераций, соответствующие декодированным инструкциям. При повторном исполнении фрагмента программного кода, находящегося в кэше трассы, процессор уже не тратит время на выборку и декодирование инструкций. В этом и заключается основная идея использования кэша трассы вместо привычного кэша инструкций. Последовательность микроопераций хранится в кэше трассы в порядке потока исполнения, так что при ветвлениях не приходится «прыгать» по строкам кэша. В память кэша трассы не попадают инструкции, которые никогда не будут исполняться. Кэш трассы способен хранить до 12 К микроопераций и за каждый такт доставлять ядру до трех микроопераций.

Исполнительное ядро имеет пиковую пропускную способность, превышающую возможности блока предварительной обработки и блока завершения. В нем используется существенно улучшенный (и ускоренный) блок FPU. Исполнительное ядро способно завершать до 6 микроопераций за такт.

Блок завершения работает практически так же, как и в P6, и позволяет выполнять до трех микроопераций за такт.

Микроархитектура NetBurst обеспечивает максимальную производительность исполнения предсказуемых (линейных и циклических) фрагментов программ, характерных для приложений, на которые и ориентирован новый процессор (см. выше). На непредсказуемо ветвящихся программах, к которым относятся, например, офисные приложения, длинный гиперконвейер оказывается менее эффективным, чем конвейер P6, если бы его удалось разогнать до частот

1,4 ГГц и выше. Но утешают две вещи — изначально высокая частота Pentium 4 и отсутствие потребности в сумасшедшей производительности для офисных приложений, в работе которых «участвует» пользователь, гораздо более медлительный по своей человеческой природе.

Интерфейс системной шины во многом напоминает шину P6 — протокол также ориентирован на одновременное выполнение нескольких транзакций, в основных цепях используются физические сигналы AGTL+. Однако здесь принят ряд мер по обеспечению высокой пропускной способности. В материалах по первым моделям (даже в сугубо техническом информационном листке) говорится о частоте шины 400 МГц с «четырехкратной накачкой» (quad pumped). Поясним, что это означает, чтобы не подвергаться соблазну излишних умножений. Для различных групп интерфейсных сигналов используются два типа синхронизации. *Общая синхронизация* осуществляется двумя парафазными сигналами BCLK0 и BCLK1 (для повышения точности применяются дифференциальные приемники). Для процессоров, у которых в качестве *частоты FSB* указано значение 400 МГц, *тактовая частота* системной шины (частота сигналов BCLK0 и BCLK1) составляет 100 МГц. Новая информация по линиям с общей синхрони

зацией может передаваться на каждом такте (в нашем случае — с частотой 100 МГц). Для 2- и 4-кратной передачи требуется *синхронизация от источника данных*: строб синхронизации, по которому приемник фиксирует данные, формируется в том же месте, что и передаваемая информация. По *шине адреса* информация передается в режиме 2-кратной передачи, и стробами являются сигналы ADSTB0# и ADSTB1#. По спаду этих стробов передается адрес, а по фронту — информация о типе транзакции. Таким образом, в каждом такте шины (за 10 нс) передаются и адрес, и тип транзакции (у P6 на это требовалось 2 такта, что занимало 15-30 нс). По *шине данных* информация передается с 4-кратной частотой, для чего используются пары стробирующих сигналов DSTBp[0:3]# и DSTBn[0:3]# (в нашем случае с периодом 5 нс, что соответствует частоте 200 МГц). Стробы сдвинуты относительно друг друга на половину своего маленького такта, синхронизация по их спадам и дает учетверенную частоту передачи (400 МГц).

Разрядность шины данных, как и в предыдущих двух поколениях процессоров, составляет 64 бита (8 байт), что для процессоров с FSB 400 МГц[1] дает максимальную пропускную способность 400 x 8 = 3,2 Гбайт/с. У процессоров Pentium III шина обеспечивала пропускную способность 133 x 8 = 1,06 Гбайт/с, так что по этому параметру у Pentium 4 улучшение троекратное. Шина адреса имеет разрядность 36 бит, что позволяет адресовать те же 64 Гбайт памяти, из которых кэшируются только первые 4 Гбайт. Более поздние модели Pentium 4 допускают «частоту шины» 533, 800 и даже 1066 МГц, что обеспечивает соответствующее повышение пропускной способности. В процессорах с 64-битным расширением (не во всех) шина адреса имеет разрядность 40 бит, что позволяет адресовать до 1024 Гбайт памяти.

Интересное решение принято для уменьшения помех при переключениях сигналов. Каждая пара байтов шины данных может передаваться в прямом или инверсном виде независимо от других байтов. Естественно, источник данных сигнализирует приемнику о текущем способе представления соответствующим сигналом. Решение о том или ином способе передачи принимается источником перед передачей каждой порции данных с таким расчетом, чтобы количество линий данных, изменяющих свое состояние, было минимальным. Это позволяет уменьшить броски тока и электромагнитные помехи. По сравнению с P6 несколько изменилось назначение сигналов контроля четности шин, изъята возможность ECC-контроля системной шины данных. В цепях питания предпринят ряд мер по снижению помех, питание аналоговых схем автоподстройки фазы изолировано от питания цифрового ядра.

Процессоры Pentium 4 по сравнению с предшественниками потребляют большую мощность и требуют установки вентилятора с радиатором «выдающихся» размеров и веса. Такой радиатор уже не повесить на корпус процессора, так что требуются специальные крепежные стойки, проходящие через системную плату и соединяемые с металлическим шасси (плата их тоже не выдерживает).

[1] Как показано ранее, тактовая частота при этом составляет 100 МГц.

Первые процессоры Pentium 4 (2001 г.) выполнялись по технологии 0,18 мкм (180 нм), тактовые частоты — 1,3-1,7 ГГц, корпус со штырьковыми выводами для сокета 423. Соответствующие им процессоры Celeron (с кэшем 128 Кбайт, частота — 1,7-2,8 ГГц) имеют аналогичный корпус. Процессоры для сокета 423 отличаются большими размерами: примерно 53 х 53 мм в плане. Конструкция корпуса OLGA (Organic Land Grid Array) довольно сложная: сам процессор с матрицей контактных площадок устанавливается на переходник (interposer) со штырьковыми выводами. Выводы расположены в шахматном порядке (четыре вывода образуют квадрат 2,54 х 2,54 мм, в центре — пятый вывод). Развитие процессоров Pentium 4 идет по нескольким направлениям: повышение тактовой частоты, применение более тонких технологий изготовления (130 нм, 90 нм, а затем и 65 нм), создание гиперпотоковых и мультиядерных процессоров, энергосбережение (IST и EIST, см. 7.4) и, наконец, 64-битное расширение.

Процессоры для сокета 478 появились в 2002 году сначала по технологии 0,18 мкм; по характеристикам они мало отличались от предшественников для сокета 423. С переходом на технологию 0,13 мкм повысили частоты ядра и системной шины, увеличили размер вторичного кэша и стали ставить на кристалл кэш 3-го уровня. Начиная с частоты 3,06 ГГц стали выпускать процессоры с поддержкой гиперпотоковости. С переходом на технологию 90 нм еще повысили частоту ядра (до 3,8 ГГц), вдвое увеличили первичный и вторичный кэш и ввели поддержку SSE3. Корпус процессора стал меньше (35 х 35 мм в плане, выводы размещаются в сетке с шагом 1,27 мм) и проще (микроPGA, без переходника).

Начиная с процессоров Pentium 4 с частотой 2,8 ГГц (и Celeron D, 2,26 ГГц) стали использовать корпус LGA 775 и соответствующий сокет. По сравнению с прежними сокетами здесь стало больше выводов для «земли» и питания, что необходимо при работе на более высоких частотах. Кроме того, изменилась конструкция: теперь в сокете расположены пружинистые контакты, которые упираются в площадки на корпусе процессора. Размер процессора остался примерно тем же (34 х 35 мм), как и шаг выводов (1,27 мм). Поскольку процессоры для этих сокетов обычно дороже системных плат, то поломка ножки (теперь сокета, а не процессора) обойдется дешевле. Для сокета 775 выпускаются много моделей процессоров, различающихся тактовой частотой, параметрами кэширования, числом физических и логических ядер, наличием (или отсутствием) 64-битного расширения.

Технология гиперпотоковости (hyperthreading), применение которой началось с процессоров Pentium 4 3,06 ГГц и Xeon, состоит в размещении на одном кристалле двух логических процессоров. Основная идея — совмещение по времени исполнения двух потоков инструкций и повышение эффективности использования дорогостоящих блоков процессора (два логических процессора их загрузят плотнее, чем один). *Мультиядерные процессоры* содержат несколько полноценных ядер. В процессорах Pentium Extreme Edition 840 находится пара гиперпотоковых ядер, каждое из которых имеет собственную двухуровневую систему кэширования. Таким образом, на одном физическом процессоре (микросхеме, установленной в сокет) можно получить 4-процессорную симметрич

ную систему (SMP-4). В облегченном варианте (Pentium D) гиперпотоковость не поддерживается, так что он выглядит как двухпроцессорный (и потребляет меньшую мощность). Гиперпотоковые и мультиядерные процессоры позволяют строить высокопроизводительные компьютеры на основе системных плат с одним обычным сокетом для LGA 775 (на нем присутствуют сигналы как бы одного процессора). Правда, для использования этих свойств требуется специальная поддержка со стороны чипсета и BIOS. Конфигурация процессора определяется значением сигналов на определенных входах в момент окончания сигнала RESET#. Так, например, установка низкого уровня сигнала A31# заставляет двухпроцессорную конструкцию работать в режиме одного логического процессора (до следующего сброса этот параметр уже не изменить).

Мультиядерность порождает специфику в управлении производительностью и потреблением, поскольку оба ядра питаются от одного источника, управляемого сигналами VID. Частота и напряжение питания связаны (см. описание IST и EIST в 7.4). Частоты ядер могут быть независимы, но общий сигнал VID будет соответствовать максимальной из двух частот, так что экономия от понижения частоты другого ядра оказывается неполной. Термоконтроль реализован на обоих ядрах, но на внешний интерфейс выводятся только один термодиод и один термодатчик (для совместимости со старыми одноядерными системами).

64-битное расширение EM64T фирма Intel стала использовать в 2005 году — значительно позже, чем компания AMD выпустила свои процессоры Athlon 64 и Opteron. 64-битные процессоры x86 от AMD и Intel практически совместимы (программно), хотя некоторые вещи (как и само расширение) у них называются по-разному. Вместе с 64-битным расширением появилась и дополнительная возможность защиты страниц памяти от исполнения программного кода. У Intel это называется функцией execute disable bit (в табл. 7.4 соответствующая графа называется «Бит NX»).

Для *серверов* фирма Intel позиционирует процессоры Xeon и Xeon MP (с поддержкой SMP). Для первых моделей этих процессоров используются корпуса (и сокеты) с числом выводов 603 *(Socket F),* для более поздних — 604. Дополнительный контакт служит ключом, препятствующим установке новых процессоров в старые системные платы. Размер процессоров в плане — 53 x 53 мм, шаг выводов — 1,27 мм. Позже появились процессоры Xeon 64 и Xeon 64 MP в таких же корпусах. Как и предыдущие процессоры Xeon (на базе Pentium II и Pentium III), эти процессоры имеют дополнительные узлы (термоконтроль, PIROM и EEPROM для идентификационной информации), доступные по отдельной шине SMBus (I^2C).

Для *мобильных компьютеров* выпускаются процессоры Pentium M и Celeron M, отличающиеся пониженным энергопотреблением и развитыми средствами энергосбережения. Среди них есть процессоры с пониженным (low voltage, 0,98-1,12 В) и даже «ультранизким» напряжением питания (ultra low voltage,

0, 81-0,94 В). Мобильные процессоры заметно дороже настольных, но привлекательно их относительно малое энергопотребление (следовательно, для них проще и тише система охлаждения). Так, например, процессор с максимальной

частотой 2,1 ГГц и вторичным кэшем 2 Мбайт потребляет от 11 до 28 Вт в зависимости от выбранной производительности (тактовой частоты и напряжения питания). Его настольный аналог потребляет раза в два большую мощность. Намечается тенденция использования мобильных процессоров и в настольных ПК. Мобильные процессоры выпускаются в малогабаритных корпусах как со штырьковыми (μPGA для установки в сокет с 478 или 479 контактами), так и с шариковыми выводами(μBGA для припайки). Заметим, что процессоры Mobile Celeron (0,13 мкм) с частотами 650-1133 МГц, устанавливаемые в сокеты с 478 и 479 контактами, относятся к P6 и по ядру (облегченный процессор Pentium III), и по интерфейсу системной шины.

7.8. Процессоры AMD

Фирма AMD начала выпускать процессоры, соответствующие шестому поколению по производительности и микроархитектуре, еще для системных плат с сокетами 5 и 7. Эти сокеты были разработаны фирмой Intel для своих процессоров пятого поколения. В отличие от Intel, «похоронившей» сокет 7 на частоте 233 МГц (последний процессор Pentium MMX), фирмы AMD, Cyrix и IBM выпускали для данного сокета процессоры с тактовой частотой до 533 МГц, догоняющие по возможностям процессоры Pentium II/III при более низких ценах. Однако в отличие от процессоров Intel ни один из них не обеспечивает работу в симметричных мультипроцессорных системах. Среди системных плат с сокетом 7 появилась категория «Super 7», подразумевающая возможность установки быстрых процессоров при частоте системной шины до 100 МГц с поддержкой порта AGP и некоторых других усовершенствований. При этом пропускная способность внешней шины не уступает шине Pentium II. Более экономичный (за счет простоты) протокол обмена частично компенсирует преимущества двойной независимой шины P6.

Фирма AMD выпускала несколько семейств процессоров, предназначенных для установки в сокет 7. Здесь кратко остановимся лишь на тех, которые тяготеют к шестому поколению.

Процессор *AMD Кб,* он же AMD K6 MMX (частота ядра до 300 МГц), выпущенный на месяц раньше Pentium II, по архитектуре ядра и свойствам напоминает Pentium II, но без встроенного вторичного кэша. Производительность AMD K6 200 для приложений Windows сопоставима с Intel Celeron 300. В этом процессоре отражены практически все достижения, имеющиеся в процессоре Pentium II, включая режимы управления энергопотреблением и тактированием. В режиме Stop Grant потребление снижается до сотен милливатт. В версии для мобильных применений AMD K6 300 потребляет всего 6,6 Вт.

Процессоры *AMD K6-2* представляют собой дальнейшее развитие Кб: введена технология 3DNow!, частота внешней шины поднята до 100 МГц. В процессоре K6-2+ (2000 г., 0,18 мкм) появился вторичный кэш на 128 Кбайт, работающий на частоте ядра (500 МГц). В 3DNow! введено расширение для DSP.

Процессор AMD K6-III (*Sharptooth*) является самым мощным процессором для сокета 7 (точнее, Super 7), и его название намекает на вызов процессору Pentium III. По результату ряда тестов производительности K6-III-450 МГц находится где-то между Pentium III-450 и 500 МГц. Главный козырь процессора K6-III — трехуровневая (!) система кэширования памяти. Здесь имеется первичный кэш размером 64 Кбайт (по 32 Кбайт для данных и инструкций) — это в два раза больше, чем у Pentium II и III. К тому же теперь к нему добавлен вторичный кэш размером 256 Кбайт, расположенный на одном кристалле с процессором и работающий на полной частоте ядра. Процессор K6-III устанавливается в плату с сокетом Super 7, на ней тоже может быть до 2 Мбайт кэш-памяти, которая становится кэшем 3-го уровня. Конечно, скорость обмена с этим кэшем уже не такая высокая (до 800 Мбайт/с), поскольку она ограничена частотой системной шины. В итоге процессор K6-III может располагать кэшем суммарным объемом до 2368 Кбайт, из которых 320 Кбайт доступны на полной частоте ядра. Команды 3DNow! имеют расширения для DSP. Для блокнотных ПК выпускались процессоры AMD-K6-III+ с частотой 500 МГц; технология 0,18 мкм.

Процессор Athlon (K7) по многим номинациям был признан лучшим процессором 1999 года. Название K7 обозначает его принадлежность к седьмому поколению (по AMD). Производительность достигается не только высокой тактовой частотой, но и особой суперконвейерной суперскалярной микроархитектурой. Эта архитектура при тех же прилагательных в названии существенно отличается от архитектуры других процессоров и AMD, и Intel. Заметим, что Athlon (и 7-е поколение) у AMD появился раньше, чем его конкурент — Pentium 4 у Intel. Система команд, помимо обычного набора инструкций 6-го поколения, включает команды MMX и *расширенной технологии 3DNow!* (3DNow!E). Преимущества процессора Athlon проявляются на приложениях с интенсивными вычислениями (особенно с плавающей точкой) и, естественно, оптимизированных под расширенную систему команд.

Процессор Athlon моделей 1 (технология 0,25 мкм) и 2 (0,18 мкм, начиная с 550 МГц) выполнен в виде картриджа с 242-контактным краевым разъемом и предназначен для установки в *слот A*. Механическая совместимость со слотом 1 была принята в угоду производителям системных плат, которые уже обзавелись всей инфраструктурой для картриджей процессоров Intel. Электрически процессоры Athlon и Pentium II/III *несовместимы*. Athlon имеет первичный кэш рекордного размера — 128 Кбайт (64 для данных и 64 для инструкций) и вторичный кэш, размещенный на картридже процессора.

В процессорах *Athlon модели 4* вторичный кэш размером 256 Кбайт размещен на одном кристалле с ядром, обмен с ним выполняется на полной частоте ядра. Однако, в отличие от Pentium III с интегрированным кэшем, в Athlon разрядность шины данных вторичного кэша осталась той же 8-байтной (а не 32-байт- ной), так что по скорости вторичного кэша Athlon проигрывает. Процессор упакован в корпус со штырьковыми выводами для *сокета A* (Socket-462), напоми

нающего сокет 370, но с дополнительными рядами контактов и механическими ключами (отсутствуют гнезда в некоторых позициях).

Процессор Duron — облегченный вариант K7 (кодовое название — Spitfire). Его вторичный кэш, уменьшенный до 64 Кбайт, но работающий на частоте ядра, располагается на кристалле ядра. Процессор выпускается в корпусе PGA для установки в сокет A.

Системная шина для Athlon и Duron, называемая *Athlon system bus*, основана на шине EV6 процессоров Alpha (фирмы Digital). В нее входят 64-битная двунаправленная шина данных с ECC-контролем, а также две встречные шины запросов и ответов для взаимодействия процессора и системы. В шине применяется синхронизация от источника данных. Частота передачи данных является удвоенной тактовой частотой. При тактовой частоте 100 МГц (частоте передачи 200 МГц) пропускная способность шины составляет 1,6 Гбайт/с. В спецификациях в качестве частоты системной шины указывают частоту передачи данных. В более поздних моделях процессоров частота передачи увеличена до 400 МГц. Протокол шины EV6 поддерживает мультипроцессирование, что является новинкой для AMD — предыдущие процессоры этой фирмы мультипроцессирование не поддерживали. При этом в мультипроцессорных конфигурациях каждому процессору выделяется собственная локальная шина.

Процессоры AMD отличаются большим размером первичного кэша, при этом размер вторичного кэша относительно невелик (Intel ставит кэш больше). Однако малый вторичный кэш используется экономнее — благодаря *эксклюзивной организаций* информация в кэшах разных уровней не дублируется.

Выпустив процессоры Athlon и Duron, фирма AMD на некоторое время заметно вырвалась вперед в гонке с Intel, однако появление и развитие Pentium 4 потеснило AMD в плане производительности. Процессоры Athlon и Duron стали недорогой альтернативой Pentium 4 — правда, и менее производительной. Новым прорывом для AMD был переход на 64-битную архитектуру, названную «x86-64» (см. http://www.x86-64.org). Эта архитектура введена в процессоры Athlon 64 и Opteron. Вдобавок к 3DNow!E процессоры с 64-битным расширением поддерживают команды SSE, SSE2, а более новые — и SSE3. Идею 64-битно- го расширения x86 с некоторым опозданием[2] подхватила и фирма Intel, правда, без явного указания на «первоисточники». Теперь блок XMM (с удвоенным количеством регистров) и команды SSE, SSE2 и SSE3 стали общепринятыми и для Intel, и для AMD. Процессоры с 64-битным расширением можно отнести к 8-му поколению процессоров x86.

Одновременно с расширением архитектуры в процессорах AMD была изменена и система подключения процессора к памяти и остальным компонентам системы. На процессорах установили контроллер DDR SDRAM с 64- или 128-бит- ной шиной памяти, к которой непосредственно подключаются модули памяти.

[1] Противоположность — инклюзивный вторичный кэш, в котором присутствуют блоки, находящиеся в первичном кэше.

[2] И с предшествовавшими заверениями о том, что этого расширения не будет.

Благодаря этому существенно уменьшилась задержка доступа к памяти. Второй интерфейс процессора — HyperTransport — связывает ядро с подсистемой ввода-вывода, а в мультипроцессорных системах — и с другими процессорами. Процессоры имеют 1 или 3 интерфейса HyperTransport 16x, каждый интерфейс обеспечивает пропускную способность 6,4 Гбайт/с.

Фирма AMD выпускает и двухъядерные процессоры Athlon 64 X2, о гиперпотоковых процессорах сообщений нет (при наличии двухъядерных Гиперпотоковые уже не так привлекательны).

По состоянию на весну 2005 года фирма AMD выпускает для *мультипроцессорных серверов* и *рабочих станций* процессоры Opteron (64-битный с широкой шиной памяти и тремя каналами HyperTransport) и Athlon MP (32-битный для сокета A).

Для настольных компьютеров предназначены 64-битные процессоры Athlon 64 FX и Athlon 64 (с 939 и 940 выводами) и 32-битные Sempron (754 или 462 вывода) и Athlon XP (462 вывода).

Для *мобильных применений* есть специальные версии 64-битных процессоров Turion 64 Mobile Technology, Mobile AMD Athlon 64 и AMD Athlon 64 for Notebooks. Есть и 32-битные мобильные процессоры Mobile AMD Sempron и Mobile AMD Athlon XP-M.

ГЛАВА 8

Электронная память

Электронная память применяется практически во всех подсистемах PC, выступая в качестве оперативной памяти, кэш-памяти, постоянной памяти, полупостоянной памяти, буферной памяти, внешней памяти.

Основная, или *оперативная*, *память* (main memory) компьютера используется для оперативного обмена информацией (командами и данными) между процессором, внешней памятью (например, дисковой) и периферийными подсистемами (графика, ввод-вывод, коммуникации и т. п.). Ее другое название — *ОЗУ* (оперативное запоминающее устройство) — примерно соответствует английскому термину *RAM* (Random Access Memory — память с произвольным доступом). Произвольность доступа подразумевает возможность операций записи в любую ячейку ОЗУ или чтения любой ячейки ОЗУ в произвольном порядке. Требования, предъявляемые к основной памяти:

- большой (для электронной памяти) объем, исчисляемый уже десятками — сотнями мегабайт и даже гигабайтами;
- быстродействие и производительность, позволяющие реализовать вычислительную мощность современных процессоров;
- высокая надежность хранения данных — ошибка даже в одном бите, в принципе, может привести к ошибкам вычислений, к искажению и потере данных, причем иногда и на внешних носителях.

Кэш-память (cache memory) — сверхоперативная память (СОЗУ), является буфером между ОЗУ и ее «клиентами» — процессором (одним или несколькими) и другими абонентами системной шины. Кэш-память не является самостоятельным хранилищем; информация в ней не адресуема клиентами подсистемы памяти, присутствие кэша для них «прозрачно». Кэш хранит копии блоков данных тех областей ОЗУ, к которым происходили последние обращения, и весьма вероятное последующее обращение к тем же данным будет обслужено кэш-памятью существенно быстрее, чем оперативной памятью. От эффективности алгоритма кэширования зависит вероятность нахождения затребованных данных в кэш-памяти и, следовательно, выигрыш в производительности памяти и компьютера в целом. Современные процессоры располагают встроенным кэшем. (Вопросы кэширования рассмотрены в главе 7.)

Постоянная память используется для энергонезависимого хранения системной информации — BIOS, таблиц знакогенераторов и т. п. Эта память при

обычной работе компьютера только считывается, а запись в нее (часто называемая программированием) осуществляется специальными устройствами — программаторами. Отсюда и ее название — *ROM* (Read Only Memory — память только для чтения), или *ПЗУ* (постоянное запоминающее устройство). Требуемый объем памяти этого типа невелик: например, объем BIOS PC/XT составлял 8 Кбайт, в современных компьютерах типовое значение от 128 Кбайт до 2 Мбайт. Быстродействие постоянной памяти обычно ниже, чем оперативной, но этот недостаток может быть исправлен применением теневой памяти (см. 4.2). В последние годы постоянную память вытесняют *флэш-память*, запись в которую возможна в самом компьютере в специальном режиме работы, и другие типы энергонезависимой памяти (EEPROM, FRAM).

Полупостоянная память в основном используется для хранения информации о конфигурации компьютера. Традиционная память конфигурации вместе с часами-календарем (CMOS Memory и CMOS RTC) имеет объем несколько десятков байт, ESCD (Extended System Configuration Data) — область энергонезависимой памяти, используемая для конфигурирования устройств Plug and Play, — несколько килобайт. Сохранность данных CMOS-памяти при отключении питания компьютера обеспечивается маломощной внутренней батарейкой или аккумулятором. В качестве полупостоянной применяется и *энергонезависимая память с произвольным доступом* (Non-Volatile Random Access Memory, NVRAM), которая хранит информацию и при отсутствии питания.

Буферная память различных адаптеров и контроллеров (коммуникационных, дисковых и пр.) обычно разделяется между процессором (точнее, абонентами системной шины) и контроллерами устройств. К этой памяти относятся и 16-байтные FIFO-буферы COM-портов, и несколькомегабайтные кэш-буферы высокопроизводительных устройств хранения. Специфическим типом буферной памяти является *видеопамять* дисплейного адаптера — к ней производятся интенсивные обращения со стороны центрального процессора и графического акселератора одновременно с непрерывным процессом регенерации изображения.

Электронная память применяется и в качестве *внешней памяти* — на флэш-каратах с различными интерфейсами и конструктивами (см. 9.10).

В зависимости от требований конкретной подсистемы ее память реализуется на микросхемах с различными принципами хранения информации, которые и рассматриваются в данной главе.

8.1. Структура оперативной памяти

Оперативная, или основная, память является одним из «трех китов», на которых держится «компьютерный мир» (процессор, память и периферийные устройства). Основной груз оперативного хранения информации ложится на динамическую память, на сегодняшний день имеющую наилучшее сочетание объема, плотности упаковки, энергопотребления и цены. Однако ей присуще невысокое (по меркам современных процессоров) быстродействие, и здесь на выручку приходит статическая память, быстродействие которой выше, но

достижимая емкость принципиально ниже, чем у динамической памяти. Обсудим основные параметры памяти — быстродействие, производительность, достоверность хранения и методы их улучшения, а также основные идеи организации кэширования памяти.

Быстродействие и производительность памяти

Быстродействие памяти определяется временем выполнения операций записи и считывания данных. Основными параметрами любых элементов памяти является минимальное время доступа и длительность цикла обращения. *Время доступа* (access time) определяется как задержка появления действительных данных на выходе памяти относительно начала цикла чтения, *длительность цикла* — как минимальный период следующих друг за другом обращений к памяти, причем циклы чтения и записи могут требовать различных затрат времени. В цикл обращения помимо активной фазы самого доступа входит и фаза восстановления (возврата памяти к исходному состоянию), которая соизмерима по времени с активной фазой. Временные характеристики самих запоминающих элементов определяются их принципом действия и технологией изготовления.

Производительность памяти можно характеризовать как скорость потока записываемых или считываемых данных и измерять в мегабайтах в секунду. Производительность подсистемы памяти наравне с производительностью процессора существенным образом определяет производительность компьютера. Выполняя определенный фрагмент программы, процессору придется, во-первых, загрузить из памяти соответствующий программный код, а во-вторых, произвести требуемые обмены данными, и чем меньше времени потребуется подсистеме памяти на обслуживание этих операций, тем лучше.

Производительность памяти, как основной, так и кэша второго уровня, обычно характеризуют *длительностью пакетных циклов чтения* (memory burst read cycle). Пакетный режим обращения является основным для процессоров, использующих кэш (класса 486 и выше); циклы чтения выполняются гораздо чаще, чем циклы записи (хотя бы потому, что процессору приходится все время считывать инструкции из памяти). Эта длительность выражается в числе тактов системной шины, требуемых для передачи очередной порции данных в пакете. Обозначение вида 5-*3*-*3*-*3* для диаграммы пакетного цикла чтения соответствует пяти тактам на считывание первого элемента в цикле и трем тактам на считывание каждого из трех последующих элементов. Первое число характеризует латентность (latency) памяти — время ожидания данных, последующие — скорость передачи. При этом, конечно же, оговаривается и частота системной шины.

Производительность подсистемы памяти зависит от *типа* и *быстродействия* применяемых запоминающих элементов, *разрядности* шины памяти и некоторых «хитростей» архитектуры. Современные типы памяти обеспечивают высокую скорость передачи внутри пакета, используя двойную и даже четырехкратную синхронизацию. При этом параметром шины, по которой передаются данные, может быть как *частота тактового сигнала*, так и *частота передачи данных*.

Последняя может в 2 (DDR SDRAM) или в 4 (DDR2 SDRAM, шина Pentium 4) раза превышать тактовую частоту. Задержка получения данных чтения процессорным ядром в современных компьютерах может составлять от 45 до нескольких сотен наносекунд в зависимости от способа подключения памяти.

Производительность микросхем или модулей памяти повышают применением различных вариантов конвейеризации, о чем подробнее рассказывается далее.

Разрядность шины памяти — это количество байтов (или битов), с которыми операция чтения или записи может быть выполнена одновременно. Разрядность основной памяти обычно согласуется с разрядностью внешней шины процессора (1 байт — для 8088; 2 байта — для 8086, 80286, 386SX; 4 байта — для 386DX, 486; 8 байт — для Pentium и выше). Вполне очевидно, что при одинаковом быстродействии микросхем или модулей памяти производительность блока с большей разрядностью будет выше, чем у малоразрядного. Именно с целью повышения производительности у 32-битных (по внутренним регистрам) процессоров класса Pentium и выше внешняя шина, связывающая процессор с памятью, имеет разрядность 64 бита. У современных процессоров пропускная способность системной шины превышает пропускную способность шины памяти. Это подталкивает к использованию *двухканальной памяти* — удвоению разрядности шины памяти относительно разрядности системной шины процессора.

Банком памяти называют комплект микросхем или модулей (а также их посадочных мест — «кроваток» для микросхем, слотов для SIMM или DIMM), обеспечивающий требуемую для данной системы разрядность хранимых данных. Работоспособным может быть только полностью заполненный банк. Внутри одного банка практически всегда должны применяться одинаковые (по типу и объему) элементы памяти.

В современных компьютерах на процессорах 6-8-го поколений банком является один модуль DIMM или RIMM (подобный модуль может содержать и несколько банков, см. далее).

Если устанавливаемый объем памяти набирается несколькими банками, появляется резерв повышения производительности за счет *чередования банков* (bank interleaving). Идея чередования заключается в том, что смежные блоки данных (разрядность такого блока данных соответствует разрядности банка) располагаются поочередно в разных банках. Тогда при весьма вероятном последовательном обращении к данным банки будут работать поочередно, причем активная фаза обращения к одному банку может выполняться во время фазы восстановления другого банка, то есть применительно к обоим банкам не будет простоя во время фазы восстановления. Частота передачи данных в системе с чередованием двух банков может быть удвоенной по отношению к максимальной частоте работы отдельного банка. Для реализации механизма чередования чипсет должен обеспечивать возможность перекоммутации адресных линий памяти в зависимости от установленного количества банков и иметь для них (банков) раздельные линии управляющих сигналов. Чем больше банков участвуют в чередовании, тем выше (теоретически) предельная производительность. Чаще всего в чередовании участвуют два банка (two way interleaving), но

их может быть и больше. Из разбиения на мелкие банки можно извлечь и другую выгоду. Поскольку современные процессоры способны параллельно выставлять несколько запросов на транзакции с памятью, обусловленные необходимым временем доступа скрытые фазы обработки запросов, относящихся к разным банкам, могут выполняться одновременно.

Микросхемы памяти SDRAM (см. далее) имеют внутреннюю мультибанковую организацию, применительно к этой памяти в данном контексте (комплекте микросхем) используют понятие *физический банк*, или *ряд* (row).

Достоверность хранения данных

В любой из многих миллионов ячеек памяти возможен случайный сбой или окончательный отказ, приводящий к ошибке. Вероятность ошибки, естественно, возрастает с увеличением объема памяти. Современные технологии позволяют выпускать высоконадежные микросхемы памяти, у которых при корректной эксплуатации (то есть при соблюдении заданных характеристик напряжения питания, температуры, временной диаграммы, уровнях сигналов, нагрузки на выходные шины...) вероятность ошибки довольно мала, но все-таки она не нулевая.

Отказ ячейки памяти — потеря ее работоспособности, обычно требующая замены элемента памяти. Отказ может быть устойчивым, но возможно и самопроизвольное восстановление работоспособности, например после повторного включения питания. Часто причиной отказов является неисправность контакта или нарушение условий эксплуатации.

Случайный *сбой* может произойти и в исправной микросхеме памяти, например при пролете через нее ионизирующей частицы (по этой причине в условиях высокого уровня радиации обычные электронные элементы неработоспособны). После сбоя следующая же запись в ячейку произойдет нормально.

В первых моделях PC, когда микросхемы памяти имели существенно худшие характеристики надежности по сравнению с современными, обязательно применялся контроль четности. В этом случае каждый байт памяти сопровождается битом четности (parity bit), дополняющим количество единиц в байте до нечетного. Значение бита четности аппаратно генерируется при записи в память и проверяется при считывании. При обнаружении ошибки четности схемой контроля вырабатывается немаскируемое прерывание (NMI), его обработчик обычно выводит на экран сообщение «Parity Check Error» (ошибка четности) с указанием адреса сбойной ячейки и останавливает процессор командой Halt.

Команда Halt (останов) останавливает выполнение текущего потока команд процессора. Из состояния останова процессор может выйти только по прерываниям, которые обработчиком ошибки четности могут и блокироваться. Останов не позволяет процессору «перепахать» всю память (в том числе и внешнюю), что, в принципе, возможно при возникновении ошибки в памяти. «Сдвинуть с места» процессор, остановленный таким образом, можно нажатием кнопки Reset, выключением и последующим включением питания, а иногда и нажатием клавиш Ctrl+Alt+Del.

Специальными битами конфигурационных регистров (порт 061h) вырабатывание прерывания NMI от схем контроля памяти можно запретить (см. 4.4), также можно изменить процедуру обработки NMI для игнорирования ошибки (в диагностических целях). При выполнении начального тестирования памяти в рамках теста POST выполняется запись во все ячейки ОЗУ, в результате чего формируются правильные биты четности (при исправной памяти).

Со временем качество применяемых микросхем памяти улучшилось, и в целях удешевления модулей памяти от контроля четности стали отказываться — сначала через установку параметра CMOS Setup предлагали на выбор, проверять или не проверять четность, а потом появилась масса моделей системных плат, в которых контроля четности нет вообще. Модули памяти (SIMM, SIPP) стали выпускать как с битом четности, так и без него, а для «ублажения» (точнее — обмана) плат, требующих наличия такого бита, стали выпускать модули с «подделкой» четности (fake parity). В этих модулях вместо дополнительной микросхемы памяти используется генератор четности (Parity Generator, PG) — логическая схема сумматора по модулю 2, формирующая всегда «хороший» бит четности независимо от ошибок в самой памяти. Его наличие можно распознать визуально — логическая микросхема на модуле заметно отличается от микросхем памяти. В обозначение типа модуля с генератором четности обычно входят буквосочетания BP, VT, GSM или MPEC. Заметим, что никакие программные средства тестирования памяти (Checklt, PCCheck и т. п.) не позволяют отличить память с настоящим битом четности от памяти с фиктивным битом четности.

Вопрос о необходимости контроля четности не имеет однозначного ответа. По мнению автора, лучше столкнуться с остановкой машины по ошибке с явным указанием на ее источник, чем искать причины непонятных зависаний и «вылетов», результатом которых, в принципе, может стать и полная потеря данных на диске. Но контроль четности не всесилен, он выявляет в пределах каждого байта ошибки только нечетной кратности (искажение одного, трех, пяти или семи битов): правда, вероятность одновременного отказа или сбоя двух битов у работающей памяти весьма мала. Кроме того, многочисленные наблюдения дают основания полагать, что при загадочных неполадках в памяти (когда все тесты проходят нормально), возможно, происходит обращение по ложным адресам ячеек памяти, что никаким контролем четности не выявляется.

Попутно заметим, что останов по ошибке четности при программном тестировании памяти компьютера с включенным контролем четности может происходить от ошибки как в информационных, так и в контрольных битах. В этом случае для выявления неисправности отключают бит четности. Схему контроля четности такими тестами проверять можно только косвенно, устанавливая заведомо исправный модуль памяти.

Побайтный контроль четности шины данных встроен в микропроцессоры, начиная с класса 486, что упрощает схемы контроля.

В компьютерах особо ответственного применения используют память с обнаружением и коррекцией ошибок (Error Checking and Correcting, ECC). В этом случае для каждого записываемого информационного слова памяти (а не байта,

как при контроле четности) по определенным правилам вычисляется функция свертки, результат которой разрядностью в несколько битов также хранится в памяти. Для 64-битного слова обычно используют 7-8 дополнительных битов. При считывании схема контроля с использованием этих избыточных битов способна обнаруживать ошибки различной кратности *и/или* исправлять однократные ошибки. Обнаружение ошибки выполняется «на лету» и само по себе дополнительного времени не требует. Однако при исправлении ошибок требуются дополнительные такты для срабатывания аппаратной логики исправления и фиксации скорректированного результата. Даже при отсутствии ошибки нужен по крайней мере один лишний такт для срабатывания схемы контроля, разрешающей использование считанных данных. Таким образом, ECC-память работает несколько медленнее неконтролируемой, и при наличии исправимых ошибок замедление становится заметнее. Функцию контроля и исправления выполняет чипсет, его реакцию на ошибки обычно можно задать настройкой CMOS Setup. Возможны различные варианты поведения, например:

- автоматически исправлять ошибки, не уведомляя об этом систему;
- исправлять однократные ошибки, уведомляя систему только о многократных;
- не исправлять ошибки, а только уведомлять об их обнаружении (самый достоверный контроль).

В отличие от памяти с контролем четности, допускающей побайтное обращение, к ECC-памяти можно обращаться только полноразрядными словами. Заботу об этом берет на себя чипсет.

В современных компьютерах схема ECC широко применяется в кэш-памяти — в процессорах P6 и выше встроенный вторичный кэш, как правило, имеет ECC-контроль, выполняемый схемами процессора. При «разгоне» процессора ECC-контроль может подтормаживать кэш, в котором начинают появляться ошибки, но отключение ECC-контроля (настройкой CMOS Setup) ради ускорения чревато неконтролируемыми ошибками.

Достоверность информации, хранимой в постоянной (ROM BIOS) и полупостоянной (CMOS RTC, ESCD) памяти, проверяется с помощью *контрольной суммы* (checksum) — обычно это байт, дополняющий до нуля сумму по модулю 256 всех байтов контролируемой области. Проверка контрольной суммы обычно выполняется однократно во время теста POST.

Кэширование оперативной памяти

Основная память компьютеров реализуется на относительно медленной динамической памяти (DRAM), обращение к ней приводит к простою процессора — появляются такты ожидания (wait states). Статическая память (SRAM), построенная, как и процессор, на триггерных ячейках, по своей природе способна догнать современные процессоры по быстродействию и сделать ненужными такты ожидания (или хотя бы сократить их количество). Разумным компромиссом для построения экономичных и производительных систем явился иерархический способ организации оперативной памяти. Идея заключается в со

четании основной памяти большого объема на DRAM с относительно небольшой кэш-памятью на быстродействующих микросхемах SRAM.

В переводе слово «cache» (кэш) означает «тайный склад», «тайник», «заначка». Тайна этого склада заключается в его «прозрачности» — адресуемой области памяти для программы он не добавляет. Кэш является дополнительным быстродействующим хранилищем копий блоков информации из основной памяти, вероятность обращения к которым в ближайшее время велика. Кэш не может хранить копию всей основной памяти, поскольку его объем во много раз меньше объема основной памяти. Он хранит лишь ограниченное количество блоков данных и *каталог* (cache directory) — список их текущего соответствия областям основной памяти. Кроме того, кэшироваться может и не вся оперативная память, доступная процессору: во-первых, из-за технических ограничений может быть ограничен максимальный объем кэшируемой памяти; во-вторых, некоторые области памяти могут быть объявлены некэшируемыми (настройкой регистров чипсета или процессора). Если установлено больше оперативной памяти, чем возможно кэшировать, обращение к некэшируемой области ОЗУ будет медленным. Таким образом, увеличение объема ОЗУ, теоретически всегда благотворно влияющее на производительность, может снизить скорость работы определенных компонентов, попавших в некэшируемую память. В ОС Windows память распределяется, начиная с верхних адресов физической памяти в результате в некоторых конфигурациях в некэшируемую область может попасть ядро ОС.

При каждом обращении к памяти контроллер кэш-памяти по каталогу проверяет, есть ли действительная копия затребованных данных в кэше. Если она там есть, то это случай *кэш-попадания* (cache hit) и данные берутся из кэш-памяти. Если действительной копии там нет, это случай *кэш-промаха* (cache miss) и данные берутся из основной памяти. В соответствии с алгоритмом кэширования блок данных, считанный из основной памяти, при определенных условиях замещает один из блоков кэша. От интеллектуальности алгоритма замещения зависит процент попаданий и, следовательно, эффективность кэширования. Поиск блока в списке должен производиться достаточно быстро, чтобы «задумчивостью» в принятии решения не свести на нет выигрыш от применения быстродействующей памяти. Обращение к основной памяти может начинаться одновременно с поиском в каталоге, а в случае попадания — прерываться (архитектура look aside). Это экономит время, но лишние обращения к основной памяти ведут к увеличению энергопотребления. Другой вариант: обращение к основной памяти начинается только после фиксации промаха (архитектура look through); при этом теряется по крайней мере один такт процессора, зато экономится энергия.

Режим пакетной передачи данных

Режим пакетной передачи (burst mode) предназначен для ускорения операций пересылки строк кэша. Этот режим появился в процессорах 486. Строка кэша процессора 486 имеет длину 16 байт, следовательно, для ее пересылки требуется четыре 32-разрядных шинных цикла. Для пересылки 32-байтной строки

кэша Pentium+ требуется тоже четыре такта, поскольку разрядность передач составляет 64 бита. Использование кэша предполагает, что строка должна в нем присутствовать целиком. Пакетный цикл (burst cycle) оптимизирован именно для операций обмена внутреннего кэша с оперативной памятью. В этом цикле адрес и сигналы идентификации типа шинного цикла выдаются только в первом такте пакета, а в каждом из последующих тактов могут передаваться данные, адрес которых уже не пересылается по шине, а вычисляется из первого адреса по правилам, известным процессору и контроллеру памяти. Пакетные циклы связаны только с кэшируемой памятью, при этом кэшируемость памяти подразумевает и поддержку пакетного режима. Один пакетный цикл не может пересекать границу строки кэша. Специфический порядок следования адресов в пакетном цикле определяется начальным адресом пакета. Порядок чередования (interleaving) адресов, который иллюстрирует табл. 8.1, в пакетном цикле характерен для всех процессоров Intel и совместимых с ними, начиная с 486. Он оптимизирован для двухбанковой организации памяти, подразумевающей чередование банков, используемых в соседних передачах пакетного цикла. С точки зрения памяти, у каждой микросхемы во время пакетного цикла могут изменяться только два младших бита адреса (независимо от разрядности шины данных процессора). Данный порядок чередования поддерживает любая память с пакетным режимом: динамическая память BEDO DRAM, SDRAM, RDRAM и статическая память Sync Burst SRAM, PB SRAM. Процессоры других семейств (например, Power PC) используют линейный (linear) порядок адресов в пакете. Микросхемы пакетной памяти обычно имеют входной сигнал или бит в регистре режима, задающий порядок следования адресов (линейный или с чередованием) для конкретного применения. Длина пакетного цикла может быть фиксированной или программируемой и составлять 2, 4 или 8 передач.

Таблица 8.1. Последовательность адресов в пакетном цикле Pentium+ (длина цикла — 4)

Первый адрес	Второй адрес	Третий адрес	Четвертый адрес
0	8	10h	18h
8	0	18h	10h
10h	18h	0	8
18h	10h	8	0

Временная диаграмма пакетных циклов обращения к памяти (главным образом, чтения) является основной характеристикой производительности памяти компьютера. Ее описывают числом тактов системной шины, требуемых для каждой передачи пакета. При этом, естественно, оговаривают и саму частоту. Так, для динамической памяти BEDO-50 нс достижимый идеал — цикл 5-1-1-1 на частоте 66 МГц.

8.2. Динамическая память

Динамическая память (Dynamic RAM, DRAM) получила свое название от принципа действия ее запоминающих ячеек, которые выполнены в виде кон-

денсаторов, образованных элементами полупроводниковых микросхем[1]. Несколько упрощая описание физических процессов, можно сказать, что при записи логической единицы в ячейку конденсатор заряжается, при записи нуля — разряжается. Схема считывания разряжает через себя этот конденсатор, и если заряд был ненулевым, выставляет на своем выходе единичное значение и подзаряжает конденсатор до прежнего уровня. При отсутствии обращения к ячейке со временем за счет токов утечки конденсатор разряжается и информация теряется, поэтому такая память требует постоянной периодической подзарядки конденсаторов (обращения к каждой ячейке), то есть память может работать только в динамическом режиме. Этим она принципиально отличается от статической памяти, реализуемой на триггерных ячейках и хранящей информацию без обращений к ней сколь угодно долго (при включенном питании). Благодаря относительной простоте ячейки динамической памяти на одном кристалле удается размещать миллионы ячеек и получать самую дешевую полупроводниковую память достаточно высокого быстродействия с умеренным энергопотреблением, используемую в качестве основной памяти компьютера. Расплатой за низкую цену являются некоторые сложности в управлении динамической памятью, которые рассматриваются далее.

Основы работы DRAM

Запоминающие ячейки микросхем DRAM организованы в виде двухмерной матрицы. Адрес строки и столбца передается по мультиплексированной шине адреса MA (Multiplexed Address) и стробируется по спаду импульсов RAS# (Row Access Strobe — строб адреса строки) и CAS# (Column Access Strobe — строб адреса столбца). Состав сигналов микросхем динамической памяти приведен в табл. 8.2.

Таблица 8.2. Сигналы микросхем динамической памяти

Сигнал	Назначение
RAS#	Row Access Strobe – строб выборки адреса строки. По спаду сигнала начинается любой цикл обращения, низкий уровень сохраняется на все время цикла. Перед началом следующего цикла сигнал должен находиться в неактивном состоянии (высокий уровень) не менее чем время предварительного заряда RAS (RAS precharge time) – T_{RP}
CAS#	Column Access Strobe – строб выборки адреса столбца. По спаду сигнала начинается цикл записи или чтения, минимальная длительность (T_{CAS}) определяется спецификацией быстродействия памяти. Минимальная длительность неактивного состояния между циклами (высокий уровень) должна быть не менее, чем время предварительного заряда CAS (CAS precharge time) – T_{CP}
MAi	Multiplexed Address – мультиплексированные линии адреса. Во время спада сигнала RAS# на этих линиях присутствует адрес строки, во время спада CAS# – адрес столбца. Адрес должен устанавливаться до спада соответствующего строба и удерживаться после него еще некоторое время. Микросхемы могут быть с симметричной организацией (например, при объеме 2^{22} ячеек) – 11 битов адреса строк и 11 битов адреса столбцов или асимметричными – 12 × 10 бит соответственно

продолжение

[1] Это «паразитная» емкость затвора полевого транзистора, которая в данном случае оказывается весьма полезной.

Таблица 8.2 *(продолжение)*

Сигнал	Назначение
WE#	Write Enable — разрешение записи. Данные записываются в выбранную ячейку либо по спаду CAS# при низком уровне WE# (Early Write — ранняя запись, обычный вариант), либо по спаду WE# при низком уровне CAS# (Delayed Write — задержанная запись). Переход WE# в низкий уровень и обратно при высоком уровне CAS# записи не вызывает, а только переводит выходной буфер EDO DRAM в высокоимпедансное состояние
OE#	Output Enable — разрешение открытия выходного буфера при операции чтения. Высокий уровень сигнала в любой момент переводит выходной буфер в высокоимпедансное состояние
DB-In	Data Bit Input — входные данные (только для микросхем с однобитной организацией)
DB-Out	Data Bit Output — выходные данные (только для микросхем с однобитной организацией). Выходные буферы стандартных микросхем открыты только при сочетании низкого уровня сигналов RAS#, CAS#, OE# и высокого уровня WE#; при невыполнении любого из этих условий буферы переходят в высокоимпедансное состояние. У микросхем EDO выходные буферы открыты и после подъема CAS#. Логика управления предусматривает возможность непосредственного объединения выходов нескольких микросхем
DQx	Data Bit — объединенные внутри микросхемы входные и выходные сигналы данных (объединение экономит количество выводов для микросхем с многобитной организацией)
N.C.	No Connection — свободный вывод

Выбранной микросхемой памяти является та, на которую во время активности (низкого уровня) сигнала RAS# приходит сигнал CAS# (тоже низким уровнем). Тип обращения определяется сигналами WE# и CAS#. Временная диаграмма «классических» циклов записи и чтения приведена на рис. 8.1. Как видно из диаграммы, при чтении данные на выходе относительно начала цикла (сигнала RAS#) появятся не раньше, чем через интервал T_{RAC}, который и является *временем доступа.*

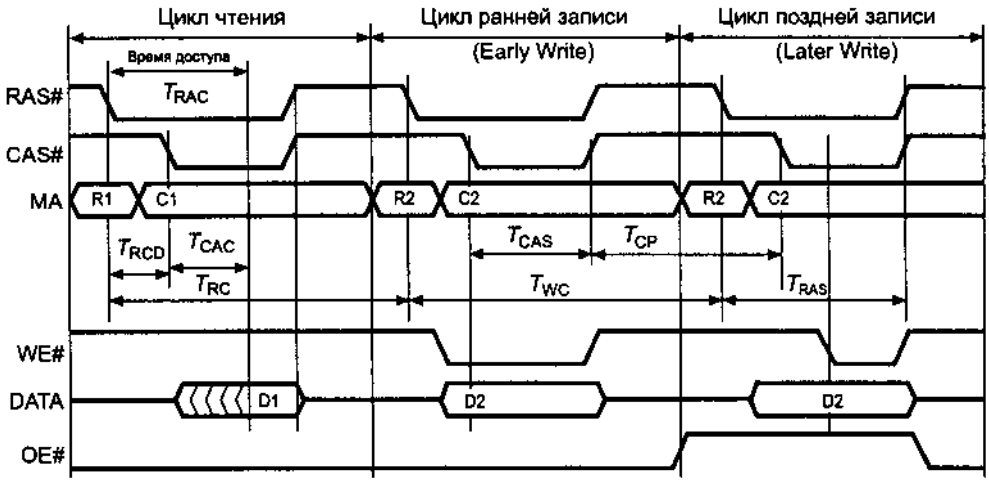


Рис. 8.1. Временные диаграммы чтения и записи динамической памяти

Микросхемы DRAM имеют много различных временных параметров. Выделим несколько важнейших, с которыми иногда приходится сталкиваться при настройке параметров циклов в CMOS Setup:

- Время доступа (RAS Access Time) — T_{RAC} — задержка появления действительных данных на выходе относительно спада импульса RAS (см. рис. 8.1). Этот основной параметр спецификации памяти, измеряемый в единицах или десятках наносекунд, обычно является последним элементом обозначения микросхем и модулей (названия xxx-7 и xxx-70 означают время доступа 70 нс). Для современных микросхем характерно время доступа 30-100 нс.
- Время цикла (cycle time) — минимальный период между началами соседних циклов обращения (T_{wc} для записи и T_{RC} для чтения). Для современных микросхем лежит в пределах 75-125 нс.
- Время цикла (период следования импульсов CAS#) в страничном режиме (Page CAS Time) — T_{PC}.
- Длительность сигналов RAS# и CAS# — T_{RAS} и T_{CAS} — минимальная длительность активной части (низкого уровня) стробирующих сигналов (см. рис. 8.1).
- Время предварительного заряда RAS и CAS (RAS и CAS Precharge Time) — T_{RP} и T_{CP} — минимальное время нахождения соответствующих сигналов в высоком состоянии.
- Время задержки между импульсами RAS# и CAS# (RAS to CAS Delay) - T_{RCD}.
- Задержка данных относительно импульса CAS# — T_{CAC}.

Все эти параметры и определяют предел производительности памяти. В табл. 8.3 приведены типовые значения временных параметров, отвечающие конкретной спецификации быстродействия. На них можно было ориентироваться при задании циклов обращений к асинхронной динамической памяти (настройкой CMOS Setup), но при этом необходимо учитывать, что микросхемы различных производителей могут несколько отличаться друг от друга по отдельным параметрам.

Таблица 8.3. Ключевые параметры временной диаграммы DRAM

Спецификация быстродействия	T_{RC}, нс	T_{RAC}, нс	T_{PC}, нс	T_{CAS}, нс	T_{CP}, нс
-4	75	40	15	6	6
-5	100	50	20	8	8
-6	104	60	25	10	10
-7	110	70	30	12	12

Ключевой параметр микросхем — время доступа — за всю историю удалось улучшить всего на порядок (с сотен до нескольких десятков наносекунд). За меньший исторический период только тактовая частота процессоров x86 выросла на 3 порядка, так что разрыв между потребностями процессоров и возможностями ячеек памяти увеличивается. Для преодоления этого разрыва, во-первых, увеличивают разрядность данных памяти, а во-вторых, строят вокруг массивов

ячеек памяти разные хитрые оболочки, ускоряющие доступ к данным. В первую очередь стараются оптимизировать чтение, поскольку операции записи по сравнению с чтением в большинстве случаев выполняются гораздо реже. Отметим, что все, даже «самые модные» типы памяти — SDRAM, DDR SDRAM и Rambus DRAM — имеют запоминающее ядро, которое обслуживается описанным выше способом.

Регенерация

Поскольку обращения (запись или чтение) к различным ячейкам памяти обычно происходят в случайном порядке, для поддержания сохранности данных применяется регенерация *памяти* (memory refresh) — регулярный циклический перебор ее ячеек (обращение к ним) с холостыми циклами. Регенерация в микросхеме происходит одновременно по всей строке матрицы при обращении к любой из ее ячеек. Максимальный период обращения к каждой строке T_{RF} (refresh time) для гарантированного сохранения информации у современной памяти лежит в пределах 8-64 мс. В зависимости от объема и организации матрицы для однократной регенерации всего объема требуется 512, 1024, 2048 или 4096 циклов обращений. При *распределенной регенерации* (distributed refresh) одиночные циклы регенерации выполняются равномерно с периодом t_{RF} (рис. 8.2, а), который для стандартной памяти принимается равным 15,6 мкс. Период этих циклов называют *частотой регенерации* (refresh rate), хотя такое название больше соответствует обратной величине — частоте циклов $f = 1/t_{RF}$ Для памяти с расширенной регенерацией (extended refresh) допустим период циклов до 125 мкс. Возможен также и вариант *пакетной регенерации* (burst refresh), когда все циклы регенерации собираются в пакет (рис. 8.2, *б*), во время которого обращение к памяти по чтению и записи блокируется. При количестве циклов 1024 эти пакеты будут периодически занимать шину памяти примерно на 130 мкс, что далеко не всегда допустимо. По этой причине, как правило, выполняется распределенная регенерация, хотя возможен и промежуточный вариант — пакетами по несколько (например, 4) циклов.

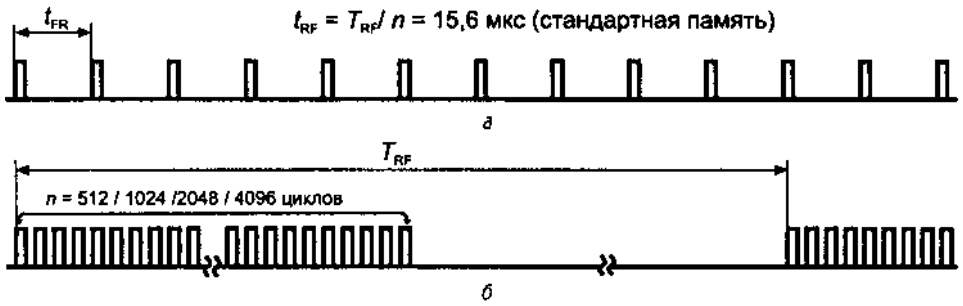


Рис. 8.2. Регенерация динамической памяти: а - распределенная, *б* — пакетная

Циклы регенерации могут организовываться разными способами. Классическим является цикл без импульса CAS# (рис. 8.3, слева), сокращенно именуемый

ROR (RAS Only Refresh — регенерация только импульсом RAS#). В этом случае адрес очередной регенерируемой строки выставляется *контроллером памяти* до спада RAS# очередного цикла регенерации, порядок перебора регенерируемых строк не важен.

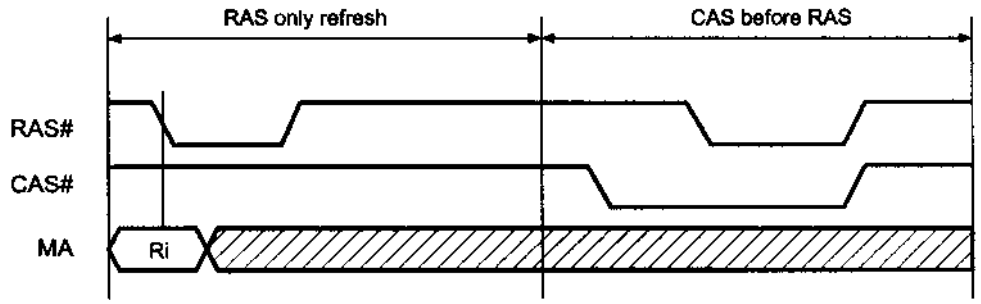


Рис. 8.3. Циклы регенерации динамической памяти: слева — ROR, справа — CBR

Другой вариант — цикл *CBR* (CAS Before RAS), поддерживаемый практически всеми современными микросхемами памяти (рис. 8.3, *справа)*. В этом цикле регенерации спад импульса RAS# происходит при низком уровне сигнала CAS# (в обычном цикле обращения такой ситуации не возникает). В этом случае микросхема выполняет регенерацию строки, адрес которой находится во *внутреннем счетчике микросхемы*, и в задачу контроллера входит только периодическое формирование таких циклов. Во время спада RAS# сигнал WE# должен находиться в состоянии высокого уровня. Дополнительным преимуществом данного цикла является экономия потребляемой мощности за счет неактивности внутренних адресных буферов.

Микросхемы синхронной динамической памяти выполняют циклы CBR по команде *Auto Refresh.* А по команде *Self Refresh* или *Sleep Mode* они производят автономную регенерацию в энергосберегающем режиме. Такой возможностью обладают некоторые современные микросхемы, имеющие внутренний генератор. Вход в режим осуществляется, как в цикл CBR, но сигнал RAS# должен быть активен более 100 мкс. Информация в таком состоянии хранится сколь угодно долго при наличии питающего напряжения. Выход из этого «спящего» состояния осуществляется по подъему сигналов RAS# и CAS#.

Цикл *скрытой регенерации* (hidden refresh) является разновидностью цикла CBR: здесь в конце полезного цикла чтения или записи сигнал CAS# удерживается на низком уровне, а RAS# поднимается и снова опускается, что и является указанием микросхеме на выполнение цикла регенерации по внутреннему счетчику (рис. 8.4). При этом слово «скрытость» не всегда означает экономию времени (затраты на регенерацию остаются теми же, что и в обычном цикле CBR, хотя, в принципе, возможно предельное укорочение активной части импульса CAS# при чтении). Во время скрытой регенерации после цикла чтения выходные буферы сохраняют только что считанные данные (в обычном цикле CBR выходные буферы находятся в высокоимпедансном состоянии).

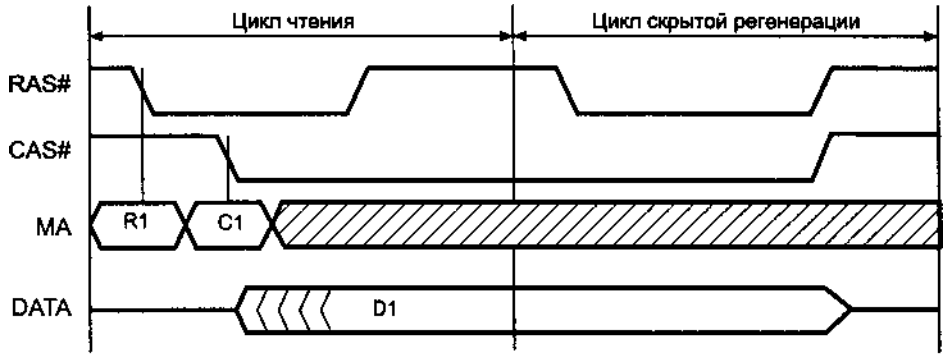


Рис. 8.4. Скрытая регенерация

Регенерация основной памяти в PC/XT осуществлялась каналом DMA-O. Сигнал Refr, вырабатываемый каждые 15,6 мкс по сигналу от первого канала таймера-счетчика 8253/9254 (порт 041h), вызывает холостой цикл обращения к памяти для регенерации очередной строки. В PC/AT контроллер регенерации усложнен. В современных компьютерах регенерацию основной памяти берет на себя чипсет, и его задача — по возможности использовать для регенерации циклы шины, не занятые ее абонентами (процессорами и активными контроллерами). Самые «ловкие» контроллеры выполняют интеллектуальную регенерацию *(smart refresh)* — ставят запросы на регенерацию в очередь, которую обслуживают в свободное для шины время, и только если запросов накапливается больше предельного количества, откладывается текущий цикл обмена по шине и цикл регенерации выполняется немедленно. Модули памяти в разных банках могут регенерироваться одновременно, но в условиях чередования для экономии времени целесообразно производить регенерацию одного банка во время полезного обращения к другому. Некоторые системные платы позволяют использовать режим пониженной частоты регенерации *(slow refresh*), однако он доступен только с модулями памяти, допускающими расширенную регенерацию.

Динамическая память, используемая в видеобуферах графических адаптеров, специальных циклов регенерации, как правило, не требует, поскольку частота ее чтения при регенерации изображения вполне достаточна для сохранения информации.

Асинхронная память — FPM, EDO и BEDO DRAM

Временная диаграмма, приведенная на рис. 8.1, может быть модифицирована для случая последовательного обращения к ячейкам, принадлежащим одной строке матрицы. В этом случае адрес строки выставляется на шине только один раз, и сигнал RAS# удерживается на низком уровне на время всех последующих обращений, которые могут быть циклами как записи, так и чтения. Такой режим обращения называется *режимом быстрого страничного обмена* (Fast Page Mode, FPM), или просто *режимом страничного обмена* (page mode), его временная диаграмма приведена на рис. 8.5. Понятие «страница» на самом деле отно

сится к строке (row), а состояние с низким уровнем сигнала RAS# называется «открытой страницей». Преимущество данного режима заключается в экономии времени за счет исключения фазы выдачи адреса строки из циклов, следующих за первым, что позволяет повысить производительность памяти. Для памяти с временем доступа 60 не время цикла обмена внутри страницы может быть сокращено до 35 нс. Способность работать в режиме FPM является «заслугой» не микросхем или модулей памяти (в этом режиме могут работать и самые «древние» микросхемы, и микросхемы EDO, о которых речь пойдет далее), а контроллера динамической памяти (то есть чипсета). Однако по сложившейся терминологии обозначение FPM относят к «стандартным» микросхемам и модулям динамической памяти, которые не являются памятью EDO, BEDO или SDRAM. Иногда их все-таки более точно называют стандартными *(Std)*.

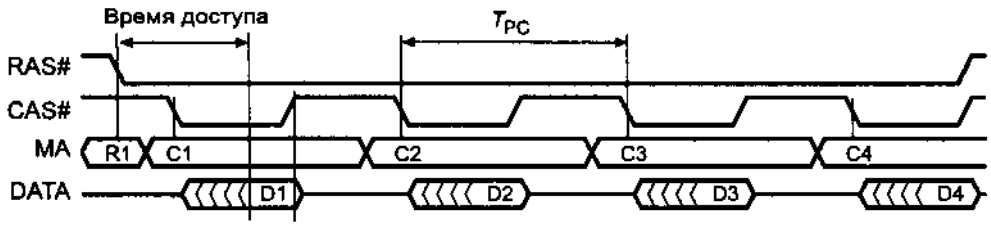


Рис. 8.5. Страничный режим считывания стандартной памяти DRAM (FPM)

Повысить производительность памяти FPM можно путем уже упоминавшегося *чередования банков* (bank interleaving): считывание (или запись) данных одного банка выполнять во время предварительного заряда другого банка (рис. 8.6).

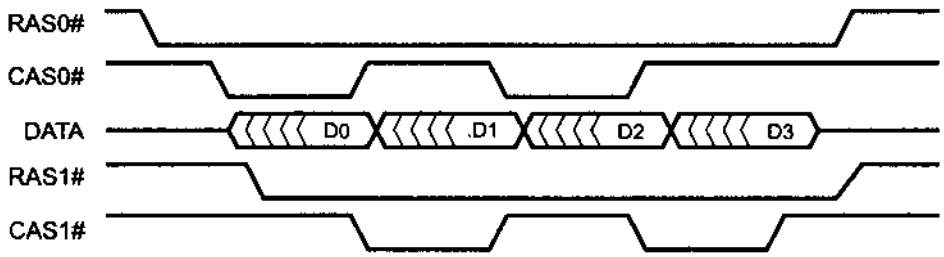


Рис. 8.6. Чередование банков DRAM в режиме страничного обмена: DO, D2 — данные из банка 0 (CAS0#); D1, D3 — данные из банка 1 (CAS1#)

Обратим внимание на то, что информация на выходе микросхем *стандартной* памяти DRAM появляется с некоторой задержкой относительно спада импульса CAS# и держится только во время низкого уровня этого сигнала. После подъема CAS# выходной буфер микросхемы переводится в третье (высокоимпедансное) состояние. Стандартная память со временем доступа 60-70 нс в режиме быстрого страничного обмена при частоте системной шины 66 МГц может обеспечить лучший пакетный цикл чтения 5-3-3-3.

Следующей модификацией памяти, направленной на повышение производительности при том же быстродействии запоминающих элементов, явилась *память EDO DRAM* (Extended, или Enhanced, Data Out). Эта память содержит регистр-защелку выходных данных (data latch), что обеспечивает некоторую конвейеризацию работы для повышения производительности при чтении. Регистр «прозрачен» при низком уровне сигнала CAS#, а по его подъему фиксирует текущее значение выходных данных до следующего его спада. Перевести выходные буферы в высокоимпедансное состояние можно либо подъемом сигнала OE# (Output Enable), либо одновременным подъемом сигналов CAS# и RAS#, либо импульсом WE#, который при высоком уровне CAS# не вызывает записи (в PC управление по входу OE# практически не используют).

Временная диаграмма работы с EDO-памятью в режиме страничного обмена приведена на рис. 8.7, этот режим иногда называют гиперстраничным режимом обмена (Hyper Page Mode, *HPM)*. Его отличие от стандартного заключается в подъеме импульса CAS# до появления действительных данных на выходе микросхемы. Считывание выходных данных может производиться внешними схемами вплоть до спада следующего импульса CAS#, что позволяет экономить время за счет сокращения длительности импульса CAS#. Время цикла внутри страницы для памяти со временем доступа 60 нс уменьшается с 35 нс (28,5 МГц) у стандартной памяти DRAM до 25 нс (40 МГц) у EDO, повышая производительность в страничном режиме на 40 %. Память EDO со временем доступа 60-70 нс в режиме гиперстраничного обмена при частоте системной шины 66 МГц может обеспечить лучший пакетный цикл чтения 5-2-2-2. Благодаря простоте данного усовершенствования при одном и том же времени доступа запоминающих элементов цена памяти EDO почти не отличается от цены стандартной памяти. Однако ее применение дает эффект, соизмеримый с эффектом от установки стандартного асинхронного внешнего кэша. Более того, установка такого кэша в систему с памятью EDO практически не дает повышения производительности. В результате распространилось мнение, что в памяти EDO содержится внутренний кэш, хотя называть кэшем простой регистр-защелку, вероятно, слишком претенциозно.

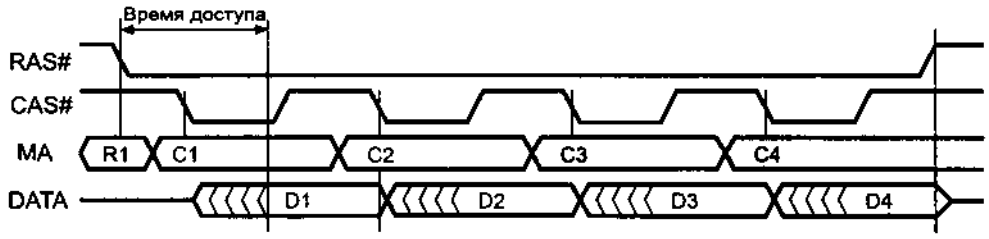


Рис. 8.7. Страничный режим считывания EDO DRAM (HPM)

Микросхемы EDO DRAM применяются в модулях SIMM-72 и DIMM, эти модули конструктивно и по назначению выводов совместимы со стандартными модулями (FPM). Все модули EDO не имеют бита четности (однобитные микросхемы EDO не выпускаются). Контрольные разряды 36-битных модулей

EDO могут использоваться только в памяти ECC, в которой доступ осуществляется всегда сразу ко всем байтам.

Установка EDO DRAM вместо стандартной памяти в не приспособленные для этого системы может вызвать конфликты выходных буферов устройств, разделяющих с памятью общую шину данных. Скорее всего, этот конфликт возникнет с соседним банком памяти при чередовании банков. Для отключения выходных буферов памяти EDO внутри страничного цикла обычно используют сигнал WE#, не вызывающий записи во время неактивной фазы CAS# (рис. 8.8, *а)*. По окончании цикла буферы отключаются лишь по снятию сигнала RAS# (рис. 8.8, б).

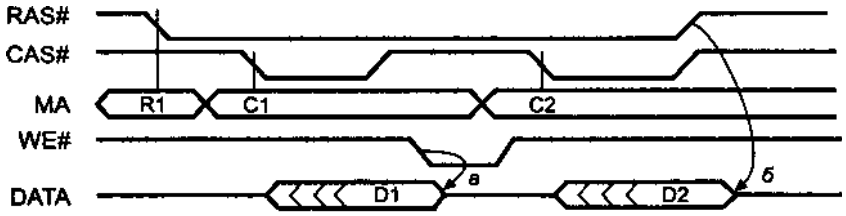


Рис. 8.8. Управление выходным буфером EDO DRAM

Из принципиального различия в работе выходных буферов следует, что в одном банке не стоит смешивать EDO и стандартные модули. Модули EDO поддерживаются не всеми чипсетами и системными платами (в большей мере это относится к системным платам для процессоров класса 486). Кроме того, не все системные платы, поддерживающие память EDO, используют потенциальный выигрыш в производительности от ее «малой конвейеризации» (это замечание больше относится к дешевым системным платам). Задержка отключения выходных буферов затрудняет чередование банков, из-за чего некоторые системные платы не поддерживают чередование для памяти EDO.

Многие чипсеты совместно с BIOS автоматически определяют тип установленных модулей и даже допускают смесь EDO и стандартных модулей в разных банках. Для определения типа чипсет организует специальный цикл обращения, в котором «прощупывает» все банки и заполняет таблицу, после чего переводится в режим нормального обращения (с таким специальным циклом возможна и обычная работа с памятью, но ее производительность оказывается на удивление низкой). В нормальном режиме обращения в зависимости от адреса, определяющего номер банка, по значению соответствующего ему поля таблицы организуется требуемый цикл.

Микросхемы EDO применялись как в основной памяти, так и в видеопамяти графических адаптеров.

Результатом дальнейшего развития конвейерной архитектуры модулей памяти явилась *память BEDO* (Burst EDO) *DRAM.* В микросхемах данного типа помимо регистра-защелки выходных данных, стробируемого по фронту импульса CAS#, содержится еще и внутренний счетчик адреса столбцов для пакетного

цикла. Это позволяет выставлять адрес столбца только в начале пакетного цикла, а во 2, 3 и 4-й передачах импульсы CAS# только запрашивают очередные данные. В результате удлинения конвейера выходные данные как бы отстают на один такт сигнала CAS#, зато следующие данные появляются без тактов ожидания процессора, чем обеспечивается лучший цикл чтения 5-1-1-1 для памяти BEDO с временем доступа 60 нс при частоте шины до 66 МГц. Задержка появления первых данных пакетного цикла окупается повышенной частотой приема последующих. Память BEDO применяется в модулях SIMM-72 и DIMM, но поддерживается далеко не всеми чипсетами.

На этом эволюция асинхронной памяти остановилась, а дальнейшие усовершенствования потребовали синхронного интерфейса. Память BEDO широкого распространения не получила, поскольку ей уже «наступала на пятки» синхронная динамическая память (SDRAM).

Вышеперечисленные типы памяти являются *асинхронными* по отношению к тактам системной шины компьютера. Это означает, что все процессы инициируются только импульсами RAS# и CAS#, а завершаются через какой-то определенный (для данных микросхем) интервал. На время этих процессов шина памяти оказывается занятой, причем, в основном, ожиданием данных.

Синхронная память — SDRAM, DDR и DDR2 SDRAM

Микросхемы *синхронной динамической памяти* (Synchronous DRAM, SDRAM) представляют собой конвейеризированные устройства, которые на основе вполне обычных ячеек[1] DRAM обеспечивают цикл 5-1-1-1, но уже при частоте шины в 100 МГц и выше. По составу сигналов интерфейс SDRAM близок к обычной динамической памяти: помимо входов синхронизации, здесь есть мультиплексированная шина адреса, линии RAS#, CAS#, WE# (разрешение записи) и CS# (выбор микросхемы), линии данных (табл. 8.4). Все сигналы стробируются по положительному перепаду синхроимпульсов, комбинация управляющих сигналов в каждом такте кодирует определенную *команду*. С помощью этих команд организуется та же последовательность внутренних сигналов RAS и CAS, что и для памяти FPM (см. выше). Для выполнения транзакции чтения или записи сначала подается *команда активации* ACT вместе с адресом строки, которая будет открыта (активирована). Далее через несколько тактов (для DRAM нужно выдержать задержку T_{RCD}) подается *команда чтения* (RD) или *записи* (WR), вместе с которой подается адрес столбца. Таким образом передается первый адрес пакетной транзакции, остальные адреса в пределах пакета (2, 4 или 8 ячеек) микросхема вычисляет сама. *Деактивировать* (закрыть) строку можно как явной командой, так и автоматически. Последний случай называется *автопредзарядом,* его можно указать в командах чтения и записи.

[1] На момент появления SDRAM время доступа было 50-70 нс, к 2005 году его удалось снизить до **30** нс.

Таблица 8.4. Назначение сигналов в микросхемах SDRAM, DDR и DDR2 SDRAM

Сигнал	I/O	Назначение
CLK (CK, CK#)	I	Clock Input — синхронизация, действует по положительному перепаду. Для DDR SDRAM используется дифференциальный вход CK, CK#
CKE	I	Clock Enable — разрешение синхронизации (высоким уровнем). Низкий уровень переводит микросхему в режим Power Down, Suspend или Self Refresh
CS#	I	Chip Select — разрешение декодирования команд (низким уровнем). При высоком уровне новые команды не декодируются, но выполнение начатых продолжается
RAS#, CAS#, WE#	I	Row Address Strobe, Column Address Strobe, Write Enable — сигналы, определяющие операцию (код команды)
BS[2:0] или BA[2:0]	I	Bank Selects или Bank Address — выбор банка, к которому адресуется команда. Для микросхем с 4-банковой структурой используются только 2 младших бита
A[0:15]	I	Address — мультиплексированная шина адреса (наличие старших битов определяется емкостью микросхемы). В циклах Bank Activate определяют адрес строки. В циклах Read/Write линии A[0:9] и A11 задают адрес столбца. Линия A10 в циклах Read/Write включает режим автопредзаряда (при A10 = 1), в цикле Precharge значение A10 = 1 задает предзаряд всех банков (независимо от BS[2:0])
DQx	I/O	Data Input/Output — двунаправленные линии данных
DQS	I/O	Data Strobe — двунаправленные линии стробирования данных (для DDR SDRAM)
DQM (DM)	I	Data Mask — маскирование данных. В цикле чтения высокий уровень переводит шину данных в высокоимпедансное состояние (действует через 2 такта). В цикле записи высокий уровень запрещает запись текущих данных, низкий — разрешает (действует без задержки)
ODT	I	On Die Termination — включение резисторов-терминаторов (для DDR2), расположенных внутри микросхемы. Через расширенный регистр режима включение терминаторов может быть запрещено
V_{SS}, V_{DD}	–	Общий провод и питание ядра (нет в DDR2)
V_{SSQ}, V_{DDQ}	–	Общий провод и питание выходных буферов. Изолированы от питания ядра для снижения помех (в DDR2 используются и для питания ядра)
V_{SSDL}, V_{DDDL}	–	Общий провод и питание цепей DLL (для DDR2). Изолированы от питания ядра для снижения помех
V_{REF}	–	Опорное напряжение интерфейса SSTL (для DDR SDRAM)

Данные для первой передачи *пакета записи* устанавливаются вместе с командой WR. В следующих тактах подаются данные для остальных передач пакета. Первые данные *пакета чтения* появляются на шине через определенное количество тактов после команды. Это число, называемое *CAS Latency* (CL), определяется временем доступа T_{CAC} и тактовой частотой. Остальные данные пакета выдаются в последующих тактах. Временные диаграммы работы SDRAM приведены на рис. 8.9. Здесь показана команда записи WR, за которой следует команда чтения RD той же страницы, предварительно открытой командой ACT. Далее страница закрывается командой PRE. Длина пакета 2, CL = 3.

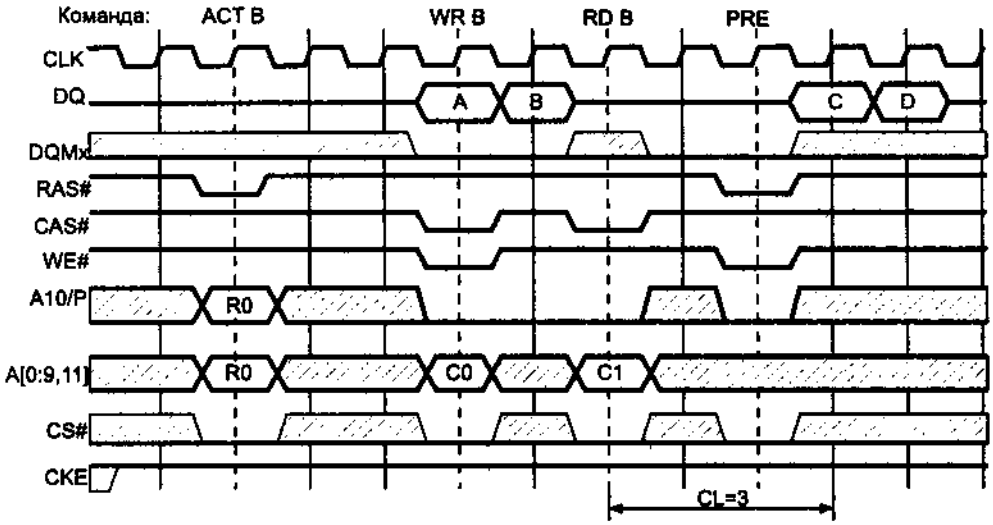


Рис. 8.9. Временные диаграммы пакетных циклов SDRAM: A и B - данные для записи по адресу R0/C0 и R0/C0+1, C и D - данные, считанные по адресу R0/C1 и R0/C1+1

Регенерация (цикл CBR с внутренним счетчиком адреса регенерируемой строки) выполняется по команде REF, которую можно вводить только при состоянии покоя *(idle)* всех банков.

На первый взгляд, из этого описания не видно никаких особых преимуществ SDRAM по сравнению с BEDO. Однако синхронный интерфейс в сочетании с внутренней мультибанковой организацией обеспечивает возможность повышения производительности памяти при множественных обращениях. Здесь имеется в виду способность современных процессоров формировать следующие запросы к памяти, не дожидаясь результатов выполнения предыдущих, а также обращения к памяти со стороны других устройств-мастеров шин (PCI, AGP). В SDRAM после выбора строки (активации банка) ее можно закрыть не сразу, а после выполнения серии обращений к ее элементам, причем как по записи, так и по чтению. Эти обращения выполняются быстрее, поскольку для них не требуется подавать команду активации и выжидать в течение времени T_{RC}. Максимальное время удержания строки открытой ограничивается периодом регенерации. Возможность работы с открытой строкой была использована уже в FPM DRAM. Однако в SDRAM можно активировать строки в нескольких банках, причем каждую своей командой ACT — это одна из ключевых особенностей SDRAM (single-pulsed RAS interface). Активировать строку можно во время выполнения любой операции с другим банком. Обращение к открытой строке требуемого банка выполняется по командам RD и WR, у которых в качестве параметров, помимо адреса столбца, фигурирует и номер банка. Таким образом, можно так спланировать транзакции, чтобы шина данных в каждом такте несла очередную порцию данных и такой поток продолжался не только в пределах одного пакета, но и для серии обращений к разным областям памяти. Кстати,

держать открытыми можно и строки в банках разных микросхем, объединенных общей шиной памяти, — для этого при адресации используются линии CS#.

Микросхемы SDRAM оптимизированы для пакетной передачи. У них при инициализации программируются длина пакета (*burst length* = 1, 2, 4, 8 элементов), порядок следования адресов в пакете (*wrap mode*: interleave/linear — чередующийся/линейный) и операционный режим. Пакетный режим может включаться как для всех операций (normal), так и только для чтения (multiple burst with single write). Этот выбор позволяет оптимизировать память либо для обратной записи (WB), либо для работы с кэшем сквозной записи (WT). Обратим внимание на то, что внутренний счетчик адреса работает по модулю, равному запрограммированной длине пакетного цикла (например, при burst length = 4 он не позволяет перейти границу обычного 4-элементного пакетного цикла).

Пакетные циклы могут прерываться (принудительно завершаться) последующими командами. При этом оставшиеся адреса отбрасываются, и прерывающий пакет получает полную длину (если его самого не прерывают).

В команде Write имеется возможность блокирования записи данных любого элемента пакета — для этого достаточно в его такте установить высокий уровень сигнала DQM. Этот же сигнал используется и для перевода в высокоимпедансное состояние буферов данных при операциях чтения.

Микросхемы SDRAM имеют *средства энергосбережения*, для управления ими используется вход разрешения синхронизации CKE.

В режиме *саморегенерации* (self refresh) микросхемы периодически выполняют циклы регенерации по внутреннему таймеру, в этом режиме они не реагируют на внешние сигналы, и внешняя синхронизация может быть остановлена.

Режимы пониженного потребления (power down mode) устанавливаются при переводе CKE в низкий уровень по команде NOP или INHBT. В этих режимах микросхема не воспринимает команд. Поскольку в данных режимах регенерация не выполняется, длительность пребывания в них ограничена периодом регенерации.

Если во время выполнения команды чтения или записи установить CKE - L, то микросхема перейдет в *режим приостановки синхронизации* (clock suspend mode), в котором «замораживается» внутренняя синхронизация (нет продвижения данных) и не воспринимаются новые команды.

Временные характеристики (*тайминг*) памяти SDRAM описываются тактовой частотой и тремя параметрами задержек CL-T_{RCD}-T_{RP}:

- CL (Cas Latency) — число скрытых тактов (2 или 3);
- T_{RCD} - задержка RAS-CAS, выраженная в тактах (2 или 3);
- T_{RP} — время предварительного заряда RAS.

По тактовой частоте для микросхемы SDRAM, применяемой в качестве ОЗУ PC-совместимых компьютеров, имеется три градации: PC66 (поначалу ее так не называли, поскольку другой и не было), PC100 и PC133 для максимальных частот 66,6, 100 и 133 МГц соответственно. Их ключевые параметры приведены в табл. 8.5. Помимо перечисленных ранее, здесь фигурирует и параметр T_{RC} —

минимальное время цикла обращений к строкам одного банка. В обозначении быстродействия микросхем SDRAM обычно присутствует и параметр T_{AC} — задержка данных относительно фронта синхросигнала; период частоты синхронизации, естественно, не может быть меньше этой задержки. Микросхемы со спецификацией -10 (T_{AC} = 10 нс) могут устойчиво работать в модулях лишь на частоте 66 МГц. Микросхемы со спецификацией -8 могут работать на частоте 100 МГц, но, в зависимости от модификации, с разной латентностью. Так, например, для памяти Micron микросхемы с маркировкой -8A...-8C могут работать на частоте 100 МГц с CL = 3, а -8D или -8E — с CL = 2.

Таблица 8.5. Ключевые параметры временной диаграммы SDRAM

Спецификация	Частота, МГц	CL - T_{RCD} - T_{RP}
PC66	66	3-2-3 2-2-2
PC100	100	3-3-3 3-2-2 2-2-2
PC133	133	3-3-3 3-2-2 2-3-2 2-2-2
DDR 200 PC1600	100	2-2-2
DDR 266 PC2100	133	2-2-2 2-3-3 2,5-3-3
DDR 333 PC2700	166	2,5-3-3
DDR 400 PC3200	200	3-3-3 3-4-4
DDR2-400	200	3-3-3 4-4-4
DDR2-533	266	3-3-3 4-4-4 5-5-5
DDR2-667	333	4-4-4 5-5-5

Естественно, память может работать и на частотах ниже максимальной. Для микросхем SDRAM, применяемых, например, в графических адаптерах, существуют и иные спецификации быстродействия.

Время доступа SDRAM (от подачи полного адреса до получения данных чтения) можно определить через задержки (в тактах) T_{RCD} + CL и частоту:

$$T_{RAC}\ [\text{нс}] = (T_{RCD} + \text{CL})/F\ [\text{ГГц}].$$

Как видно из таблицы, микросхемы с наилучшим таймингом 2-2-2 при частоте 133 МГц обеспечивают время доступа (2 + 2)/0,133 = 30 нс (это очень быстрая

память). Самый худший вариант 3-2-3 на 66 МГц соответствует 75 нс, 3-2-3 на 100 МГц — 50 нс.

Синхронный интерфейс позволяет довольно эффективно использовать шину и обеспечить на частоте 100 МГц пиковую производительность 100 Мбит/с/пин на 1 вывод шины данных. SDRAM применяют в составе модулей DIMM с 8-байт- ной разрядностью, что дает производительность 800 Мбайт/с. При частоте шины 133 МГц пиковая производительность уже достигла 1064 Мбайт/с. Однако это — теоретическая производительность, в ней не учтены накладные расходы на регенерацию и подразумевается, что требуемые страницы уже открыты. Из-за указанных выше ограничений на реальном произвольном потоке запросов производительность, конечно же, будет ниже.

Если частота системной шины совпадает с частотой памяти, то SDRAM обеспечивает параметры пакетного цикла от 5-1-1-1 до 7-1-1-1. Даже при самой быстрой памяти (2-2-2) хотя бы один лишний такт обязательно вводит контроллер памяти, соединяющий шину памяти с системной шиной.

Память DDR SDRAM (стандарт JEDEC JESD-92D) представляет собой дальнейшее развитие SDRAM. Здесь используется ядро (ячейки DRAM) с разрядностью вдвое большей, чем разрядность шины данных. Внешний интерфейс мультиплексирует данные, обеспечивая их быструю передачу. Как и следует из названия (Dual Data Rate — удвоенная скорость передачи данных), у микросхем DDR SDRAM данные внутри пакета передаются с удвоенной скоростью — они переключаются по обоим перепадам синхроимпульсов (рис. 8.10). На частоте 100 МГц DDR SDRAM имеет пиковую производительность 200 Мбит/с на вывод, что в составе 8-байтных модулей DIMM дает производительность 1600 Мбайт/с. На высоких тактовых частотах (100-200 МГц) двойная синхронизация (по фронту и спаду) предъявляет очень высокие требования к точности временных диаграмм. Для повышения точности синхронизации предпринят ряд мер:

- Сигнал синхронизации микросхемы подается в дифференциальной форме по двум линиям, CLK и CLK# (differential clock inputs). Это позволяет снизить влияние смещения уровней на точность определения момента синхронизации — дифференциальный приемник срабатывает в момент равенства уровней напряжения.
- Для синхронизации данных в интерфейс введен новый двунаправленный стробирующий сигнал DQS. Строб генерируется источником данных: при операциях чтения DQS генерируется микросхемой памяти, при записи — контроллером памяти (чипсетом). При чтении фронты и спады этого сигнала точно центруются в моменты смены данных, приемник должен стробировать данные с небольшой задержкой относительно переключений DQS. При записи фронты и спады центруются точно посередине окна действительности данных и масок DQM.
- Для синхронизации DQS с системной тактовой частотой (CLK) микросхемы имеют встроенные схемы автоподстройки по задержке (Delay Locked Loop,

DLL) сигнала DQS относительно CLK. Схема DLL работает наподобие фазовой автоподстройки и способна выполнять синхронизацию (обеспечивать совпадение фронтов DQS и CLK) лишь в некотором ограниченном диапазоне частот синхронизации.

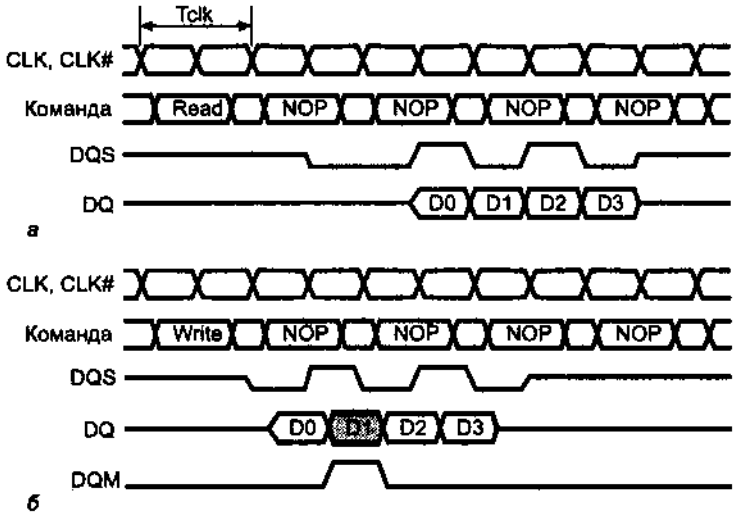


Рис. 8.10. Временные диаграммы пакетных циклов DDR SDRAM: а — чтение, CL = 2, длина пакета 4; *б* — запись, длина пакета 4, данные D1 не записываются

Есть микросхемы DDR SDRAM с возможностью отключения схем DLL, для этого они имеют дополнительный расширенный регистр режима. Отключение DLL необходимо при снижении тактовой частоты (в целях энергосбережения). При отключенной схеме DLL стробы DQS не привязаны к синхросигналу CLK, и у разных микросхем, работающих в одной системе, они будут иметь разные частоты.

Микросхемы с частотой 100, 133, 166 МГц по стандарту JEDEC обозначаются как DDR 200, DDR 266 и DDR 333; они питаются напряжением 2,5 или 3,3 В. Микросхемы с частотой 200 МГц (DDR 400) питаются напряжением 2,6 В. Микросхемы DDR SDRAM выпускаются как в корпусах TSOP и LSOJ (с обычным двухрядным расположением выводов), так и в корпусах BGA (шариковые выводы почти по всей поверхности дна).

В отличие от обычных микросхем SDRAM, у которых данные для записи передаются одновременно с командой, в DDR SDRAM данные для записи (и маски DQM) подаются с задержкой на один такт (write latency). Значение CAS Latency может быть и дробным (CL = 2, 2,5, 3).

Развитием DDR SDRAM стала память *DDR2 SDRAM* (стандарт JEDEC JESD79-2). Вопреки расхожему заблуждению, здесь обмен данными происходит также на удвоенной частоте синхронизации. Временные диаграммы чтения и записи по виду аналогичны DDR SDRAM. Однако в DDR2 тактовые частоты уже выше (200-333 МГц), что предъявляет особые требования к передаче сиг

налов. Для исключения отводов от шин печатных проводников, неизбежных при использовании корпусов с двухрядным расположением выводов, микросхемы DDR2 выпускаются только в корпусах BGA с матрицей выводов. Кроме того, микросхемы имеют внутренние резисторы-терминаторы, подключенные к линиям данных (DQx), стробов (DQS) и масок (DM) — всем сигнальным линиям, работающим на удвоенной частоте. Эти терминаторы управляются через внешний вход ODT и внутренний расширенный регистр режима. Назначение терминаторов — улучшение качества сигналов, передаваемых на высоких частотах. При инициализации микросхем выбирается сопротивление терминаторов (75 Ом, 150 Ом или отключены). Для прежних типов памяти терминаторы устанавливались на системной плате, теперь они приближены к источникам и потребителям информации. При инициализации выбирается и мощность буферных формирователей данных и стробов, а также выполняется калибровка приемопередатчиков. Такая забота о качестве сигнала еще в большей мере проявляется в RDRAM (см. далее).

Память DDR2 SDRAM разрабатывалась с учетом обратной логической совместимости: контроллер DDR2 может работать как с памятью DDR, так и с DDR2. Набор команд DDR2 является расширением набора DDR. В DDR2 появились некоторые упрощения, облегчающие тестирование компонентов. Так, например, исключена возможность задания дробных значений CL, а длина пакетного цикла может быть только 4 (значения 2 и 8 упразднены). Запоминающее ядро (DRAM) работает на частоте, в 4 раза меньшей частоты передачи данных (в два раза ниже тактовой), однако разрядность ядра в 4 раза выше, чем разрядность шины данных микросхемы. Такое решение обеспечивает высокую скорость передачи, а понижение частоты ядра — снижение потребляемой мощности.

Для облегчения работы планировщика транзакции (в контроллере памяти) и повышения коэффициента использования шины данных применяются «отправленные стробы CAS» (posted CAS). Во всех версиях SDRAM транзакция запускается двумя командами — командой активацией банка (посылка RAS) и собственно командой чтения или записи (посылка CAS), между которыми должна быть пауза в 2-4 такта (задержка T_{RCD}). Из-за этого в ряде случаев не удается своевременно (оптимально) подать команду (шина управления могла быть занята), так что на шине данных возникает простой — «пузырь» (рис. 8.11, а). Программируемая *дополнительная задержка* (Additional Latency, AL) в DDR2 позволяет подавать обе части запроса в соседних тактах, но фактическая подача CAS (и отсчет CL) начнется позже на заданное, число тактов. Таким образом, «пузыри» исключаются — эффективность использования шины данных повышается (рис. 8.11, *б*). Дополнительная задержка фигурирует и в циклах записи. Обычная память DDR SDRAM, фактически, работает с нулевой дополнительной задержкой.

Физической совместимости DDR2 и DDR нет; для DDR2 разработаны специальные модули с иными ключами и более мелким шагом контактов. Разводка сигналов этих модулей изменена, чтобы оптимизировать разводку проводников на печатных платах.

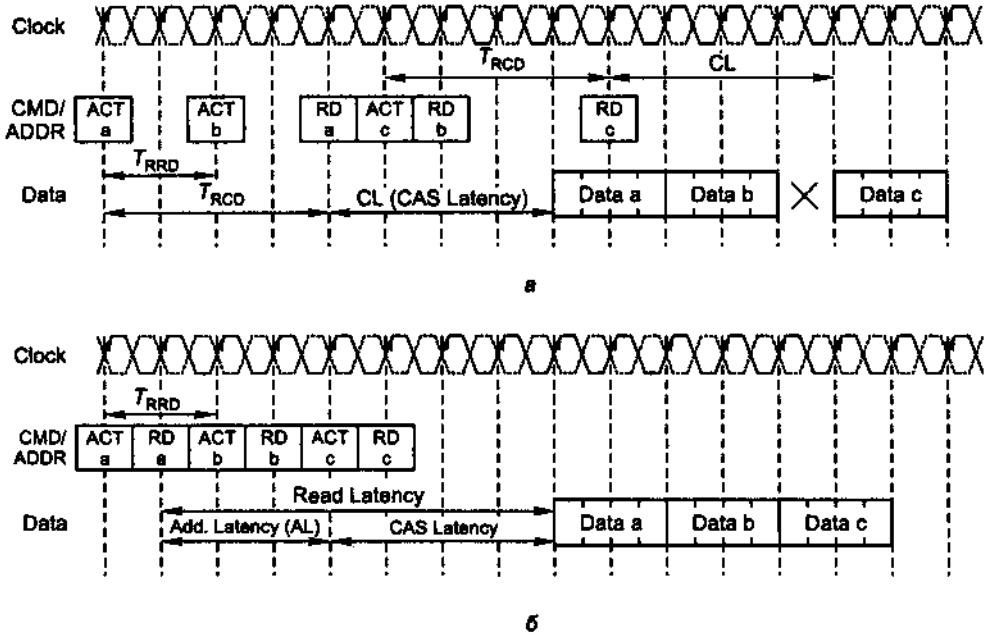


Рис. 8.11. Временные диаграммы выполнения двух команд чтения: а - для DDR SDRAM, *б* - для DDR2 SDRAM (CL = 2, AL = 2)

Микросхемы DDR2 с частотой 200, 266, 333 и 400 МГц по стандарту JEDEC обозначаются как DDR2-400, DDR2-533, DDR2-667 и DDR2-800

Для DDR2 SDRAM на 2005 год освоены частоты 200, 266, 333 МГц, при этом тайминг принимает значение от 3-3-3 до 5-5-5: чем выше частота, тем больше задержки, выраженные в числе тактов (физические свойства запоминающих ячеек не слишком различаются). На частоте 200 МГц тайминг 3-3-3 соответствует T_{AC} = (3 + 3)/0,2 = 30 нс, той же задержке на частоте 333 МГц соответствует тайминг 5-5-5. Микросхемы питаются напряжением 1,8 В.

Для локальной памяти графических адаптеров используются специальные типы SDRAM с более высокими тактовыми частотами и некоторыми особенностями временных диаграмм. Далее приводятся параметры такой памяти, выпускаемой фирмой Samsung в 2005 году:

- *GDDR SDRAM с* тактовыми частотами 166-300 МГц обеспечивает пропускную способность 333-600 Мбит/с на вывод, питание — 2,5-2,8 В (болшее напряжение для более высоких частот). Логически эта память соответствует обычной памяти DDR SDRAM, CL = 3 (величина фиксирована), длина пакета — 2, 4 или 8.
- *GDDR2 SDRAM с* тактовыми частотами 266-333 МГц обеспечивает пропускную способность 533-667 Мбит/с на вывод. Питание — 1,8 В. Эта память примерно соответствует DDR2 SDRAM: имеется дополнительная программируемая латентность (AL = 0...4), CL = 3...5 (величина программируема),

латентность записи на 1 такт меньше латентности чтения. Длина пакета — 4 или 8.

- *GDR3 SDRAM* заметно отличается от DDR2: латентность записи программируется (1...7 тактов, независимо от чтения); программируемая дополнительная задержка AL = 0 или 1, CL = 4... 11. Длина пакета — 4 или 8. Тактовые частоты — 500-800 МГц, пропускная способность — 1-1,6 Гбит/с на контакт. Питание — 1,8 или 2 В (для самых быстродействующих).

Микросхемы SDRAM до «штатного» использования обязательно инициализируются. После подачи питания и установления синхросигнала должна быть выполнена предварительная зарядка всех банков, после чего запрограммирован регистр режима, определяющий параметры циклов обмена. К основным параметрам относятся порядок адресов пакетного цикла (линейный или чередующийся), длина пакета, CL. Кроме того, программируются параметры энергосбережения и некоторые другие. В DDR SDRAM дополнительно требуется программирование режима работы цепей DLL, и для этого здесь появился расширенный регистр режима. В DDR2 SDRAM задается дополнительная задержка AL (0-4), а также требуется управление терминаторами и выходными формирователями. Для этого в расширенном регистре режима определены дополнительные биты. Параметр CL выбирают, исходя из спецификации микросхем и тактовой частоты так, чтобы задержка, обусловленная CL, была бы минимальной, но не меньше T_{CAC}. В DDR SDRAM возможны и дробные значения CL, так что настройка может быть более тонкой. В DDR2 дробные значения не используют, но здесь фигурируют более высокие частоты. Значение AL определяют, исходя из тактовой частоты и особенностей работы контроллера памяти.

Для программирования регистров режима (основного и расширенных) контроллер устанавливает значение битов регистра на шине адреса A[15:0], линиями BA[2:0] выбирает нужный регистр и подает специальную команду (низкий уровень CS#, RAS#, CAS# и WE#).

По причине существенного отличия интерфейса от традиционной асинхронной памяти микросхемы SDRAM не могут быть установлены в модули SIMM, они применяются в модулях DIMM или устанавливаются прямо на системную (или графическую) плату. Интерфейсы (SDR) SDRAM, DDR и DDR2 SDRAM значительно отличаются и друг от друга; модули, на которых они устанавливаются, также конструктивно различаются. Возможность использования этих типов памяти определяется чипсетом системной платы. Память SDRAM в конце 90-х годов стала самой распространенной, затем ее потеснила DDR SDRAM. В 2004-2005 годах стали переходить на DDR2 SDRAM.

Память Rambus DRAM — RDRAM и XDRAM

Память RDRAM (Rambus DRAM) имеет синхронный интерфейс, отличающийся от традиционного. Запоминающее ядро этой памяти построено на все тех же КМОП-ячейках динамической памяти, но пути повышения производительности интерфейса совершенно иные. Первые микросхемы RDRAM применялись

в некоторых моделях видеокарт и игровых приставок. Их интерфейс — Rambus Channel — имел разрядность шины данных в 1 байт, но, работая на частоте 250-300 МГц, обеспечивал производительность 500-600 Мбайт/с. Его сменил интерфейс CRDRAM (Concurrent RDRAM) с частотами 300-350 МГц и производительностью 600-700 Мбайт/с. Дальнейшим развитием интерфейса стал фирменный (Rambus) стандарт DRDRAM (Direct Rambus DRAM), обеспечивающий производительность до 1600 Мбайт/с на двухбайтной шине данных при частоте 400 МГц. Стандарт RDRAM (точнее, DRDRAM, но для краткости первую букву опустили) на словах был поддержан множеством производителей микросхем и модулей памяти; как и DDR SDRAM, он претендовал на роль основного высокопроизводительного стандарта для памяти компьютеров любого размера.

Подсистема памяти (ОЗУ) RDRAM состоит из контроллера памяти, канала и собственно микросхем памяти. По сравнению с DDR SDRAM при той же производительности RDRAM имеет более компактный интерфейс и обеспечивает лучшую масштабируемость. Разрядность ОЗУ RDRAM (16 байт) не зависит от числа установленных микросхем, а число банков, доступных контроллеру, и объем памяти суммируются по всем микросхемам канала. При этом в канале могут присутствовать микросхемы разной емкости в любых сочетаниях.

Организация *запоминающего ядра* микросхем мультибанковая: 64-мегабитные микросхемы имеют 8 банков, 256-мегабитные — 32 банка. У каждого банка свои усилители считывания, благодаря чему в микросхеме может быть активировано несколько банков. Разрядность ядра — 16 байтов, что составляет 128 или (с контрольными разрядами) 144 бит. Ядро работает на 1/8 частоты канала, взаимодействие с ядром осуществляется по внутренним сигналам RAS и CAS. В RDRAM применяются ячейки памяти с временем доступа 32-53 нс.

Канал RDRAM (Rambus channel) представляет собой последовательно-параллельную шину, благодаря которой можно ограничить количество линий интерфейса, что позволяет упорядочить разводку проводников ради повышения частоты передачи сигналов. Небольшое количество сигналов обеспечивает возможность при не очень высокой цене применить сверхбыстродействующие интерфейсные схемы. Тактовая частота канала — до 533 МГц[1], стробирование информации осуществляется по обоим фронтам синхросигнала. Таким образом, пропускная способность одной линии достигает 1066 Мбит/с. Канал состоит из 30 основных линий с интерфейсом *RSL* (Rambus System Logic) и 4 вспомогательных линий КМОП, используемых для инициализации микросхем. Стандарт требует соблюдения топологических правил, структура подсистемы памяти приведена на рис. 8.12. Все основные интерфейсные линии, кроме линий синхронизации, начинаются от интерфейсной микросхемы контроллера памяти и заканчиваются терминаторами на противоположном конце канала. Терминаторы не позволяют сигналам отражаться от концов канала. Микросхемы памяти подключаются к каналу без Т-образных ответвлений проводников, что облегчается их упаковкой в корпуса BGA. Интерфейсные линии должны

[1] На 2005 год.

идти строго параллельно друг другу, чтобы задержки распространения сигналов по разным линиям совпадали. На канале может быть установлено до 32 микросхем, все микросхемы соединяются параллельно. Для того чтобы контроллер мог адресоваться к определенной микросхеме, каждой из них назначается свой уникальный адрес DEVID. Нумерация микросхем (device enumeration) осуществляется в процессе инициализации, которая выполняется с использованием вспомогательного последовательного КМОП-интерфейса. Этот интерфейс имеет линии синхронизации (SCK), команд (CMD) и данных (SIO). По линиям SCK и CMD все микросхемы запараллеливаются. Каждая микросхема имеет два вывода — SI00 и SI01, которые обычно объединены внутренним коммутатором, но по команде могут быть разорваны. По этим линиям микросхемы соединяются в цепочку.

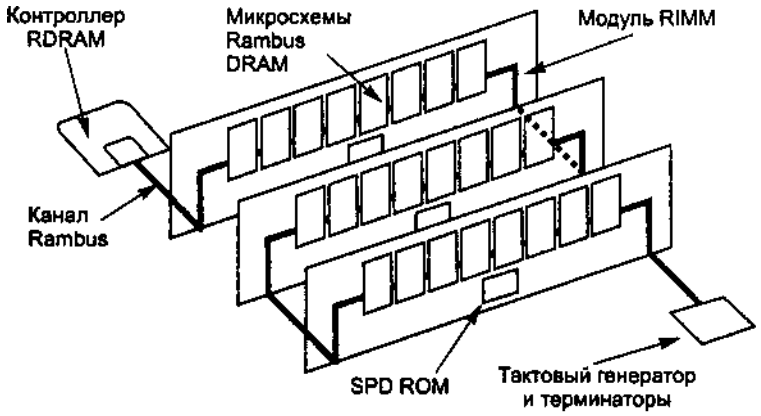


Рис. 8.12. Память Direct RDRAM

Синхросигнал вводится в канал с «дальнего конца» и распространяется в сторону контроллера по линии CTM (Clock To Master). По этому сигналу микросхемы памяти стробируют данные, посылаемые к контроллеру (при чтении). Распространяясь по каналу, эти данные сохраняют свою привязку к синхроимпульсам до самого контроллера. Дойдя до контроллера, синхросигнал выходит на линию CFM (Clock From Master) и идет по каналу до терминатора, установленного на конце. По этой линии синхронизируется информация, посылаемая от контроллера к микросхемам памяти, и ее привязка к синхросигналу также сохраняется в любом месте канала. Микросхемы привязывают данные чтения к синхросигналу с помощью встроенных схем DLL для автоподстройки задержки сигнала DQS относительно CLK.

Физический уровень интерфейса учитывает волновой характер процессов распространения сигналов в канале. Передатчики микросхем памяти формируют сигналы с половинной амплитудой. Эти сигналы распространяются по шине в обе стороны и на конце терминатора полностью поглощаются (отражения нет). На конце контроллера импеданс приемников высокий (терминаторов нет), и амплитуда сигнала из-за отражения удваивается. Таким образом приемник

контроллера принимает сигнал полной амплитуды. Отраженный от контроллера сигнал доходит до терминатора и поглощается им. По пути он никому не мешает, поскольку сигнал, передаваемый микросхемой памяти, «интересен» только контроллеру. Контроллер генерирует сигналы полной амплитуды, и по пути к терминаторам они в таком виде проходят по всем микросхемам памяти. Сигнал синхронизации передается в дифференциальной форме, что снижает погрешность стробирования, вызванную смещением уровней сигналов.

Канал разделен на *три независимые шины*: шину строк, шину столбцов и двухбайтную (2 x 9 бит) шину данных. Дополнительный бит байта данных (имеется не у всех микросхем RDRAM) может использоваться для контроля достоверности. По каждой шине информация передается *пакетами*. Высокая производительность шины управления (строк и столбцов) позволяет отказаться от пакетных (в терминологии BEDO и SDRAM) передач и упростить протокол шины. Память может одновременно обслуживать до четырех транзакций на полной скорости передачи данных.

В микросхемах RDRAM применяется механизм *отложенной*, или *буферированной записи*. Данные для записи сначала помещаются в буфер, из которого они выгружаются в усилители считывания-записи (sens amp) несколько позже по явной команде выгрузки (retire) или автоматически. Буфер записи хранит сами данные, а также номер банка и адрес столбца (но не строки). Буферизация записи позволяет контроллеру посылать команду записи на T_{RTR} раньше, чем этого требует параметр T_{RCD}, что повышает коэффициент использования шины.

Конвейерное выполнение операций RDRAM обеспечивается мультибанковой организацией с отдельными усилителями считывания. Пакеты команд по шинам строк и столбцов могут идти сплошным потоком, при этом на шине может выполняться до четырех транзакций. При произвольных обращениях повышению производительности способствует большое количество банков, практически недостижимое в памяти SDRAM. Банковые зависимости обращений приводят к необходимости «лишних» предварительных зарядов. Чем больше независимых банков, тем, в принципе, больше вероятность попадания соседних запросов в разные банки. При последовательных обращениях чтения (RD) или записи (WR) к ячейкам, расположенным в различных (несмежных) банках, эффективность использования полосы шины данных (1600 Мбайт/с) достигает 100 %. При цепочке обращений RD-RD-WR-WR к несмежным банкам одной микросхемы эффективность составляет 76 %, а при обращениях к разным микросхемам канала она достигает 94 %.

Регенерация осуществляется по команде, адресуемой к определенному банку одной или всех микросхем. За период регенерации T_{REF} (32 мс) должны быть перебраны все строки всех банков. В режимах пониженного потребления микросхемы осуществляют саморегенерацию.

Средства управления энергопотреблением отключают питание неиспользуемых узлов. В самом экономичном состоянии — *PDN* (Power Down) — микросхемы потребляют мощность в 110 раз меньшую, чем в состоянии полной готовности *STBY* (Standby). При этом время доступа в состоянии *PDN* в 250 раз больше, чем в *STBY*. Есть еще энергосберегающее состояние *NAP*, в котором доступ про

исходит быстрее, но потребление больше. В обоих энергосберегающих состояниях данные сохраняются саморегенерацией (в состоянии *NAP* возможна и регенерация командами REFA).

Микросхемы RDRAM требуют периодической (раз в 100 мс) подстройки выходного тока и термокалибровки, для этих целей имеются специальные команды. При подстройке тока микросхемы способны сообщать о своем перегреве.

Вспомогательная шина с сигналами $CK, CMD и SI0 служит для обмена данными с управляющими регистрами и вывода микросхем из состояния пониженного потребления (*PDN* и *NAP).* Информация по этой шине тоже передается пакетами.

Управляющие регистры хранят информацию об адресе микросхемы, управляют работой микросхемы в различных режимах, содержат счетчики регенерации для банков и строк, параметры временных циклов. В них же можно прочитать информацию о конкретной микросхеме — организацию, версию протокола и т. п. В составе управляющих есть и тестовые регистры.

Обязательным «фирменным» компонентом ОЗУ на RDRAM является *контроллер памяти.* В его функции входит обслуживание микросхем памяти, установленных в канале, по запросам, поступающим со стороны интерфейса системной шины компьютера. Часть контроллера, обращенная к каналу, инвариантна к архитектуре компьютера. Именно она «знает» протокол RDRAM и является продуктом фирмы Rambus.

Микросхемы RDRAM устанавливаются в модули RIMM. В соответствии со спецификацией RDRAM в одном канале может быть до трех слотов под RIMM, и их интерфейсные линии соединяются змейкой. В слоты могут устанавливаться RIMM различной емкости (сейчас они выпускаются на 64, 96, 128 и 256 Мбайт). Однако для устойчивой работы канала в ряде случаев приходится ограничиться двумя. В памяти появился новый элемент-пустышка *continuity module.* Это как бы модуль RIMM, но без микросхем памяти, и нужен он для того, чтобы замыкать цепь канала Rambus. Такая «затычка» должна устанавливаться во все слоты канала, не занятые под модули RIMM. Если используются не все слоты, то память выгоднее ставить ближе к контроллеру — она будет работать быстрее (см. выше).

Контроллер RDRAM встраивался в чипсеты для процессоров P6 (например, i820, i840), Pentium 4 (i850 с 32-разрядным каналом, то есть уже под пары модулей RIMM) и других архитектурных линий. Производством RDRAM занимаются немного фирм, эта память не выдержала конкуренции с массовой памятью DDR RDRAM и DDR2 SDRAM. Эти традиционные «широкоразрядные» типы памяти используются и в двухканальных конфигурациях, так что малая разрядность канала Rambus оказалась не таким уж важным преимуществом. Неприятно и слишком большое энергопотребление RDRAM — на модулях RIMM автору впервые (для памяти) довелось увидеть радиаторы. В настоящее время память RDRAM в новых системных платах для PC практически не используется.

На 2005 год память RDRAM выпускается с частотой передачи данных 800, 1066 и 1200 МГц с перспективами 1333 и 1600 МГц (тактовая частота — в два раза ниже). Самые высокопроизводительные модули RIMM 4800 (32-битные, в PC не используются) содержат два раздельных канала RDRAM с частотой передачи 1200 МГц (тактовая частота — 600 МГц).

Фирма Rambus развивает свою канальную архитектуру, и в 2004 - 2005 годах появилась новая *память XDRAM* (eXtreme Data Rate DRAM), в которой используется дифференциальный сигнальный интерфейс Rambus. Ширина канала осталась той же (16 бит), но данные передаются на 8-кратной тактовой частоте. При тактовых частотах 300, 400 и 500 МГц пропускная способность каждой линии данных составляет 2,4, 3,2 и 4 Гбит/с. Для 16-битного канала это означает пиковую скорость 4,8, 6,4 и 8 Гбайт/с, а в двухканальном варианте можно получить память с пиковой скоростью до 16 Мбайт/с. В перспективе частоту предполагается удвоить. Микросхемы XDRAM уже выпускаются различными фирмами (например, Samsung), их намечается использовать в системах, требующих экстремальной производительности памяти. В PC такая пропускная способность пока не востребована.

Память с виртуальными каналами — VC DRAM

Идея архитектуры памяти с виртуальными каналами (VirtualChannel memory architecture, не путать с виртуальной памятью!) заключается в помещении между массивом запоминающих ячеек и внешним интерфейсом микросхемы памяти набора *канальных буферов*. При этом операции обмена данными разделяются на два процесса: «фасадный» обмен данными с каналами и «тыловой» обмен между каналами и массивом запоминающих ячеек. Оба процесса выполняются по командам со стороны внешнего интерфейса почти независимо друг от друга. Архитектура виртуальных каналов применима к памяти любого типа, включая ПЗУ и флэш-память, но наиболее интересна она в приложении к динамической памяти — VC DRAM. Именно ее подразумевают под аббревиатурой VCM (VirtualChannel Memory). Название VirtualChannel является зарегистрированной торговой маркой фирмы NEC, а информацию по этой архитектуре предоставляет фирма Elpida Memory, Inc.

Устройство VC DRAM рассмотрим на примере микросхем емкостью 128 Мбит, на которых строятся модули DIMM VC DRAM. По интерфейсу (составу и уровням сигналов) микросхемы и модули VC DRAM аналогичны обычным микросхемам SDRAM, но отличаются системой команд. Микросхемы имеют такую же внешнюю организацию по 4, 8 или 16 бит данных, но совершенно иную внутреннюю архитектуру. У них две матрицы (два банка) запоминающих ячеек размером 8К x 8К, то есть каждая строка имеет объем 8 Кбит и состоит из четырех *сегментов* размером по 2 Кбит. Между матрицами и внешним интерфейсом имеются 16 *канальных буферов*, каждый объемом 2 Кбит. За одно обращение к матрице выполняется параллельная передача 2 Кбит данных между одним из буферов и сегментом выбранной строки. Этот «тыловой» обмен реализуют команды PRF (Prefetch — чтение массива в буфер) и RST (Restore — сохранение буфера в массиве), в которых микросхеме указываются номера банка, сегмента

и канала. Предварительно командой ACT должна быть активирована требуемая строка матрицы (при подаче этой команды задаются банк и адрес строки). Деактивация строк (предварительный заряд) может быть автоматической, сразу после обращения к массиву (для этого имеются специальные команды предвы- борки и сохранения — PRFA и RSTA), или же по специальным командам, деактивирующим выбранный банк или оба банка сразу.

«Фасадный» обмен с канальными буферами выполняется по командам чтения и записи (READ и WRIT), в которых указываются номер канала и та часть адреса, которая соответствует адресу столбца в обычной микросхеме DRAM или SDRAM. Этот обмен выполняется в пакетном режиме, длина пакета программируется (1, 2, 4, 8 или 16 передач), но пакет может быть укорочен подачей следующей команды обращения к каналу. Первые данные при чтении канала появляются с задержкой чтения (read latency) в 2 такта относительно команды чтения, следующие идут в каждом такте. В некоторых моделях микросхем имеется поддержка комбинированной команды PFR (перед которой тоже должна идти команда ACT) — предвыборка с автопредзарядом и чтение буфера. После подачи этой команды первые данное появляются на 4-м такте — не раньше и не позже, чем при поочередной подаче команд PRF(A) и READ.

Регенерация VC DRAM выполняется так же, как и в SDRAM, — либо периодической подачей команд REF (авторегенерация по внутреннему счетчику адреса регенерируемых строк), либо в энергосберегающем режиме саморегенерации, в который микросхемы переходят по команде SELF.

Как видно из этого описания, работа микросхем VC DRAM и SDRAM очень похожа, но в VC DRAM операции обмена данными разделены на две сравнительно независимые фазы. Активация-деактивация банков выглядит так же, но при чтении VC DRAM данные появляются даже позже, чем в случае SDRAM: у SDRAM эта задержка (CL) составляет 2-3 такта, у VC DRAM — 4 такта. Тем не менее, применение VC DRAM дает прирост производительности памяти почти по всем тестам. Этот выигрыш получается за счет поддержки многозадачности в самих микросхемах и в контроллере памяти. Памяти приходится обслуживать обращения от нескольких абонентов (процессор, порт AGP, шина PCI), и контроллер памяти, являющийся центральной фигурой северного моста (хаба) чипсета системной платы, «знает», от кого приходит конкретное обращение. Для каждого абонента в отдельности обращения к памяти носят не совсем произвольный характер — скорее их можно рассматривать как потоки последовательных обращений или их смесь (в многозадачных системах). Контроллер может связать каждый поток со своим канальным буфером, и поток с хранящими матрицами микросхем будет обмениваться весьма крупными блоками данных. Применительно к 8-чиповым модулям DIMM, в которых используются 8-битные микросхемы VC DRAM, этот блок имеет размер 2 Кбайт. Передача этих данных между контроллером памяти и канальными буферами выполняется быстро, без всяких активаций и деактиваций — задержка первых данных составляет всего 2 такта, а не 5, как в SDRAM. При типовой длине пакета (4) потери времени на обращение к матрице (по 6 тактов) возникают лишь раз на 32 пакета. Обращения к канальным буферам требуют небольших затрат

энергии, так что VC DRAM еще и выгодно отличается от SDRAM по потреблению. Большое количество канальных буферов (16) позволяет устанавливать виртуальный канал для каждого потока, выявленного контроллером памяти. В виртуальные каналы могут объединяться и несколько буферов. С точки зрения повышения производительности строку выгодно читать целиком: активировать ее, затем последовательными командами считать (сохранить) все ее 4 сегмента, использовав 4 канальных буфера, и деактивировать. Но это уже детали оптимизации, а более существенно то, что для работы с микросхемой VC DRAM контроллер памяти должен «знать» ее систему команд, не имеющую прямой совместимости с командами SDRAM. Поддержка VC DRAM реализована далеко не во всех чипсетах — ее вводят, например, VIA и SiS, но Intel эту память игнорирует. Механически и электрически модули VC DRAM совместимы с обычными модулями DRAM. Во время начального тестирования (POST) модули VC DRAM могут быть опознаны по информации, хранящейся в микросхеме EEPROM последовательной идентификации модуля, либо по поведению после инициализации. Модули VCM выпускают фирмы NEC и Kingston, информацию о наличии этих модулей у других производителей памяти автору найти не удалось.

Память VC DRAM по сравнению с другими типами динамической памяти обеспечивает меньшее среднее время задержки данных в многозадачных системах. Однако по пиковой скорости передачи она не имеет преимуществ перед SDRAM и проигрывает RDRAM и DDR SDRAM.

Сравнительные характеристики и перспективные типы динамической памяти

Асинхронная память FPM, EDO и BEDO развивалась и широко применялась при частотах системной шины компьютеров до 66 МГц. При наименьшем по тем временам времени доступа 40 нс эти типы памяти обеспечивали пакетные циклы с параметрами 5-3-3-3, 5-2-2-2 и 5-1-1-1 соответственно. Динамической памяти с действительно произвольным доступом, выполняемым даже с частотой 66 МГц, нет — для этого требуется время доступа 15 нс, что пока недостижимо. Быстродействие памяти определяется не только собственно микросхемами памяти, но и задержками окружающих их элементов (коммутирующих элементов чипсета, внешних буферов), длины проводников, емкостной нагрузки на шины (зависящей и от количества посадочных мест и установленных элементов памяти). Дальнейшее повышение тактовой частоты для памяти DRAM приводит к появлению тактов ожидания в середине передачи пакета (циклы 6-2-2-2 и хуже), что сводит на нет смысл повышения частоты. Память BEDO широкого распространения не получила (перспектив повышения частоты для нее не было), ее вытеснила память SDRAM с циклом 5-1-1-1 на частотах даже выше 100 МГц.

Применительно к системной памяти SDRAM эффективно используется для частоты системной шины вплоть до 133 МГц. Однако появление режима 4x для порта AGP, а позже и 8x поставило задачу повышения производительности па

мяти, что, как видно из вышеизложенного, уже реализуется двумя путями: во-первых, в памяти DDR и DDR2 SDRAM, во-вторых, в памяти RDRAM. Первый путь является эволюционным — это просто повышение скорости доставки данных SDRAM без смены общей идеологии построения памяти. Второй путь — революционный, поскольку идеология канала Rambus весьма своеобразна. В этой памяти высокая производительность достигается при сокращении разрядности шины данных (в модулях DIMM SDRAM разрядность составляет 8 байт, а в RIMM RDRAM — 2 байта). Уменьшение числа линий интерфейса позволяет контролировать топологию разводки печатных проводников, правда, с повышением частоты и требования ужесточаются. Память RDRAM позволяет организовать чередование большого числа банков, что для традиционной памяти и памяти DDR SDRAM становится проблематичным из-за разрастания числа линий интерфейса.

При всей, казалось бы, перспективности память RDRAM не завоевала мир PC-совместимых компьютеров — из-за проблем с поставками и высокой цены. Фирма Intel (основной «толкач» этой технологии) выпустила несколько чипсетов для RDRAM, но из-за отсутствия самой памяти для них пришлось делать временные «заплаты» MTH (Memory Translator Hub) — преобразователи интерфейса RDRAM (от чипсета) в интерфейс SDRAM (распространенных модулей). Эти преобразователи устанавливались либо на системную плату, либо на модуль памяти (который, таким образом, переставал быть стандартным). Производительности памяти, естественно, они не добавляют.

Память SDRAM использовалась для частот 66-133 МГц. При традиционной разрядности шины 8 байт частоты 100 и 133 МГц (модули PC100 и PC133 соответственно) обеспечивают пиковую производительность 800 и 1064 Мбайт/с соответственно. Модули DDR SRDAM с маркировкой PC1600, PC2100, PC2700 и PC3200 обеспечивают пиковую производительность от 1600 до 3200 Мбайт/с, у них данные передаются с удвоенной частотой 100, 133, 166 и 200 МГц соответственно. Память DDR2-400, DDR2-533 и DDR2-666 обеспечивает пропускную способность 3200, 4260 и 5328 Мбайт/с. У модулей RDRAM числа в названии (600, 700, 800, 1066 и 1200) обозначают округленную частоту схода двухбайтных данных с конвейера RDRAM. Таким образом, их пиковая производительность составляет 1200, 1424, 1600, 2132 и 2400 Мбайт/с — до появления DDR SDRAM эти цифры казались внушительными.

В этой теоретической производительности не учитываются накладные расходы на регенерацию, подразумевается, что требуемые страницы уже открыты. Пиковая производительность интересна на передаче пакетов, причем удлинение пакетов уменьшает влияние начальной задержки на производительность памяти. Однако все это хорошо только при *последовательных* обращениях. При *произвольных* обращениях время доступа «обойти» все равно не удается. А поскольку к памяти обращается не только процессор(ы), но и другие абоненты (порт AGP, мастера шины PCI), характер обращений к памяти в современных компьютерах далек от последовательного. На реальном произвольном потоке запросов производительность, конечно же, ниже. Повышение частоты SDRAM с 66 до

100 МГц, на первый взгляд, должно дать повышение производительности памяти на 50 %, но реальный прирост составляет около 20 %.

Потенциальные возможности почти одновременного обслуживания множества запросов, предоставляемые микросхемами SDRAM и RDRAM, могут быть реализованы лишь при достаточно «умном» контроллере памяти. От его предусмотрительности эффективность памяти зависит, пожалуй, больше, чем у простых модулей FPM и EDO DRAM. Контроллер может и не закрывать страницу после каждого обращения к памяти, а держать открытыми несколько страниц в разных банках памяти. Но что потребуется дальше, контроллер не знает. Если он оставит страницу открытой и следующее обращение произойдет к той же странице, время будет выиграно (случай попадания). Если же произойдет обращение к другой странице того же банка, придется сначала деактивировать банк (закрыть страницу), выждать некоторое время предварительного заряда и только после этого начинать полный цикл обращения (случай промаха). Применение виртуальных каналов (VC DRAM) открывает чипсету больше возможностей для оптимизации планирования обращений к памяти, чем в случае SDRAM (не говоря уже о DRAM). Заметим, что внедрение канальных буферов изменяет политику контроллера памяти в плане удержания строк в открытом состоянии, поскольку цена выигрыша и потерь времени от попаданий и промахов меняется.

Одним из способов повышения производительности динамической памяти является помещение статической кэш-памяти прямо на кристалл памяти. Такая двухступенчатая память выпускается и применяется в некоторых моделях PC. Эта архитектура реализована в микросхемах *CDRAM* (Cached DRAM) — продуктах фирм Mitsubishi и Samsung. Микросхемы CDRAM емкостью 4 и 16 Мбит имеют 16-килобайтный кэш статической памяти со 128-битной внутренней шиной данных. Другая линия — *EDRAM* (Enhanced DRAM) — продукция фирмы Ramtron International. Микросхемы EDRAM емкостью 4 Мбит имеют 8-килобайтный кэш статической памяти с разрядностью внутренней шины данных 2048 бит. Память с внутренним кэшем существенно эффективнее обычной комбинации DRAM и вторичного кэша, особенно в многозадачных системах, где переключение задач приводит к высокой вероятности кэш-промахов обычного кэша.

Динамическая память для графических адаптеров *(VRAM, WRAM, MDRAM, SGRAM)* рассмотрена в [1].

8.1. Применение модулей DRAM в оперативной памяти

Динамическая память в настоящее время практически незаменима в качестве основной (оперативной) памяти компьютеров. Имея общее представление о работе разных типов динамической памяти, можно обсудить варианты построения оперативной памяти и «организационные» способы повышения ее производительности.

Ниже перечислены меры, применяемые для повышения производительности памяти:

- Повышают быстродействие ядра — снижают время доступа запоминающих ячеек, пока остановились на 30 нс.
- Применяют конвейеризацию, внешнюю (память EDO) или внутреннюю (BEDO, SDRAM, RDRAM).
- Увеличивают количество независимых банков, внешне (увеличивая число интерфейсных сигналов) или внутренне (в SDRAM — до четырех, в DDR2 — до восьми). Чем больше будет независимых банков в ОЗУ, тем больше вероятность их одновременного использования при обслуживании произвольных конкурирующих запросов.
- Увеличивают разрядность шины данных, для современных процессоров — до 8 байт.
- Повышают скорость передачи данных по интерфейсу памяти — в SDRAM частота «схода с конвейера» доходит до 100-133 МГц, в DDR SDRAM — до 2 х 200 = 400 МГц, в DDR2 - до 2 х 400 = 800 МГц.
- Вводят канальные буферы между запоминающим ядром и внешним интерфейсом (DDR, DDR2 SDRAM; RDRAM; VC DRAM).

Увеличение количества независимых банков и разрядности шины данных очень мешают повышению скорости передачи данных по интерфейсу памяти — 96 цепей к одному модулю развести без «перекосов» довольно сложно. Широкая разрядность интерфейса сковывает и масштабируемость памяти: нельзя увеличивать объем ОЗУ, добавляя по одной микросхеме — можно только по четыре (а чаще по восемь). Это противоречие как раз и должна была разрешить технология Rambus DRAM. Однако, как показало время, удвоенную (относительно «классической», 8-байтной) разрядность используют и для DDR, и DDR2 SDRAM, а наращивание объема «по чуть-чуть» уже неактуально.

Массовая память настоящего времени — DDR SDRAM 100-200 МГц; ее более мощная альтернатива — DDR2 SDRAM с частотой 200-400 МГц.

Банк памяти набирается из модулей (или микросхем) DRAM, количество которых обеспечивает разрядность, требуемую микропроцессором (и чипсетом), включая (в случае контроля четности или ECC-контроля) контрольные биты. В банке все одноименные адресные входы микросхем и линии RAS# соединяются параллельно. Каждый банк выбирается своим сигналом RAS#. Линии CAS# *щи* (и) WE должны быть индивидуальными для каждого байта, чтобы обеспечить возможность индивидуальной записи в любой байт банка. Модули могут содержать один или два *банка микросхем* (двусторонние модули).

На системных платах с 8-байтной шиной памяти (Pentium и выше) банк состоит из одного модуля DIMM (или пары модулей SIMM-72, причем оба модуля каждого банка используют общие линии RASx#).

Память PC (кроме систем с ECC) допускает возможность побайтного обращения (что существенно для операций записи). Выбор байтов, участвующих в операциях, осуществляется сигналами CASx#, следовательно, каждый байт

банка должен иметь собственную линию для этого сигнала. В системах без чередования банков линии CASx# для одноименных байтов всех банков обычно объединяются. Чередование увеличивает потребность в этих линиях в зависимости от схемы чередования (two-way interleaving — в два раза, three-way interleaving, — соответственно, в три и т. д.).
Увеличению числа слотов препятствует ограниченная нагрузочная способность шины памяти — каждый слот (тем более с модулем) вносит паразитную емкость и индуктивность, ограничивающие быстродействие шины. Из-за влияния этой нагрузки для работы модулей SDRAM на частоте шины 100 МГц была разработана спецификация PC 100, в которой помимо требований к быстродействию микросхем памяти задаются правила разводки сигнальных и питающих проводников и прочие конструктивные нюансы. Затем появилась и аналогичная спецификация PC133 для частоты шины 133 МГц. Однако повышение тактовой частоты традиционной шины памяти технически сложно из-за большого числа сигнальных проводников. В модулях DIMM SDRAM используют 96 сигнальных цепей: 32 адресных и управляющих линии и 64 (с контрольными — 72 или 80) линии данных, при этом каждый дополнительный слот памяти требует еще несколько управляющих линий. На высоких частотах приходится учитывать задержки распространения сигналов в проводниках и, что самое неприятное, — неодинаковость, или перекос (skew), этих задержек.
Повышение производительности памяти за счет чередования банков требует некоторого усложнения контроллера памяти и обеспечения независимости банков (возможности активации одного банка до предварительного заряда предыдущего). Независимость банков для асинхронной памяти (DRAM) достигается сугубо экстенсивным способом — значительным увеличением числа линий интерфейса. Микросхемы синхронной памяти SDRAM могут иметь внутреннюю 4-банковую организацию, независимость банков поддерживается синхронным интерфейсом. В памяти SDRAM для выбора физических банков микросхем вместо нескольких сигналов RASi# используются сигналы Si#, и проблемы дефицита управляющих линий (в случае двусторонних модулей DIMM) относятся уже к этим сигналам. Здесь чередование банков выполняется внутри микросхем и не требует дополнительных интерфейсных сигналов.

Модули динамической памяти

Динамическая память чаще всего применяется в виде модулей с разрядностью
1, 2, 4 или 8 байт, которые могут устанавливаться пользователем без каких-либо приспособлений. Модули стандартизованы, поэтому обеспечивается взаимная совместимость.

- *SIPP* и *SIMM-30* — самые первые модули с однобайтной организацией, применялись вплоть до процессоров класса 486.
- *SIMM-72-pin* — 4-байтные модули, применявшиеся на системных платах для процессоров классов 486 и Pentium.
- *DIMM-168* — 8-байтные модули для Pentium и выше. Существуют два поколения, существенно различные по интерфейсу. Модули DIMM 168-pin Buf

fered (1-го поколения), как и слоты для них, встречаются редко и с широко распространенными модулями DIMM 2-го поколения несовместимы даже механически (по ключам). Наиболее популярно второе поколения с микросхемами SDRAM. Различают модификации в зависимости от наличие буферов или регистров на управляющих сигналах: Unbuffered, Buffered и Registered.

- *DIMM-184* — 8-байтные модули DDR SDRAM для системных плат 6-8-го поколений процессоров.
- *DIMM-240* — 8-байтные модули DDR2 SDRAM для системных плат 7-8-го поколений процессоров.
- *RIMM* — 2-байтные модули RDRAM для системных плат 6-7-го поколений процессоров.
- *SO DIMM* (72-, 144- и 200-pin) и *SO RIMM* — малогабаритные варианты модулей (для блокнотных ПК).
- *AIMM* (AGP Inline Memory Module), они же *GPA Card* (GPA расшифровывается как Graphics Performance Accelerator) — 66-контактные 32- или 16-битные модули SDRAM, предназначенные для расширения памяти графических адаптеров, встроенных в системную плату.

Не пересчитывая контакты, отличить «короткие» модули SIMM от «длинных» и от модулей DIMM легко по их размеру: длина модуля SIMM-30 pin — примерно 89 мм, SIMM-72 — 108 мм. Модули DIMM-168, DIMM-184 и DIMM-240 имеют одинаковую длину около 134 мм (5,25"), но у 168-контактных модулей два ключа, у 184- и 240-контактных — один (за счет чего больше контактов); кроме того, у DIMM-184 и DIMM-240 по две прорези по бокам, а не по одной. У модулей DIMM-240 шаг контактов мельче (1 мм), чем у DIMM-184 (1,27 мм), и ключи питания расположены выше. Модули RIMM имеют ту же длину, но их легко отличить по меньшему числу контактов — середина краевого разъема свободна от ламелей. У модулей RIMM микросхемы памяти закрыты пластиной радиатора. Кроме того, их левый ключ гораздо ближе к центру, чем у DIMM.

В компьютерах, предназначенных для использования в качестве серверов или мощных станций, нередко применяются специальные платы памяти, позволяющие устанавливать большие объемы ОЗУ. На такие платы также устанавливаются модули SIMM, DIMM или SO DIMM. Модули памяти применяются и в принтерах (лазерных) — DIMM-168, 100-Pin DIMM, AIMM, SO DIMM-144, но иногда для них требуются и специальные (по конструктиву или параметрам) модули.

Современные модули памяти имеют шину данных разрядностью 1, 4 или 8 байтов. Помимо основных информационных битов, модули могут иметь дополнительные контрольные биты с различной организацией:

- Модули без контрольных битов *(non Parity)* имеют разрядность 8, 32 или 64 бита и допускают независимое побайтное обращение по отдельным для каждого байта линиям CAS#.

- Модули с контролем четности (*Parity*) имеют разрядность 9, 36 или 72 бита и также допускают независимое побайтное обращение, контрольные биты по обращению приписаны к соответствующим байтам.
- Модули с генератором четности *(Fake Parity, Parity Generator, Logical Parity)* также допускают независимое побайтное обращение, логические генераторы четности по чтению приписаны к соответствующим байтам. Действительного контроля памяти они не обеспечивают.
- Модули с контролем по схеме *ECC* имеют разрядность 36, 40, 72 или 80 битов. Обычно они допускают побайтное обращение к информационным битам, но контрольные биты у них привязаны к одному или нескольким сигналам CAS#, поскольку ECC подразумевает обращение сразу к целому слову.
- *ECC-Optimized* — модули, оптимизированные под режим ECC. От обычных модулей ECC они отличаются тем, что могут не обеспечивать побайтное обращение к информационным битам.
- *ECC-On-Simm* (EOS) — модули со встроенной схемой исправления ошибок. Каждый байт модуля имеет встроенные средства контроля и исправления ошибок, работающие прозрачно. Для системы модули функционируют как обычные с контролем четности — в случае обнаружения неисправимой ошибки они генерируют ошибочный бит четности. Эти модули обеспечивают отказоустойчивость по памяти (kill protected memory) для системных плат, поддерживающих только контроль четности. По «благородству» поведения (делают больше, чем «говорят») они являются прямой противоположностью модулям с генератором четности.

Набор сигналов модуля SIMM в основном совпадает с сигналами одиночных микросхем динамической памяти. Основные характеристики распространенных модулей приведены в табл. 8.6, более подробное описание см. в следующих разделах.

Таблица 8.6. Основные характеристики модулей памяти

Модуль	Разрядность[1], бит	Объем, Мбайт	Тип	Питание, В	Спецификация
SIMM-30, SIPP	8 (9)	0,25–4	FPM, EDO	5	60, 70, 80 нс
SIMM-72	32 (36)	1–32	FPM, EDO, BEDO	5	50, 60, 70 нс
DIMM-168-I	64 (72, 80)	8–256	FPM, EDO	5	50, 60, 70 нс
DIMM-168-II	64 (72, 80)	8–512	FPM, EDO	5, 3,3	50, 60, 70 нс
DIMM-168-II	64 (72, 80)	8–2048	SDRAM	3,3	PC66, PC100, PC133 (CL = 2, 3)
DIMM-184	64 (72, 80)	128–4096	DDR SDRAM	2,5	PC1600, PC2100, PC2700, PC3200 100, 133, 166, 200 МГц (2-2-2, 2,5-3-3, 3-3-3, 3-4-4)

Модуль	Разрядность[1], бит	Объем, Мбайт	Тип	Питание, В	Спецификация
DIMM-240	64 (72, 80)	256–1024	DDR2 SDRAM	1,8	PC2-3200, PC2-4300, PC2-5300, PC2-5300, PC2-6400
					200, 266, 333, 400 МГц
					3-3-3, 4-4-4, 5-5-5
SODIMM-200	64 (72, 80)	256–1024	DDR2 SDRAM	1,8	200, 266, 333 МГц
					3-3-3, 4-4-4, 5-5-5
AIMM	32	4	SDRAM	3,3	166 МГц
100-pin DIMM	32	4–128	SDRAM	3,3	100, 125 МГц
100-pin DIMM	32	4–32	FPM, EDO	3,3	50, 60 нс
SO DIMM-72	32 (36)	4–32	FPM, EDO	3,3	50, 60 нс
SO DIMM-144	64 (72)	32, 64	FPM, EDO	3,3	50, 60 нс
SO DIMM-144	64 (72)	32–512	SDRAM	3,3	66, 100, 125, 133 МГц (CL = 2, 3)
µSO DIMM-144	64 (72)	64–256	SDRAM	3,3	100, 133 МГц (CL=2, 3)
RIMM	16 (18)	64, 96, 128, 256	RDRAM	2,5	PC600, PC700, PC800, PC1066
SO RIMM	18	64–384	RDRAM	2,5	PC600, PC700, PC800, PC1066

[1] В этом столбце в скобках указана разрядность с учетом битов четности или ECC.

Спецификация быстродействия у разных типов памяти отражает различные параметры и выбирается, исходя из технических и маркетинговых соображений. Для асинхронной памяти указывают время доступа (в наносекундах). Для памяти SDRAM указывается тактовая частота, на которой она работает с достойным значением латентности (на более высокой частоте она, возможно, и будет работать, но с большим значением CL). В обозначениях PC66, PC100 и PC133 здесь тоже имеется в виду частота (отсутствие обозначения соответствует 66 МГц — поначалу иных спецификаций не было), а также соответствие спецификациям Intel. Для DDR SDRAM и DDR2 SDRAM числа в спецификации (PC1600-PC3200, PC2-3200-PC2-6400) отражают пиковую пропускную способность (Мбайт/с). Из этих чисел можно определить *тактовую частоту* (например, для PC3200 и PC2-3200 — 200 МГц) и частоту передачи данных (для них же — 400 МГц).

Для RDRAM числа в названии обозначают округленную частоту передачи пар байтов, то есть соответствуют половине пиковой производительности (Мбайт/с).

Существуют *адаптеры*, преобразующие форматы модулей SIMM (*SIMMVerter, SIMMSaver).* Они позволяют, например, сложить из четырех модулей SIMM-30 один SIMM-72 или из двух односторонних модулей SIMM-72 сложить один двусторонний. Трудно назвать такие конструктивные решения элегантными и надежными (появляется слишком много механических соединений и контактов), но их применение бывало оправдано при дефиците гнезд на плате. Или,

например, при наличии четырех 4-мегабайтных модулей SIMM-30 можно было сделать 16-мегабайтный модуль SIMM-72. Следует помнить о повышенной нагрузке на шины, вносимой такими «супермодулями» с непомерным количеством микросхем и проводников.

Идентификация модулей

Для автоматической идентификации наличия и типа установленного модуля применяются различные методы, основанные на считывании конфигурационной информации с модуля (параллельная или последовательная идентификация) или «исследовании» свойств модуля во время начального тестирования по включении питания.

Метод *параллельной идентификации* начал применяться с модулями SIPP и SIMM-30 фирмы IBM. В интерфейс этих модулей были введены два дополнительных вывода, и по заземленным (на модуле) сигналам системная плата могла распознать наличие и объем установленной памяти. В SIMM-72 для идентификации предназначались 4 вывода (для ECC-модулей — 5), которые должны были нести информацию об объеме, быстродействии и типе применяемой памяти. Этот метод не выдержал натиска новых типов памяти, поскольку описать их важнейшие параметры четырьмя битами невозможно. На системных платах эти линии обычно заземлены, а чипсет распознает модули, выполняя специальные циклы диагностических обращений к, памяти. В модулях SO DIMM-72 используются 7 битов параллельной идентификации, а в DIMM-168 первого поколения — 10. Однако и десяти битов недостаточно для радикального решения проблемы — таблицы типов модулей памяти получаются неоднозначными и противоречивыми. Кроме того, метод предполагает наличие линий идентификации, собственных для каждого гнезда модуля. Эти линии надо провести по плате и считать через какой-либо порт. Некоторое облегчение принесла буферизация битов идентификации, позволяющая объединять одноименные выходы всех модулей и обращаться к конкретному модулю по разрешающему сигналу. Такая схема для восьми из десяти битов применяется в DIMM-168.

В новых модулях памяти — DIMM-168 второго поколения, DIMM-184, DIMM- 240, SO DIMM-144, SO DIMM-200 — используют последовательную идентификацию (serial presence detection). На модуль устанавливается микросхема специальной энергонезависимой памяти с последовательным доступом по двухпроводному интерфейсу I^2C, хранящая исчерпывающую конфигурационную информацию. Формат конфигурационных данных стандартизован JEDEC, из доступных 256 байт под параметры пока определены только первые 32 и еще 32 зарезервированы, 64 байта отданы под информацию производителя (табл. 8.7). Основные параметры описываются в явном виде, например, временные — в наносекундах, количество битов адреса задается числами. Интерфейс I^2C позволяет легко объединять интерфейсные сигналы со всех модулей, что существенно проще, чем коммутация 4-10 линий параллельной идентификации. На разъем модулей DIMM-168 выведены 3 бита адреса SA[0:2], что позволяет разводкой этих выводов адресовать до восьми модулей с объединенными линиями синхронизации и данных. При необходимости расширения следую-

щие восемь модулей потребуют от контроллера (чипсета) еще только одной двунаправленной или выходной линии. Адрес в SO DIMM-144 фиксирован, так что двухпроводный интерфейс позволяет опрашивать только один модуль, а каждый следующий модуль требует по одной дополнительной линии. В таблице приведена структура идентификатора для модулей «классических» вариантов DRAM, для RD RAM формат описания иной.

Таблица 8.7. Назначение байтов последовательной идентификации

Байт	Назначение
Стандартизованная информация о микросхеме	
0	Число записанных байтов конфигурационной памяти
1	Разрядность адреса микросхемы Serial PD (определяет объем конфигурационной памяти: 1 — 2 байта, 2 — 4 байта, ..., 0Dh — 8 Кбайт)
2	Тип памяти: 00 — резерв, 01 — Std FPM, 02 — EDO, 03 — Pipelined Nibble (BEDO), 04 — SDRAM, 07 — DDR SDRAM, 08 — DDR2 SDRAM
3	Количество битов адреса строк в банке 1 (биты 0–3) и банке 2 (биты 4–7) по модулю 16 (0 — не определено, 1 — 1 или 16, 2 — 2 или 17 и т. д.). Если банки одинаковые, то биты 4–7 нулевые
4	Количество битов адреса столбцов (аналогично предыдущему)
5	Количество банков (рядов микросхем)
6–7	Разрядность данных с учетом контрольных битов (если менее 255, байт 7 — 0)
8	Уровень напряжения интерфейса: 0 — TTL/5B, 01 — LVTTL (не допускает 5 В), 02 — HSTL 1.5 В, 03 — SSTL 3.3 В, 04 — SSTL 2.5 В, 05 — SSTL 1.85 В
9	Для DRAM — время доступа (RAS Access time) в наносекундах. Для SDRAM — минимальное время цикла (T_{clk}) для максимального значения CL (десятые доли наносекунд в BCD-коде)
10	Для DRAM — время доступа (CAS Access time) в наносекундах. Для SDRAM — время доступа относительно тактового импульса (T_{ac}), аналогично предыдущему
11	Схема контроля: 00 — Non-Parity, 01 — Parity, 02 — ECC
12	Частота (тип) регенерации: 00 — Normal (распределенный цикл 156 мкс), 01 — Reduced 0.25x (39 мкс), 02 — Reduced 0.5x (78 мкс), 03 — Extended 2x (313 мкс), 04 — Extended 4x (625 мкс), 05 — Extended 8x (125 мкс). Бит 7 является признаком саморегенерации (биты 6:0 кодируют те же периоды)
13	Разрядность микросхем основной памяти в битах. Бит 7 равен 1, если имеется второй банк с удвоенной разрядностью микросхем. Если банк один или оба банка одинаковы, бит 7 равен 0
14	Разрядность микросхем контрольных разрядов в битах (аналогично предыдущему)
15–30	Детальное описание временных и организационных параметров SDRAM
31	Объемы банков (рядов микросхем): бит 0 — 4 Мбайт, бит 1 — 8 Мбайт, бит 7 — 512 Мбайт, единичное значение устанавливается в одном или нескольких (двух) битах
32–35	Время предварительной установки и удержания входных сигналов
36–61	Резерв
62	Ревизия SPD (две BCD-цифры)
63	Контрольная сумма байтов 0–62 по модулю 256

продолжение

Таблица 8.7 (*продолжение)*

Байт	Назначение
Информация изготовителя	
64–71	Идентификатор производителя по JEDEC
72	Код страны производителя
73–90	Код изделия (ASCII)
91–92	Код модификации
93–94	Дата изготовления (wwyy – неделя, год)
95–98	Серийный номер
99–127	Специальные данные изготовителя
126	Спецификация частоты (для Intel) DIMM SDRAM. Частота 66 МГц задается кодом 66h, более высокие значения – числом мегагерц (100 = 64h)
127	Детализация для SDRAM 100 МГц (для Intel)

Преимущество последовательной идентификации перед параллельной состоит в том, что появление новых типов устройств и новых параметров не требует конструктивных изменений — все нововведения могут учитываться чисто программно.

Байты 128-255 конфигурационной памяти свободны. Эту область в принципе можно занимать для пометки компьютера (точнее, модуля памяти) с целью привязки программного обеспечения к конкретному экземпляру PC. Однако при неосторожном использовании модулей с микросхемами без защиты от модификации случайная запись в ячейки 0-127 может привести к недоступности модуля памяти. «Оживить» его можно будет только записью корректных данных.

Модули SIMM-30, SIPP, SIMM-72

Модули SIMM (Single In-Line Memory Module) и SIPP (Single In-Line Pin Package) представляют собой небольшие печатные платы с односторонним краевым разъемом (табл. 8.8). Контактами модулей SIMM являются позолоченные (или покрытые специальным сплавом) площадки, расположенные на обеих поверхностях вдоль одной из сторон. Слово «single» (одиночный) в названии подразумевает, что пары площадок на обеих сторонах эквивалентны (электрически соединяются между собой). У малораспространенных модулей SIPP контакты штырьковые (pin — иголка), эти контакты при необходимости можно припаять к площадкам модулей SIMM (такие контакты когда-то даже продавались в комплекте с модулями SIMM). Модули SIPP оказались непрактичными — их контакты не выдерживают транспортировки и многократной установки.

Таблица 8.8. Организация модулей SIMM

Емкость, Мбайт	С четностью		Без четности	
	30-pin	72-pin	30-pin	72-pin
256 Кбайт	256 K × 9	–	256 K × 8	–
1	1 M × 9	256 K × 36	1 M × 8	256 K × 32

Емкость, Мбайт	С четностью		Без четности	
	30-pin	72-pin	30-pin	72-pin
2	–	512 K × 36	–	512 K × 32
4	4 M × 9	1 M × 36	4 M × 8	1 M × 32
8	–	2 M × 36	–	2 M × 32
16	–	4 M × 36	–	4 M × 32
32	–	8 M × 36	–	8 M × 32
64	–	16 M × 36	–	16 M × 32

На модулях смонтированы микросхемы памяти в корпусах SOJ или TSOP, их адресные входы объединены. Количество и тип микросхем определяются требуемой разрядностью и объемом хранимых данных. Архитектура модулей обеспечивает возможность побайтного обращения, что существенно для записи (byte-write); выбор байтов производится отдельным сигналом на входе CAS# для каждого байта. Распространенные модули имеют напряжение питания 5 В. По логической организации различают *односторонние* и *двусторонние* модули. У односторонних модулей микросхемы смонтированы на одной (передней) поверхности, у двусторонних *двойной комплект (два банка*) микросхем смонтирован на обеих сторонах платы. Эти названия не совсем точны, но имеют прочные позиции и иностранное происхождение (single side и double side). Часто встречаются модули, у которых на второй стороне смонтировано несколько микросхем, дополняющих набор первой стороны до требуемой разрядности (чаще там размещаются контрольные биты). Такие модули являются логически односторонними. У «истинно двусторонних» на обеих сторонах обычно симметрично расположены одинаковые комплекты микросхем.

«Короткие», или *SIMM 30-pin,* модули SIMM (старый тип) имеют 30 печатных выводов и однобайтную организацию (рис. 8.13). Разводка выводов у модулей фирмы IBM (для компьютеров IBM PS/2) отличается от общепринятых стандартных. Различия делают несовместимыми модули с объемом более 1 Мбайт: модули IBM могут быть двусторонними (2 Мбайт), стандартные — только односторонними. Малораспространенные модули SIPP имеют 30 штырьковых выводов и совпадают по разводке со стандартными модулями SIMM 30-pin (SIMM-30). Применение однобайтных модулей (особенно в 32-битных системных платах) в значительной степени сковывает свободу выбора объема памяти. Назначение выводов SIMM-30 и SIPP приведено в [1].

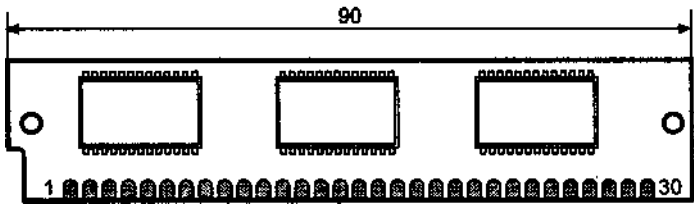


Рис. 8.13. Модуль SIMM-30

«Длинные», или *SIMM 72-pin* (SIMM-72), модули SIMM имеют 72 печатных вывода (рис. 8.14) и 4-байтную организацию с возможностью независимого по-

байтного обращения по сигналам CASx#. По сигналам выборки строк биты данных делятся на два слова, биты DQ[0:15] выбираются сигналом RAS0# для первого банка, биты RAS1# — для второго. Биты DQ[16:31] выбираются сигналами RAS2# и RAS3# соответственно. В односторонних модулях (1, 4, 16, 64 Мбайт — 1 банк) задействуется только одна пара сигналов выборки RAS0# и RAS2#, в двусторонних (2, 8, 32 Мбайт — 2 банка) — две пары сигналов RAS#. Заметим, что использование всеми модулями обеих пар линий RAS# поддерживается не всеми системными платами. Контрольные биты модулей с четностью по выборке приписываются к соответствующим байтам, в ECC-модулях возможны различные варианты. Модули без четности имеют разрядность 32 бит, с четностью — 36 бит, модули ECC — 36 или 40 бит. Модули ECC-36 и ECC-40 (ECC-optimised) не Допускают побайтного обращения и существенно отличаются от 32-битных и модулей с четностью.

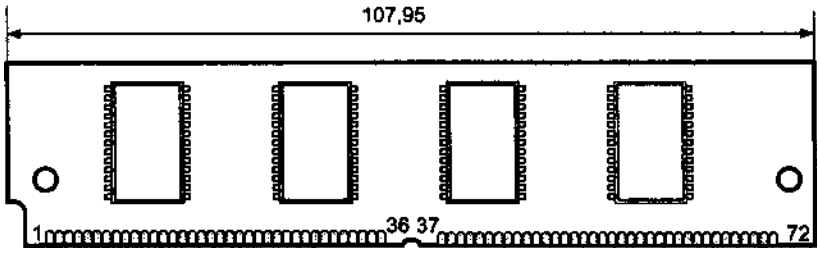


Рис. 8.14. Модуль SIMM-72

Сигналы модулей SIMM в основном совпадают с сигналами микросхем динамической памяти. Для идентификации модулей предназначены сигналы PD[1:5]. По заземленным (на модуле) сигналам системная плата может распознать быстродействие (тип) и объем установленной памяти.

Модули DIMM-168

Модуль памяти *DIMM* (Dual-In-line-Memory Module) имеет 168 независимых печатных выводов (шаг 1,27 мм), расположенных с обеих сторон (контакты 1-84 — с фронтальной стороны, 85-168 — с тыльной). Разрядность шины данных — 8 байт, организация рассчитана на применение в компьютерах с 4- и 8-байтной шиной данных. Конструкция и интерфейс модулей соответствуют стандарту JEDEC 21-C. Модули устанавливаются на плату вертикально в специальные разъемы (слоты) с ключевыми перегородками, задающими допустимое питающее напряжение и тип (поколение) применимых модулей. Модули выпускаются для напряжения питания 3,3 и 5 В. Модули и сочетания ключей представлены на рис. 8.15. Толщина модулей с микросхемами в корпусах SOJ не превышает 9 мм, в корпусах TSOP — 4 мм.

По внутренней архитектуре модули близки к SIMM-72, но имеют удвоенную разрядность и, соответственно, удвоенное количество линий CAS#. Также удвоено число сигналов разрешения записи и разрешения выходных буферов, что позволяет организовывать модули в виде двух 4-байтных банков с возможностью их чередования (bank interleaving). Модули могут иметь разрядность 64,

72 или 80 бит, дополнительные разряды 72-битных модулей организуются либо по схеме контроля четности (приписываясь к соответствующим байтам), либо по схеме ECC; 80-битные — только по схеме ECC.

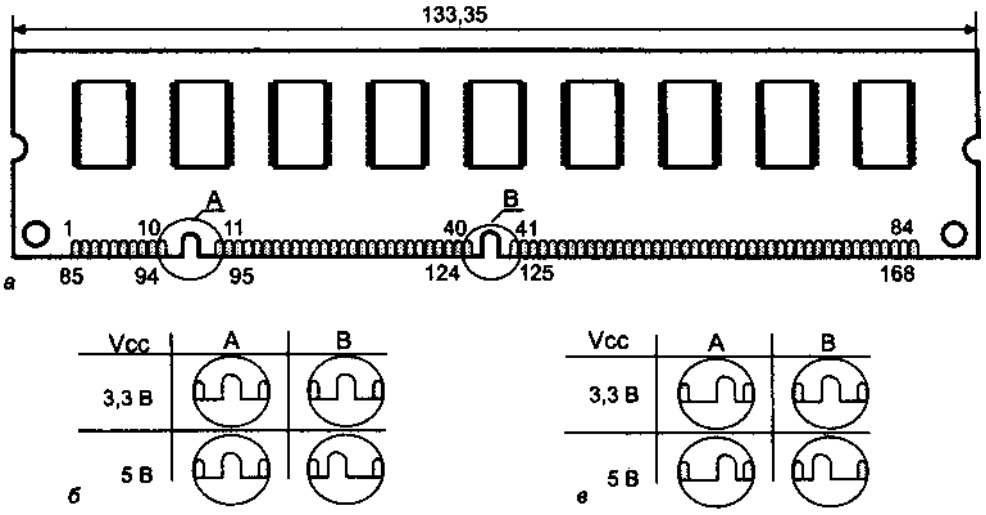


Рис. 8.15. Модули DIMM: а - вид модуля DIMM-168, *б* - ключи для модулей первого поколения, *в* — ключи для модулей второго поколения

Модули на микросхемах емкостью 4-64 Мбит имеют объем от 8 до 256 Мбайт. Модули на 256-мегабитных микросхемах могут иметь емкость до 512 Мбайт. Применение DIMM в системах с 64-битной шиной данных (Pentium, Pentium Pro...) сняло проблему подбора «парных» (идентичных) модулей и открыло перспективы использования новых разновидностей памяти. Высокая плотность упаковки позволяет уменьшить площадь, занимаемую на системной плате памятью большого объема. *168-pin Buffered DIMM* — модули DIMM первого поколения (по IBM), у которых входные адресные и управляющие (кроме RAS#) сигналы буферизованы. Эти модули создают минимальную нагрузку на шину памяти, но буферные микросхемы вносят дополнительную задержку порядка 5 нс (для простоты временные параметры модулей указываются уже с учетом этой задержки). Модули комплектуются микросхемами асинхронной динамической памяти (FPM, EDO и BEDO) и по архитектуре напоминают SIMM-72.

В модулях применяется параллельная идентификация — параметры быстродействия и объема передаются через 8 буферизованных выводов идентификации (presence detect pins).

Наличие сигнала разрешения выходов буфера сигналов идентификации позволяет объединять выводы идентификации нескольких модулей. Два дополнительных небуферизованных вывода несут информацию о разрядности шины и саморегенерации.

Для процессоров с 4-байтной шиной данных в качестве двухбанковых модулей с возможностью чередования могут использоваться 64-битные модули без контроля четности (2 x 32), 72-битные с контролем четности (2 x 36) и 80-битные

ECC (2 x 39/32), за исключением 80-битных на микросхемах 16-бит DRAM. Все модули ECC-72 и модули ECC-80 бит на микросхемах 16-бит DRAM предназначены только для 8-байтных процессоров (72/64 ECC и 80/64 ECC). С точки зрения пользователей PC это ограничение несущественно, поскольку слоты для DIMM появились только на системных платах для процессоров Pentium и выше. Модули первого поколения не получили широкого распространения, поскольку не принесли принципиальных новшеств в подсистему памяти.

Модули второго поколения отличаются тем, что позволяют использовать микросхемы как асинхронной (FPM и EDO), так и синхронной динамической памяти (SDRAM). Внешне они похожи на модули первого поколения, но отличаются ключом, не допускающим ошибочную установку. Унифицированное назначение выводов позволяет в одни и те же слоты устанавливать как модули DRAM, так и SDRAM. Нумерация битов данных единая для всех типов организации — контрольные биты CBx имеют отдельную нумерацию, их наличие зависит от организации (четность, ECC-72, ECC-80). В модулях имеет место *последовательная идентификация параметров* на двухпроводном интерфейсе (I^2C) для чтения атрибутов (идентификации) из специальной конфигурационной памяти (обычно EEPROM 24C02), установленной на модулях.

168-pin Unbuffered DIMM — модули, у которых все цепи не буферизованы (одноименные адресные и управляющие сигналы микросхем соединены параллельно и заводятся прямо с контактов модуля). Эти модули больше нагружают шину памяти, но позволяют добиться максимального быстродействия. Они предназначены для системных плат с небольшим (1-4) количеством слотов DIMM или имеющих шину памяти, буферизованную на плате. Модули выполняются на микросхемах DRAM или SDRAM. Высота модулей не превышает 51 мм. Объем — 8-512 Мбайт.

168-pin Registered DIMM — модули синхронной памяти (SDRAM), у которых адресные и управляющие сигналы буферизованы регистрами, синхронизируемыми тактовыми импульсами системной шины. По виду модули этого типа DIMM легко отличимы — помимо микросхем памяти и EEPROM на них установлено по несколько микросхем регистров-защелок. За счет регистров эти модули меньше нагружают шину памяти, что позволяет набирать больший объем памяти. Применение регистров повышает точность синхронизации, что позволяет повысить тактовую частоту. Однако каждый регистр вносит дополнительный такт задержки. Кроме того, на них может быть установлена микросхема ФАПЧ (PLL), формирующая тактовые сигналы для микросхем памяти и регистров-защелок. Это делается для разгрузки линий синхронизации, причем в отличие от обычной буферизации сигнала, вводящей задержку между входом и выходом, схема PLL обеспечивает синфазность выходных сигналов (их на выходе PLL несколько, каждый для своей группы микросхем) с опорным сигналом (линия CK0). Модули на 64 Мбайт могут быть и без схем PLL — в них линии CK[0:3] разводятся прямо на свои группы микросхем памяти. Регистры могут быть переведены в режим асинхронных буферов (только на 66 МГц), для чего на вход REGE нужно подать низкий уровень. Для модулей на 66 МГц возможна замена регистров асинхронными буферами. Объем — 64-1024 Мбайт.

Модули DIMM-184 DDR SDRAM

Модули DIMM-184 предназначены для микросхем DDR SDRAM. По габаритам они аналогичны модулям DIMM-168, но у них имеются дополнительные вырезы по бокам (рис. 8.16) и отсутствует левый ключ. Разрядность — 64 или 72 битов (ECC), имеются варианты с регистрами в адресных и управляющих цепях (*Registered*) и без них (*Unbuffered*). Напряжение питания — 2,5 В. Идентификация последовательная (SPD). Состав сигналов в основном повторяет набор для DDDR SDRAM. Модули отличаются большим количеством стробирующих сигналов DQSx — по линии на каждые 4 бита данных. Вход тактовой частоты только один, но дифференциальный — раздачу сигналов по микросхемам памяти и регистрам осуществляет микросхема DLL.

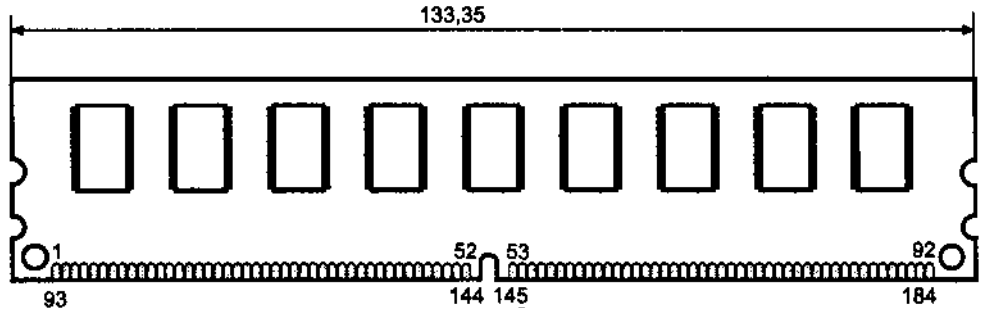


Рис. 8.16. Модуль DIMM-184

Модули DIMM-240 DDR2 SDRAM

Модули DIMM-240 предназначены для микросхем DDR2 SDRAM. По габаритам они аналогичны модулям DIMM-184, но у них используется более мелкий шаг выводов (1 мм) и вырезы по бокам находятся несколько выше (рис. 8.17). Разрядность — 64 или 72 бита (ECC), имеются варианты с регистрами в адресных и управляющих цепях (*Registered*) и без них (*Unbuffered*). Напряжение питания — 1,8 В. Идентификация последовательная (SPD).

Модули RIMM

Модули *RIMM* (Rambus Interface Memory Module), по форме похожие на обычные модули памяти (рис. 8.18), специально предназначены для памяти RDRAM. У них 30-проводная шина проходит вдоль модуля слева направо, и на эту шину без ответвлений напаиваются микросхемы RDRAM в корпусах BGA. Модуль RIMM содержит до 16 микросхем RDRAM, которые всеми выводами (кроме двух) соединяются параллельно. Микросхемы памяти закрыты пластиной радиатора. В отличие от модулей SIMM и DIMM, у которых объем памяти кратен степени двойки, модули RIMM могут иметь более равномерный ряд объемов — в канал RDRAM память можно добавлять даже по одной микросхеме.

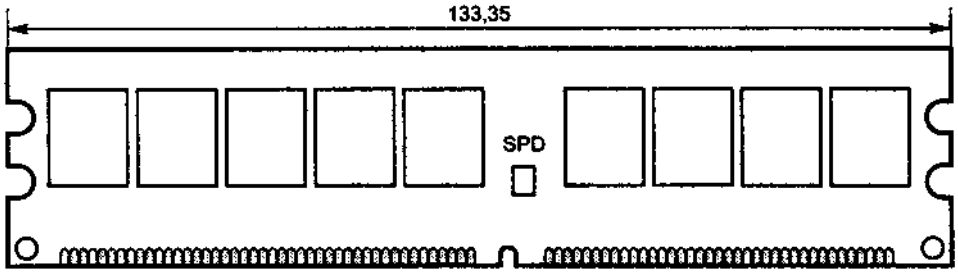


Рис. 8.17. Модуль DIMM-240

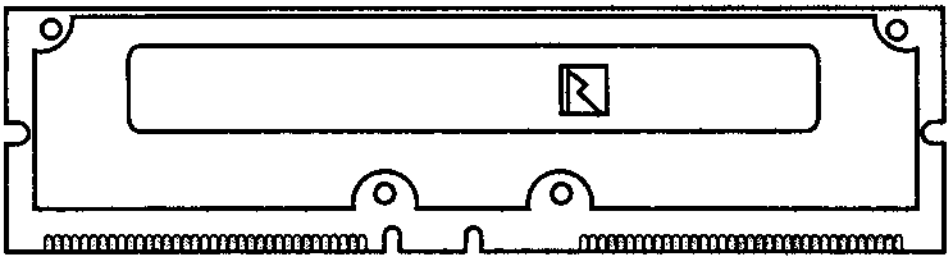


Рис. 8.18. Модуль RIMM

Модули SO DIMM 72-pin

72-pin SO DIMM (Small-Outline-Dual-Inline-Memory Module) — малогабаритный (длина 2,35" — 60 мм) модуль с двусторонним 72-контактным разъемом, нечетные контакты расположены с фронтальной стороны, четные — с тыльной (рис. 8.19). Модули комплектуются микросхемами DRAM в корпусах TSOP, емкость 2-32 Мбайт, разрядность данных — 32 или 36 бит (с контролем четности), 36-битные модули отличаются только наличием дополнительных битов PQx. Память организована в виде двух 2-байтных слов с возможностью побайтного обращения и предназначена для 2- и 4-байтных применений. Информация об объеме, организации, адресации, быстродействии и регенерации передается через семь линий параллельной идентификации.

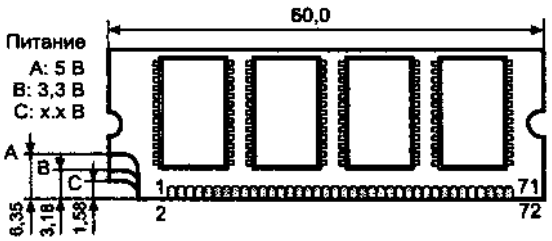


Рис. 8.19. Модуль SO DIMM 72-pin

Модули SO DIMM 144-pin

144-pin SO DIMM — малогабаритный модуль (длина — 67,6 мм, шаг 0,8 мм) с двусторонним 144-контактным разъемом (рис. 8.20), емкость 8-512 Мбайт,

разрядность данных — 64 или 72 бита (ECC). Модули обеспечивают побайтное обращение по сигналам CAS[0:7]#, сигнал RAS0# выбирает банк 0, сигнал RAS1# — банк 1 (при его наличии). Напряжение питания — 5 или 3,3 В, механический ключ напряжения питания расположен между контактами 59-60 и 61-62. Нечетные контакты находятся с фронтальной стороны, четные — с тыльной. Идентификация последовательная. Модули могут содержать микросхемы как DRAM, так и SDRAM 8-256 Мбайт.

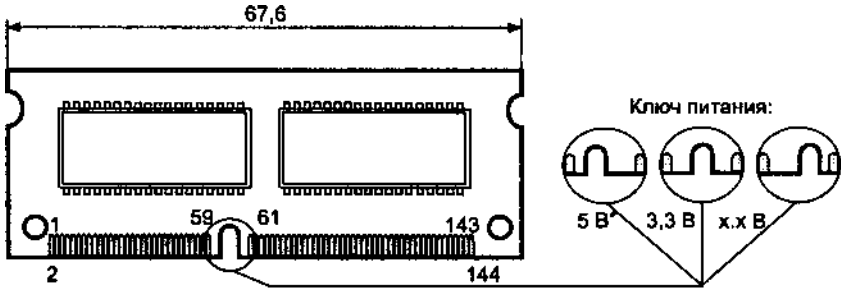


Рис. 8.20. Модуль SO DIMM 144-pin

Модули µSO DIMM 144-pin

µSO DIMM 144-pin — миниатюрный модуль длиной 42 мм (рис. 8.21) с двусторонним 144-контактным разъемом с мелким шагом (0,37 мм). Предназначен для микросхем SDRAM, организация 8-байтная, питание — 2,5 В.

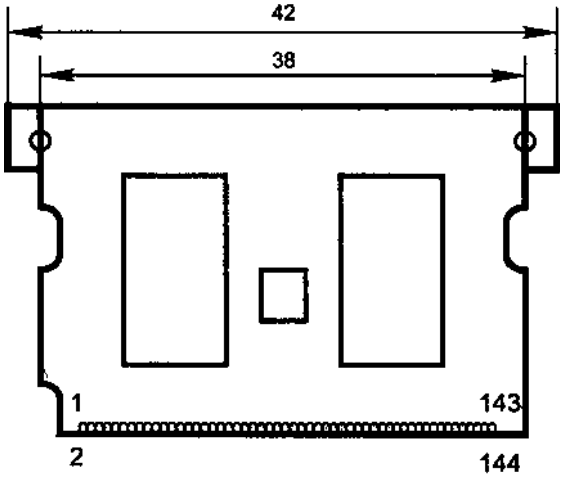


Рис. 8.21. Модуль µSO DIMM 144-pin

Модули SO DIMM 200-pin

200-pin SO DIMM — малогабаритный модуль DDR2 SDRAM (рис. 8.22), разрядность данных — 64 или 72 бита (ECC) Шаг выводов — 0,6 мм. Напряжение питания — 1,8 В. Идентификация последовательная.

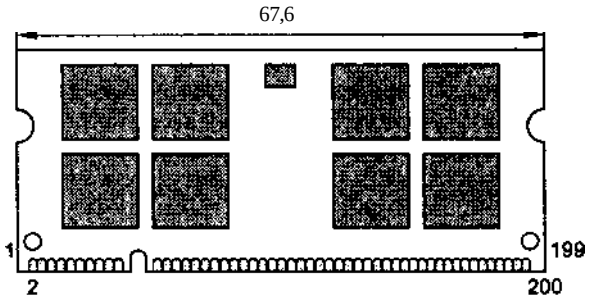


Рис. 8.22. Модуль SO DIMM 200-pin

Модули SO RIMM

SO RIMM — малогабаритный модуль RDRAM (рис. 8.23), разрядность данных — 16 бит. Напряжение питания — 2,5 В.

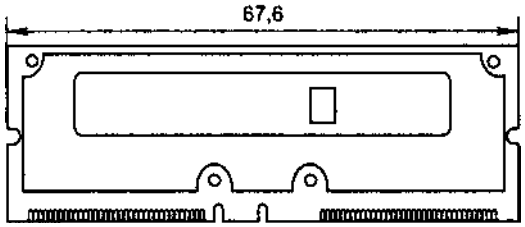


Рис. 8.23. Модуль SO RIMM

Модули DRAM cards 88-pin

88-pin DRAM cards — миниатюрные модули (3,37" x 2,13" x 0,13" — 85,5 x x 54 x 3,3 мм) в пластиковом корпусе размером с карту PCMCIA (PC Card). Имеют 88-контактный разъем (не PCMCIA!), разрядность — 18, 32 или 36 бит, емкость — 2-36 Мбайт. Комплектуются микросхемами DRAM в корпусах TSOP. Информация о быстродействии и объеме передается по восьми выводам. Внутренняя архитектура близка к архитектуре SIMM-72. Напряжение питания — 5 или 3,3 В. Применяются в малогабаритных компьютерах, легко устанавливаются и снимаются.

Нюансы применения DRAM

Микросхемы динамической памяти весьма критичны к *форме управляющих импульсов* — крутизне фронтов и величине выбросов (старые микросхемы можно было даже физически вывести из строя недостаточно крутыми фронтами импульсов). Задача формирования этих импульсов осложняется тем, что они поступают на соединенные вместе входы большого количества микросхем памяти. Для улучшения формы в этих линиях обычно применяют последовательные согласующие резисторы с небольшим сопротивлением. Логически управляющие импульсы формируются микросхемами чипсета. Если плата рассчитана на ус

тановку большого числа модулей памяти, то в ряде случаев ставят внешние буферные микросхемы. Они имеют большую, чем у БИС чипсета, нагрузочную способность и могут использоваться еще и как логические разветвители сигналов. На дешевых платах без буферов могут возникать проблемы при установке большого количества модулей, особенно если модули содержат много микросхем.

Количество микросхем памяти на модулях определяет нагрузку (активную и паразитную емкостную), которую вносят модули на управляющие и особенно адресные линии. Однобайтные (короткие) модули могут быть «девятичиповыми» («восьмичиповыми», если нет контроля четности) на однобитных микросхемах или «трехчиповыми» (2 x 4 бит + 1 бит для четности). «Длинные» модули SIMM собирают из 4- или 16-битных (а для контроля четности — и 18-битных микросхем), контрольные биты могут собираться в одном 4-битном корпусе (для ECC-памяти). При большом суммарном количестве микросхем на установленных модулях возможно превышение нагрузочной способности шины памяти системной платы, результатом которого будет неустойчивая работа памяти. Особенно сильно это проявляется на дешевых системных платах, у которых адресные и управляющие сигналы от микросхем чипсета подводятся непосредственно к банкам памяти, а не заводятся через внешние буферные микросхемы с повышенной нагрузочной способностью. По этим причинам из модулей одинаковой емкости предпочтительнее модули с меньшим количеством микросхем.

Память критична к питанию. Помимо традиционных микросхем с напряжением питания 5 В существуют и низковольтные микросхемы с номиналом питания 3,3 В. Номинал питания, естественно, должен обязательно соблюдаться. Причиной неустойчивой работы памяти может быть и некачественная фильтрация питающего напряжения (по причине неисправности блока питания или выхода из строя фильтрующих конденсаторов).

Микросхемы одного и того же объема могут иметь различный *формат матрицы*. Например, матрица из 4М ячеек может иметь формат (количество строк х количество столбцов) 1К х 4К, 2К х 2К или 4К х 1К. Поскольку при регенерации должны перебираться все строки, эти форматы обозначаются как 1K-Refresh, 2K-Refresh или 4K-Refresh соответственно. Микросхемы с меньшим количеством столбцов потребляют меньшую мощность, особенно при одиночных обращениях. Микросхемы одного и того же объема формата 4K-Refresh потребляют в 2,3 раза меньше мощности, чем формата 1K-Refresh в режиме одиночных обращений и в 1,2 раза — в страничном режиме. В обычных ПК, как правило, применяются микросхемы с количеством строк 1К или 2К. Микросхемы с количеством строк 4К в основном характерны для портативных компьютеров (экономится энергия), а также серверов и мощных станций (большой объем памяти ставит проблему ее охлаждения).

С точки зрения регенерации при цикле CBR формат матрицы несуществен, от контроллера требуется только соблюдение периода генерации циклов. Контроллер регенерации, осуществляющий классический цикл RAS Only Refresh, должен иметь разрядность счетчика адреса, соответствующую количеству строк применяемых микросхем. Одни контроллеры поддерживают регенерацию для

количества строк 2К, другие — для 4К. Микросхемы с меньшей разрядностью адреса строки регенерируются нормально, поскольку за весь цикл счетчика их матрица будет пройдена два или четыре раза. Микросхемы с большей разрядностью адреса строки в полном объеме не регенерируются, что ведет к неработоспособности половины или даже трех четвертей памяти, причем тест POST и некоторые простые тесты эту ситуацию могут не зафиксировать.

Для корректного обращения ко всему объему памяти чипсет должен разделять полный адрес на соответствующее количество битов адресов строк и столбцов. Формат используемых микросхем обычно хранится в регистрах чипсета и задается для каждого банка. Из этого следует, что в банке должны находиться модули одного формата. В принципе, возможна установка микросхем с разрядностью адреса строки, меньшей, чем вырабатывает контроллер памяти. Для этого старший бит (или биты) мультиплексированного адреса в каждом цикле обращения защелкивается специальным регистром по спаду RAS# и в дальнейшем используется как бит адреса столбца. Таким образом может осуществляться преобразование форматов некоторых (или всех) микросхем модуля памяти, однако не для всех систем этот способ успешно работает. Чаще такие преобразователи применяются в модулях с контролем четности или ECC, у которых состав микросхем неоднороден по объему и организации.

Асимметричные модули (у которых в микросхемах количество строк не равно количеству столбцов матрицы) поддерживаются *не всеми чипсетами*, *симметричные — всеми*. Существуют модули псевдосимметричные (на асимметричных микросхемах), их отличительной особенностью является наличие микросхемы 74F08 (ТТЛ-логика), которую легко распознать по надписи и четырнадцати ножкам среди многоножечных микросхем памяти. Этих модулей следует избегать, поскольку круг поддерживающих их чипсетов невелик.

Рекомендации по выбору модулей динамической памяти

В этом разделе перечислены некоторые моменты, на которые надо обращать внимание, приобретая и устанавливая модули памяти:

- Конструктив (SIMM-30, SIMM-72, DIMM-168, DIMM-184, DIMM-240, RIMM) и тип (FPM, EDO, BEDO, SDRAM, DDR SDRAM, DDR2 SDRAM, VC DRAM, RDRAM) должны поддерживаться системной платой.
- Спецификация быстродействия асинхронной памяти (время доступа) должна быть не хуже требуемой на заданной частоте системной шины (и с учетом возможных перспектив замены процессора). Использование модулей со временем доступа большим, чем указано в документации на системную плату, обычно требует увеличения количества тактов в циклах памяти, что не всегда поддерживается параметрами CMOS Setup. «Разогнанная» память имеет полное право работать неустойчиво. Установка модулей более быстродействующих, чем требуется, повышения производительности может и не дать, поскольку циклы обращения к памяти можно (если позволяет чипсет и BIOS) укорачивать только на целое количество тактов и ближайшие возмож

ные значения времени цикла могут «не вписаться» в быстродействие модуля.

- Спецификация частоты модуля синхронной памяти должна быть не ниже требуемой частоты шины памяти. При этом следует отдавать предпочтение модулям с меньшей латентностью на требуемой частоте: память с CL = 2 будет работать несколько быстрее памяти с CL = 3.
- Каждый банк памяти[1] должен быть заполнен однотипными модулями. Некоторые «капризные» платы требуют применения только «родных» фирменных модулей. Не полностью заполненный банк в лучшем случае игнорируется.
- В системах с двухканальной памятью модули в каналах должны быть попарно идентичными. В противном случае контроллер перейдет на одноканальный режим (с потерей производительности).
- В односторонних модулях DRAM (1, 4, 16, 64, 256, 1024, 4096 Мбайт) используется только одна пара сигналов выборки RAS#, в двусторонних (2, 8, 32, 128, 512, 2048 Мбайт) — две пары сигналов RAS#. Некоторые системные платы не могут полностью задействовать объем двусторонних модулей. Иногда установка двустороннего модуля в одном банке исключает возможность использования соседнего банка. Для памяти SDRAM физические банки (стороны модулей) выбираются сигналами CSi#, и для них существует та же проблема дефицита (у чипсета) линий управляющих сигналов.
- Не все системные платы поддерживают асимметричные или псевдосимметричные модули.
- Смешение на плате (и, тем более, в одном банке) модулей, разнотипных по организации (симметричность матрицы и количество занятых линий RAS#), может приводить к неработоспособности или неполному использованию установленной памяти.
- Модули SDRAM различаются по числу используемых входов синхронизации (1, 2 или 4). Старые системные платы, рассчитанные только на модули PC66, могут подавать синхронизацию только на 2 входа модуля — на таких платах модули с 4 входами (4-Clock PC100 и PC133 без PLL) работать не будут.
- Одновременное использование памяти DRAM и SDRAM может не поддерживаться системной платой, а может приводить к снижению производительности SDRAM.
- Возможность применения разнотипных (например, смесь EDO и FPM) модулей в разных банках есть не всегда.
- При использовании в разных банках модулей с разным быстродействием часто производится выравнивание временной диаграммы по самому медленному.

[1] Относится к старым платам, где банк набирался модулями SIMM-72 или даже SIMM-30.

- При установке большого количества модулей с большим количеством микросхем возможна неустойчивость в работе памяти. В этом случае предпочтение следует отдавать модулям с меньшим количеством микросхем. Если системная плата поддерживает буферированную память (buffered, или registered, SDRAM), то имеет смысл применять именно эти модули (производительность немного снизится, но надежность возрастет). Смешивать буферированные и небуферированные модули обычно нельзя.
- На плате, поддерживающей чередование банков (bank interleaving), с точки зрения повышения производительности целесообразно стремиться к заполнению всех банков, участвующих в чередовании, и разрешить чередование настройкой CMOS Setup. Например, в Pentium объем 16 Мбайт памяти можно получить одной парой модулей SIMM-72 по 8 Мбайт и иметь свободный банк на перспективу, а можно набрать и двумя парами по 4 Мбайт и разрешить чередование банков, но дальнейшее наращивание памяти (замена модулей) обойдется дороже. Можно также установить один модуль DIMM SDRAM, в котором чередование встроенное, да и цикл 5-1-1-1.
- При установке более 16 Мбайт памяти для обеспечения непрерывности основной памяти, возможно, потребуется (настройкой CMOS Setup) убрать образ ROM BIOS и/или «дырку в памяти» (memory hole) из-под границы 16 Мбайт.
- Если системная плата поддерживает память с битом четности или ECC, для ответственных применений есть смысл в установке всей памяти с битами четности (но не генераторами четности!) или ECC и разрешении контроля (настройкой CMOS Setup).

Тестирование оперативной памяти

Оперативная память современного компьютера представляет собой взаимосвязанную подсистему основной динамической памяти и обычно двухуровневой статической кэш-памяти. Возможные неполадки с памятью могут иметь источники на любом уровне (правда, отказы внутреннего кэша неразогнанного процессора случаются редко, поскольку выходной контроль процессоров обычно достаточно жесткий). Весьма уязвимым местом памяти являются контактные соединения модулей (микросхем) памяти с системной платой. Здесь возможны как нарушения контактов (полные, которые выявляются легко, и частичные — повышение сопротивления окислившихся контактов, что выявляется с трудом), так и замыкание соседних цепей токопроводящим мусором или погнутым контактом.
Несмотря на однородность и регулярность структуры массива памяти, его тестирование в полном объеме является довольно сложной задачей. Простейшие тесты проверяют способность правильного считывания данных, записанных в ячейку. Для проверки на отсутствие замыкания во внешних цепях или паразитных связей внутри микросхемы служат тесты типа Walk Bit Left/Right, Inverted Walk Bit Left/Right, в которых по ячейкам с нулевыми битами «пробегает» единица или наоборот. Примерно таким же образом проверяется и шина

адреса памяти. Тесты Pseudo Random Read/Write записывают эталонные данные и считывают их в псевдослучайном порядке. Тест регенерации проверяет сохранность данных при отсутствии в течение некоторого времени обращения к хранящим их ячейкам со стороны процессора.

Первоначальное тестирование динамической памяти по включении питания или аппаратному сбросу выполняется процедурой POST. Этот тест определяет объем работоспособной памяти и сообщает его системе. Тест выполняется довольно быстро и обычно выявляет только полный отказ ячеек. Некоторые версии CMOS Setup позволяют ускорять (quick test) или даже отключать тестирование, в этом случае производятся только определение объема и инициализация — «прописывание» всего объема памяти, например, нулями для установки корректных битов четности. Тестирование преходит в два этапа — сначала тестируется минимальный объем (64 Кбайт), необходимый для вывода диагностических сообщений на экран. Если это не удается, тест подает звуковой сигнал. Тестирование полного объема может сопровождаться щелчками (их можно отключить через Setup) при переходе к каждой следующей странице и индикации успешно протестированного объема. По частоте щелчков, которые на современных компьютерах сливаются в непрерывный звук, можно судить
о производительности компьютера и положении переключателя Turbo. При обнаружении ошибки тестирование останавливается со звуковым сигналом (два гудка) и сообщением адреса сбойной ячейки. Распространенной причиной ошибок является разрешение (в Setup) контроля четности при установке обычных модулей или установке вместо них модулей ECC. С применением модулей ECC могут быть проблемы, связанные с различной организацией этих модулей у разных производителей.

При установке памяти свыше 16 Мбайт возможна остановка тестирования POST на такой границе. Обычно это происходит, когда включен параметр Memory Hole At 15M-16M, System BIOS Alias Below 16M или ему подобный, из-за чего теряется непрерывность основной памяти (дань совместимости с компьютерами AT-286, у которых шина адреса была 24-разрядной). При этом и размер памяти, сообщаемый системе, также усекается.

Если доступного (работоспособного) объема памяти достаточно для загрузки ОС (пусть даже простейшей MS-DOS 3.x), можно перейти к тестированию памяти диагностическими программами типа Checklt, PCCheck, QAPlus и т. п. Для тестирования компьютера ОС должна загружаться в минимальном варианте без использования верхней памяти (драйверы типа HIMEM.SYS и EMM386.EXE загружать не следует). Тесты могут выполняться как в ускоренном, так и в полном варианте. Полезно зацикливание (многократные проходы). В случае ошибки тестовая программа сообщает адрес сбойной ячейки, ожидаемый и полученный результат, по которому можно определить характер неисправности. Самопроизвольный рестарт компьютера во время тестирования памяти тоже является указанием на неполадки памяти. Однако успешное прохождение теста — еще не абсолютно надежный показатель ее исправности. Реальное тестирование выполняется на реальных задачах, но при этом трудно определить виновника зависаний и «вылетов» — им может быть и прикладная программа, и операци

онная система, и что-либо другое (адаптер, программа, драйвер, некорректные параметры...). Расширенная память весьма эффективно тестируется при загрузке драйвера HIMEM.SYS, этот тест иногда выявляет ошибки, не фиксируемые специальными диагностическими программами.

Обнаружив ошибку, следует повторить тестирование — это может быть и случайный сбой, вызванный даже попаданием ионизирующей частицы в микросхему памяти (эффективной защитой от таких явлений выступает ECC- или EOS- контроль). Если при повторном тестировании адрес сбойной ячейки устойчиво повторяется, неисправность следует искать в конкретном модуле или микросхеме памяти, а его местоположение можно определить по адресу. В этом случае следует первым делом проверить контакты: особенно ненадежны контакты модулей с луженными площадками, самые надежные — золоченые. Выявить неисправный модуль (или микросхему) можно поочередной заменой модулей на заведомо исправные или перестановками. Плохой контакт при перестановках может устраниться. Поскольку тест POST обычно не сообщает, в каком бите произошла ошибка, а адрес чаще всего указывает на начальный адрес слова со сбойной ячейкой, перестановка неисправного модуля или микросхемы в пределах одного банка может и не привести к заметным изменениям поведения компьютера. Если удается загрузить DOS и тестовую программу, поиск упрощается. Ради возможности загрузки теста стоит попытаться поменять местами банки памяти, если сбой происходит в младших адресах.

Если устойчивости поведения при повторных тестах не наблюдается, следует искать причины в общих узлах и настройке памяти. Первым делом стоит проверить соответствие быстродействия установленных элементов памяти (и динамической, и статической) частоте системной шины и параметрам CMOS Setup. Не вскрывая системный блок, можно для пробы замедлить работу памяти, задав большее количество тактов ожидания на выполнение обращения к памяти (если это позволяет Setup). Диапазон поиска можно сузить, запрещая кэширование (иногда приходится и физически вынимать микросхемы или модули кэш-памяти). И наконец, следует проверить напряжение питания (+5, +3,3 или +2,5 В) на системной плате около модулей памяти. Причиной неработоспособности может быть и обломанный конденсатор в цепи питания. Все возникающие ситуации описать невозможно, но знание вышеизложенных общих принципов работы компонентов и их «капризов» поможет найти выход из лабиринта загадок памяти, которые вносят, пожалуй, основной вклад в «букет» причин полной или частичной неработоспособности компьютера.

8.4. Статическая память

Статическая память (Static Random Access Memory, SRAM), как и следует из ее названия, способна хранить информацию в статическом режиме — то есть сколь угодно долго при отсутствии обращений (но при наличии питающего напряжения). Ячейки статической памяти реализуются на триггерах — элементах с двумя устойчивыми состояниями. По сравнению с динамической памятью эти ячейки более сложные и занимают больше места на кристалле, однако они

проще в управлении и не требуют регенерации. Быстродействие и энергопотребление статической памяти определяется технологией изготовления и схемотехникой запоминающих ячеек. Самая экономичная КМОП-память (CMOS memory) имеет значительное время доступа (более 100 нс), но зато пригодна для длительного хранения информации при питании от маломощной батареи и применяется в PC. Самая быстродействующая статическая память имеет время доступа в несколько наносекунд, что позволяет ей работать на частоте системной шины процессора, не требуя от него тактов ожидания. Объем памяти микросхем SRAM уже достиг 32 Мбит. Относительно высокие удельная стоимость хранения информации и энергопотребление при низкой плотности упаковки не позволяют использовать SRAM в качестве основной памяти компьютеров. В PC микросхемы SRAM в основном применяются для построения вторичного кэша; они могут располагаться как на системной плате, так и на картридже процессора. Разновидности статической памяти — Async SRAM, Sync Burst SRAM и Pipelined Burst SRAM — мы рассмотрим именно с точки зрения этого применения.

Разновидности статической памяти

Асинхронная статическая память (Asynchronous SRAM, Async SRAM), она же обычная, или стандартная, подразумевается под термином SRAM по умолчанию, когда тип памяти не указан (до некоторых пор ему действительно не было альтернативы).

Микросхемы этого типа имеют простейший асинхронный интерфейс, включающий шину адреса, шину данных и сигналы управления CS#, 0E# и WE#. Микросхема выбирается низким уровнем сигнала CS# (Chip Select), низкий уровень сигнала OE# (Output Enable) открывает выходные буферы для считывания данных, низким уровнем WE# (Write Enable) разрешается запись. Временные диаграммы циклов обращения приведены на рис. 8.24. При записи управление выходными буферами может производиться как сигналом OE# (цикл 1), так и сигналом WE# (цикл 2). Для удобства объединения микросхем внутренний сигнал CS# может собираться по схеме «И» из нескольких внешних, например CSO#, CS1 и CS2# — в таком случае микросхема выбирается сочетанием логических сигналов 0, 1, 0 на соответствующих входах.

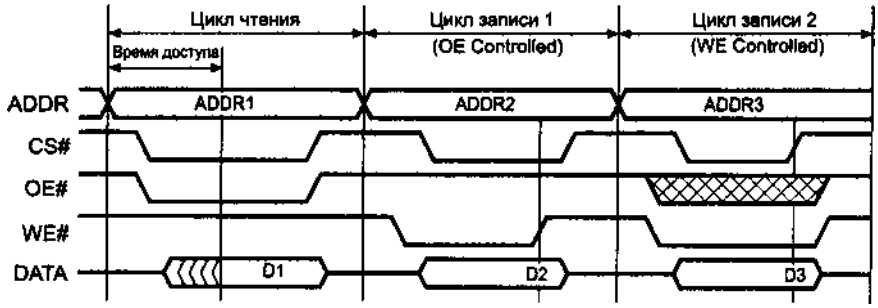


Рис. 8.24. Временные диаграммы чтения и записи асинхронной статической памяти

Время доступа — задержка появления действительных данных на выходе относительно момента установления адреса — у стандартных микросхем SRAM составляет 10, 12, 15 или 20 нс, что позволяет процессору выполнять пакетный цикл чтения *2-1-1-1* (то есть без тактов ожидания) на частоте системной шины до 33 МГц. Наиболее быстрая память имеет время доступа 8 нс. Объем микросхемы SRAM достиг 4 Мбит.

Синхронная пакетная статическая память (Sync Burst SRAM) оптимизирована под выполнение пакетных (burst) операций обмена, свойственных кэш-памяти. В ее структуру введен внутренний 2-битный счетчик адреса. В дополнение к сигналам, характерным для асинхронной памяти (адрес, данные, CS#, OE# и WE#), синхронная память использует сигнал CLC (Clock) для синхронизации с системной шиной и сигналы управления пакетным циклом ADSP#, CADS# и ADV#. Сигналы CADS# (Cache ADdress Strobe) и ADSP# (ADdress Status of Processor), которыми процессор или кэш-контроллер отмечает фазу адреса очередного цикла, являются стробами записи начального адреса цикла во внутренний регистр адреса. Любой из этих сигналов инициирует цикл обращения, одиночный (single) или пакетный (burst), а сигнал ADV# (ADVance) используется для перехода к следующему адресу пакетного цикла. Все сигналы, кроме сигнала управления выходными буферами OE#, синхронизируются по положительному перепаду сигнала CLK. Это означает, что значение входных сигналов должно установиться до перепада и удерживаться после него еще некоторое время. Выходные данные при считывании во время этого перепада остаются действительными. На рис. 8.25 приведены диаграммы нескольких вариантов циклов чтения синхронной статической памяти. Обратим внимание на то что, что 2-битный счетчик адреса не позволяет перейти границу четырехэлементного пакетного цикла. Кроме того, порядок счета адресов внутри пакетного цикла соответствует

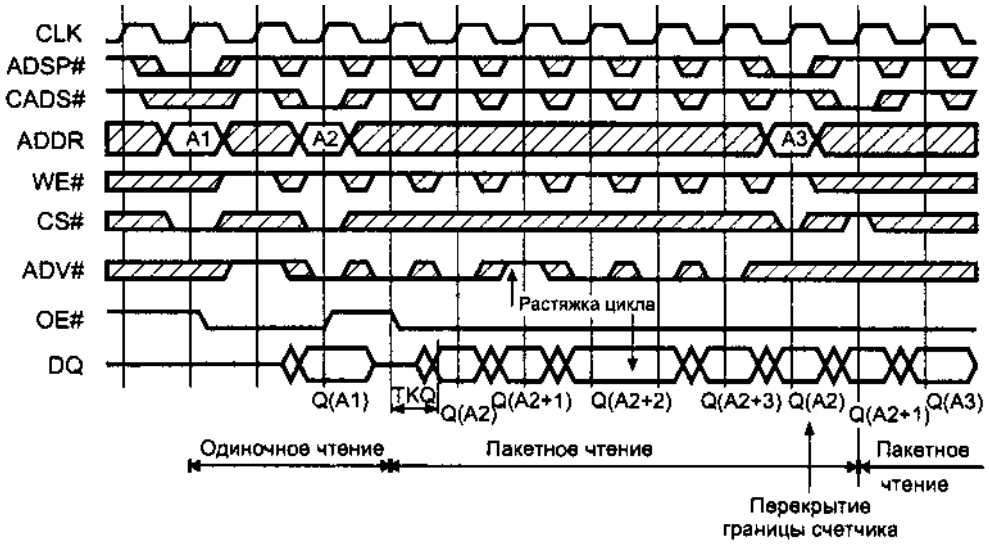


Рис. 8.25. Временные диаграммы чтения синхронной статической памяти

специфическому порядку (interleaved), принятому в процессорах i486 и выше. Микросхемы синхронной статической памяти, как и SDRAM, обычно имеют сигнал, выбирающий режим счета адреса: чередование (для процессоров Intel) или последовательный счет (для Power PC).

Синхронный интерфейс с таким набором сигналов позволяет памяти узнавать о намерениях процессора раньше и при задержке данных на выходе SRAM относительно синхронизирующего перепада T_{KQ} (Clock-to-Output Access Time) 8,5, 10 и 13,5 нс обеспечивать цикл *2-1-1-1* на частотах 66, 60 и 50 МГц соответственно. Однако на частотах 75 МГц и выше получается цикл *3-2-2*-2.

Конвейерно-пакетная статическая память (Pipelined Burst SRAM, PB SRAM) — усовершенствование синхронной памяти (слово «синхронная» из ее названия для краткости изъяли, но оно обязательно подразумевается). Конвейером является дополнительный внутренний регистр данных, который, требуя дополнительного такта в первой пересылке цикла, позволяет остальные данные получать без тактов ожидания даже на частотах выше 75 МГц. Задержка данных относительно синхронизирующего перепада у современных микросхем PB SRAM составляет 2,6-5 нс! Но, как и в случае памяти Sync Burst SRAM, этот параметр не является временем доступа в чистом виде (не следует забывать

о двух-трех тактах в первой передаче), а отражает появление действительных данных относительно очередного перепада сигнала синхронизации. Интерфейс PB SRAM аналогичен интерфейсу Sync Burst SRAM. Современные микросхемы способны работать на частоте до 250 МГц, первые данные пакета чтения появляются через один такт, то есть через 8-10 нс.

Развитием этой памяти стала статическая *память DDR SRAM*, работающая на частотах 275-375 МГц, а также *DDRII SRAM*. Эта память может работать как в режиме SDR (однократная синхронизация), так и в режиме DDR; в режиме DDR частота передачи данных удваивается относительно тактовой (то есть достигает 2 x 375 = 750 МГц). Плотность упаковки достигает 32 Мбит в микросхеме.

В *памяти QDR SRAM* четырехкратная скорость (относительно тактовой частоты) обеспечивается тем, что одновременно возможна передача данных чтения и записи (шина ввода отделена от шины вывода). По каждой шине передача идет на удвоенной частоте. Длина пакета ограничивается двумя передачами, адрес и команда (READ, WRITE, N0P) подаются в обоих полутактах синхронизации (рис. 8.26). Команды READ подаются одновременно с адресом по фронту синхросигнала, первые данные чтения появятся через один такт (4 нс при частоте 250 МГц). Команды WRITE и адрес подаются по спаду синхросигнала, к этому времени уже должна быть подана вторая часть данных пакета. Объем микросхемы составляет 18-72 Мбит, частоты — 166-300 МГц.

В *памяти QDRII SRAM* длина пакета увеличена до четырех передач, время цикла чтения — от 3,3 до 6 нс, частоты составляют 166-300 МГц. Временные диаграммы несколько иные: из-за удлиненного цикла все команды подаются по фронту синхросигнала.

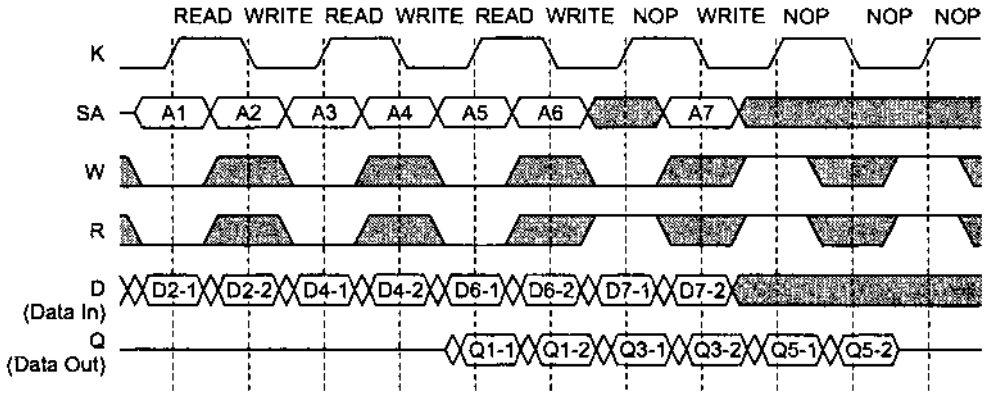


Рис. 8.26. Временные диаграммы работы памяти QDR SRAM

Применение статической памяти для кэширования ОЗУ

Самое распространенное применение статической памяти — кэширование ОЗУ. На микросхемах статической памяти обычно строится *внешний кэш*, в котором используется архитектура прямого отображения или наборно-ассоциативная архитектура (см. 7.3). Внешний кэш характерен для системных плат процессоров 386, 486, Pentium и совместимых с ними по интерфейсу. При этом функции кэш-контроллера выполняет чипсет. Процессоры P6 и выше имеют собственный вторичный кэш, расположенный либо на кристалле ядра, либо на картридже процессора. Кэш на картридже выполняется на почти таких же микросхемах статической памяти, какие устанавливаются на системной плате, но функции контроллера кэша уже ложатся на процессор, у которого для кэша имеется выделенная шина. Данный раздел в основном относится к кэш-памяти системных плат с сокетами 5 и 7 (и Super 7), предназначенных для процессоров Pentium и совместимых с ним по интерфейсу. У некоторых из них есть и собственный вторичный кэш, но для чипсета уже не важно, один или два уровня кэша имеет процессор. Микросхемы хранения данных кэша организуются в *банки*, число микросхем в банке должно соответствовать разрядности системной шины процессора. Банк должен заполняться микросхемами одного объема, требуемое быстродействие микросхем зависит от частоты системной шины. Банков может быть и несколько, количество заполненных банков и организация установленных микросхем, которые определяют объем кэш-памяти (V_{CACHE}), задаются джамперами или автоматически.

Для хранения тегов обычно используется отдельная микросхема асинхронной памяти SRAM — *Tag SRAM*, а для более чем 8-битного тега — пара микросхем. Здесь как для асинхронного, так и для синхронного кэша требуется асинхронная память. Ее объем может и превышать минимально необходимый для установленной кэш-памяти. Быстродействие определяется тактовой частотой системной шины. Необходимый *объем памяти тегов* (количество ячеек) можно вычислить, разделив объем установленной кэш-памяти на длину строки кэша,

определяемой чипсетом. Длина строки обычно равна количеству байтов, передаваемых за один стандартный пакетный цикл (4 x 4 = 16 байт для i486, 4 x 8 = 32 байта для Pentium).
Максимальный *объем кэшируемой памяти* (M_{CACHED}) ограничен как архитектурными особенностями чипсета и системной платы, так и объемом установленной кэш-памяти данных и разрядностью памяти тегов. Для обычных 8-бит- ных тегов он не может превышать 256 x V_{CACHE}. Так, для V_{CACHE} = 256 Кбайт M_{CACHE} = 64 Мбайт. Увеличение кэшируемого объема требует увеличения объема кэш-памяти или/и разрядности тегов (конечно, в рамках поддержки чипсетом).
Для кэша с обратной записью (WB) необходима еще и память для хранения признака «чистоты» строки. Признак может храниться в отдельной микросхеме *Dirty SRAM* или занимать один бит в *Tag SRAM*. Обратная запись во вторичном кэше применяется не всегда (она появилась несколько позже начала выпуска процессоров класса 486), ее реализация сложнее, чем сквозной.
Микросхемы асинхронной памяти обычно исполняются в DIP-корпусах с 8-битной организацией. *Микросхемы синхронной памяти* обычно имеют разрядность 16 или 32 бита (18 или 36 — с контролем четности).
Для системных плат с сокетами 5 и 7 были распространены модули *COAST* (Cache On A Stick — «кэш на палочке»). Это небольшой модуль с двусторонним печатным разъемом, устанавливаемый в специальный слот. Модуль содержит собственно кэш-память необходимой разрядности (асинхронную, синхронную пакетную или синхронную конвейерно-пакетную), на нем же может быть установлена и асинхронная память тегов. Модуль может использоваться и как расширение кэша, запаянного на системной плате.

Напряжение питания SRAM

Микросхемы SRAM, применяемые во вторичном кэше, своими адресными входами и двунаправленными линиями данных подключаются непосредственно к системной шине (host bus) — то есть к выводам процессора. Поскольку современные процессоры имеют различные номиналы питающего напряжения (но все гораздо ниже 5 В), возникает необходимость согласования уровней их сигналов с уровнями сигналов SRAM. Микросхемы SRAM исполняются *в* нескольких модификациях, различающихся напряжением питания и уровнями сигналов. По уровням выходных сигналов они все совместимы со стандартными микросхемами ТТЛ, для которых логический ноль — ниже 0,8 В, логическая единица — выше 2 В. В табл. 8.9 приведены электрические характеристики *стандартных* и *смешанных* (mixed-mode) *микросхем SRAM*, а также *микросхем SRAM* с напряжением питания *3,3 В*.

Таблица 8.9. Электрические характеристики микросхем SRAM

Характеристика	Стандартные	Смешанные	3,3 В
Питание (V_{CC})	+5 В	+5 В	+3,3 В
Напряжение на входах	0 ~ +5 В	0 ~ +5 В	0 ~ +3,3 В
Напряжение на выходах	0 ~ +5 В	0 ~ +3,3 В	0 ~ +3,3 В

При замене или добавлении микросхем (модулей) SRAM следует учитывать, следующее:

- на системной плате, рассчитанной на *стандартную микросхему SRAM*, последнюю можно без каких-либо изменений заменить *смешанной*, поскольку ее выходные уровни нормально распознаются ТТЛ-логикой (уровень логической единицы выше 2 В);
- на системной плате, рассчитанной на *смешанную микросхему SRAM*, ее *нельзя заменять* ни стандартной микросхемой (поскольку выходные уровни стандартной микросхемы SRAM недопустимы для устройств, рассчитанных на 3,3 В), ни микросхемой SRAM с напряжением питания 3,3 В (поскольку на входы микросхемы будут поданы сигналы ТТЛ 5 В, которые для нее недопустимы);
- на системной плате, рассчитанной на микросхему *SRAM* с напряжением питания *3,3 В,* ее *нельзя заменять* ни смешанной, ни стандартной, поскольку для этих микросхем питания 3,3 В недостаточно;
- у микросхем памяти тегов (*tag SRAM)* линии данных с системной шиной не соединяются, поэтому их модификация *может отличаться* от модификации самого кэша, но правила замены остаются теми же.

8.5. Энергонезависимая память

Обобщенно под энергонезависимой памятью (*NV Storage)* подразумевается любое устройство, хранящее записанные данные даже при отсутствии питающего напряжения (в отличие от статической и динамической полупроводниковой памяти). В данном разделе рассматриваются только электронные устройства энергонезависимой памяти, хотя к энергонезависимой памяти относятся и устройства с подвижным магнитным или оптическим носителем. Существует множество типов энергонезависимой памяти (ROM, PROM, EPROM, EEPROM, Flash Memory, FRAM), различающихся по своим потребительским свойствам, обусловленным способом построения запоминающих ячеек и сферами применения. Запись информации в энергонезависимую память, называемая *программированием*, обычно существенно сложнее и требует больших затрат времени и энергии, чем считывание. Программирование ячейки (или блока) — это целая процедура, в которую могут быть вовлечены специальные команды записи и верификации. Основным режимом работы такой памяти является считывание данных, а некоторые типы памяти после программирования допускают только считывание, что и обусловливает их общее название — «память только для чтения» (Read Only Memory, ROM), или ПЗУ (постоянное запоминающее устройство). Самые первые постоянные запоминающие устройства выполнялись на магнитных сердечниках, где информация заносилась их прошивкой проводниками считывания. С тех пор применительно к программированию ПЗУ укоренилось понятие «прошивка».

Запоминающие ячейки энергонезависимой памяти по своей природе обычно асимметричны и, как правило, позволяют записывать только нули в нужные

биты предварительно стертых (чистых) ячеек, содержащие единицы. Для некоторых типов памяти чистым считается нулевое состояние ячеек. Однократно программируемые микросхемы позволяют изменять только исходное (после изготовления) состояние ячеек. Для стирания (если оно возможно) требуются значительные затраты энергии (мощности и времени), и процедура стирания обычно существенно дольше записи. Стирание ячеек выполняется либо для всей микросхемы, либо для определенного блока, либо для одной ячейки (байта). Стирание приводит все биты стираемой области в одно состояние (обычно во все единицы, реже — во все нули).

Процедура программирования многих типов памяти требует относительно высокого напряжения программирования (12-26 В), а для однократно программируемых (прожигаемых) микросхем — и специального (не ТТЛ) интерфейса управления. После программирования нужна верификация — сравнение записанной информации с оригиналом, причем некачественное управление программированием (или брак микросхемы) может приводить к «зарастанию» записанной ячейки, что потребует повторного (возможно, и неудачного) ее программирования. Возможен и обратный вариант, когда «пробиваются» соседние ячейки, что требует повторного стирания (тоже, возможно, неудачного). Стирание и программирование микросхем может выполняться либо в специальном устройстве — программаторе, либо в самом целевом устройстве, если у него предусмотрены соответствующие средства. Микросхемы различают по возможности программирования:

- Микросхемы, программируемые при изготовлении, — масочные ПЗУ, содержимое которых определяется рисунком технологического шаблона. Такие микросхемы используют лишь при выпуске больших партий устройств с одной и той же прошивкой.
- Микросхемы, программируемые однократно после изготовления перед установкой в целевое устройство, — *ППЗУ* (программируемые ПЗУ), или *PROM* (Programmable ROM). Программирование осуществляется прожиганием определенных хранящих элементов на специальных устройствах-программаторах.
- Микросхемы, стираемые и программируемые многократно, — *РПЗУ* (непрограммируемые ПЗУ), или *EPROM* (Erasable PROM — стираемые ПЗУ). Для стирания и программирования требуется специальное оборудование. Микросхемы программируются в программаторе. Иногда возможно программировать микросхемы прямо в целевом устройстве, подключая внешний программатор, — так называемый метод *ОВР* (On-Board Programming). Наиболее распространены микросхемы *УФРПЗУ,* стираемые ультрафиолетовым облучением, — их обычно называют просто *EPROM,* или *UV-EPROM* (Ultra-Violet EPROM). В этом классе имеются и электрически стираемые ПЗУ (*ЭСПЗУ*), или *EEPROM* (Electrical Erasable PROM).
- Микросхемы, перепрограммируемые многократно в целевом устройстве с использованием программы его процессора, — по так называемому методу *ISW* (In-System Write). К этому классу относятся чисто электрически пере

программируемые микросхемы NVRAM и FRAM, но наибольшее распространение получила флэш-память.

Энергонезависимая память перечисленных типов в основном применяется для хранения неизменяемой (или редко изменяемой) информации — системного программного обеспечения (BIOS), таблиц (например, знакогенераторов графических адаптеров), памяти конфигурации устройств (ESCD, EEPROM адаптеров). Эта информация обычно является ключевой для функционирования PC (без BIOS компьютер представляет собой только коробку с дорогими комплектующими), поэтому весьма существенна забота о ее сохранности и предотвращении несанкционированного изменения. Нежелательное (ошибочное или под действием вируса) изменение содержимого становится возможным при использовании для хранения BIOS флэш-памяти, программируемой в целевом устройстве (на системной плате PC). Важными параметрами энергонезависимой памяти являются *время хранения* и *устойчивость к электромагнитным воз- действиям*, а для перепрограммируемой памяти еще и гарантированное *количество циклов перепрограммирования*.

Описываемая энергонезависимая память не является памятью с произвольным доступом, поскольку запись в нее выполняется не рядовым обращением по адресу, а процедурой программирования. Существует и *энергонезависимая память с произвольным доступом* (Non-Volatile Random Access Memory, NVRAM). Это название подразумевает возможность произвольной смены информации не только во всей области или блоке памяти, но и в отдельной ячейке, причем не процедурой, а обычным шинным циклом. К этому классу относят микросхемы FRAM и EEPROM, но у последних время записи обычно довольно большое. Флэш-память к этому классу относить нельзя, поскольку изменение информации, недаром называемое программированием, в этой памяти осуществляется, специальной программной процедурой.

Ферроэлектрическая память (Ferroelectric RAM, FRAM) — энергонезависимая память с истинно произвольным доступом, ее запись и чтение осуществляются как в обычных микросхемах статической памяти. При ее изготовлении используется железо — ее можно считать эхом старинной памяти на магнитных сердечниках больших машин. Ячейки FRAM по структуре напоминают DRAM, но информация хранится не в виде заряда конденсатора (который нужно поддерживать регенерацией), а в виде направления поляризации кристаллов. Запись производится непосредственно, предварительного стирания не требуется. Как и флэш-память, она используется в самых портативных системах класса PDA (Personal Digital Assistants — персональный электронный секретарь). Над этими устройствами активно работают фирма Hitachi совместно с Ramtron (www. ramtron.com) и фирма Matsushita с фирмой Symetrix. В 2001 году выпускались микросхемы емкостью 4-256 Кбит (технология 0,35 мкм) с параллельным интерфейсом (как SRAM) и временем доступа 70-120 нс, а также с последовательным интерфейсом I^2C. На 2005 были доступны микросхемы 1 Мбит (128K x 8) с временем доступа 60 нс (время цикла — от 150 нс), питание — 3,3 В (20 мА в активном режиме). Число циклов перезаписи неограниченно, время хранения — 10 лет. Помимо массивов памяти FRAM используется и в специ

альных энергонезависимых регистрах — есть, например, микросхемы FM573 и FM574, которые при включенном питании ведут себя аналогично стандартным 8-битным регистрам ‘573 и '574, но при выключении питания помнят свое состояние. У микросхем FRAM КМОП-интерфейс, питание 5 В, но имеются изделия и на 2,7 В. В отличие от флэш-памяти, у которой число циклов перезаписи принципиально ограничено (хотя и очень велико), ячейки FRAM практически не деградируют в процессе записи — гарантируется до 10^{10} циклов перезаписи. Провозглашается замена на FRAM даже динамической памяти, однако в PC память FRAM автору пока встречать не доводилось.
Энергонезависимая память в компьютерах применяется двояко:

- Включается в пространство памяти, и тогда процессор может использовать ее как для хранения (считывания) данных, так и для непосредственного исполнения программного кода. Таким способом энергонезависимая память, например, обеспечивает хранение информации BIOS (системной и модулей расширения).
- Используется в качестве носителя устройств хранения данных (карты USB Flash, CFA, SMC, MMC, SD и др.). В этом случае для доступа к ней требуются интерфейсные адаптеры и контроллеры, с помощью которых обеспечивается произвольный доступ к *блокам данных* устройства хранения.

Постоянная и полупостоянная память — ROM, PROM, EPROM

Масочные постоянные запоминающие устройства (ПЗУ, или ROM) имеют самое высокое быстродействие (время доступа 30-70 нс). Эти микросхемы в PC широкого применения не получили ввиду сложности модификации содержимого (только путем изготовления новых микросхем), они иногда применялись в качестве знакогенераторов в некоторых моделях графических адаптеров CGA, MDA, HGC.
Однократно программируемые постоянные запоминающие устройства (ППЗУ, или PROM) имеют аналогичные параметры и благодаря возможности программирования изготовителем оборудования (а не микросхем) находят более широкое применение для хранения кодов BIOS и в графических адаптерах. Программирование этих микросхем осуществляется только с помощью специальных программаторов, в целевых устройствах они устанавливаются в «кроватки» или запаиваются. Как и масочные, эти микросхемы практически не чувствительны к электромагнитным полям (в том числе к рентгеновскому облучению), и несанкционированное изменение их содержимого в устройстве исключено (конечно, не считая отказа).
Репрограммируемые постоянные запоминающие устройства (РПЗУ, или EPROM) до недавних пор были самыми распространенными носителями BIOS как на системных платах, так и в адаптерах, а также использовались в качестве знакогенераторов. Наиболее популярные микросхемы имеют 8-битную организацию и обозначение вида 27xx-tt или 27Cxx-tt для микросхем CMOS. Здесь xx определяет емкость в килобитах: 2708 — 1K x 8 — родоначальник семейства,

2716/32/64/128/256/512 имеют емкость 2/4/8/16/32/64 Кбайт соответственно, 27010 и 27020 — 128 и 256 Кбайт. Время доступа tt лежит в диапазоне 50—250 нс. 16-битные микросхемы (например, 27001 или 27002 емкостью 64К или 128К 16-битных слов) в PC применяются редко.

Микросхемы EPROM тоже программируются на программаторах, но относительно простой интерфейс записи позволяет их программировать и в устройстве (но не в штатном его режиме работы, а при подключении внешнего программатора). *Стирание* микросхем осуществляется ультрафиолетовым облучением в течение нескольких минут. Специально для стирания микросхемы имеют стеклянные окошки. После программирования эти окошки заклеивают, предотвращая стирание под действием солнечного или люминесцентного облучения. Время стирания зависит от расстояния до источника облучения, его мощности и объема микросхемы (более емкие микросхемы стираются быстрее). Вместо штатных стирающих устройств можно пользоваться и обычной медицинской ультрафиолетовой лампой с расстояния порядка 10 см. Для микросхем 2764 ориентировочное время стирания составляет 5 минут. Стирание переводит все биты в единичное состояние. «Недотертые» микросхемы при программировании могут давать ошибки, передержка при стирании снижает количество возможных циклов перепрограммирования (в пределе — до нуля).

Некоторые микросхемы, похожие по виду и обозначению на стираемые ультрафиолетом, не имеют окон (они упакованы в дешевый пластмассовый корпус). Эти микросхемы либо стираются рентгеновским облучением (что не совсем удобно), либо допускают лишь однократное программирование, которое может выполняться и по заказу фирмой-производителем микросхем. Их интерфейс полностью совпадает с интерфейсом обычных микросхем EPROM 27xx.

С программированием ПЗУ приходится сталкиваться при русификации графических адаптеров (CGA, MDA, HGC) и принтеров с незагружаемыми знакогенераторами, а также при замене (или восстановлении) системной микросхемы BIOS или микросхемы удаленной загрузки для адаптера локальной сети. Распространенные *программаторы EPROM* имеют интерфейс подключения к COM- или LPT-порту PC или подключаются через собственную карту расширения (обычно с шиной ISA). Время программирования зависит от типа и объема микросхемы и применяемого алгоритма программирования. Классический алгоритм с 50-миллисекундными импульсами записи каждой ячейки для современных микросхем практически не используется. Более быстрые «интеллигентные» алгоритмы позволяют записывать 8 Кбайт (микросхему 2764) менее чем за минуту. Вся процедура программирования может затягиваться в случае медленного интерфейса связи программатора с PC (например, COM-порт на скорости 2400 бит/с) за счет длительной процедуры копирования данных в буфер программатора.

Интерфейс микросхем постоянной памяти в режиме *чтения* совпадает с интерфейсом статической памяти. Для *программирования* (записи) требуется приложение ко входу Vpp напряжения программирования, которое для различных типов EPROM лежит в диапазоне 12-26 В (обычно указывается на корпусе микросхемы). Комбинации управляющих сигналов, формирующие импульсы

записи для EPROM разной емкости, различны. При напряжении на входе V_{PP} 5 В и ниже модификация памяти (запись) невозможна ни при каких комбинациях управляющих сигналов, и микросхемы работают строго в режиме ROM. Этот режим и используется для микросхем BIOS, так что никакой вирус им не страшен.
В PC чаще всего применяют микросхемы EPROM в корпусах DIP и PLCC (табл. 8.10).

Таблица 8.10. Популярные микросхемы EPROM

Микросхема и организация	Корпус	Примечание
2716 — 2K × 8	DIP-24	20 = OE#; 21 = V_{PP}
2732 — 4K × 8	DIP-24	20 = OE#/V_{PP}, 21 = A11
2764 — 8K × 8	DIP-28	1 = V_{PP}, 22 = OE#; 26 = NC, 27 = PGM#
27128 — 16K × 8	DIP-28	1 = V_{PP}, 22 = OE#; 26 = A13, 27 = PGM#
27256 — 32K × 8	DIP-28	1 = V_{PP}, 22 = OE#; 26 = A13, 27 = A14
27512 — 64K × 8	DIP-28	1 = A15, 22 = OE#/V_{PP}, 26 = A13, 27 = A14
27010 — 128K × 8	DIP-32	30 = NC
27010 — 128K × 8	TSOP-32	6 = NC
27010 — 128K × 8	PLCC-32	30 = NC
27020 — 256K × 8	DIP-32	–
27020 — 256K × 8	TSOP-32	–
27020 — 256K × 8	PLCC-32	–

Назначение выводов микросхем EPROM приведено в табл. 8.11.

Таблица 8.11. Назначение выводов микросхем EPROM

Сигнал	Назначение
CE#	Chip Enable — разрешение доступа. Низкий уровень разрешает обращение к микросхеме, высокий уровень переводит микросхему в режим пониженного потребления
OE#	Output Enable — разрешение выходных буферов. Низкий уровень при низком уровне CE# разрешает чтение данных из микросхемы. У некоторых типов микросхем на этот же вывод в режиме программирования подается напряжение V_{PP}
DQx	Data Input/Output — двунаправленные линии шины данных. Время доступа при чтении отсчитывается от установки действительного адреса или сигнала CE# (в зависимости от того, что происходит позднее)
Ax	Address — входные линии шины адреса. Линия A9 допускает подачу высокого (12 В) напряжения для чтения кода производителя (A0 = 0) и устройства (A0 = 1), при этом на остальные адресные линии подается логический ноль
PGM#	Programm — импульс программирования (некоторые микросхемы не имеют этого сигнала, их программирование осуществляется по сигналу CE# при высоком уровне V_{PP})
V_{PP}	Программирующее напряжение питания (для некоторых типов — импульс)
V_{CC}	Питание (+5 В)

Отметим *основные свойства EPROM*:

- Стирание информации происходит сразу для всей микросхемы под воздействием облучения и занимает несколько минут. Стертые ячейки имеют единичные значения всех битов.
- Запись может производиться в любую часть микросхемы побайтно, в пределах байта можно маскировать запись отдельных битов, устанавливая им единичные значения данных.
- Защита от записи осуществляется подачей низкого (5 В) напряжения на вход Vpp в рабочем режиме (только чтение).
- Защита от стирания производится заклейкой окна.

Флэш-память и EEPROM

Стирать микросхемы постоянной памяти можно электрическим способом. Однако этот процесс требует значительного расхода энергии, который выражается в необходимости приложения относительно высокого напряжения стирания (10-30 В) и длительности импульса стирания более десятка микросекунд. Интерфейс традиционных микросхем EEPROM имел временную диаграмму режима записи с большой длительностью импульса, что не позволяло непосредственно использовать сигнал записи системной шины. Кроме того, перед записью информации в ячейку обычно требовалось предварительное стирание, тоже довольно длительное. Микросхемы EEPROM относительно небольшого объема широко применяются в качестве энергонезависимой памяти конфигурирования различных адаптеров. Современные микросхемы EEPROM имеют довольно сложную внутреннюю структуру, в которую входит управляющий автомат. Это позволяет упростить внешний интерфейс, делая возможным непосредственное подключение к микропроцессорной шине или иному интерфейсу, и скрыть специфические (и не нужные пользователю) вспомогательные операции типа стирания и верификации. Микросхемы EEPROM позволяют считывать и перезаписывать (стирать) любую ячейку памяти, но перезапись требует довольно много времени.

Флэш-память по определению относится к классу EEPROM (электрическое стирание), но в ней используется особая технология построения запоминающих ячеек. Стирание во флэш-памяти производится сразу для целой области ячеек (блоками или полностью всей микросхемы). Это позволяет существенно повысить производительность в режиме записи (программирования). Во флэш-памяти сочетаются высокая плотность упаковки (ее ячейки на 30 % меньше ячеек DRAM), энергонезависимое хранение, электрические стирание и запись, низкое потребление, высокая надежность и небольшая стоимость. Первые микросхемы флэш-памяти были предложены фирмой Intel в 1988 году и с тех пор претерпели существенные изменения в архитектуре, интерфейсе и напряжении питания.

Каждая ячейка флэш-памяти состоит всего из одного униполярного (полевого) транзистора. Ячейки организованы в матрицу; разрядность данных внешнего интерфейса — 8 или 16 бит (ряд микросхем позволяет переключать разряд

ность). Чистые (стертые) ячейки содержат единицу во всех битах; при записи (программировании) нужные биты обнуляются. Возможно последующее программирование и уже записанных ячеек, но при этом можно только обнулять единичные биты и никак не наоборот. В единичное состояние ячейки переводятся только при стирании. Стирание выполняется для всей матрицы ячеек; стирание одиночной ячейки невозможно. Чтение флэш-памяти ничем не отличается от чтения любой другой памяти — подается адрес ячейки, и через некоторое время доступа (десятки — сотни наносекунд) на выходе появляются данные. Запись выглядит несколько сложнее — для программирования каждого байта (слова) приходится выполнять процедуру, состоящую из операций записи и считывания, адресованных к микросхеме флэш-памяти. Однако при этом шинные циклы обращения к микросхеме являются нормальными для процессора, а не растянутыми, как для EPROM и EEPROM. Таким образом в устройстве с флэш-памятью легко реализуется возможность перепрограммирования без извлечения микросхем из устройства. Большинство микросхем флэш-памяти имеет такой же интерфейс, как у асинхронной статической памяти (SRAM), а при чтении он упрощается до интерфейса ROM/PROM/EPROM. Существуют версии с интерфейсом динамической памяти, асинхронным и синхронным, а также и со специальными интерфейсами, в том числе и I^2C. Первые микросхемы работали только при напряжении питания 5 В, а для программирования и стирания требовали дополнительного питания V_{PP} = + 12 В. Затем появились микросхемы всего с одним питанием +5 В. Дальнейшее развитие технологии позволило снизить напряжение питания до 2,7-3,3 В и 1,65-2,2 В, а V_{PP} — до 5, 3,3, 2,7 и даже 1,65 В. Технологические процессы производства микросхем позволяют достичь разрешения 0,3, 0,22, 0,18 мкм (чем мельче ячейки, тем они экономичнее). Микросхемы первых выпусков (1990 г.) имели гарантированное число циклов стирания-программирования 10 000, современные — 100 000.

Флэш-память имеет время доступа при чтении 35-200 нс. Стирание информации (поблочное или во всей микросхеме) у микросхем середины 90-х годов занимает 1-2 с, программирование (запись) байта — порядка 10 мкс. У современных микросхем время стирания и записи заметно сократилось. Процедура записи от поколения к поколению упрощается (см. далее). Применяются различные методы программной и аппаратной защиты от ошибочного стирания (записи). Программной защитой является ключевая последовательность команд, нарушение которой не позволяет начать операции стирания и записи. Аппаратная защита не дает выполнять стирание и запись, если на определенные входы не поданы требуемые уровни напряжения. Аппаратная защита может защищать как весь массив целиком, так и отдельные блоки.

По организации массива в плане стирания групп ячеек различают следующие архитектуры:

- *Bulk Erase* (BE) — все ячейки памяти образуют единый массив. Запись возможна в произвольную ячейку. Стереть можно только сразу весь массив.
- *Boot Block* (BB) — массив разделен на несколько блоков разного размера, стираемых независимо, причем один из блоков имеет дополнительные средства защиты от стирания и записи.

- *Flash File* — массив разделен на несколько равноправных независимо стираемых блоков обычно одинакового размера, что позволяет их называть микросхемами с симметричной архитектурой (Symmetrical Architecture, SA).

Архитектура BE применялась только в микросхемах первого поколения, ее недостатки вполне очевидны (получается просто аналог EEPROM с более удобным способом стирания и интерфейсом программирования). Все современные микросхемы секторированы (разбиты на отдельно стираемые блоки), так что остается лишь деление на симметричную и несимметричную архитектуры.

В *симметричной архитектуре* (SA), как правило, память разбивается на блоки по 64 Кбайт; один из крайних блоков (с самым большим или самым маленьким адресом) может иметь дополнительные средства защиты.

В *асимметричной архитектуре* один из 64-килобайтных блоков разбивается на 8 блоков по 8 Кбайт. Один из блоков имеет дополнительные аппаратные средства защиты от модификации и предназначается для хранения жизненно важных данных, не изменяемых при запланированных модификациях остальных рбластей. Эти микросхемы специально предназначены для хранения системного программного обеспечения (BIOS), а привилегированный блок *(Boot Block)* — для хранения минимального загрузчика, позволяющего загрузить (например, с дискеты) и выполнить утилиту программирования основного блока флэш-памяти. В обозначении этих микросхем присутствует суффикс *T* (Top) или *B* (Bottom), определяющий положение загрузочного блока либо в старших, либо в младших адресах соответственно. Первые предназначены для процессоров, стартующих со старших адресов (в том числе x86, Pentium), вторые — для стартующих с нулевого адреса, хотя возможны и противоположные варианты, когда некоторые биты шины адреса перед подачей на микросхему памяти инвертируются. Старые микросхемы BB малого объема имели немного другое распределение. Например, микросхема *28F001BX-T (28F001BN-T*), часто применяемая для флэш-BIOS в PC, содержит:

- основной блок (main block) объемом 112 Кбайт (00000h-1BFFFh);
- два блока параметров (parameter block) объемом по 4 Кбайт (1C000h- 1CFFFh и 1D000h-1DFFFh);
- загрузочный блок (boot block) объемом 8 Кбайт (1E000h-1FFFFh), стирание и программирование которого возможны лишь при особых условиях.

Основной блок и блоки параметров по защите равноправны; выделение небольших блоков параметров позволяет хранить в них часто сменяемую информацию, например ESCD системы Plug and Play.

Выпускают и комбинированные микросхемы: например, MT28C3214P2FL представляет собой комбинацию флэш-памяти 2M x 16 и SRAM 256K x 16.

По организации матрицы ячеек различают архитектуры NOR и NAND. В традиционной *архитектуре NOR* транзисторы на одном проводе объединяются своими стоками параллельно, как бы образуя логический элемент ИЛИ-НЕ (Not OR, NOR). Эта архитектура обеспечивает высокое быстродействие произвольного считывания, что позволяет исполнять программы прямо из флэш-памяти (не копируя в ОЗУ) без потери производительности. Для современной

памяти NOR характерны объем 16-256 Мбит (технология 120 или 90 нм) и время доступа 70-90 нс. Интерфейс аналогичен SRAM, причем как асинхронной, так и синхронной (пакетной) памяти.

В *архитектуре NAND* несколько транзисторов разных ячеек соединяются последовательно, образуя логический элемент И-НЕ (Not AND, NAND), что дает высокую скорость последовательных обращений (но не произвольных). Для памяти NAND характерна высокая плотность упаковки: от 64 Мбит до 64 Гбит (2 Гбайт) в одном корпусе, при этом время цикла последовательного чтения составляет порядка 50 нс. Время программирования страницы в 2112 байт (2048 байт данных + 64 служебных байта) — 200 мкс, время стирания 128-кило- байтного блока — 2 мс. Такую память применяют в твердотельных устройствах хранения.

Появились и комбинированные типы памяти, например OneNAND (Samsung), в которых ядро флэш-памяти NAND объединено с буфером SRAM. В результате получается память, сочетающая высокую скорость с возможностью произвольного доступа. В этой памяти еще и встроен ECC-контроль, позволяющий исправлять однократные и обнаруживать двукратные ошибки.

В первых микросхемах флэш-памяти каждая ячейка (всего один транзистор) предназначалась для хранения одного бита информации (1 — стерта, 0 — «прошита»). Позже появилась технология хранения двух битов в одной ячейке — благодаря совершенствованию технологии удалось надежно различать 4 состояния ячейки, что и требуется для хранения двух битов. Два бита в ячейке хранит память Intel StrataFlash, емкость одной такой микросхемы уже достигла 256 Мбайт.

Флэш-память постоянно развивается в плане как повышения емкости и снижения потребления, так и расширения возможностей и повышения производительности. Так, например, в ряде микросхем AMD имеется возможность чтения одновременно с записью других блоков (чтение во время стирания стало возможным еще со второго поколения флэш-памяти).

Некоторые микросхемы обеспечивают быстрый обмен в *страничном режиме* (page mode). Страницей являются 4 или 8 смежных ячеек; первое чтение в странице выполняется с временем доступа 70 нс. Если микросхема остается выбранной, то другие ячейки этой страницы (отличающиеся значением младших битов адреса) можно считывать циклами длительностью по 10-20 нс. Доступ к одиночным ячейкам не отличается от обычного. Микросхемы с *пакетным режимом* (burst mode) вдобавок к одиночному и страничному асинхронным режимам могут работать и в синхронном режиме (интерфейс PB Sync SRAM, см. выше). Для этого они имеют вход синхронизации CLK. Адрес начала пакета передается вместе с сигналом ADV# (фиксируются по положительному перепаду CLK). Первые данные на выходе появятся через 2-3 такта, после чего в каждом следующем такте выдаются очередные данные (тактовая частота до 66 МГц).

Выпускалась и *синхронная флэш-память* с интерфейсом SDRAM (и даже с совпадающей упаковкой в корпуса), например, MT28S4M16LC (2001 г., частота до 66 МГц) — 1M x 16 x 4 банка. Такая память удобна для хранения ПО, испол

няемого прямо на месте (без копирования в ОЗУ) во встраиваемых компьютерах. В 2005 году информацию о новых микросхемах такой памяти автору в Сети найти не удалось.

Микросхемы флэш-памяти с симметричной архитектурой выпускаются и с *интерфейсом DRAM* (динамической памяти) — с мультиплексированной шиной адреса, стробируемой сигналами RAS# и CAS#. Они предназначены для применения в модулях SIMM или DIMM, устанавливаемых в гнезда микросхем обычной динамической памяти. Таким образом реализуются, например, модули PostScript для лазерных принтеров и любые резидентные программные модули. Эти модули, естественно, не будут определяться системой как основная память — на попытку обычной записи и считывания, предпринимаемую в тесте POST при определении установленной памяти, они ответят весьма своеобразно. Также они не будут восприниматься и как модули дополнительной системы BIOS, поскольку займут не подходящие для этого физические адреса. Использовать эти модули можно только с помощью специального драйвера, который «объяснит» чипсету, какому диапазону адресов пространства памяти соответствуют сигналы выборки банков флэш-памяти. Поскольку интерфейс модулей SIMM и DIMM не предполагает сигналов защиты записи, системного сброса и дополнительного питания +12 В, все вопросы, связанные с программированием и защитой, решаются дополнительными элементами, устанавливаемыми на модулях. При использовании 16-битных микросхем такие модули непосредственно не обеспечивают независимую побайтную запись, но она может обеспечиваться программно, маскированием (записью 0FFh) немодифицируемых байтов.

Для хранения BIOS появились микросхемы *флэш-памяти с интерфейсом LPC*, называемые хабами (firmware hub).

Для некоторых сфер применения требуются специальные меры по блокированию изменения информации пользователем. Так, Intel в некоторые микросхемы вводит однократно записываемые (One-Time-Programmable, OTP) регистры. Один 64-битный регистр содержит уникальный заводской номер, другой может программироваться пользователем (изготовителем устройства) только однажды.

Фирма Intel выпускает микросхемы «Wireless Flash Memory» — за интригующим названием (беспроводная флэш-память) скрывается, конечно же, обычный электрический интерфейс (с проводами). Они ориентированы на применение в средствах беспроводной связи (сотовые телефоны с доступом в Интернет): питание — 1,85 В, наличие регистров OTP для защиты от мошенничества и т. п.

Корпуса, интерфейс и обозначение микросхем флэш-памяти

Микросхемы флэш-памяти упаковывают в корпуса со стандартизованным назначением выводов. Первые микросхемы выпускались в корпусах DIP, что обеспечивало легкость замены микросхем (E)EPROM флэш-памятью. Затем в целях миниатюризации перешли к корпусам PLCC, TSOP и TSOP-II. Применение

корпусов FBGA (Fine Pitch ball Grid Array) — матрицы из 6 x 8 шариковых выводов с шагом 0,8 мм — позволяет уменьшить размер корпуса до минимума, требуемого для упаковки кристалла. Для микросхем, используемых в картах SmartMedia, применяют и оригинальную упаковку KGD (Known Good Die).

Интерфейс микросхем флэш-памяти хорошо сочетается со стандартными сигналами, используемыми в микропроцессорных системах. *Внутренние циклы* стирания, записи и верификации выполняются автономно от *шинных циклов* внешнего интерфейса, что является существенным преимуществом перед микросхемами EPROM и EEPROM. В режиме чтения они полностью совместимы с EPROM, совпадая с ними и по расположению основных выводов.

Обозначение микросхем для изделий лидеров в области разработки и производства флэш-памяти — фирм Intel и AMD — несколько различаются. Остальные производители для своих аналогичных по свойствам изделий в основном придерживаются системы обозначений лидеров.

Обозначение микросхем флэш-памяти Intel начинается с признака 28F, за которым следует трехзначный код объема (табл. 8.12), а за ними — два символа технологии и архитектуры:

- B5, BC, BX, BR — Boot Block с питанием 5 В;
- C3 — Boot Block с питанием 3 В;
- F3 — Boot Block с питанием 3 В, повышенное быстродействие;
- J3 и J5 — StrataFlash (SA) с питанием 3 и 5 В соответственно;
- S3 и S5 — Flash File (SA) с питанием 3 и 5 В соответственно.

Для флэш-памяти AMD первая часть обозначения определяет тип и характеристики микросхем:

- Am29BDS — 1,8 В, считывание одновременно с записью, пакетный режим чтения;
- Am29DS — 1,8 В, считывание одновременно с записью;
- Am29SL - 1,8 В;
- Am29LV — 3 В;
- Am29DL — 3 В, считывание одновременно с записью;
- Am29BL — 3 В, пакетный режим чтения;
- Am29PL — 3 В, страничный режим чтения;
- Am30LV - 3 В, UltraNAND;
- Am29F — 5 В.

Далее следует трехзначный код объема, за ним — символ технологии изготовления (B, C или D), за которым идет символ архитектуры:

- T — boot sector, верхний;
- B — boot sector, нижний;
- H — симметричная архитектура, защищен блок со старшим адресом;

- L — симметричная архитектура, защищен блок с младшим адресом;
- U (нет символа) — симметричная архитектура;
- J40 — число 100-процентно годных блоков (только для UltraNAND).

Оставшаяся часть определяет параметры питания, быстродействие, тип корпуса, температурный диапазон и некоторые особенности.

Таблица 8.12. Объем и организация популярных микросхем флэш-памяти

Обозначение	Организация[1]
256	32K × 8 BE
512	64K × 8 BE
010	128K × 8 BE
020	256K × 8 BE
001	128K × 8 BB
002	256K × 8 BB
004	512K × 8 BB, SA
008	1M × 8 BB, SA
016	2M × 8 BB, SA
200	256K × 8 / 128K × 16 BB
400	512K × 8 / 256K × 16 BB
800	1024K × 8 / 512K × 16 BB
160	2M × 8 / 1M × 16 SA, BB
320	4M × 8 / 2M × 16 SA
640	8M × 8 / 4M × 16 SA

[1] BE — Bulk Erase (стираемые целиком), BB — Boot Block (несимметричные блоки), SA — Symmetric Architecture (симметричные блоки). Через косую черту указана архитектура для микросхем с переключаемой разрядностью данных.

Флэш-память с интерфейсом PCMCIA (PC Card) оптимизирована для построения внешней памяти миниатюрных ПК. Модуль флэш-памяти в формате PC Card имеет интерфейс дисков IDE (ATA) как на уровне электрических сигналов, так и по системе команд. Помимо собственно микросхем накопителя этот модуль обычно содержит управляющую микросхему программируемой логики. Флэш-память в стандарте PC Card логически является устройством *внешней* памяти. Ее не следует путать с похожей по виду памятью в формате Credit Card, которая является оперативной и вставляется в специальный (не PCMCIA) слот компьютера. Внешнюю память, в отличие от оперативной, в принципе, можно вставлять и вынимать без перезагрузки ОС.

Организация и программирование флэш-памяти Intel

По организации и программированию можно выделить 3 поколения флэш-памяти Intel.

Микросхемы *первого поколения* (*28F256, 28F512, 28F010, 28F020)* представляют собой единый массив памяти, стираемый целиком (*Bulk Erase).* Для стирания и записи микросхемы имеют внутренний регистр команд и управляющий авто-

мат *WSM* (Write State Machine). Стирание и программирование флэш-памяти возможны только при подаче на вход V_{PP} напряжения 12 В по командам, записываемым во внутренний регистр в шинном цикле записи по сигналу WE#.

Выполнение команд инициируется записью их кодов во внутренний регистр, не имеющий конкретного адреса.

По включении питания внутренний регистр команд обнуляется, что соответствует команде чтения, и микросхема работает, как обычная микросхема PROM или EPROM. Это позволяет устанавливать микросхемы флэш-памяти вместо EPROM аналогичной емкости. При подаче на вход V_{PP} низкого напряжения (0-6,5 В) стирание и программирование невозможны, и микросхема ведет себя, как обычная микросхема EPROM.

Микросхемы *второго поколения* секторированы — ячейки группируются в блоки, допускающие независимое стирание (асимметричное разбиение — *Boot Block*, симметричное — *Flash File*). Длительная операция стирания одного блока может прерываться для считывания данных других блоков, что значительно повышает гибкость и производительность устройства. Микросхемы имеют более сложный внутренний управляющий автомат и регистр состояния, что позволяет разгрузить внешний процессор и программу от забот по отслеживанию длительности операции программирования и стирания, а также упростить эти процедуры.

Внутренние операции стирания и программирования выполняются после посылки соответствующих кодов во внутренний *регистр команд*. Отработка операций внутренним управляющим автоматом отображается соответствующими битами *регистра состояния* (Status Register, SR), по значению которых внешняя программа может получить информацию о результате выполнения и возможности посылки следующих команд.

Программирование и стирание загрузочного блока, в отличие от операций с другими блоками, требует подачи высокого потенциала V_{HH} (не ТТЛ, а +12 В) на вход PWD# перед выдачей команды стирания или программирования и удержания его до успешного завершения операции. Альтернативный способ — подача такого же потенциала, но на вход 0E# на время пар шинных циклов записи команд стирания или программирования.

Микросхемы *Flash File* организованы в виде набора одинаковых блоков, равноправных по защите (SA). Защита от модификации для *28F008SA* может осуществляться только для всей микросхемы подачей низкого напряжения на вход V_{PP}. По интерфейсу и командам микросхемы совпадают с микросхемами Boot Block (исключая специфику загрузочного блока).

Архитектура микросхем *28F016SA* существенно изменена, что значительно повышает производительность программирования (до 28,6 Мбайт/с в пакетном режиме) и обеспечивает поблочную защиту. Для записи микросхема имеет два буфера данных по 256 Кбайт. Введены новые команды, обеспечивающие расширение функций. Далее перечислены дополнительные возможности микросхем:

- Буферированное страничное программирование. Помимо обычного побайтного или двухбайтного программирования возможно быстрое заполнение буфера шинными циклами записи. Далее переписывание его содержимого (всего или фрагмента) во флэш-память выполняется одной командой. Содержимое буфера может быть считано после подачи соответствующей команды.
- Поддержка очереди команд.
- Автоматическая запись из буфера во флэш-массив во время стирания другого блока.
- Программная защита позволяет для любого блока установить бит защиты в специальную энергонезависимую область. Запись и стирание защищенного блока могут осуществляться только после снятия общей защиты записи по сигналу WP#. Сброс бита защиты блока осуществляется только при успешном стирании или перезаписи защищенного блока.
- Стирание всех незащищенных блоков может выполняться одной командой.
- Перевод микросхемы в режим ожидания (Sleep) с пониженным потреблением. В этом режиме возможно считывание информации состояния и получение команд.

Микросхема *28F032SA* представляет собой два параллельно соединенных кристалла 28F016SA в одном корпусе. Входы CE# одного из них соединены с выводами CE0# и CE1#, второго — с CE0# и CE2#.

Третье поколение — современные микросхемы, выполненные по технологии *SmartVoltage*, допускают стирание и программирование при напряжении V_{PP} как 12 В, так и 5 В. В последнем случае эти операции занимают больше времени. Кроме того, операции чтения возможны при пониженном (3,3 и даже 2,7 В) напряжении питания V_{CC}, при этом снижается потребление, но увеличивается время доступа.

Флэш-память фирмы AMD

Фирмой AMD выпускается несколько семейств микросхем флэш-памяти. Первые из них *(Am28F256/512/010/020)* были близки по характеристикам к флэш-памяти Intel первого поколения (Bulk Erase, стирание и программирование 12 В). В отличие от аналогичных микросхем Intel, микросхемы Am28F256/512 не имели стоп-таймера, что требовало точной выдержки при программирования и стирании. Следующими были микросхемы Am*28F256A/ 512A/010A/020A* со встроенным алгоритмом программирования, отличающимся от алгоритма микросхем Intel второго поколения как последовательностью команд, так и способом определения окончания операций. Состояние операций стирания или программирования определяется по результату данных, полученных в шинном цикле чтения по адресу ячейки, участвующей в операции. При этом для определения факта окончания операций может использоваться метод Data# Polling или Toggle Bit. Метод Data# Polling основан на анализе бита DQ7 считанных данных. В начале выполнения внутреннего цикла он устанавливается инверсным по отношению к тому, что должно быть записано в ячейку. По ус-

пешном окончании операции он принимает желаемое значение (при стирании — 1). Метод Toggle Bit основан на анализе бита DQ6, который при каждом шинном цикле считывания во время выполнения операции меняет свое значение на противоположное. По окончании операции он остается в каком-либо состоянии, при этом об успешности можно судить по биту 7. Единичное значение бита DQ5 — Exceeded Timing Limits — указывает на превышение допустимого времени выполнения операции.

Микросхемы семейства *Am29Fxxx* выполняют все операции при одном питающем напряжении 5 В и имеют секторированную структуру (Sector Erase), симметричную (аналогично Flash File) или несимметричную (Boot Block), с верхним (T) и нижним (B) положением загрузочного блока. С помощью программатора каждый сектор может быть защищен от модификации в целевой системе (в отличие от Intel, способ установки и снятия защиты фирмой AMD широко не раскрывается). По расположению выводов и интерфейсу микросхемы соответствуют стандарту JEDEC для флэш-памяти с одним питающим напряжением. Для защиты от случайного выполнения цепочки команд состоят из 3-6 шинных циклов, причем для них существен и адрес. Микросхемы позволяют выполнять одновременное стирание группы секторов. Все эти микросхемы, кроме Am29F010, обеспечивают приостанов стирания сектора (Erase Suspend) для чтения других секторов, а микросхемы Am29F080/016 позволяют еще и программировать байты во время приостанова стирания.

Следующей появилась секторированная флэш-память *Am29LVxxx с* одним питающим напряжением (3,0 В) для всех операций. У этих микросхем защита любого сектора также устанавливается с помощью программатора стандартной памяти EPROM, и возможно временное снятие защиты в целевой системе. Помимо программной (биты состояния) индикации окончания операции имеется и аппаратная (сигнал RY/BY#). Также присутствует сигнал аппаратного сброса, переводящий микросхему в режим чтения.

Вышеперечисленные микросхемы имеют традиционную архитектуру NOR. От них значительно отличается микросхема Am30LV0064D — 64 Мбит (8M x 8) с архитектурой UltraNAND, обеспечивающей быстрый последовательный доступ к данным выбранной страницы. Каждая страница имеет 512 байт данных и 16 дополнительных байтов, используемых, например, для хранения ECC-ко- да. Для выбора страницы при чтении (загрузки во внутренний 528-байтный регистр) требуется около 7 мкс, после чего данные считываются последовательно со скоростью до 20 Мбайт/с (50 нс/байт). Таким образом, среднее время на чтение одного байта составляет всего 65 нс. Для записи данные (страница полностью или частично) загружаются в регистр с той же скоростью, после чего запись их в массив хранящих ячеек требует всего 200 мкс. Таким образом, среднее время на запись одного байта составляет всего 430 нс — в 20 раз быстрее обычной (NOR) флэш-памяти (скорость записи — 2,3 Мбайт/с). Стирание выполняется блоками по 8 Кбайт за 2 мс (в обычной — 600 мс). Микросхема питается от 3 В. Планируется достижение объема микросхемы до 1 Гбит. Надежность хранения — 10 лет, 10^4 циклов безошибочного программирования, более 10^6 циклов

программирования с коррекцией ошибок. Применение — «твердые диски», цифровые камеры, диктофоны и т. п.

Флэш-память других фирм

Микросхемы флэш-памяти выпускаются многими фирмами. Они различаются по организации, интерфейсу, напряжению питания и программирования, методам защиты и другим параметрам. Лидеры в области разработки и производства флэш-памяти — фирмы AMD, Fujitsu, Intel и Sharp летом 1996 года приняли спецификацию *CFI* (Common Flash Interface), обеспечивающую совместимость разрабатываемого программного обеспечения с существующими и новыми моделями флэш-памяти. Для микросхем с дополнительными архитектурными расширениями используется расширенный (масштабируемый) набор команд (Scalable Command Set, SCS).

Для большинства изделий справедливы те же тенденции, что и для рассмотренных микросхем Intel и AMD, а именно — повышение объема, снижение напряжений питания и потребляемой мощности, повышение производительности и упрощение внешнего интерфейса для операций стирания и программирования.

Микросхемы с буферированным программированием или страничной записью могут не иметь в своей системе команд отдельной операции стирания сектора. Внутренняя операция стирания (и предварительного обнуления сектора) выполняется при страничном программировании.

Для защиты от случайного выполнения ключевые последовательности команд требуют от 2 до 6 шинных циклов, причем у них может быть важен и адрес (как в микросхемах AMD). Методы защиты секторов имеют различную программную и аппаратную реализацию. Для временного снятия защиты используют разные способы, одним из которых является ключевая последовательность семи шинных циклов чтения.

Микросхемы флэш-памяти Micron совместимы с Intel и обозначаются аналогично, но начинаются с признака MT28F. Среди них есть и особенные, например: MT28F321P2FG - 2M x 16 Page Flash Memory, MT28F322D18FH - 2M x 16 Burst Flash Memory.

Фирма Silicon Storage Technology выпускает разнообразные микросхемы флэш-памяти с одним напряжением питания для всех операций. Их свойства можно определить по обозначению вида SST xx YY zzz — ttt, где xx — семейство:

- *28* — побайтное программирование, посекторное стирание;
- *29* — страничное программирование с прозрачным стиранием (команда стирания сектора отсутствует, внутренняя операция выполняется автоматически перед записью страницы в массив).

Символы YY обозначают функциональный тип и напряжение питания:

- *EE* — EEPROM-совместимые, выполнение одной инструкции, V_{cc} = 5 В;
- *LE* — то же, что и EE, V_{cc} = 3 В;
- *VE* — то же, что и EE, V_{cc} = 2,7 В;
- *SF* — операции Super Flash Command Register, V_{cc} = 5 В;

- *LF* — то же, что и SF, V_{cc} = 3 В;
- *VF* — то же, что и SF, V_{CC} = 2,7 В;
- *DM* — Disk Media (для флэш-дисков, требует внешнего контроллера), V_{cc} = 5 В;
- *LM* — то же, что и DM, V_{cc} = 3 В;
- *VM* — то же, что и DM, V_{cc} = 2,7 В;
- *PC* — PCMCIA (интерфейс и протоколы), VCC = 5 В.

Символы zzz обозначают объем микросхемы:

- *512* - 512 Кбит (64К x 8);
- *010* - 1 Мбит (128К x 8);
- *040* - 4 Мбит (512К x 8);
- *080* — 8 Мбит (1М x 8);
- *016* - 16 Мбит (2М x 8);
- *032* — 32 Мбит (4М x 8).

Символы ttt обозначают время доступа при чтении.

Микросхемы *SST 29EE010, 29LE010* и *29VE010,* часто применяемые в качестве носителей флэш-BIOS, организованы как 1024 страницы по 128 байт с программной и аппаратной защитой. Каждая страница может быть защищена независимо от других. Временные диаграммы стирания и программирования, а также необходимое напряжение программирования генерируются внутри микросхемы. Окончание операции определяется по алгоритму Toggle Bit или Data# Polling.

Аналогичные параметры имеют мйкросхемы *29EE011, 29LE011, 29VE011* фирмы Winbond.

Энергонезависимая память с последовательными интерфейсами

Для микросхем энергонезависимой памяти малого объема, от которых не требуется высокой производительности обмена данными, часто применяют последовательный интерфейс. Двухпроводной (считаются только сигнальные линии, общий провод подразумевается) интерфейс I^2C обеспечивает низкую скорость передачи (частота до 100, 400 КГц и 1 МГц у самых быстрых устройств). Трех- проводной интерфейс SPI (Serial Peripheral Interface) с линией синхронизации SCK и раздельными линиями входных (SI) и выходных (SO) данных обеспечивает более высокую скорость, чем I^2C: SPI Mode 0 — частота 2,1 МГц, Mode 3 —

4 МГц.

Применение последовательного интерфейса позволяет упаковывать микросхемы памяти любого объема в корпус, имеющий всего 8 выводов (табл. 8.13). С таким интерфейсом выпускаются микросхемы EEPROM, FRAM и флэш-па- мяти. Микросхемы EEPROM и флэш-памяти выполняют внутренние операции записи автономно, о завершении операции можно судить по результатам

опроса ее состояния. Более сложные микросхемы имеют блочную организацию и средства управления доступом к каждому блоку с помощью программируемых регистров состояния и внешнего вывода управления записью (программированием). Микросхемы FRAM выполняют все операции на скорости интерфейса (на то они и RAM). Существуют модификации микросхем, позволяющие блокировать запись данных пользователем в определенную область (или всю микросхему, что превращает ее в ROM). Вывод управления защитой у разных типов микросхем функционирует и называется по-разному WP# — Write Protect, WC — Write Control, PP — Programm Protect. Для выбора микросхемы используются либо входы задания внутреннего адреса A[0:2], либо сигнал выборки CS#, с помощью которого контроллер может обратиться к одному из требуемых устройств. Для упрощения внешних схем могут использоваться и несколько сигналов выборки S[0:2], один из которых (S1) иногда инвертирован.

Таблица 8.13. Популярные микросхемы памяти с последовательным интерфейсом

Микросхема	Организация	Интерфейс
24C001, 24C01	16 × 8, 128 × 8	I^2C
24C02, 24C164	256 × 8, 2K × 8	I^2C
24F016	2K × 8	I^2C
24F128	16K × 8	I^2C
X76F041	512 × 8	I^2C
FM24C04, FM24C16, FM24C64, FM24C256	512 × 8, 2K × 8, 8K × 8, 32K × 8	I^2C
FM25040, FM25160, FM25256	512 × 8, 2K × 8, 64K × 8	SPI

Помимо обычных устройств энергонезависимой памяти с интерфейсом I^2C выпускают и специализированные *устройства защиты* (security devices). Например, микросхема X76F041, представляющая собой 4 блока памяти по 128 байт, имеет 64-битный регистр пароля, доступный только по записи. Обращение к микросхеме возможно только при предъявлении правильного пароля (который рассчитать невозможно в принципе). Программируемый управляющий регистр (тоже энергонезависимый) позволяет для каждого блока установить свой режим доступа (полный доступ, только чтение, возможность только обнуления битов при записи, доступ только по предъявлении пароля конфигурации). Кроме того, есть возможность включения режима саморазрушения после превышения заданного количества попыток доступа с неверным паролем. Такие устройства могут применяться в аппаратных ключах, защищающих программные средства от несанкционированного исполнения и пиратского копирования.

Микросхемы EEPROM 24C02 с интерфейсом I^2C объемом 256 байт применяются для последовательной идентификации модулей DIMM-168 второго поколения и более новых.

Часть III

Периферийные устройства

ГЛАВА 9

Устройства хранения данных

Устройства хранения данных относятся к внешней памяти компьютера — они позволяют сохранять информацию для последующего ее использования независимо от состояния (включен или выключен) компьютера. В этих устройствах могут быть реализованы различные физические принципы хранения информации — магнитный, оптический, электронный — в любых их сочетаниях. Внешняя память принципиально отличается от внутренней (оперативной, постоянной и специальной) способом доступа к этой памяти процессора (исполняемой программы). Устройства внешней памяти оперируют *блоками* информации, но никак не байтами или словами, как, например, оперативная память. Процедуры обмена с устройствами внешней памяти привязаны к типу устройства, его контроллеру и способу подключения устройства к системе (интерфейсу).

В этой главе рассматриваются устройства хранения различных типов — с подвижными носителями, дисковыми и ленточными, магнитными и оптическими, а также статические (электронные). Основной акцент сделан на дисковых устройствах — основных устройствах хранения данных в современных компьютерах. Помимо собственно устройств рассматриваются их интерфейсы, а также организация систем и сетей хранения данных.

9.1. Принцип действия и назначение устройств хранения

Устройства хранения, относящиеся к внешней памяти компьютера, обеспечивают энергонезависимое хранение блоков информации на каком-либо физическом носителе. Физические принципы энергонезависимого хранения и соответствующие им носители разнообразны. Наибольшее распространение получили следующие:

- Магнитный принцип основан на перемагничивании участков носителя в соответствии со значениями битов записываемой информации. Этот принцип

реализуется в *устройствах с подвижным носителем* в виде диска или ленты, где запись и считывание производится на дорожку (трек). Головка записи вызывает изменение намагниченности участков трека в соответствии с записываемой битовой последовательностью. При считывании регистрируется изменение магнитного поля, связанное с прохождением под головкой участков трека, и из этих изменений извлекается ранее записанная информация. Существуют магнитные устройства хранения и с неподвижным носителем. В «древней» истории компьютеров применялись матрицы (кубы) памяти на магнитных сердечниках. В настоящее время используются (но пока еще не широко) микросхемы памяти FRAM (Ferroelectric Random Access Memory — ферроэлектрическая оперативная память). В магнитооптических устройствах принцип хранения — магнитный, оптика (лазер) используется лишь для разогрева перемагничиваемого участка при записи (это позволяет значительно уменьшить размер участка — повысить плотность записи) и считывании (свойства отраженного луча зависят от состояния магнитной «ячейки»).

- Оптический принцип основан на изменении оптических свойств участка носителя: степени прозрачности или коэффициента отражения. Способы, какими эти изменения достигаются, различны. В первых оптических устройствах использовался механический способ записи — пробивали отверстия в перфолентах и перфокартах. В современных оптических устройствах на дисках CD и DVD изменение оптических свойств достигается с помощью лазера, выжигающего лунки (необратимо, однократно) или изменяющего состояние участка (обратимо, многократно). Выпуск массового тиража оптических носителей с информацией возможен и с помощью различных технологий печати.
- Электрический принцип основан на пороговых эффектах в полупроводниковых структурах. Этот принцип используется в *твердотельной памяти* — флэш-памяти и EEPROM. Здесь для изменения состояния хранящей ячейки требуется значительная энергия (довольно длительное воздействие сильного электрического поля), что и происходит в процессе записи, называемом программированием. Считывание требует значительно меньших затрат как энергии, так и времени. Под твердотельностью в этих устройствах подразумевается отсутствие относительного движения носителя и головок записи- считывания.

Устройство хранения тем или иным способом подключается к *хосту* — компьютеру, в котором, как минимум, присутствуют процессор и оперативная память. Для хоста устройство хранения должно обеспечивать возможность записи блоков данных из внутренней памяти (как правило, ОЗУ) в устройство и считывание этих блоков из устройства в ОЗУ. Взаимодействие с устройством хранения выполняется по инициативе хоста (программы, выполняемой его процессором). В отличие от взаимодействия с внутренней памятью, с которой можно оперировать на уровне записи-чтения отдельных байтов, операции обмена с устройствами хранения всегда блочные. Блок может быть как фиксированного, так и произвольного размера. В настоящее время большее распространение получили устройства с фиксированным размером блока — это упрощает

многие аспекты взаимодействия. Самый популярный размер блока — 512 байт, хотя в ряде устройств используются и иные размеры блока. Блок может быть переписан из внутренней памяти во внешнюю или обратно только целиком, и для выполнения любой операции обмена с внешней памятью требуется специальная процедура (подпрограмма).
Блоки в устройстве могут адресоваться различными способами. Наиболее простой и удобной является *линейная адресация логических блоков*, при которой каждый блок хранимых данных адресуется одномерным адресом (числом) *LBA* (Logical Block Address — адрес логического блока). Исторически сложилось использование и иных способов адресации; для дисковых устройств это *трехмерная адресация CHS* (Cylinder-Head-Sector — цилиндр-головка-сектор).

9.2. Основные характеристики и конструктивы устройств хранения

По *методу доступа* к информации устройства внешней памяти разделяются на устройства с прямым (или непосредственным) доступом и устройства с последовательным доступом. В *устройстве хранения с прямым доступом* (Direct Access Storage Device, DASD) есть возможность обращения к блокам по их адресам в произвольном порядке и, что важно, допускается произвольное чередование операций записи и чтения блоков. Традиционными устройствами с прямым доступом являются дисковые накопители, и часто в понятие «диск», или «дисковое устройство» (disk device), вкладывают значение «устройство внешней памяти прямого доступа». Так, например, виртуальный диск в ОЗУ и электронный диск на флэш-памяти[1] отнюдь не имеют круглых, а тем более вращающихся деталей.
В *устройствах последовательного доступа* произвольное чередование операций записи и чтения, относящихся к произвольным адресам блоков, либо невозможно, либо затруднительно (требует дополнительных внутренних операций, занимающих длительное время). Традиционными устройствами с последовательным доступом являются *накопители на магнитной ленте* (tape device), они же *стримеры*. Здесь для доступа к блокам информации с произвольными адресами приходится вхолостую считывать (или ускоренно перематывать) все блоки, находящиеся между ними. Необходимость последовательного сканирования блоков (вперед или назад) — неотъемлемое свойство устройств последовательного доступа с подвижным носителем. Несмотря на очевидный проигрыш во времени доступа к требуемым данным, ленточные устройства последовательного доступа в качестве внешней памяти находят применение для хранения очень больших массивов информации и эффективно используются для чтения-записи длинных последовательностей блоков.

[1] Внутренне устройства хранения на флэш-памяти не являются устройствами прямого доступа из-за невозможности непосредственной перезаписи одиночных блоков.

Устройствами с последовательным доступом являются и оптические диски (CD, DVD). В этих устройствах информация записывается последовательно на один длинный спиральный трек. Конечно, устройство позиционирования головки позволяет ее довольно быстро (по сравнению с ленточными устройствами) перемещать на любой участок трека, обеспечивая произвольную адресацию. Однако по признаку невозможности произвольного чередования операций чтения-записи блоков (минимальная записываемая единица больше блока хранения) эти устройства являются последовательными. Программная эмуляция жесткого диска создает лишь иллюзию прямого доступа, скрывая от пользователя подробности непосредственной работы с устройством.

Главная характеристика устройства хранения — *емкость* (capacity), измеряемая в килобайтах, мегабайтах, гигабайтах и терабайтах (Кбайт, Мбайт, Гбайт, Тбайт, или в английской транскрипции KB, MB, GB, TB, или, еще короче — K, M, G, T). Здесь, как правило, приставки кило-, мега-, гига-, тера- имеют *десятичные* значения — 10^3, 10^6, 10^9 и 10^{12} соответственно. Емкость устройства в первую очередь определяется его носителем, однако она может ограничиваться и пределом возможности адресации блоков, свойственным тому или иному интерфейсу подключения.

Устройства внешней памяти могут иметь *сменные* или *фиксированные носители* информации. Применение сменных носителей (removable media) позволяет хранить неограниченный объем информации, а если носитель и формат записи стандартизованы, то они позволяют еще и обмениваться информацией между компьютерами. Существуют устройства с автоматической сменой носителя — ленточные карусели, дисковые устройства JukeBox. Эти довольно дорогие устройства применяют в крупных хранилищах данных. Для настольных машин имеются накопители CD-ROM с несколькими дисками (CD-changer), сменяемыми автоматически. Сменным может быть и целое устройство хранения.

Важнейшими общими параметрами устройств являются время доступа, скорость передачи данных и удельная стоимость хранения информации.

Время доступа (access time) определяется как усредненный интервал от получения устройством запроса на запись или чтение блока данных до фактического начала передачи данных. Дисковые устройства имеют время доступа от единиц до сотен миллисекунд. Для электронных устройств внешней памяти время доступа определяется быстродействием используемых микросхем памяти и при чтении составляет доли микросекунд, причем запись может продолжаться значительно дольше, что объясняется природой энергонезависимой электронной памяти. Для устройств с подвижными носителями основной расход времени имеет место в процессе *позиционирования* головок (seek time — время поиска) и *ожидания* подхода к ним требуемого участка носителей (latency — скрытый период). От того, что может делать система (хост и другие устройства) во время этой неизбежной задержки, предшествующей передаче запрашиваемых данных, зависит эффективность (общая производительность) компьютерной системы. В значительной степени эти возможности зависят от интерфейса устройства хранения.

Скорость записи и считывания определяется как отношение объема записываемых или считываемых данных ко времени, затрачиваемому на эту операцию. В затраты времени входят и время доступа, и время передачи данных. При этом оговаривается характер запросов — линейный или случайный, что сильно сказывается на величине скорости из-за влияния времени доступа. При определении скорости линейных запросов чтения-записи (linear transfer rate read/write) производится обращение к длинной цепочке блоков с последовательным нарастанием адреса. При определении скорости случайных запросов чтения-записи (random transfer rate read/write) соседние запросы направляются во все точки носителя. Для современных многозадачных ОС характерно чередующееся выполнение нескольких потоков запросов, и в каждом потоке высока вероятность последовательного нарастания адреса. Способность устройств хранения отрабатывать множество запросов, помещая их в свои внутренние очереди, существенно влияет на производительность системы в целом. Возможность эффективной работы с очередями существенно зависит от интерфейса устройства хранения.

Скорость передачи данных (Transfer Speed, Transfer Rate, или сокращенно XFER) определяется как производительность обмена данными, измеряемая после завершения поиска данных. Однако в способе измерения этого параметра возможны разночтения, поскольку современные устройства имеют в своем составе буферную память существенных размеров. Скорости обмена буферной памяти с собственно носителем (внутренняя скорость) и с внешним интерфейсом могут существенно различаться. Если *скорость* работы *внешнего интерфейса* ограничивается быстродействием электронных схем и достижимой частотой передаваемых сигналов, то *внутренняя скорость* более жестко ограничивается возможностями электромеханических устройств (скоростью движения носителя и плотностью записи). При измерениях скорости передачи на небольших объемах пересылок проявится ограничение внешнего интерфейса буферной памяти, при средних объемах — ограничение внутренней скорости, а при больших объемах проявится еще и время поиска последующих блоков информации. Бывает, что в качестве скорости передачи данных указывают лишь максимальную скорость интерфейса, а о внутренней скорости можно судить по частоте вращения дисковых носителей и числу секторов на треке (об этих понятиях рассказывается далее).

В табл. 9.1 приведены основные параметры распространенных устройств внешней памяти (поскольку эти компоненты развиваются весьма динамично, данные таблицы, фиксирующие состояние на весну 2005 года, не стоит рассматривать с коммерческой точки зрения). Они позволяют сопоставить возможности различных решений задачи хранения и переноса данных. А о динамике развития можно сказать, что до 2003 года, например, «модный» объем винчестера ежегодно удваивался даже при некотором снижении цены, так что удельная стоимость хранения снижалась более чем вдвое. К 2005 году темп роста замедлился. Поскольку емкость и скорость постоянно растут, в таблице эти возможности роста обозначены символом + (плюс). В последней колонке приведена стоимость хранения; для устройств со сменным носителем она определена без учета стоимости привода (который может быть очень дорогим).

Таблица 9.1. Характеристики устройств внешней памяти

Тип и размер диска	Емкость носителя, байт	Время доступа, мс	Скорость чтения/ интерфейса, Мбайт/с	Цена устройства, $	Цена хранения, $/Гбайт
FDD 3,5"	1,44M	100	0,055	8	200
HDD IDE	10–250Г+	7,5–10	2–70/133	110 (200 Гбайт)	0,55
HDD SATA	10–400Г+	7,5–10	20–70/150	120 (200 Г)	0,6
HDD USB, FireWire (ZIV)	10–100Г+	7,5–10	0,5–30	45 (100 Г)	0,4
HDD SCSI	20–250Г+	3–10	10–100/320	480 (146 Гбайт)	3,3
CD-ROM 1x	650M	240–500	0,15	Не выпускаются	Менее 0,05
CD-ROM 52x	650M	85	До 7,2	15	Менее 0,05
CD-RW 52/32/52 (IDE)	650M	100	До 7,2	30	0,5 (R) 1,3 (RW)
DVD-ROM 16x	4,7–17,08Г	90	До 21,6	30	Менее 0,05
DVD±RW 16x	4,7 Г	130	До 21,6	75	0,25 (R) 0,45 (RW)
MOD 3,5"	540/640M (до 1,3 Гбайт)	28	До 3,7	100	9
MOD 5,25"	2,6Г	25	10	200	10,5
Compact Flash	32M–4Г	<1	До 33	50 (512 M)	95
USB Flash	64M–2Г	<1	До 24	55 (512 M)	105
MMC, SD, SMC	64M–2Г	<1	До 5	55 (512 M)	105
LS-120	120M	70	0,1–0,5	130	100
Iomega Zip 100	100M	29	1,4	100	115
Iomega Zip 250	250M	29	2,4	100	52
Ultrim	200(400)Г	<1 мин.	80		0,2
DLT	20(40)Г	<1 мин.	1,5		1,2
DAT-72	36(72)Г	<1 мин.	3,5	400	0,4
DDS-3	8(4)Г	<1 мин.	3		0,7

Все устройства внешней памяти, применяемые в современных компьютерах, имеют *унифицированные конструктивные исполнения* (конечно, за исключением крупногабаритных устройств с автоматической сменой носителя). Их типоразмеры стандартизованы: наиболее жестко задаются ширина и высота устройств (поскольку их лицевые панели могут «выглядывать» из лицевой панели компьютера), глубина ограничена только максимально допустимым значением. Единообразное расположение резьбовых крепежных отверстий позволяет унифицировать конструкцию *отсеков* (bay) корпусов PC, предназначенных для установки накопителей. Первые накопители на гибких магнитных дисках диаметром 5,25", применяемые в PC, имели размеры лицевой панели 146,1 x 82,6 мм (5,75” x 3,25”) и глубину около 203 мм (8”). Этот формат называется полновысотным пятидюймовым форматом (доли дюйма для краткости опускают) — 5” full-height form-factor. Такого же размера были и первые винчестеры. Вскоре

высоту накопителя удалось уменьшить вдвое (half-height — половинная высота) — до 41,4 мм, и этот формат до сих пор используется как стандартный для многих типов устройств: НГМД с дискетами диаметром 5", приводов CD и DVD, стримеров, магнитооптики, накопителей на жестких дисках (старых и новых значительной емкости) и др. НГМД с дискетами диаметром 3,5" имеют ширину и высоту 101,6 х 25,4 мм (4” х 1") и длину (глубину) около 146 мм (5,74”). Этот формат широко используется и в современных трехдюймовых винчестерах, хотя среди них встречаются и «тонкие» модели высотой 20 мм (0,75[м]), и «толстые» — высотой 41 мм (1,6"). Для портативных компьютеров используют формат 2,75" (его называют и 2,5") с размерами 70 х 12,7 х 100 мм (2,76" х 0,5" х х 3,95") и более тонкие — 70 х 9 х 100 мм. Накопители формата 1,8" имеют габариты 54 х 7 х 71 мм, а особо тонкие — 54 х 5 х 71 мм.

Стандартизованы также и *разъемы подключения питания* (рис. 9.1). Миниатюрные разъемы используются только для питания трехдюймовых НГМД, практически для всех остальных устройств трех- и пятидюймовых форматов применяются большие разъемы. Напряжение +5 В задействуют для питания электронных схем, напряжение +12 В — для питания двигателей, хотя в некоторых накопителях приводы могут питаться и от шины +5 В.

ВНИМАНИЕ ----------

Редко, но попадаются блоки питания с ошибочно собранными колодками — у них шины +5 В и +12 В перепутаны, что легко заметить по цветовой маркировке проводов. Подключение накопителя к такой колодке чревато выгоранием его электроники — она рассчитана на питание 5 В, а не 12 В.

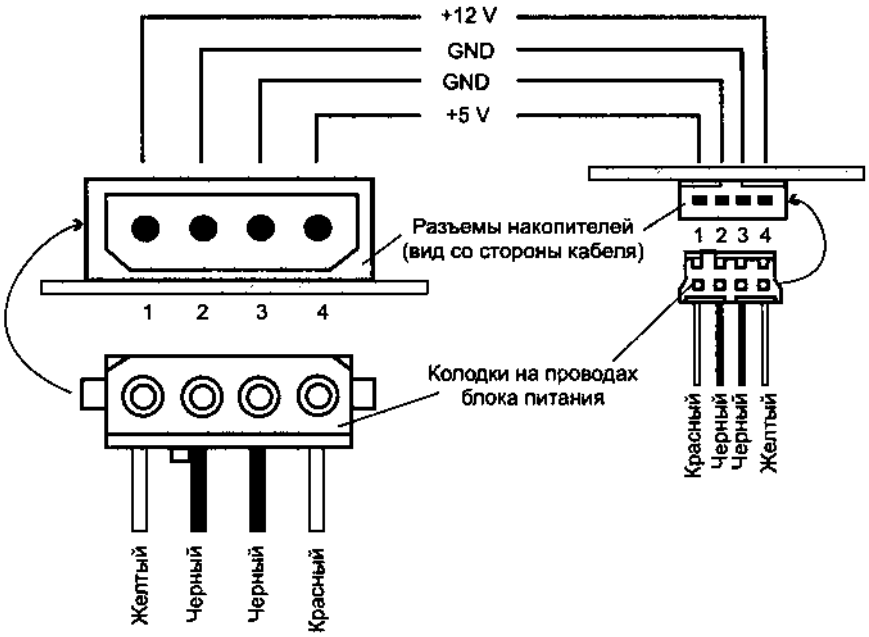


Рис. 9.1. Разъемы питания накопителей

В момент запуска двигателей ток потребления по цепи +12 В может превышать установившееся значение в несколько раз. В компьютерах и системах хранения с большим числом накопителей моменты их запуска стараются разнести во времени, что позволяет снизить пиковую нагрузку на источник питания. Возможность управления запуском двигателей зависит от интерфейса устройств хранения.

По отношению к корпусу компьютера устройства могут быть внутренними (internal) и внешними (external). *Внутренние устройства* помещаются в специальные трех- или пятидюймовые отсеки корпуса компьютера и питаются от его же блока питания. В описании корпусов компьютеров отсеки также подразделяются на внешние и внутренние, но они различаются лишь наличием или отсутствием выхода передней панели устройства, установленного в отсек, на лицевую панель корпуса. *Внешние устройства* помещают в отдельный корпус, а питаются они от собственного блока питания или от интерфейса (USB или FireWire). Есть внешние устройства, подключаемые к LPT-порту, которые перехватывают питание +5 В от разъема клавиатуры компьютера. Внешнее исполнение имеют как малогабаритные портативные устройства, так и особо крупные дисковые массивы. Сами приводы для внешних и внутренних устройств обычно имеют одинаковый конструктив одного из распространенных форматов.

Выпускаются и микровинчестеры (microdrive) в формате карт CFA Type II (42,8 x 36,4 x 5,0 мм) весом всего 16 г. Они предназначены в основном для цифровых фотокамер, но могут подключаться к LPT-порту или шине USB через адаптер интерфейса CompactFlash (по сигналам это интерфейс ATA), правда, с потерей в скорости передачи. Фирма выпускает и адаптеры для подключения микровинчестеров к шине PC Card (адаптер вместе с винчестером имеют габариты карты PCMCIA Type II, см. 14.11).

Твердотельные устройства хранения на флэш-памяти выпускаются в разнообразных конструктивных исполнениях. Первые «статические диски» выполнялись в виде устройств формата 3,5" с интерфейсом ATA. Затем появились флэш-карты расширения с интерфейсом PC Card (PCMCIA), Card Bus, которые используются в блокнотных ПК, а также в ряде бытовых электронных устройств, например в цифровых фотокамерах. Современные малогабаритные карты (Compact Flash, SmartMedia, MMC, SD и др.) имеют разнообразные (собственные) конструктивы, и для их подключения к компьютеру требуются специальные устройства, оборудованные соответствующими слотами. Очень популярными стали твердотельные устройства хранения с интерфейсом USB: устройства размером с брелок для ключей (или свисток) вставляются прямо в гнездо порта USB. Емкость такого «свистка» уже перевалила за гигабайт.

9.3. Интерфейсы устройств хранения

Как минимум, устройство хранения состоит из собственно носителя (фиксированного или сменного) и средств доступа к носителю. Под средствами доступа подразумеваются необходимые узлы записи и считывания, а также — для

подвижных носителей — привод и механизмы позиционирования. Для твердотельных устройств аналогом средств позиционирования являются средства адресации (выбора микросхемы, банка памяти, адреса). *Контроллер устройства хранения* занимается управлением носителем, избыточным кодированием и декодированием, исправлением ошибок или/и организацией повторных обращений к носителю и другими вспомогательными операциями. Для хоста контроллер совместно со своим программным *драйвером* должен обеспечивать базовые операции:

- сохранение (запись) информации из указанной области внутренней памяти хоста (размером в целое количество блоков) в указанное место на носителе устройств;
- считывание указанных блоков с носителя устройства в указанную область внутренней памяти хоста;
- вспомогательные операции, включая определение состояния и параметров носителя, форматирование носителя (если требуется), тестирование и т. п.

Соотношение интеллекта аппаратного (с точки зрения хоста) контроллера и сложности его программного драйвера (объема работы, выполняемой процессором хоста) зависит от типа устройства хранения. Для оптимизации производительности системы в целом (хоста и его устройств хранения) стремятся повышать интеллект контроллера. Для удешевления устройства хранения контроллер могут и упрощать до простейшего интерфейсного адаптера.

Физическое местоположение контроллера зависит от реализации устройства. Обобщенная схема подключения устройства к хосту приведена на рис. 9.2. Если контроллер располагается отдельно от устройства, то интерфейс устройства хранения бывает сугубо специфическим. Если контроллер встроен в устройство хранения, то вся специфика взаимодействия с носителем скрывается внутри устройства — во внутреннем интерфейсе между средствами доступа к носителю и контроллером. При этом появляется свобода в выборе интерфейса подключения устройства (фактически — его контроллера) к хосту.

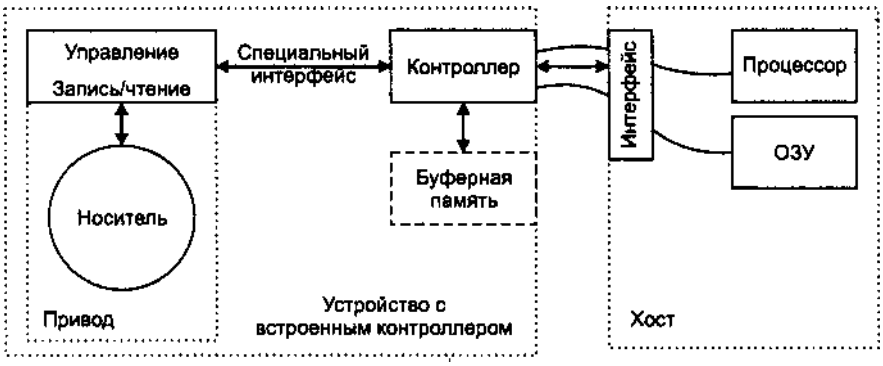


Рис. 9.2. Устройство хранения, подключенное к хосту, и его интерфейсы

Существенным параметром интерфейса подключения устройства хранения является скорость передачи данных. Если интерфейс подключения устройства

хранения обеспечивает связь средств доступа к носителю с контроллером, то этот интерфейс должен также обеспечивать передачу данных со скоростью доступа (записи и чтения) к носителю. В современных устройствах контроллер, расположенный вместе с носителем, обладает собственной буферной памятью. При этом появляется независимость пропускной способности внешнего интерфейса от скорости доступа к носителю. Это дает дополнительную свободу в выборе интерфейса подключения. Конечно, чем выше быстродействие внешнего интерфейса, тем быстрее происходит обмен данными с устройством хранения: задержка, требуемая для передачи данных между буферной памятью устройства и памятью хоста, уменьшается. В самых высокоскоростных современных винчестерах внутренняя скорость обмена (между носителем и контроллером) приближается к гигабиту в секунду. Скорость, обеспечиваемая внешним интерфейсом, как правило, выше внутренней. Однако и при медленном внешнем интерфейсе устройство хранения не теряет работоспособности, так что в ряде случаев ради удобства, дешевизны и доступности подключения жертвуют производительностью обмена с устройством хранения. Есть устройства хранения, критичные к скорости внешнего интерфейса: записывающие устройства оптических дисков не могут останавливать процесс записи в произвольном месте. Однако и эта проблема решается: объем буферной памяти увеличивается до такого размера, чтобы в нем умещался фрагмент, требующий непрерывной записи. К быстродействию внешнего интерфейса критичны и ленточные устройства хранения: несвоевременность доставки данных может приводить к их переходу в стартстопный режим, что вызывает дополнительное снижение их производительности.

Первые устройства хранения в ПК — накопители на гибких магнитных дисках (НГМД) — подключались интерфейсным кабелем-шлейфом к контроллеру, отделенному от самих устройств. Этот специализированный интерфейс (FDD) сохранился до сих пор (см. 9.7), им подключают дисководы, требующие скорости передачи всего 500 Кбит/с (около 60 Кбайт/с). К интерфейсу дисководов подключали и старые стримеры (очень тихоходные).

Аналогичный интерфейс (опять-таки специфический ST-506) поначалу использовался и для подключения винчестеров (так называемых MFM и RLL), по нему передавались «сырые» данные записи чтения с головок диска (правда, усиленные) и сигналы управления приводом. Позже накопители слегка «интеллектуализировали», и появился (ненадолго) интерфейс дисков ESDI, обеспечивающий скорость передачи данных до 1 Мбайт/с.

Параллельная шина ATA (IDE) — самый массовый интерфейс, применяемый для устройств хранения, — также является специфическим. Но эта специфика обусловлена историей его появления: чтобы не вводить новых интерфейсов, контроллер НЖМД, подключаемый к шине ISA, «разрезали» около интерфейса шины ISA. Цель этой операции — приблизить контроллер к приводу, из-за чего пришлось удлинить связь между основной частью контроллера (устройством ATA) и точкой его подключения (буферным адаптером, подключенным к шине ISA). По этому интерфейсу в параллельном виде передаются байты сохраняемых данных и содержимого регистров контроллера. Со временем место

подключения изменилось (теперь это шина PCI и ее последователи), но ради совместимости и преемственности интерфейс ATA сохранялся (постепенно модернизируясь). Для устройств, логически отличающихся от жестких дисков — оптических, магнитооптических, ленточных и любых других, — в 1996 году была принята спецификация ATAPI. Это пакетное расширение интерфейса, которое позволяет передавать по шине ATA устройству блоки командной информации, структура которых позаимствована из SCSI. Достигнутый потолок скорости ATA/ATAPI — 133 Мбайт/с (Ultra DMA Mode 6). Первоначально интерфейс ATA обладал ограничением адресуемого объема данных в 137 Гбайт, в последних версиях (ATA/ATAPI-6) это ограничение преодолено, нынешний «потолок» — 144 Пбайт (петабайт) (2^{48} блоков). Физически интерфейс ATA — это ленточный кабель-шлейф, предназначенный для подключения устройств внутри системного блока компьютера. Первоначально интерфейс не допускал «горячего» подключения-отключения; эта возможность появилась в реализациях ATA для блокнотных, а затем и для настольных ПК (для съемных устройств).

Последовательный интерфейс SerialATA (SATA) — преемник своего параллельного предшественника. Здесь повышается скорость обмена с устройством, решается проблема одновременной работы с несколькими устройствами, сразу используется расширенная адресация. Кабели и разъемы последовательного интерфейса SATA компактны, «горячее» подключение реализуется естественным образом: в SATA каждое устройство подключается к собственному порту хост-контроллера, а не к общей шине. Поначалу интерфейс SATA предназначался только для подключения внутренних устройств, позже он стал и внешним интерфейсом. В SATA-II появились новые элементы: мультиплексоры, позволяющие подключать к одному порту хоста несколько устройств, и селекторы портов, позволяющие подключать одно устройство (или мультиплексор) к двум хостам (но работать устройство может только с одним хостом).

Интерфейс SCSI — главный конкурент ATA для устройств хранения — является универсальным. Он предназначен для подключения устройств различных классов: дисковых, ленточных, оптических и других устройств хранения, принтеров, сканеров, коммуникационных и прочих устройств. В интерфейсе SCSI определена идеология взаимодействия хоста с устройствами, эффективная при работе с множеством устройств в многозадачных системах. Интерфейс SCSI поначалу существовал только в виде параллельной шины. Согласно современным стандартам протоколы интерфейса SCSI позволяют работать не только с параллельной шиной или последовательным интерфейсом *SAS* (недавно появившийся вариант SCSI), но и с другими средствами доставки: последовательной шиной *IEEE 1394* (FireWire), *Fibre Channel*, *SSA*, а также любыми IP-сетя- ми — *iSCSI*. Все варианты SCSI пригодны как для внутреннего, так и для внешнего подключения; они имеют поддержку горячего подключения-отключения, необходимую в больших и ответственных системах хранения данных. Предел адресации данных для устройств SCSI первоначально составлял 2 Тбайт (32-разрядная адресация блоков), позже ввели возможность 64-разрядной адресации блоков (объем хранения — до 9 444 732 965 739 290 427 392 байт).

Интерфейс SAS создан на основе дешевого интерфейса SATA и даже обеспечивает совместимость устройств SATA с контроллерами SAS (но не наоборот). Сферы применений этих интерфейсов различны. Устройства SAS предназначены для систем хранения данных предприятий (Enterprise-class), они имеют од- но- или двухпортовые интерфейсы. Устройства SATA (только однопортовые) предназначены для настольных систем, они дешевле устройств SAS. По схемотехнике и встроенному ПО (firmware) устройства SAS близки к устройствам Fibre Channel (двухпортовым), применяемым в сетях хранения данных (Storage Area Network, SAN) масштабов предприятия.

Для внешних устройств хранения с успехом применяют подключение и к шине USB, и к LPT-порту. Интерфейс LPT-порта обеспечивает невысокую скорость передачи (до 2 Мбайт/с), но он присутствует практически на всех компьютерах (даже очень старых). Шина USB 1.0 для устройства хранения может предоставить пропускную способность до 1,2 Мбайт/с, шина USB 2.0 — до 25 Мбайт/с. Более эффективна для подключения внешних устройств шина FireWire, выступающая в роли среды доставки SCSI.

Сравнительные характеристики интерфейсов устройств хранения данных приведены в табл. 9.2. Здесь показаны максимальные значения основных параметров. Реальная скорость передачи данных, естественно, всегда ниже. Топологические ограничения (число устройств и максимальное удаление) для ряда интерфейсов указаны формально (в пределах работоспособности). Реальные значения, отвечающие эффективной конфигурации, могут быть скромнее (например, считается, что FC-AL эффективно работает при числе узлов до 30 и длине кольца до 100 м). Под возможностью одновременного обмена подразумеваются физически одновременно выполняемые передачи информации, относящиеся к выполнению разных заданий (от одного или нескольких инициаторов с одним или несколькими целевыми устройствами). Работа с несколькими инициаторами тоже имеется в виду одновременная (селектор порта SATA эту возможность не обеспечивает). Подробности функционирования различных интерфейсов рассмотрены в соответствующих разделах.

Таблица 9.2. Характеристики интерфейсов устройств хранения

Интерфейс	FDD	ATA	SATA	SCSI	SAS	FC-SW	FC-AL	1394	iSCSI	USB 1.x/2.0	LPT
Скорость, Мбайт/с	0,06	133	150	160 320 (640)	150 300	100 200	100 200	40 80 160	1–100	1,2/24	2
Число устройств	2	2	4 и выше	16	16 384	2^{24}	126	63	Не ограничено	127	1
Длина кабеля, м	0,5	0,5	1	25	1	60 500 10 км	60 500 10 км	4,5 (100)	100 500	5	5
Максимальное удаление, м	0,5	0,5	1	25 (400 нс)	1	10 км	10 км	72	Не ограничено	25	5

продолжение

Таблица 9.2 (*продолжение)*

Интерфейс	FDD	ATA	SATA	SCSI	SAS	FC-SW	FC-AL	1394	ISCSI	USB 1.x/2.0	LPT
Одновременный обмен	–	–	–	–	+	+	–	–	+	–	–
Работа с несколькими инициаторами	–	–	–	+	+	+	+	+	+	–	–

При всем разнообразии интерфейсов в большинстве случаев выбирать приходится между двумя основными — ATA (IDE) и SCSI, правда, теперь у каждого есть последовательный вариант (SATA и SAS). Шина ATA в современных системах работает в режиме UltraDMA, так что по скорости и защищенности от ошибок она не отстает от конкурентов. Для внешних устройств широко применяются интерфейсы USB и FireWire. Достоинства и недостатки основных интерфейсов подключения устройств хранения сведены в табл. 9.3. Не попавший в таблицу интерфейс *Fibre Channel* применяют для подключения устройств внешней памяти в больших системах. Этот интерфейс позволяет значительно разносить устройства памяти и компьютеры (допускается разделяемое использование устройств). Эти свойства ценны для особо ответственных применений, когда требуется обеспечить живучесть системы даже при частичных разрушениях (при стихийных бедствиях, катастрофах и прочих «радостях» современного бытия).

Таблица 9.3. Сравнение интерфейсов устройств хранения

Интерфейс	Достоинства	Недостатки
ATA	Массовость. Низкая цена устройств и кабелей. Простота конфигурирования. Эффективность при малом числе устройств	Малое число устройств: до 2 на шине (обычно есть 2 шины). Только для внутренних устройств. Низкая эффективность при работе с двумя устройствами на шине. Высокая загрузка ЦП и большое число прерываний при отработке запросов
SATA	То же. Независимость устройств. Высокая эффективность при поддержке NCQ и устройствами и контроллером	Малая распространенность устройств и контроллеров с поддержкой NCQ
SCSI	Большое число подключаемых устройств, внутренних и внешних. Высокая эффективность в многозадачных системах при большом числе устройств	Высокая цена контроллера, устройств и аксессуаров (кабели, терминаторы). Сложность установки и конфигурирования
SAS	То же. Возможность физически одновременного обмена с несколькими устройствами. Простота подключения и конфигурирования	Высокая цена оборудования
USB	Удобство подключения. Распространенность. Средняя скорость (USB 2.0)	Загрузка ЦП, возрастающая с увеличением числа устройств. Низкая скорость (USB 1.0)
FireWire	Удобство подключения. Высокая скорость. Эффективная работа с множеством устройств	Малая распространенность в мире PC

9.3. Преодоление физических ограничений — массивы RAID

У любого физического устройства хранения данных есть пределы возможностей, обусловленные современными ему технологическими достижениями и приемлемой ценой. В ряде случаев требуются устройства хранения данных с «запредельными» параметрами:

- емкостью хранилища, превышающей емкость физического устройства хранения;
- длительной скоростью передачи данных, превышающей внутреннюю скорость передачи устройства;
- надежностью, превышающей надежность физического устройства.

Такое выдающееся устройство можно получить за счет избыточности — параллельного использования множества обычных устройств. Применительно к дисковым устройствам применяют термин RAID (Redundant Array of Inexpensive Disks — избыточный массив недорогих дисков). Идея заключается в подключении группы обычных, как правило, однотипных, дисков к RAID-контроллеру — устройству, которое для хост-компьютера этот массив представляет как один *виртуальный диск* с улучшенными свойствами. Улучшения касаются вышеперечисленных параметров в различных сочетаниях, хотя какими-то параметрами иногда приходится жертвовать. Аналогичные массивы применяются и для ленточных устройств хранения, но называются они RAIT (Redundant Array of Inexpensive Tapes — избыточный массив недорогих ленточных устройств хранения).

В зависимости от алгоритма представления диска различают следующие схемы (типы, уровни) RAID:

- RAID 0 — дисковый массив без избыточности и отказоустойчивости, простейшее средство повышения производительности и увеличения объема. Виртуальный диск разбивается на *зоны*, или *полосы* (strips), которые равномерно распределяются по всем дискам массива. Размер зон кратен размеру сектора диска. При обращениях хоста к большому блоку данных, занимающему несколько зон, RAID-контроллер посылает запросы одновременно к нескольким дискам — обращение реально распараллеливаются, что повышает производительность как по чтению, так и по записи. Для коротких запросов выигрыша в скорости нет. Пространство всех дисков используется полностью (избыточности нет). Отказ любого диска приводит к отказу всего массива, надежность виртуального диска *ниже,* чем у одного большого устройства, поскольку вероятность отказа хотя бы одного из дисков выше, чем вероятность отказа каждого из них в отдельности.
- RAID 1 — зеркальное отражение (mirroring). Два (или более) диска дублируют друг друга. Запись информации выполняется одновременно на все диски, чтение — с любого свободного, в результате чего производительность чтения повышается. Отказ одного диска приводит только к снижению скорости

чтения. Отказавший диск может быть заменен, и для ввода его в действие требуется просто копирование данных с оставшегося диска. Эффективность использования пространства дисков низкая (при двух дисках эффективный объем составляет лишь половину их суммарного, при большем числе — и того меньше). Надежность виртуального диска тем выше, чем больше дисков в массиве, и она превышает надежность одиночного диска.

- RAID 2 — избыточный массив, в котором *биты данных* распределяются по нескольким дискам и еще несколько дисков несут проверочные коды Хэмминга (ECC). Проверочные коды позволяют исправлять битовые ошибки, возникающие при отказе одного из дисков. Отказ физического устройства не приводит к отказу виртуального диска. Для получения высокой производительности диски в массиве должны быть синхронизированы по позиционированию головок и по вращению шпинделей, поскольку данные записываются параллельно сразу на все диски. Надежность ниже, чем у RAID 1, но и избыточность меньше. Эффективность кода Хэмминга (избыточность, обеспечивающая исправление) зависит от числа информационных битов: выгоднее более длинные слова, то есть массивы с большим числом дисков.
- RAID 3 — избыточный массив, отличающийся от RAID 2 тем, что вместо кодов Хэмминга (несколько дополнительных битов и, соответственно, дисков) используется лишь бит четности (1 диск). Поскольку отказ (ошибка чтения) каждого диска определяется его встроенным контроллером, RAID-контрол- лер «видит» ошибочный диск и его бит вычисляет через бит четности. Это и позволяет отказаться от кода Хэмминга, с помощью которого вычисляются и позиция ошибочного бита, и его значение. Производительность, отказоустойчивость и эффективность использования пространства довольно высокие, но для достижения высокой скорости требуется синхронизация устройств.
- RAID 4 — избыточный массив, в нем данные разбиты на зоны, размер которых кратен размеру сектора (как у RAID 0), и дополнительно выделен диск для размещения зоны четности для всех зон данных. В случае отказа любого из дисков его данные восстанавливаются с помощью зон четности и соответствующих зон данных на «живых» дисках. Программная реализация довольно сложная. Короткие запросы записи обслуживаются медленно: изменение одного сектора (зоны) требует чтения соответствующего сектора (зоны) на всех дисках данных с тем, чтобы вычислить и записать новое значение зоны четности. Возможна оптимизация записи, при которой вычисляется изменение значения четности на основе старых и новых данных изменяемого сектора данных.
- RAID 5 — распределение и чередование данных и четности по дискам, но для битов четности не выделяется специальный диск (биты четности распределяются по всем дискам по кругу). Обеспечивает более высокую скорость записи, чем RAID 4. В случае отказа одного диска потерянные данные восстановимы, но не так быстро и просто, как в RAID 4.
- RAID 6 — более сложная схема, устойчивая даже к двойным отказам (но за счет более низкой производительности).

Возможны и более сложные массивы, в которых используются двухступенчатые комбинации данных уровней:

- RAID 10 — массив RAID 0, собранный из пар зеркальных дисков (RAID 1). Обеспечивает высокую скорость и надежность, но ценой большой избыточности.
- RAID 30 — массив RAID 0, собранный из блоков RAID 3.
- RAID 50 — массив RAID 0, собранный из блоков RAID 5. Обеспечивает отказоустойчивость и высокую производительность.

Для организации эффективных схем массивов RAID требуется значительное количество дисков. Так, например, RAID 10 требует не менее 4 дисков, что для настольных компьютеров не всегда приемлемо. Фирма Intel в своих «настольных» чипсетах i925X/915 применяет *технологию Matrix Storage*, позволяющую всего на паре дисков SATA организовывать пару независимых массивов RAID: RAID 0 (striping) для повышения скорости и RAID 1 (mirroring) для повышения надежности. При этом диски, фактически, используются по частям (что характерно и для «больших» RAID-контроллеров на SCSI-дисках),

В качестве интерфейса подключения физических устройств чаще всего фигурирует интерфейс SCSI, который обеспечивает подключение большого числа устройств и высокую эффективность использования шины. Еще лучшие результаты дает применение последовательного интерфейса SCSI (SAS), поскольку он обеспечивает возможность физически одновременных обменов с несколькими устройствами. Есть и RAID-контроллеры с интерфейсом ATA, но они менее эффективны. Последовательный интерфейс SATA эффективнее своего параллельного предшественника, однако он уступает SAS по суммарной производительности массива. Применение мультиплексоров SATA позволяет увеличить число физических устройств, используемых в массиве, но не повысить пропускную способность.

Массивы RAID могут быть реализованы для хост-компьютера как аппаратно, так и программно. *Аппаратный RAID-контроллер* представляет собой интеллектуальное устройство со своим мощным микропроцессором, имеющее интерфейсы для подключения дисков, а также интерфейс подключения к хост- компьютеру. *Внешний массив RAID* представляет собой отдельное устройство (блок, шасси, стойку) со своим блоком питания, отсеками для установки дисков и, естественно, RAID-контроллером. В качестве интерфейса с хост-компьютером для внешних массивов используют интерфейсы SCSI (параллельный или последовательный), Fibre Channel, FireWire, SATA. *Внутренний RAID-контроллер* подключается к шине PCI, PCI-X или PCI Express (раньше задействовали EISA и даже ISA, встречалась и VLB). Конструктивно это либо карта расширения, устанавливаемая в соответствующий слот, либо интегрированный компонент, расположенный на системной плате. К внутреннему контроллеру могут подключаться как внутренние, так и внешние физические устройства хранения (в зависимости от интерфейса).

Функции RAID могут быть реализованы и программно, средствами ОС компьютера (например, эти функции поддерживаются в Windows NT/XP/2000). При

этом в качестве интерфейсов дисков применяют интерфейсы существующих контроллеров (предпочтительно SCSI, но используют и ATA). Программный массив RAID вполне справляется с увеличением объема и повышением надежности, но для повышения скорости требуется большая вычислительная мощность процессора.

Массив RAID для операционной системы хоста выглядит как одно логическое устройство хранения. Однако в отличие от обычного устройства, готового к использованию сразу после подключения, массив RAID требует предварительного конфигурирования и обслуживания с помощью специальных утилит. В случае внешних RAID-контроллеров конфигурирование и обслуживание могут выполняться и с автономного пульта управления массивом. Заметим, что архитектура SCSI позволяет использовать также иерархическую адресацию и через одно SCSI-устройство, подключенное к хосту, обращаться к подчиненным ему логическому устройству.

Отказоустойчивые схемы RAID позволяют продолжать работу хранилища даже при отказе одного физического устройства. В таких системах, как правило, реализована «горячая замена» устройств. Для вновь установленного устройства должна быть выполнена синхронизация данных. Для зеркального диска это просто копирование данных с его зеркальной копии, для более сложных схем — вычисления, выполняемые в соответствии с установленной схемой избыточности. Синхронизация, обычно занимающая довольно много времени, как правило, может выполняться на фоне обращений к «живой» части массива.

9.4. Устройства, системы и сети хранения данных

Компьютерные системы разных классов нуждаются и в разных классах устройств хранения. В простейшем случае к одному хосту подключается одно или несколько собственных (локальных) устройств хранения. В случае объединения компьютеров в сети появляются более сложные конфигурации и варианты соединения устройств хранения и хостов. В этом разделе приводится объяснение некоторых терминов, в том числе таких, как SAS, NAS и SAN.

Хост (host) — это компьютер (процессор и память), к которому подключается устройство хранения.

Физическое устройство храпения — элементарное устройство, позволяющее выполнять энергонезависимое хранение блоков данных на одном носителе (возможно, сменном). Примеры — винчестер, привод CD/DVD, флэш-память с интерфейсом USB и т. п.

Подсистема (subsystem) *хранения* представляет собой множество устройств, подключенных к *концентратору*, имеющему интерфейс связи с хостом. Пример подсистемы хранения — массив RAID. Подсистемы хранения, как правило, конструктивно оформляются в виде внешнего (по отношению к компьютеру) *блока хранения* (storage enclosure) или шасси, в состав которого, кроме концентратора, устройств и, разумеется, блока питания, могут входить дополнитель

ные *средства обслуживания* (enclosure management). Среди решаемых ими задач могут быть поддержание климата, контроль питания, несанкционированного доступа и т. п. Управляет этими средствами *обслуживающий процессор* (Storage Enclosure Processor, *SEP), с* которым хост должен иметь возможность взаимодействия.

Система хранения данных — это конструкция, обеспечивающая автоматический доступ к более чем одному сменному носителю:

- *Автозагрузчик* — конструкция с одним приводом (ленточным или дисковым), снабженным магазином, в который устанавливаются несколько носителей. Загрузка носителя в привод выполняется автоматически (по команде к устройству) в произвольном порядке.
- *Стекер* — конструкция с одним приводом и механизмом подачи, в который устанавливается стопка носителей. Носители в привод загружаются в том порядке, в котором они установлены (стек), и лишь однократно. Стекеры используются для автоматических систем резервного копирования и архивирования.
- *Библиотека* — конструкция с несколькими приводами и большим количеством носителей, связанных роботом, который может любой носитель поместить в любой привод (и извлечь). Библиотека обеспечивает одновременный доступ к нескольким носителям, а также повышение надежности (допускается отказ одного привода).

В дальнейшем для краткости под *хранилищем* будем подразумевать как одиночное физическое устройство хранения, так и систему (подсистему) хранения. Отдельно стоящий компьютер может пользоваться только своим хранилищем. В компьютерных сетях отношения между хостами (компьютерами, имеющими, как минимум, процессор и память) и хранилищами данных более разнообразны.

Хост, предоставляющий доступ к своим хранилищам другим хостам, связанным с ним по сети, является *сервером. Файл-сервер* (file server) предоставляет возможность выполнения файловых операций своим клиентам (удаленным хостам): открытие, чтение, запись, создание, удаление файлов и каталогов, расположенных на доступных ему хранилищах. *Сервер приложений* (application server) позволяет своим клиентам работать с данными, расположенными на доступных ему хранилищах. В этом случае клиенты обращаются к серверу с иными запросами, которые вызывают запросы сервера к своим хранилищам. *Сервер резервного копирования* (backup server) обеспечивает своим клиентам (агентам, работающим как на других серверах, так и на компьютерах конечных пользователей) возможность сохранения (и восстановления) данных на доступных ему хранилищах.

Традиционными являются *хранилища, непосредственно подключенные к серверу* (Server Attached Storage, SAS); чаще всего хранилищами являются внутренние винчестеры компьютера-сервера. Если в процессе эксплуатации требуется увеличение объема хранилища, то к серверу приходится подключать дополнительные устройства, что сопровождается его остановкой и реконфигурированием. В сети, к которой подключен сервер SAS, передаются запросы файловых опера

ций и ответы на них. Блочные запросы (к физическим блокам хранилища), в которые преобразуются файловые запросы, передаются между сервером и подключенным устройством, но не по сети.

С конца 90-х годов стали использовать *хранилища, подсоединенные к сети* (Network Attached Storage, NAS). Фактически, NAS — это файл-сервер, обеспечивающий доступ к своему хранилищу, как правило, по протоколу NFS (Network File System — сетевая файловая система) или CIFS (Common Internet File System — общая межсетевая файловая система). Однако для потребителя NAS выглядит не как компьютер (с монитором и клавиатурой), а просто как устройство. Функции управления сервером автоматизированы, упрощены и, в основном, скрыты от пользователя. Преимущество NAS — удобство интеграции дополнительных хранилищ в существующие компьютерные сети. Увеличение объема хранимых данных требует лишь подключения дополнительных устройств (NAS) без вмешательства в работу уже существующих. С точки зрения сетевого взаимодействия SAS и NAS существенно не различаются. Интерфейс, которым подключается хранилище в топологиях SAS и NAS, может быть любой разновидностью ATA или SCSI.

Иная модель взаимодействия используется в уже упоминавшихся *сетях хранения данных* (SAN), объединяющих хосты (компьютеры) и хранилища данных. В этих сетях к устройствам хранения передаются блочные запросы, а не файловые. Сеть SAN отделяют от общей сети (LAN/WAN), что снижает зависимость эффективной скорости передачи данных для устройств хранения в распределенных системах от непредсказуемого трафика в общей сети. Кроме того, это повышает надежность и безопасность самой ответственной части компьютерных сетей — системы хранения данных. Предсказуемость скорости передачи (и гарантированное ее значение), которую обеспечивает сеть на основе Fibre Channel, в ряде случаев весьма существенна для эффективной работы хранилищ данных. Так, несвоевременная доставка блоков из-за перепадов скорости передачи данных может вызвать стартстопный режим работы стримера, что иногда в разы снижает его эффективную производительность. В системах хранения данных, в которых широко используются стримеры для резервного копирования и архивирования, такие перепады могут быть неприемлемы.

Сеть SAN может быть реализована на основе как массовой технологии Ethernet, так и более эффективных (но и дорогих) технологий, например Fibre Channel, лучше приспособленных для передачи SCSI-команд. Технология Fibre Channel применяется в локальных сетях SAN (хотя допускает удаление узлов до 10 км). Локальные «островки» SAN на Fibre Channel могут объединяться в более крупные сети, используя различные транспортные магистрали: выделенные оптические соединения (CWDM, DWDM), ATM, SONET/SDH. Практически безграничное удаление устройств хранения обеспечивает технология iSCSI — передача SCSI-команд по любым IP-сетям (даже через Интернет).

В SAN для передачи файловых запросов NFS и CIFS используется 1P-прото- кол, и доступ к файлам сопровождается задержками. Эти задержки уменьшаются после установления IP-соединения, а также после просмотра каталога с фай

лом. Для сокращения задержек предлагаются различные решения. Одно из них — переход на *файловые системы прямого доступа* (Direct Access File System, DAFS) с использованием *удаленного прямого доступа к памяти* (Remote Direct Memory Access, RDMA). Это один из примеров, иллюстрирующий сближение и взаимное проникновение технологий SAN и NAS.

9.6. Логическая структура дисков

С аппаратной точки зрения любое устройство хранения прямого доступа (диск) можно представить как совокупность секторов, адресуемых тем или иным способом (CHS или LBA), и каждый сектор может быть записан и считан только целиком и независимо от других. Однако для большинства прикладных программ интерес представляет обращение не к отдельным секторам, а к файлам, которые могут занимать произвольное (возможно, не целое) количество секторов. Для облегчения обращения к файлам и упорядочения использования пространства секторов диска в состав любой операционной системы входит *файловая система*, тесно связанная с логической структурой диска.

Разделы и логические диски

Операционная система представляет внешнюю память в виде набора *логических дисков* (logical drive). Каждому логическому диску присваивается свое логическое имя: A, B — для дискет, C, D, E и следующие буквы — для жестких дисков, CD-ROM и прочих устройств. Рассмотрим логическую организацию физических дисков, чтобы понять, каким образом несколько логических дисков (например, C, D и E) могут оказаться на одном винчестере.

Логический диск — это совокупность секторов с последовательно нарастающими номерами. Самый первый сектор логического диска называется *загрузочным* (boot sector). В этом секторе всегда хранится описатель параметров диска и файловой системы. Дополнительно может содержаться программа загрузки операционной системы (загрузчик). Если на диске с загрузчиком присутствуют еще и сами файлы ОС, что обеспечивает возможность загрузки этой ОС на компьютер, такой диск называется *системным*.

Поскольку жесткий диск (hard drive) в общем случае не является съемным (иное его название — fixed disk — фиксированный диск), а операционных систем, которые хочется использоваться на одном компьютере, может быть несколько, и каждая из них претендует на свою логическую организацию диска, договорились о возможности разбиения жесткого диска на несколько независимых *разделов*. Дискеты в таком разбиении не нуждаются, поскольку их легко сменить, да и маленький объем не располагает к делению. Помимо обеспечения «плюрализма» операционных систем, разбиение диска на разделы позволяет уменьшить размеры логических дисков, что бывает выгодно из-за ограничений файловых систем. Путем разбиения винчестера на логические диски добиваются упорядочивания использования дискового пространства. Структуру жесткого диска поясняет рис. 9.3.

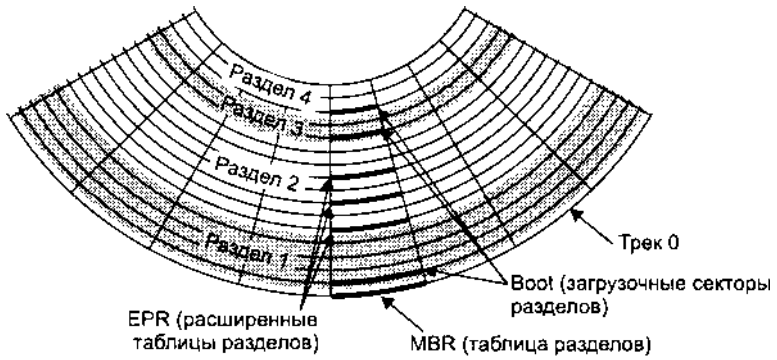


Рис. 9.3. Структура жесткого диска

Физический жесткий диск может быть разбит на несколько *разделов* (partition). Информация о структуре диска — *таблица разделов* (partition table) — хранится в *главной загрузочной записи* (Master Boot Record, MBR), находящейся в общеизвестном месте — цилиндр 0, головка 0, сектор 1. В начале этого сектора располагается программа *главного загрузчика* (master boot), а за ней — таблица разделов, содержащая четыре описателя разделов. Каждый описатель задает границы разделов, причем в двух системах: CHS (координаты начала и конца) и LBA (начало и длина). Разделы, как правило, начинаются точно по границе цилиндра (координаты N, 0, 1), кроме первого, начинающегося обычно с первой головки нулевого цилиндра (0, 1, 1), поскольку под нулевой головкой расположен сектор с MBR. Заканчиваться разделы должны на границе цилиндра, что позволяет через номера конечной головки и конечного сектора определить число головок и секторов на треке.

Описатель задает и атрибуты раздела — системный код и флаг активности. *Флаг активности* указывает главному загрузчику, какой раздел ему следует загружать. Флаг активности может быть установлен только для одного раздела диска (или вообще не устанавливаться). *Системный код* определяет тип раздела; операционная система для своей файловой системы может использовать разделы только известных ей типов. Таблица разделов может заполняться как с начала, так и с конца. Если разделов меньше четырех, то свободные описатели обнуляются. Свободные описатели, равно как и занятые, могут располагаться в любом месте таблицы (в начале, середине, конце). Формирование таблицы разделов — *конфигурирование жесткого диска* — как правило, выполняется утилитой FDISK.

Структура раздела зависит от его типа. Некоторые системные коды (типы разделов) приведены в табл. 9.4. Далее ограничимся описанием разделов и логических дисков для DOS/Windows 9x.

Таблица 9.4. Коды и типы разделов жесткого диска

Код	Раздел	Объем
01	DOS FAT12	До 15 Мбайт
04	DOS FAT16	До 32 Мбайт

Код	Раздел	Объем
05	(DOS) Extended	До 2 Гбайт
06	DOS FAT16 (Big DOS)	До 2 Гбайт
07	OS/2 HPFS, Windows NT NTFS	До 2^{64} байт
0B	Win95 FAT32	От 512 Мбайт до 2 Тбайт
0C	Win95 FAT32 (LBA)	От 512 Мбайт до 2 Тбайт
0E	Win95 FAT16 (LBA)	От 32 Мбайт до 2 Гбайт
0F	(Win95) Extended (LBA)	До 2 Тбайт

Разделы с кодами (01, 04, 06, 0B, 0C, 0E) являются *первичными разделами* (primary partition) DOS/Windows. Утилита FDISK для MS-DOS и Windows 9x позволяет создавать не более одного первичного раздела, хотя, в принципе, их может быть и больше. Первичный раздел содержит один логический диск. В стандартном случае, когда на диске имеется один первичный раздел, для первого винчестера на нем будет диск C, для второго — D и т. д. Операционные системы MS-DOS и Windows 9x «не любят», когда на одном диске более одного первичного раздела, а также когда первичный раздел не является первым в таблице разделов. Другие ОС (например, Linux) более лояльны к количеству и расположению разделов. В первом секторе логического диска находится загрузчик, а также описатель типа файловой системы (FAT12, FAT16, FAT32) и структуры диска. Загрузчик загружает ОС, расположенную на данном диске (если он системный). За загрузчиком располагаются несколько копий таблицы размещения файлов (File Allocation Table, FAT), корневой каталог и собственно область данных диска. Разные коды первичных разделов указывают на различную разрядность FAT (см. далее), новые типы вводились по мере роста размеров винчестера. С Windows 95 OSR2 появились новые типы разделов для FAT32 и FAT16 (0Ch, 0Eh) специально для дисков, поддерживающих адресацию LBA. Заметим, что в каждом описателе разделов задаются как трехмерные границы раздела (начальные и конечные номера цилиндра, головки и сектора), так и линейные (номер начального сектора и их количество), но долгое время по старинке использовали только трехмерные описатели.

Среди разделов DOS/Windows 9x активным может быть только первичный раздел (может выполняться только его загрузчик).

Расширенный раздел (extended partition), имеющий в табл. 9.4 код 05 или 0F, служит для организации произвольного количества логических дисков (на рис. 9.3 раздел 2 является расширенным). Первый сектор расширенного раздела аналогичен MBR (но загрузчик отсутствует) и содержит *расширенную таблицу разделов* (Extended Partition Record, EPR) той же структуры, но с некоторыми оговорками. Первый описатель задает *вторичный* (secondary) раздел, отведенный под очередной логический диск; в нем указывается код раздела с файловой системой (для DOS/Windows это FAT с кодами 04h, 06h, 0Bh, 0Ch или 0Eh, для других ОС — свои). В этом описателе, как обычно, задаются координаты начала и конца раздела с логическим диском (трехмерные и линейные). Если этот логический диск занимает не весь объем расширенного раздела, то второй описатель тоже имеет код 05 или 0F и указывает на положение сектора

со следующей расширенной таблицей разделов. Остальные описатели не используются (их коды нулевые). Если свободного места в разделе уже нет, то и второй описатель не используется. В следующей расширенной таблице разделов действуют те же правила. Эта цепочка заканчивается на расширенной таблице, у которой во втором описателе стоит нулевой код раздела. Заметим, что второй описатель в расширенных таблицах может указывать только на положение следующей расширенной таблицы. Часть пространства расширенного раздела может оставаться не распределенной, в дальнейшем она может быть выделена под логические диски. Цепочка расширенных таблиц разделов должна быть непрерывной, неветвящейся (используются только два описателя, и только второй может указывать на следующую таблицу) и незацикленной (второй описатель не должен ссылаться на ту же таблицу или предыдущую в цепочке). Несоблюдение первых двух условий ведет «только» к потере логических дисков (их система не найдет). Несоблюдение последнего условия может привести к «зависанию» ОС при загрузке: она зациклится на бесконечном определении повторяющихся логических дисков. Код (05 или 0F) расширенного раздела не несет никакой информации о файловой системе, и данный тип раздела используется как указатель на расширенную таблицу рядом ОС, в том числе и отличных от DOS/Windows. Координаты расширенных таблиц разделов обычно имеют вид N, 0, 1.

По расположению на физическом диске расширенные разделы являются вложенными друг в друга; все они располагаются в области, описанной в главной таблице разделов как расширенный раздел. В главной таблице может быть описан лишь один расширенный раздел.

В трактовке описателей расширенных разделов, к сожалению, возможны разночтения. Например, FDISK из MS-DOS 6.22 «честно» расставляет трехмерные описатели (в логической геометрии CHS), но это возможно лишь для дисков объемом не более 8,4 Гбайт. При этом в расширенных таблицах линейные описатели логических дисков (вторичных разделов) ставятся не относительно начала физического диска (сектора с MBR), а относительно сектора с EPR. Здесь же линейные ссылки на очередную таблицу EPR (с кодом 05) ставятся относительно первой в цепочке (линейного адреса расширенного раздела, описанного в MBR). Поле длины в описателе логического диска (вторичного раздела) относится именно к нему. В ссылке на очередную таблицу EPR поле длины определяет расстояние до следующей таблицы EPR. Сборка цепочки логических дисков именно по этому полю позволяет удалять логические диски из середины цепочки, не разрывая ее.

Если расширенные разделы имеют код 0Fh, то линейные адреса всех элементов таблиц указываются относительно начала физического диска (так поступает новая версия утилиты FDISK, и это более естественно, поскольку при этом описатель LBA является эквивалентом описателя CHS).

Каждый логический диск из расширенного раздела имеет ту же структуру, что и вышеописанный первичный раздел. Он также начинается с загрузочного сектора (только загрузчик никогда не исполняется), в котором имеется описание структуры логического диска. Координаты загрузочных секторов логических

дисков обычно имеют вид N, 1, 1. Операционная система назначает логическим дискам расширенных разделов имена (буквы), остающиеся после дисков первичных разделов. Так, если имеется один жесткий диск и у него есть первичный и вторичный разделы, причем последний разбит на два логических диска, то мы увидим следующее:

- C — первичный раздел;
- D — первый логический диск расширенного раздела;
- E — второй логический диск расширенного раздела.

Теперь если добавить второй жесткий диск (всего с одним первичным разделом), то картина изменится:

- C — первичный раздел первого диска (остался на месте);
- D — первичный раздел второго диска (новый);
- E — первый логический диск расширенного раздела первого диска (тот, что имел букву D);
- F — второй логический диск расширенного раздела первого диска (тот, что имел букву E).

Если бы у нового диска был расширенный раздел со своими логическими дисками, то они бы заняли следующие буквы (G, H...). О механизме присвоения логических имен следует помнить, устанавливая программы на компьютер, к которому эпизодически подключают дополнительные винчестеры. Незыблемое имя (C) останется только у первичного раздела винчестера, подключенного ведущим к первому контроллеру ATA (если используется SCSI, то все немного сложнее).

Логический диск с файловой системой FAT

Структуру логического диска рассмотрим на примере первичного раздела DOS с файловой системой FAT16. Логический диск начинается с *загрузочного сектора*, в котором помимо собственно загрузчика располагается таблица, описывающая все параметры диска. После загрузочного сектора находятся одна или несколько копий *таблицы размещения файлов* (FAT), *корневой каталог* (root) и собственно *область данных*. Между загрузочным сектором и первой копией FAT может находиться некоторое количество зарезервированных секторов.

Загрузчик является короткой программой, загружающей операционную систему или только ее ядро; кроме того, он может быть инструментом выбора загружаемой ОС (boot manager). В отличие от инвариантного главного загрузчика, этот загрузчик привязан к своей ОС и записывается на диск при форматировании данного диска средствами этой ОС.

Инициализация областей логического диска называется *форматированием верхнего уровня* (низкоуровневое форматирование — это формирование наборов секторов и их заголовков на каждом треке). Форматирование (верхнего уровня) выполняется утилитой FORMAT или иным средством операционной системы. Дискета форматируется сразу целиком; на жестком диске каждый раздел,

содержащий логический диск, форматируется отдельно. До форматирования жесткий диск должен быть сконфигурирован (разбит на разделы).

Область данных диска разбита на *кластеры* (cluster) — группы смежных секторов, называемые также *единицами распределения пространства* (allocation unit). Каждый кластер имеет свой номер; размер кластера (число секторов) выбирается кратным степени двойки в зависимости от объема диска и размера FAT. Файл на диске занимает целое число кластеров, от одного до всех кластеров, входящих в область данных. Если файл занимает более одного кластера, то все занятые кластеры организуются в *цепочку кластеров* (cluster chain). Количество файлов на диске не может превышать количества кластеров (элементов FAT). Обращения к файлам, занимающим цепочку смежных кластеров, выполняются гораздо быстрее, чем к файлам, у которых кластеры раскиданы по всему диску, — меньше времени тратится на позиционирование головок. Файлы, которые располагаются в цепочках из несмежных кластеров, называются *фрагментированными*. Соответственно, процедура наведения порядка на диске, повышающая производительность файловой системы ОС и компьютера в целом, называется *дефрагментацией диска*. Специальные утилиты дефрагментации (SPEDDISK, DEFRAG) занимаются тем, что разрозненные фрагменты файлов собирают в единую, по возможности непрерывную цепь смежных кластеров.

Помимо свободных, занятых и плохих кластеров на дисках могут образовываться *потерянные кластеры* (lost clusters). Это отдельные кластеры или даже цепочки, помеченные как занятые, но не принадлежащие ни одному из файлов (на них нет ссылок из элементов каталога). Их происхождение легко объяснимо: если при записи нового файла происходит внезапное отключение питания или аппаратный сбой, может оказаться, что в FAT уже внесены изменения (элементы уже заняты), а в каталог новый элемент со ссылкой на начало цепочки не внесен. Эти «бесхозные» кластеры уже не могут использоваться ОС, они просто «съедают» доступное дисковое пространство. Поиском потерянных кластеров занимаются специальные утилиты, например NDD, SCANDISK. Найденные «бесхозные» цепочки они предлагают либо пометить как свободные кластеры, либо преобразовать в файлы (дать на них ссылку из корневого каталога). Эти файлы иногда содержат ценную информацию, которую можно использовать для восстановления пропавших данных, но чаще их просто удаляют, освобождая место на диске. В системе FAT могут встречаться и иные ошибки, например пересечения цепочек кластеров. Ошибки в файловой системе обычно происходят из-за неисправностей в любом из компонентов тракта «память — контроллер — диск» или в связывающих их шинах. «Лечение» этих ошибок выполняют специальные утилиты (опять же NDD, SCANDISK), которые пользуются информацией копий FAT (на диске их, как правило, не менее двух) и элементов каталогов.

Подробнее системы FAT, FAT16 и FAT32 рассмотрены в [6]. Конечно же, есть и другие файловые системы, более сложные, защищенные и эффективные на больших объемах дисков и при большом количестве файлов. К ним можно отнести HPFS (OS/2), NTFS (Windows NT), Nowell NetWare, Unix, Linux и некоторые другие. Они построены иначе, используют другие механизмы распреде

ления дискового пространства. Отметим, что за полезные свойства часто приходится расплачиваться большим расходом системных ресурсов (особенно оперативной памяти). Существуют средства как в составе ОС, так и от сторонних производителей, предназначенные для взаимных преобразований файловых систем.

Возможность работы ОС с диском, созданным не ее средствами, зависит от типа файловой системы, типа раздела, на котором он расположен, и размера раздела. Так, диск с файловой системой FAT16 в Windows NT поддерживается на разделах типов 04, 05 и 06, но не поддерживается на разделах типов 0E и 0F, а FAT32 не поддерживается ни на каких.

Для повышения эффективности использования дисковой памяти иногда применяют *дисковые компрессоры* — программные средства, сжимающие данные на диске «прозрачно» для приложений (и пользователя). Каждый раз при записи файла (или его фрагмента) выполняется компрессия, при чтении — декомпрессия. Конечно, для исполнения в реальном времени пригодны не всякие алгоритмы компрессии, и ради экономии времени жертвуют достижимой степенью сжатия. Возможность сжатия заложена в такие сложные файловые системы, как Novell NetWare (начиная с версий 4.x) и NTFS. Для файловой системы FAT (MS-DOS и Windows 9x) встроенных компрессоров не предусмотрено, но с ними широко используются загружаемые компрессоры типа Stacker, DoubleSpace и DriveSpace. Идея этих компрессоров заключается в следующем. На обычном логическом диске, называемом *несущим,* размещается большой файл- образ сжатого диска (Compressed Volume File, CVF). В этом образе есть своя система каталогов и таблица размещения файлов. Во время загрузки (до исполнения команд файла CONFIG.SYS) в оперативную память помещается специальный резидентный драйвер из файла DBLSPACE.BIN (или DRVSPACE.BIN), находящегося в корневом каталоге загрузочного диска. Этот драйвер эмулирует обращения к реальному диску операциями доступа к файлу-образу, на ходу осуществляя компрессию/декомпрессию. Для ОС эмулируемый диск выглядит как обычный логический диск, и для удобства пользователя ему может назначаться логическое имя (буква), ранее принадлежавшее несущему диску. Несущий диск при этом получает новое (ранее неиспользованное) имя, и к нему, в принципе, тоже можно обращаться обычным способом. Несущий диск можно скрыть от приложений и пользователя (чтобы не было поползновений удалить «никому не нужный» громадный файл-образ). На одном несущем диске может размещаться несколько файлов-образов, каждый из которых представляет свой сжатый логический диск (том).

9.5. Устройства хранения на магнитных дисках

Накопитель на магнитных дисках удобно использовать для иллюстрации принципов работы устройств хранения, принципов взаимодействия с такими устройствами, а также функций, выполняемых их контроллерами. Иные устройства

с подвижными носителями можно рассматривать как вариации накопителя на магнитных дисках: возможны другие принципы функционирования и конструкции головок чтения-записи (оптические с лазерами и фотоприемниками), нюансы привода носителей (например, переменная скорость и стартстопный режим), особенности системы позиционирования.

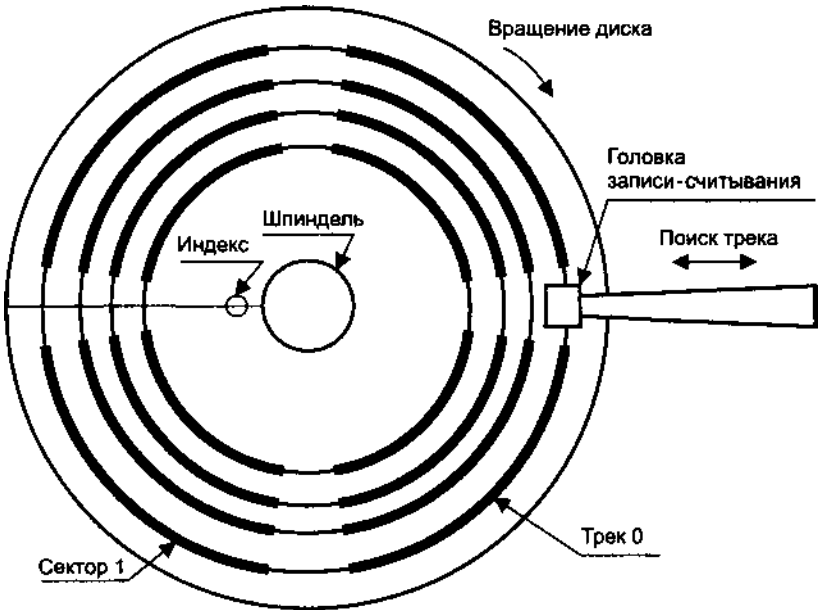


Рис. 9.4. Устройство дискового накопителя

Схематически устройство классического дискового накопителя представлено на рис. 9.4. Носителем информации является диск (один или несколько), на который нанесен слой вещества, способного намагничиваться (чаще всего ферромагнитный). Хранимую информацию представляет состояние намагниченности отдельных участков рабочей поверхности. Диски вращаются с помощью *двигателя шпинделя* (spindle motor), обеспечивающего требуемую частоту вращения в рабочем режиме. На диске имеется *индексный маркер*, который, проходя мимо специального датчика, отмечает начало каждого оборота диска. Информация на диске располагается на концентрических *треках* (дорожках), нумерация которых начинается с внешнего трека (track 00). Каждый трек разбит на *секторы* (sector) фиксированного размера. Сектор и является минимальным блоком информации, который может быть записан на диск или считан с него. Нумерация секторов начинается с единицы и привязывается к индексному маркеру. Если накопитель имеет несколько рабочих поверхностей (на шпинделе может быть размещен пакет дисков, а у каждого диска могут использоваться обе поверхности), то совокупность всех треков с одинаковыми номерами составляет *цилиндр* (cylinder). Для каждой рабочей поверхности в накопителе имеется своя *головка* (head), обеспечивающая запись и считывание информации. Головки нумеруют

ся, начиная с нуля. Для того чтобы произвести элементарную операцию обмена — запись или чтение сектора, — шпиндель должен вращаться с заданной скоростью, блок головок должен быть подведен к требуемому цилиндру, и только когда нужный сектор подойдет к выбранной головке, начнется физическая операция обмена «полезными» данными между головкой и блоком электроники накопителя. Кроме того, головки считывают служебную информацию (адресную и сервисную), позволяющую определить и установить их текущее местоположение. Для записи информации на носитель используются различные *методы модуляции*, позволяющие кодировать двоичную информацию, намагничивая зоны магнитного слоя, проходящие под головкой. При считывании намагниченные зоны наводят в головке электрический сигнал, из которого декодируется ранее записанная информация. *Контроллер накопителя* выполняет сборку и разборку блоков информации (секторов или целых треков), включая формирование и проверку контрольных кодов, осуществляет модуляцию и демодуляцию сигналов головок и управляет всеми механизмами накопителя.

Несмотря на кажущуюся простоту конструкции, записать и потом достоверно считать информацию с диска не так-то просто. Для записи данных необходимо сформировать последовательный код, который должен быть самосинхронизи- рующимся: при последующем считывании из него должны извлекаться и данные, и синхросигнал, что позволяет восстановить записанную цепочку битов (этим занимается сепаратор данных — узел дискового контроллера). Кроме того, напомним, что индуктивные считывающие головки воспринимают только изменение намагниченности участков трека. Также учтем, что физическое исполнение — магнитные свойства носителя, конструкция головок, скорость движения, «высота полета» головок и т. п. — задает предельно достижимую плотность изменения состояния намагниченности (flux density), которую хотелось бы использовать максимально эффективно. Эта плотность измеряется количеством зон с различным состоянием намагниченности на дюйм длины трека — *FCI* (Flux Changes per Inch — изменений потока на дюйм) и в современных накопителях достигает десятков тысяч. Для записи на диск применяют различные *схемы кодирования* (data encoding scheme), различающиеся сложностью реализации и эффективностью работы. В первых моделях накопителей использовалась *частотная модуляция* (Frequency Modulation, FM), при которой для каждого бита данных на треке отводится ячейка с окнами для представления бита и синхросигнала, что весьма неэффективно расходует величину FCI. Более эффективна *модифицированная частотная модуляция* (Modified Frequency Modulation, MFM), при которой синхросигнал вводится только в процессе кодирования смежных нулевых битов, что позволяет удвоить плотность записи при той же плотности изменения потока. Обе схемы (FM и MFM) являются схемами с побитным кодированием. Более эффективны схемы группового кодирования, в которых цепочка байтов данных (сектор) предварительно разбивается на группы по несколько битов, кодирующихся по определенным правилам. Схема кодирования *RLL* (Run-Length Limited encoding — кодирование с ограничением длины серий), как это следует из названия, построена на ограничении длины неперемагничиваемых участков трека. Наиболее популярна схема RLL 2.7 — в ней число неперемагничиваемых ячеек лежит в диапазоне от 2 до 7. Для

накопителей с высокой плотностью используется схема RLL 1.7, обеспечивающая большую надежность считывания. Существует и схема ARLL (Advanced RLL) — малораспространенный вариант схемы RLL 3.9. Схемы RLL стали работоспособными только при определенном уровне качества (стабильности характеристик), достигнутом в области технологии создания магнитных накопителей. По этим схемам происходят упаковка данных и исключение избыточных синхросигналов. Кстати сказать, FM и MFM являются разновидностями RLL: схема FM эквивалентна RLL 0.1; MFM — RLL 1.3. Соотношение полезной плотности записи BPI (Bit Per Inch — битов на дюйм) при одинаковой плотности FCI в популярных схемах кодирования следующее:

FM : MFM : RLL 1.7 : RLL 2.7 = 1 : 2 : 2,54 : 3.

Из-за того что линейная скорость носителя относительно головки на внутренних цилиндрах меньше, чем на внешних, для нормальной записи при меньшей скорости приходится применять предварительную компенсацию записи. Для жестких дисков в CMOS Setup имеется параметр WPcom (Write Precompensation) — номер цилиндра, начиная с которого контроллер должен вырабатывать сигнал предварительной компенсации. Для накопителей со встроенным контроллером этот параметр игнорируется, поскольку они сами «знают», как работать со своими дисками.

Информация на дисках записывается и считывается посекторно, и каждый сектор имеет определенную *структуру* (формат). Не слишком вдаваясь в подробности, отметим, что в начале каждого сектора имеется заголовок, за которым следует поле данных и поле контрольного кода. В *заголовке* имеется поле идентификатора, включающее номер цилиндра, головки и собственно сектора. В этом же идентификаторе может содержаться и пометка о дефектности сектора, служащая указанием на невозможность его использования для хранения данных. Достоверность поля идентификатора проверяется с помощью контрольного кода заголовка. Заголовки секторов записываются только в ходе низкоуровневого форматирования, причем для всего трека сразу. При обращении к сектору по чтению или записи заголовок только считывается. *Поле данных* сектора отделено от заголовка небольшим зазором (gap), необходимым для того, чтобы при записи головка (точнее, обслуживающая ее схема) могла успеть переключиться из режима чтения (заголовка) в режим записи (данных). Сектор завершается *контрольным кодом поля данных — CRC* (Cyclic Redundancy Check — контроль с помощью циклического избыточного кода) или *ECC* (Error Checking and Correcting — обнаружение и коррекция ошибок). CRC-код позволяет только обнаруживать ошибки, а ECC-код — еще и исправлять ошибки небольшой кратности. В межсекторных промежутках может размещаться сервоинформация, служащая для точного наведения головки на трек.

Для того чтобы диск можно было использовать для записи и считывания информации, он должен быть *отформатирован*. Форматирование может разделяться на два уровня:

- Низкоуровневое форматирование (Low-Level Formatting, LLF) — формирование заголовков и пустых (размеченных заполнителем) полей данных всех секторов всех треков. При форматировании выполняется и верификация

(проверка читабельности) каждого сектора, и в случае обнаружения неисправимых ошибок считывания в заголовке сектора делается пометка о его дефектности.

- Форматирование верхнего уровня заключается в формировании логической структуры диска (таблиц размещения файлов, корневого каталога и т. п.),. соответствующей файловой подсистеме применяемой ОС. Эта процедура выполнима только после низкоуровневого форматирования.

Итак, структура трека — последовательность секторов — задается при его форматировании, а начало трека определяется контроллером по сигналу от индексного датчика или иным способом. Нумерация секторов, которая задается контроллеру при форматировании, может быть произвольной — важно лишь, чтобы все секторы трека имели уникальные номера в пределах допустимого диапазона. При обращении к сектору он ищется по идентификатору, а если за оборот диска (или за несколько оборотов) сектор с указанным номером не обнаруживается, контроллер фиксирует ошибку (Sector Not Found — сектор не найден). Забота о поиске сектора по его заголовку, помещении в его поле данных записываемой информации, снабженной контрольным кодом, а также считывании этой информации и ее проверке с помощью CRC- или ECC-кода лежит на контроллере накопителя. И, конечно же, контроллер управляет поиском затребованного цилиндра и коммутацией головок, выбирая нужный трек.

Современные жесткие диски внутренне могут быть организованы несколько иначе, чем в вышеописанной схеме. Индексные датчики теперь не используются — начало трека определяется по считываемому сигналу. Физическая разбивка на секторы (по 512 байтов данных, которым предшествует идентификатор) может отсутствовать — группа секторов трека представляет собой единый битовый поток, защищенный избыточным кодированием, из которого вычисляется блок данных, находящийся в требуемой позиции (так называемый ID-less format). Для обеспечения достоверности хранения данных (исправления ошибок) применяются избыточные коды Рида - Соломона (Reed - Solomon code), позволяющие большинство ошибок исправлять «на лету», не требуя повторного считывания блока данных (и дополнительного оборота диска). Заметим, что заметная вероятность искажения информации свойственна любым носителям информации, в том числе твердотельным.

Накопители на гибких магнитных дисках

Накопители на гибких магнитных дисках (НГМД), или дискетах, применялись с первых моделей PC, у которых они были единственным средством хранения и переноса информации. С тех пор эти устройства претерпели относительно небольшие изменения — размер дискеты уменьшился почти в два раза, а емкость возросла всего на порядок. По сравнению с другими компонентами прогресс невелик. Английское сокращенное название НГМД — *FDD* (Flexible, или Floppy, Disk Drive).

Устройство НГМД полностью соответствует схеме на рис. 9.4. Носителем информации является гибкий майларовый диск (дискета), на который нанесен

ферромагнитный слой. В настоящее время дискеты формата 5[м] практически не используют (рис. 9.5, а), а используют (все меньше) дискеты формата 3,5", заключенные в жесткий пластмассовый конверт (рис. 9.5, *б)*. Помимо более высокой, чем у дискет 5", плотности хранения информации, они лучше защищены от внешних воздействий (пыли и деформации). Все накопители 3,5" имеют 80 треков. По продольной плотности существуют три градации, обеспечивающие хранение 9, 18 или 36 секторов на треке в стандартном режиме форматирования с емкостью 720 Кбайт, 1,44 и 2,88 Мбайт соответственно. В настоящее время наиболее распространенными являются накопители 3,5" и дискеты с форматированной (для PC) емкостью 1,44 Мбайт. Для получения емкости 2,88 Мбайт применяют так называемую перпендикулярную, или вертикальную, запись с расположением доменов перпендикулярно плоскости дискеты, а не в плоскости, как при обычной записи. Перпендикулярная запись требует как специальных головок, так и специальных дискет. Устройства и дискеты на 2,88 Мбайт, поддерживаемые контроллерами большинства системных плат, широкого распространения не получили.

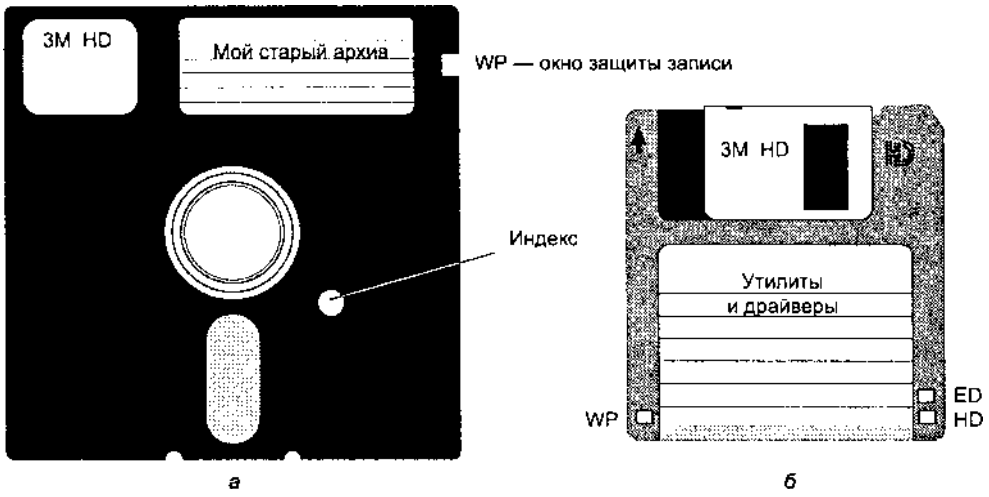


Рис. 9.5. Дискеты для НГМД: а — дискета формата 5", *б* — дискета формата 3,5"

Многофазные шпиндельные двигатели современных накопителей совместно с платой электроники автоматически поддерживают требуемую *частоту вращения* 360 об./мин[1]. Из-за невысокой стабильности частоты вращения, которую обеспечивали первые накопители, был принят формат трека с существенным запасом по числу секторов. Более точное поддержание частоты позволило увеличить число секторов (для обычной плотности 10 вместо 9 на трек) без риска «наползания» последнего сектора на первый при отклонении скорости вращения от номинальной (в сторону больших значений).

[1] 5-дюймовые дисководы работали на скорости 300 об./мин.

В качестве *привода позиционирования* головок на нужный цилиндр в НГМД применяют *шаговые двигатели.* Эти двигатели под действием серии импульсов, подаваемых на их обмотки, способны поворачивать свой вал на определенный угол. Этот угол кратен минимальному шагу, определяемому конструкцией двигателя. Поворот вала двигателя на один шаг приводит к перемещению блока головок на один цилиндр. С точки зрения теории автоматического управления, привод с шаговым двигателем является *разомкнутой системой* (то есть системой без обратной связи). Такая система не позволяет корректировать ошибки позиционирования, вызванные, например, температурным изменением размеров дисков. Конечно, при всех операциях обмена проверяется адресный маркер цилиндра, и в случае его несовпадения делается повторная попытка позиционирования — возврат к нулевому цилиндру и подача требуемого количества шаговых импульсов. При обращении к сбойным секторам дискеты эти повторные попытки, выполняемые драйвером НГМД, заметны по «рычанию», с которым устройство как будто бы «пилит» дискету.

Выход на нулевую дорожку определяется по *датчику нулевого цилиндра*, которым обычно является оптоэлектронная пара с флажком, связанным с блоком головок. Для накопителей со сменными носителями положение нулевого цилиндра существенно — чтобы обеспечить их совместимость, оно должно совпадать у всех устройств. Однако датчик задает положение нулевого цилиндра лишь грубо — он определяет только номер шага привода, на котором головки находятся напротив нулевого цилиндра. Более точно положение можно отрегулировать вращением корпуса шагового двигателя в пределах нескольких градусов (не больше, чем угловой шаг двигателя).

Головки записи-считывания — индуктивные. Головка с нулевым номером располагается снизу диска, первая головка — сверху. Головки несколько смещены относительно друг друга в радиальном направлении, так что «цилиндр» дискеты на самом деле больше похож на конус. В нерабочем положении головки подняты над поверхностью диска на несколько миллиметров, а в рабочем прижимаются к поверхности диска пружинами. При недостаточно сильном прижиме запись (особенно при высокой плотности) становится неустойчивой, при слишком сильном прижиме увеличивается износ головок и дискет.

В дисководах имеется несколько датчиков, которые могут быть как оптоэлектронными, так и механическими микровыключателями:

- Датчик индекса формирует выходной (для дисковода) импульс Index на каждый оборот диска. У дисководов 5" он оптоэлектронный, работает на просвет индексного отверстия в носителе. У дисководов 3,5" он магнитный, для него имеется отверстие в металлическом «пятачке» дискеты.
- Датчик защиты от записи, оптоэлектронный или механический, формирует выходной сигнал Wprot, когда на дискете 5" окошко заклеено, а на дискете 3,5" окошко открыто.
- Датчик нулевого трека, оптоэлектронный или механический, формирует выходной сигнал TR 00, когда головки достигают соответствующего положения (при движении от центра к краю).

- Датчик смены носителя (только у приводов HD) в момент установки дискеты вызывает срабатывание триггера с выработкой сигнала DC.
- Датчики типа дискеты (только у приводов 3,5") выходных сигналов не формируют. Датчик типа HD автоматически (независимо от интерфейсного сигнала Low Current) должен снижать ток записи, когда в привод HD установлена дискета QD. Датчик ED аналогичным образом задает специальный режим записи для дискет емкостью 2,88 Мбайт.

Размеры, форматы и правила использования дискет

Дискеты различаются по диаметру диска и плотности хранения информации, их параметры обычно входят в обозначение типа. По числу рабочих поверхностей раньше различали односторонние (Single Side, SS) и двусторонние (Double Side, 2S, или DS) дискеты. Поскольку односторонние дискеты давно уже не используются, на дискетах 3,5" от этого обозначения осталась лишь цифра 2. По *плотности* (density) *записи* различают следующие типы дискет 3,5[м]:

- *QD* (Quadr-Density) — емкость 720 Кбайт (устарели);
- *HD* (High-Density) — емкость 1,44 Мбайт (основной тип для AT);
- *ED* (Extra-High Density) — емкость 2,88 Мбайт (почти не используются).

Соотношения плотностей записи и форматированной емкости (в стандартном для PC режиме форматирования) приведены в табл. 9.5.

Таблица 9.5. Обозначение и емкость дискет 3,5"

Обозначение	Число треков	Число секторов	Поперечная плотность, TPI	Емкость (параметры команды FORMAT)
2QD	80	9	135	720 Кбайт (/T:80 /N:9)
2HD	80	18	135	1,44 Мбайт (/T:80 /N:18)
				1,6 Мбайт (/T:80 /N:20, нестандартный)
2ED	80	36	135	2,88 Мбайт (/T:80 /N:36)

На дискетах иногда пишут словосочетание *Soft Sector*, что указывает на наличие лишь одного индексного отверстия на диске, отмечающего начало трека. Положение остальных секторов определяется «мягким» способом — программой с помощью контроллера.

Перед использованием дискета должна быть отформатирована (дискеты часто продаются уже отформатированными). Стандартизация форматов записи на дискеты обеспечивает переносимость данных между компьютерами. На нижнем уровне для каждого типа дискет стандартный формат однозначен: размер сектора — 512 байт; на каждом треке секторы нумеруются последовательно, от 1 до числа секторов на треке (9 или 15 для дискет 5[м] и 9, 18 или 36 для дискет 3,5"); число цилиндров — 40 или 80. Контроллеры НГМД позволяют использовать и иной размер сектора — 256 или 1024 байт, варьировать число секторов на треке и их нумерацию, задействовать несколько дополнительных треков за «официально» последним. Помимо стандартных форматов, для повы-

шения надежности использующих поверхность диска не совсем полностью, распространены и некоторые другие форматы, доступные после загрузки драйверов типа 800.C0M и им подобных. Современные версии BIOS поддерживают нестандартные параметры (иногда только при чтении) и без дополнительных драйверов. Наиболее популярным (в разных ОС) форматом верхнего уровня для дискет является файловая система FAT-12, пришедшая из MS-DOS.

Форматирование дискет выполняется утилитой FORMAT или другими средствами, предоставляемыми операционной системой или прикладным ПО. Утилита FORMAT обеспечивает форматирование нижнего и верхнего уровней, а начиная с MS-DOS 5, предварительно анализирует и сохраняет существующий формат, что позволяет отменить форматирование. Подавить анализ существующего формата можно ключом /и (Unconditional formatting — безусловное форматирование), что экономит время (но не позволяет восстановить затертые данные). Параметрами нижнего уровня управляют ключи /T и /N, задающие количество треков и секторов соответственно. По умолчанию низкоуровневое форматирование не выполняется — вместо него производится только верификация секторов (холостое считывание), что несколько быстрее. Низкоуровневое форматирование выполняется лишь при безусловном форматировании (ключ /U) либо на неформатированной дискете. Оно может быть и пропущено по ключу /Q (Quick formatting — быстрое форматирование), при этом только очищаются таблица FAT и корневой каталог, что гораздо быстрее.

Нестандартное форматирование применяют в разных целях. В приведенных в таблице форматах емкость повышается благодаря увеличению числа секторов на треке. Кроме того, иногда задействуют треки, следующие за официально последним. Большинство дисководов позволяют головкам «прошагать» более чем 80 (или 40) треков, а на дискетах рабочий слой нанесен с запасом. Дополнительные треки в основном используют не для повышения емкости, а для создания ключевых *дискет, защищенных от копирования.* Стандартные программы копирования дискет не будут копировать эти лишние треки, что и обеспечивает защиту от наивных пиратов. Естественно, существуют программы копирования, учитывающие этот трюк.

Более хитрый способ защиты заключается в нестандартном форматировании треков. Здесь можно использовать секторы с необычными номерами, нестандартное количество секторов на треке, нестандартный размер сектора и проверку последовательности номеров секторов. Номера секторов, имеющихся на каждом треке, и их количество программа пиратского копирования может определить последовательными попытками чтения всех возможных номеров. Поскольку номеров не так уж много (не более 255), эта процедура займет обозримое время. Несколько труднее «расколоть» ключ, основанный на последовательности номеров секторов. Она может проверяться измерением задержки между появлением данных, прочитанных из секторов с определенными номерами, — эти измерения нетрудно выполнить программным способом. Однако производительности компьютеров давно уже хватает для того, чтобы выполнить полный переучет секторов на треке. Для этого следует быстро несколько раз подряд подавать команду чтения идентификаторов (Read ID) контроллеру

НГМД — так будет получена последовательность номеров и размеров всех секторов на треке (этот прием подходит и для идентификации легальной дискеты, и для пиратского копирования). Допускаются также использование секторов, специально помеченных как удаленные, и прочие ухищрения. Но, конечно же, при определенных знаниях и умениях подбор любого ключа — вопрос только времени.

Дискеты HD и ED имеют более высокую коэрцитивную силу (магнитное поле, необходимое для их перемагничивания), поэтому в режимах, ориентированных на DD и QD (и на соответствующих дисководах), их отформатировать не удастся. Дискеты DD и QD на дисководах HD и в режиме HD форматироваться кое-как будут, но с большим количеством дефектных блоков и низкой надежностью хранения. У дискет 3” имеются ключевые отверстия, по которым датчик накопителя может определить тип носителя (см. рис. 9.5, *б)*. У дискет 3" QD отверстий нет, что провоцирует любителей экспериментов «увеличить» их емкость, просверлив дырочку. Однако по причине различия коэрцитивной силы надежность записи и хранения информации на таких дискетах вызывает большие сомнения.

Дискеты имеют *ключ защиты от записи.* У дискет 3" есть выдвижная шторка, перекрытое окно разрешает запись. Защита дискет может и не сработать, если провод 28 интерфейсного шлейфа оборван.

Вне зависимости от типа все дискеты в той или иной степени критичны к воздействию сильных внешних магнитных полей. Вопреки расхожему заблуждению, рентгеновское облучение в смотровых аппаратах аэропортов для дискет безопасно. Магнитные металлоискатели, через которые проходят пассажиры, тоже не могут причинить вреда, поскольку напряженность создаваемого ими магнитного поля невелика (иначе «испортились» бы проходящие люди).

При сильном охлаждении (переносе по морозу) перед использованием дискетам нужно дать отогреться до комнатной температуры — чтобы войти в норму по размерам и хрупкости.

При записи на дискету данных, которые не хотелось бы потерять (бывают ли другие?), есть смысл включать *верификацию записи.* И в MS-DOS, и в Windows 9x (в окне MS-DOS) работает команда VERIFY ON, включающая контрольное считывание после каждой записи. Конечно, при этом запись идет несколько дольше, но риск больших потерь снижается. Чтобы впоследствии верификация не тормозила процесс записи на жесткий диск, имеется обратная команда VERIFY OFF, которую вводят после завершения записи файлов на дискету.

Интерфейс НГМД

На плате электроники, установленной на корпусе НГМД, расположены только схемы управления двигателями, усилители-формирователи сигналов записи/ считывания и формирователи сигналов от датчиков. Контроллер гибких дисков обычно размещается на системной плате компьютера или же вынесен на специальную карту расширения («в компании» с интерфейсом НЖМД). НГМД под

ключается к контроллеру через специальный стандартный интерфейс. Все сигналы являются логическими с уровнями ТТЛ, активный уровень — низкий.

Логически интерфейс довольно прост. Для того чтобы заставить накопитель работать, его нужно выбрать сигналом Drive Sel и запустить мотор шпинделя сигналом Motor On. Для выборки накопитель имеет четыре сигнала DS0...DS3, но отзывается только на один из них, определенный установкой джамперов. Только выбранный накопитель воспринимает управляющие сигналы от контроллера и передает контроллеру свои выходные сигналы. О том, что накопитель выбран, свидетельствует светодиодный индикатор на его лицевой панели.

Для перемещения головок на один шаг контроллер должен подать импульс Step, направление перемещения определяется уровнем сигнала Direction: при низком уровне (сигнал активен) перемещение происходит в сторону центра диска (номер трека увеличивается). Нулевой трек контроллер находит, перемещая головки от центра до появления сигнала Track 00. Выбор номера головки производится сигналом Side 1. Начало трека накопитель отмечает импульсом Index, который вырабатывается при прохождении индексного отверстия вращающейся дискеты мимо датчика. Считываемые данные в закодированном (MFM) виде (но усиленные и сформированные в ТТЛ-сигнал) поступают от накопителя по линии Read Data. Для включения режима записи служит сигнал Write Gate, закодированные данные в цифровом виде поступают от контроллера по линии Write Data. Если установлена дискета, защищенная от записи, накопитель сообщит об этом сигналом Write Protect. Для снижения тока записи, что требуется при работе накопителей HD с дискетами DD и QD, предназначен сигнал Reduce Write, его иное название Low Current, или FDHDIN. Для переключения головок на «вертикальную запись» служит сигнал FDEDIN. Оба эти сигнала вырабатываются контроллером, но для самого дисковода они дублируются сигналами от датчиков типа дискеты (сигнал FDEDIN не обязателен, дисковод сам переключится по сигналу от датчика). Некоторые модели дисководов позволяют настраивать режим работы датчиков типа дискеты: вместо вышеописанного режима работы, принятого для PC-совместимых ПК, они могут быть отключены или работать на информирование контроллера. Однако практически все контроллеры сами управляют линиями интерфейса, соответствующими сигналам от этих датчиков. В этом управлении учитываются тип дисковода, описанный в CMOS Setup, и заказанный формат дискеты. Сигнал Reduce Write (низкий уровень) формируется контроллером при любом обращении к дисководу, описанному в CMOS как HD (1,2 или 1,44 Мбайт), для работы с дискетами DD или QD (360 или 720 Кбайт).

Накопители HD при *смене дискеты* устанавливают сигнал Disk Changed, который сбрасывается после обращения к этому накопителю. Этот сигнал заслуживает особого внимания. Он имеется только у дисководов HD и ED, причем его использование может определяться джамперами дисковода. В PC соответствующий джампер устанавливается в положение *DC* (Disk Change). Альтернативное применение этой линии — сигнализация готовности устройства, что может обозначаться как *RY, RDY* или *SR* — для PC непригодно.

Заметим, что в интерфейсе нет никаких сигналов, прямо информирующих контроллер о готовности — наличии установленной дискеты. Контроллер может определить готовность, лишь выбрав накопитель и запустив мотор. Тогда отсутствие импульсов Index означает неготовность — отсутствие дискеты или ее фиксации на шпинделе или же неподключенность дисковода (интерфейса или питания). Наличие дисковода контроллер может определить с помощью команды рекалибровки (см. далее) — при ее выполнении дисковод должен подать сигнал Track00.

Все НГМД, применяемые в PC, независимо от типа и размера имеют одинаковый интерфейс и *унифицированные 34-контактные разъемы* двух типов: с печатными двусторонними ламелями у устройств 5" и двухрядными штырьковыми контактами у устройств 3,5[м]. Используемый в PC кабель-шлейф имеет перевернутый фрагмент из 7 проводов с номерами 10-16 (рис. 9.6). Этот поворот позволяет подключать к контроллеру одним шлейфом до двух НГМД, причем адрес накопителя определяется его положением на шлейфе: для привода А фрагмент перевернут, для В — нет. Универсальный шлейф с пятью разъемами, изображенный на рисунке, позволяет подключить пару любых дисководов, которые должны располагаться в разных зонах шлейфа. Табл. 9.6 описывает интерфейсный кабель; здесь показано, как сигналы приходят на разные накопители. Направление сигналов (I/O — ввод-вывод) указано относительно контроллера. Нечетные контакты 1-33 — земля. Для дисководов 5" ключ располагается между контактами 4-5 и 6-7.

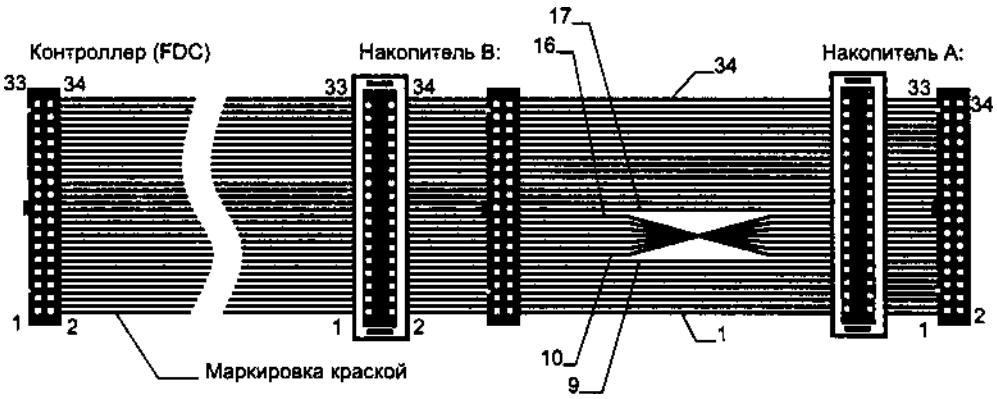


Рис. 9.6. Кабель интерфейса НГМД

Таблица 9.6. Кабель интерфейса НГМД

Контроллер			Дисковод В		Дисковод А	
Контакт	**Сигнал**	**I/O**	**Контакт**	**Сигнал**	**Контакт**	**Сигнал**
2	FDHDIN (Reduce Write)	O	2	Low Current	2	Low Current
4	Резерв	–	4	Резерв	4	Резерв

Контроллер			Дисковод B		Дисковод A	
Контакт	Сигнал	I/O	Контакт	Сигнал	Контакт	Сигнал
6	FDEDEIN	–	6	FDEDIN (DS3)	6	FDEDIN (DS3)
8	Index	I	8	Index	8	Index
10	Motor On A	O	10	DS0	16	Motor[1]
12	Drive Sel 1	O	12	DS1[1]	14	DS2
14	Drive Sel 0	O	14	DS2	12	DS1[1]
16	Motor On B	O	16	Motor[1]	10	DS0
18	Direction	O	18	Direction	18	Direction
20	Step	O	20	Step	20	Step
22	Write Data	O	22	WData	22	Wdata
24	Write Gate	O	24	WGate	24	Wgate
26	Track 00	I	26	TR 00	26	TR 00
28	Write Protect	I	28	WProt	28	WProt
30	Read Data	I	30	RData	30	Rdata
32	Side 1	O	32	Side 1	32	Side 1
34[2]	Disk Changed	I	34	DC	34	DC

[1] Пара сигналов, обеспечивающая выборку FDD (Motor On A и Drive Sel 0 для дисковода A и Motor On B и Drive Sel 1 для дисковода B).
[2] Контакт 34 в XT не используется.

Контроллер НГМД и интерфейсный кабель, принятый в PC, позволяют адресоваться к одному из двух накопителей и включать мотор сигналами Drive Sel O и Motor On A для накопителя A и сигналами Drive Sel 1 и Motor On B для накопителя B. При этом на обоих накопителях джамперы устанавливаются так, что они отзываются на сигнал DS 1 (контакт 12 разъема). Обычно джамперы на дисководе обозначаются *DS0, DS1, DS2, DS3*, а установить следует джампер *DS1*. Если джамперы обозначаются как *DS1, DS2, DS3, DS4,* что встречается нечасто, то следует установить джампер *DS2*. Принятая система выборки позволяет все дисководы конфигурировать однотипно, а адрес задавать положением на шлейфе. В некоторых специфических клонах PC задействуют иную систему выборки накопителей и «прямой» кабель-шлейф. При этом используется выборка устройства сигналом DS0, но переключение выборки на эту линию некоторыми накопителями не поддерживается, в результате замена накопителей в этих «фирменных» машинах может стать хлопотным делом, особенно при отсутствии технической документации.

Контроллеры НГМД

Контроллер накопителей на гибких дисках (Floppy Drive Controller, FDC) всегда является внешним по отношению к накопителю. Все контроллеры обеспечивают совместимость с «исторической» микросхемой контроллера NEC PD765, которая является аналогом i8272. Контроллер *FDC AT* поддерживает

два накопителя; помимо НГМД он позволяет работать со стримерами (старыми с низкой скоростью передачи). Для трехдюймовых дисководов контроллер поддерживает скорости записи/считывания 300, 500 и 1000 кбит/с; соответствующие режимы обозначаются как 1M (QD), 2M (HD) и 4M (EHD). В карте ресурсов AT имеется место под два контроллера НГМД:

- контроллер FDC AT#1 (стандартный или основной) занимает порты с адресами 3F0h-3F7h (как и FDC XT);
- контроллер FDC AT#2 (дополнительный) занимает порты с адресами 370h-377h.

Контроллеры вырабатывают запрос *аппаратного прерывания* IRQ6 (в BIOS — прерывание Int 0Eh) по окончании выполнения внутренних операций. Для обмена данными может использоваться канал DMA2.

Контроллер НГМД поддерживается дисковым сервисом BIOS Int 13h (см. 9.11). С контроллером общается и обработчик аппаратного прерывания от таймера IRQ0 (BIOS Int 08h), который декрементирует счетчик времени работы мотора НГМД (BIOS Data Area, ячейка 0:0440) и по его обнулении отключает мотор. Адрес дисковода, мотор которого включен, в позиционном коде содержит ячейка памяти 0:043F. При каждом обращении к дискете в регистре контроллера устанавливается бит включения мотора и в счетчик времени заносится константа, соответствующая выдержке на отключение (по умолчанию — 2 с). Таким образом, если в течение этого интервала нет следующих обращений, мотор автоматически выключается (если BIOS отрабатывает аппаратные прерывания от таймера). Не отключающийся мотор может быть косвенным признаком «зависания» компьютера.

Программирование контроллера НГМД для операций с дискетами — довольно хлопотное занятие, и игнорирование сервисов BIOS и даже ОС оправдано в основном лишь для нетривиальных задач наподобие работы с ключевыми дискетами и т. п. Программный интерфейс контроллера подробно описан в [6]; правда, в описании команды чтения идентификаторов в данной книге допущена неточность: эта команда считывает координаты и код размера из первого успешно считанного идентификатора адреса сектора — с ее помощью можно произвести полный переучет всех секторов на каждом треке.

Проблемы с накопителями на гибких дисках

Установленные (и используемые) дисководы должны быть корректно описаны в CMOS Setup. *Типы установленных дисководов* задаются стандартными параметрами CMOS Setup и хранятся в памяти CMOS. Установка неверного типа приводит к ошибкам обращения к дисководам.

Когда в дисковод устанавливается отформатированная дискета, BIOS определяет плотность записи (ищет скорость передачи данных, на которых дискета читается), читает ее первый сектор, определяет по нему формат и строит таблицу параметров дискеты. Эти параметры используются при дальнейших обращениях к дискете (до ее смены). При форматировании совсем чистой дискеты, когда первый сектор прочитать не удается, по умолчанию формат определяется

по типу дисковода, описанного в CMOS. Если носитель не соответствует этому формату, форматирование завершается аварийно (с сообщением «Track 0 Bad») или с получением большого числа сбойных блоков. Формат, отличающийся от стандартного для данного типа привода, можно задать принудительно ключами /T и /N (см. табл. 9.5).

При отказе датчика смены носителя (обычно это микровыключатель), обрыве в провода в 34-м шлейфе (смена носителя) или неправильном конфигурировании дисковода (установкой джампера или типа в CMOS) возникают «фантомные» каталоги: после смены дискеты система видит каталог предыдущей и при записи, естественно, испортит данные. Это явление объясняется тем, что, не получив сигнала смены носителя, операционная система пользуется копией каталога предыдущей дискеты, хранящейся в буфере (в ОЗУ). На скорую руку от таких «призраков», часто принимаемых за действие вируса, в среде DOS можно спасаться нажатием комбинации Ctrl+C (Ctrl+Break) при каждой смене диска, что приводит к принудительной очистке буферов. «Фантомы» могут быть вызваны и ошибочным заданием типа дисковода.

При работе с дискетами приходится сталкиваться с проблемой *совместимости накопителей*. Несовместимость главным образом обусловлена неточностью позиционирования: как ошибкой определения нулевого трека, так и отклонением шага треков от номинального. Другое чувствительное место накопителей — *поверхность головок*. Ее загрязнение первым делом приводит к ухудшению параметров записи (поскольку этот процесс более критичен к зазору между головкой и носителем, чем процесс чтения).

Не следует использовать дискеты с помятым (и порванным) носителем — они могут повредить головки.

Накопители на жестких магнитных дисках — винчестеры

Накопители на жестких магнитных дисках (НЖМД), они же HDD (Hard Disk Drive), являются главными устройствами дисковой памяти большинства компьютеров. По случайному совпадению цифр в названии первой модели НЖМД окрестили «винчестером» (просто игра слов), и это неофициальное название закрепилось в качестве синонима терминов HDD и НЖМД. Наряду с процессором и оперативной памятью винчестер определяет мощность компьютера. От него требуются большой объем хранимой информации (десятки и сотни гигабайт), малое время доступа (единицы миллисекунд), большая скорость передачи данных (десятки — сотни мегабайт в секунду), высокая надежность, умеренная стоимость и ряд других полезных свойств. Прогресс в области производства винчестеров устойчивый и стремительный: от 10-мегабайтного диска XT уже пришли к сотням гигабайт, скорость передачи данных возросла на три порядка. «Модная» емкость винчестера ПК год от года увеличивается, при этом цена устройства постепенно снижается.

Конструкция НЖМД

Принципиально конструкция НЖМД соответствует рис. 9.3. Вся электромеханическая часть накопителя — пакет дисков со шпиндельным двигателем и блок головок с приводом — находится в *гермоблоке.* Англоязычное сокращенное название этой сборки — *HDA* (Head Disk Assembly — диск с головками в сборке). На корпусе гермоблока размещается и плата электроники накопителя.

В качестве *привода шпинделя* используют, как правило, трехфазные синхронные двигатели. Схема управления двигателем обеспечивает пуск и останов шпинделя, а также поддерживает требуемую скорость с довольно высокой точностью. Шпиндельный двигатель является основным потребителем мощности по шине + 12 В. В качестве датчика скорости вращения у старых винчестеров использовались сигналы индексного датчика, в современных винчестерах для точного управления шпиндельным двигателем служит информация сервометок. *Скорость вращения* долгие годы держалась на уровне 3600 об./мин, теперь же чаще встречается скорость 4500, 5400 и 7200 об./мин. Самые быстрые винчестеры имеют скорость 10 000 и даже 15 000 об./мин. Чем выше скорость вращения, тем больше скорость обмена информацией с диском. Однако высокие скорости вращения порождают проблемы, связанные с балансировкой, тепловыделением, гироскопическим эффектом и аэродинамикой головок. Трение воздуха о пластины при скорости вращения 10 000 об./мин и выше[1] является основной причиной нагрева винчестеров, заставляя заботиться об их охлаждении. Из-за гироскопического эффекта не рекомендуется перемещение (точнее, смена ориентации оси шпинделя) включенных накопителей с вращающимся шпинделем. Накопители для портативных компьютеров разрабатываются с учетом этих эффектов.

Пластины (platter) жестких дисков обычно изготавливают из алюминиевых сплавов, иногда из керамики или стекла. *Рабочий магнитный слой* основан на оксиде железа или оксиде хрома (более прочный). Поверхности пластин должны быть максимально плоскими, иначе высота «полета» головок будет отклоняться от номинальной, что ухудшает условия записи и считывания. Количество пластин у большинства современных винчестеров невелико (1-4), слишком большие пакеты пластин громоздки и конструктивно неудобны (для блока головок). Однако есть модели с десятком пластин в пакете. Емкость одной пластины винчестера формата 3,5" уже достигла 160 Гбайт, и это, похоже, еще не предел возможностей магнитной записи. Правда, темпы роста снизились: последнее удвоение объема потребовало двух лет, хотя раньше объем удваивался ежегодно. Не снижая емкости, диаметр пластин уменьшают и до 3" — этим повышается быстродействие позиционирования и снижается потребляемая мощность.

Традиционно для записи и считывания информации используются *магнитные головки*, представляющие собой миниатюрные катушки индуктивности, намотанные на магнитном сердечнике с зазором. Требования к оптимальной кон

[1] Линейная скорость головки относительно поверхности у винчестеров на 7200 об./мин достигает 140 км/ч!

струкции головок у процессов записи и считывания различаются, так что универсальная головка представляет собой некоторый компромисс. Первые индуктивные головки содержали проволочные обмотки, их сменили головки, выполненные по *тонкопленочной* (Think Film, TF) технологии.

Для магнитных головок весьма критично расстояние от головки до поверхности магнитного слоя носителя. Непосредственный контакт головки с поверхностью допустим лишь при малых скоростях движения носителя (как в НГМД). Головки винчестеров поддерживаются на микроскопическом расстоянии от рабочей поверхности аэродинамической подъемной силой. Высота «полета» головки должна выдерживаться довольно строго, иначе магнитные поля головок будут «промахиваться» мимо рабочего слоя. Высота определяется тем положением, при котором подъемная сила, определяемая скоростью вращения, формой «крыла» головки и плотностью воздуха, уравновесит давление прижимающей головку пружины.

В современных накопителях для считывания часто применяют *магниторезистивные головки* (Magneto-Resistance Head, MRH), основанные на эффекте анизотропии сопротивления полупроводников в магнитном поле (Anisotropic Magneto-Resistance, AMR). В них через магниторезистивный датчик пропускают измерительный ток, и величина падения напряжения пропорциональна намагниченности находящегося под головкой участка магнитной поверхности. Сигнал с магниторезистивной головки повторяет форму записанного сигнала, а не является его производной (как у индуктивной головки). Магниторезистивная головка считывания хорошо «уживается» с индуктивной головкой записи, что позволяет достигать высокой плотности записи информации на магнитный носитель. Однако по технологии изготовления она сложнее тонкопленочной индуктивной, поскольку в ней сочетаются разнородные компоненты. От каждой комбинированной головки отходит четыре проводника: одна пара от электромагнитной головки записи (сопротивление постоянному току 8-10 Ом), вторая — от магниторезистивной головки чтения (около 30 Ом). Головки AMR позволяют считывать состояние намагниченности очень маленького пятна поверхности, с ними достигается плотность записи до 3 Гбит на квадратный дюйм. Изменение сопротивления при считывании составляет около 3 %. Следующий шаг — применение головок со сверхвысоким магниторезистивным эффектом (Giant Magneto-Resistive, GMR). Они позволяют добиваться плотности порядка 10 Гбит на квадратный дюйм, а изменение сопротивления достигает 7-8 %. Чем больше расстояние, головки считывания до поверхности, тем слабее поле и тем больше должно быть пятно магнитной поверхности, с которой головка может считать информацию, сохраняя приемлемый уровень ошибок (при одной и той же чувствительности). Правда, при этом повышается надежность (снижается вероятность повреждения поверхности головкой). Благодаря высокой чувствительности и узкой направленности головки GMR позволяют увеличить «высоту полета» без роста вероятности ошибок считывания.

При «падении» головки на рабочую поверхность, которое произойдет, если диск остановится, можно повредить как головку, так и поверхность диска. На современных дисках чистота обработки поверхности в рабочей зоне настолько

высока, что если головки лягут на неподвижную поверхность, то из-за эффекта молекулярного притяжения они просто прилипнут к ней. Чтобы этого не происходило, в нерабочем положении головки *паркуются* (park) — отводятся в нерабочую зону, где допустимо их «приземление». Поверхность парковочной зоны имеет повышенную шероховатость, благодаря чему прилипания не происходит и диск может свободно набрать скорость до «взлета» головок.

В старых винчестерах парковка обеспечивалась программно. Для ее выполнения в параметрах жестких дисков, хранимых в CMOS, присутствовал номер цилиндра для парковки (Landing Zone, или LZone). Парковка выполнялась запуском утилиты PARK или других утилит. В современных накопителях парковка осуществляется автоматически, когда напряжение питания или скорость вращения шпинделя падают ниже предельно допустимого значения. Для таких накопителей указанное в CMOS Setup значение параметра LZone игнорируется. Контроллеры современных дисков к тому же не выпускают головки из зоны парковки, пока шпиндель не наберет заданных оборотов.

Для *позиционирования головок* на требуемый цилиндр в старых винчестерах применялись шаговые двигатели с червячной передачей, зубчатой рейкой или ленточной передачей. У шагового привода есть масса недостатков: ему свойственно большое время поиска — при произвольных обращениях нужно «прошагать» в среднем половину общего числа цилиндров; с ним не добиться высокой плотности записи, поскольку повышение плотности повышает требования к точности позиционирования. Система позиционирования получается разомкнутой (без обратной связи), и все отклонения размеров (температурные, связанные со старением и т. п.) приводят к уходу головок от исходных дорожек. Правда, позже появились винчестеры с шаговым приводом, в которых положение головки подстраивается на наилучшее считывание служебной информации.

В современных накопителях для головок применяют привод с *подвижной катушкой* (voice coil actuator), работающий по принципу звуковой катушки динамика. Этот тип привода называют еще *соленоидным.* В таком приводе блок головок связан с катушкой индуктивности, помещенной в магнитное поле постоянного магнита. При протекании тока через катушку на нее начинает действовать сила, пропорциональная силе тока, которая вызывает перемещение катушки, а следовательно, и блока головок. Привод может быть линейным или поворотным. В накопителе с линейным приводом катушка с блоком головок перемещается строго по радиусу дисков. Такой привод применялся в накопителях больших машин. В накопителе с поворотным приводом блок головок с катушкой размещен на поворотной рамке (рис. 9.7), и траектория головок отличается от радиальной. При этом азимут головки относительно трека меняется при перемещении головки, и эта азимутальная погрешность нежелательна для работы головок. Тем не менее, с этой неприятностью, отсутствующей у линейного привода, мирятся из-за относительной простоты исполнения, меньших габаритов, а следовательно, и меньшей инерционности поворотного привода. В большинстве современных накопителей на жестких дисках применяется поворотный привод. Управляя направлением и силой тока, можно быстро перевести блок

головок в любое положение — произвольное, а не по фиксированным шагам. Но в такой системе позиционирования необходима обратная связь — информация о текущем положении головок, по которой контроллер может управлять приводом. Привод, обеспечивающий точное позиционирование по сигналу обратной связи, называется *сервоприводом.* С точки зрения теории автоматического управления, соленоидный привод является *замкнутой системой.* Управление сервоприводом может быть оптимизировано по времени установления головок на требуемую позицию: когда отклонение от заданного положения велико, можно подавать больший ток, вызывающий большее ускорение блока головок. По мере приближения ток уменьшается, а для компенсации инерции в конце позиционирования ток может и поменять направление (активное торможение). Такая система привода позволяет сократить время доступа до единиц миллисекунд против сотен миллисекунд, характерных для шагового привода. На требуемом цилиндре головки удерживаются следящей системой точного наведения. Остается только решить вопрос об источнике сигнала обратной связи для сервопривода.

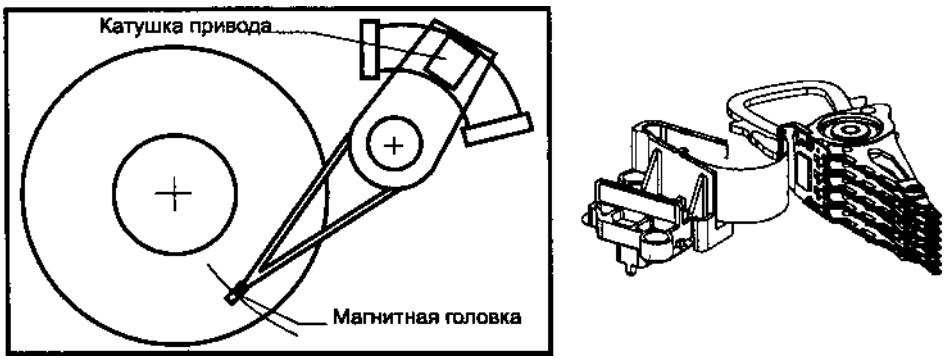


Рис. 9.7. Блок головок с поворотным приводом

В первых накопителях с линейным приводом использовались специальная зубчатая рейка и магнитный датчик, по сигналу которого отсчитывался номер трека. Однако по отношению к диску такая система привода все равно оставалась разомкнутой — привод позиционировал головки, руководствуясь собственными соображениями о координатах. Замкнуть систему управления позволило размещение прямо на диске *сервометок* — вспомогательной информации для «системы наведения». Благодаря сервометкам изменение размеров диска и привода под действием температуры и других факторов перестает существенно влиять на точность позиционирования, поскольку сервометки располагаются рядом с теми же искомыми треками. Сервометки записываются в расположенных между треками областях при сборке накопителя, когда для позиционирования используется внешний «прицел» специального технологического оборудования. Четные и нечетные серводорожки различаются, хотя у них имеются совпадающие фрагменты синхронизации. Когда головка считывания находится точно посередине между парой соседних серводорожек, сигналы, наводимые от

четной и нечетной серводорожек, имеют одинаковую амплитуду (фрагменты синхронизации синфазны и дают суммарный сигнал синхронизации сервометок). Если головка отклоняется от трека в сторону, например, четной серводорожки, амплитуда ее сигнала оказывается больше, что распознается детектором, и по сигналу рассогласования вырабатывается токовое воздействие на катушку, стремящееся вернуть головку на середину, то есть точно на трек. Этим обеспечивается хороший сигнал считывания, в составе которого имеются и адресные маркеры, содержащие номер цилиндра. По ним контроллер определяет, туда ли пришли головки, и, если необходимо, формирует импульсные сигналы позиционирования. После них снова вступает в дело система точного наведения по сервометкам, и в конце концов поиск успешно завершается. В процессе эксплуатации сервометки только считываются. По месту размещения сервометок различают накопители с *выделенной сервоповерхностью* (dedicated servo) и со *встроенными сервометками* (embedded servo). В первом случае в пакете дисков выделяется одна поверхность, используемая исключительно для хранения сервометок, и соответствующая ей головка является сервоголовкой. Ошибка позиционирования в такой системе может возникать вследствие изменения взаимного положения (перекоса) головок в блоке. Сервоголовка для следящей системы дает информацию практически непрерывно (на одном треке могут быть записаны сотни и тысячи сервометок), что повышает скорость и улучшает качество поиска и слежения. У накопителей с выделенной сервоповерхностью, как правило, число «полезных» головок нечетное.

В накопителях со встроенными сервометками информация для сервопривода записывается на рабочих поверхностях между секторами с данными. Она может размещаться в начале каждого трека — при этом на диске появляется клиновидная область сервометок. Однако из-за такого размещения сервоинформация (сигнал обратной связи) доступна только дискретно с периодичностью в один оборот диска, что при скорости 3600 об./мин составляет 16,6 мс. До точного позиционирования приходится выжидать несколько оборотов диска. Более быстродействующий вариант — размещение сервометок перед каждым сектором — позволяет выйти на заданный трек даже за доли оборота шпинделя. Преимущество встроенных сервометок в том, что они обеспечивают компенсацию любых изменений в геометрии, поскольку система наводит головки именно по тому треку, к которому выполняется доступ. При этом снижаются требования к жесткости блока головок, что позволяет его облегчить и снизить инерционность. Однако даже переключение головок (без смены номера цилиндра) требует некоторого времени поиска — следящая система должна перестроиться на сервометки с другой головки. Существуют накопители с гибридной сервосистемой, где помимо выделенной сервоповерхности используются сервометки, размещенные на рабочих поверхностях.

При выполнении операций записи и форматирования сигнал записи на время прохождения сервометок должен обязательно блокироваться. Если из-за неисправности устройства сервометки на какой-либо дорожке «накроет» сигнал записи, эта дорожка (а возможно, и соседние с ней) навсегда станет сбойной.

Для накопителей с соленоидным приводом проблема автопарковки решается легко, поскольку энергии для перемещения поворотного привода требуется немного. В парковочном положении головки удерживаются магнитной защелкой или механическим фиксатором.

Совершенно очевидно, что в гермоблоке воздух должен быть чистым — мелкая частичка, попавшая под головку, под которой пролетает носитель со скоростью несколько десятков километров в час, может поцарапать и головку, и диск. Вопреки названию гермоблок может и не быть герметичным — в нем имеется отверстие, закрытое фильтром и обеспечивающее выравнивание давления внутри сборки с атмосферным. Помимо этого фильтра, называемого барометрическим, имеется еще и внутренний рециркуляционный фильтр. Этот фильтр устанавливается на пути потока воздуха, увлекаемого вращающимся пакетом дисков. Он улавливает частички, которые могут выбиваться из поверхности дисков при «взлете» и «посадке» головок. Существуют, конечно, и специальные накопители для работы в особых климатических условиях. Они могут иметь герметический корпус, призванный выдерживать разность внутреннего и наружного давлений. В PC, как правило, применяют накопители обычного исполнения.

Помимо блока механики дисковый накопитель должен иметь и блок *электрони*- ки, управляющий приводами шпинделя и головок, а также обслуживающий сигналы рабочих головок записи-считывания. Обычно в гермоблоке на маленькой плате устанавливают предварительные усилители считывания, коммутаторы и формирователи сигналов записи. К этой плате подключаются провода, идущие к головкам, а также кабель, связывающий ее с контроллером. *Контроллером накопителя* называют электронное устройство, на одной (интерфейсной) стороне которого идет обмен байтами команд, состояния и, конечно же, записываемой и считываемой информации, а другая его сторона связывается с гермоблоком. В современных накопителях на жестких дисках контроллер расположен на плате электроники, смонтированной вместе с гермоблоком. В старинных накопителях на жестких дисках с интерфейсами ST506/412 и ESDI контроллер был внешним и располагался на специальной карте расширения. Объединение контроллера с гермоблоком позволило решить многие проблемы оптимизации накопителей, о которых речь пойдет далее. Контроллер современного винчестера состоит из нескольких основных блоков:

- Управляющий микроконтроллер (однокристальный) обеспечивает взаимодействие всех блоков накопителя и связь с внешним интерфейсом (ATA, SCSI, USB, 1394, Fibre Channel...). Здесь, как правило, используется 8- или 16-разрядный микроконтроллер общего назначения. Его программное обеспечение может храниться во внутреннем ПЗУ контроллера или в отдельной микросхеме памяти (EPROM или флэш). Часть ПО может загружаться во внутреннее ОЗУ, доступное микроконтроллеру, со служебной области диска.
- Внутреннее ОЗУ (буферная память накопителя) используется для считывания и записи секторов и локального кэширования. Объем внутреннего ОЗУ может быть от десятков килобайт до единиц мегабайт. Этот объем в сочетании

с эффективностью алгоритмов кэширования заметно влияет на производительность винчестера.

- Блок управления шпиндельным двигателем обеспечивает запуск и останов шпинделя по команде от микроконтроллера и поддерживает заданную скорость вращения по сигналам от датчиков индекса, специальных датчиков вращения или/и сервометок. Блок сигнализирует о достижении минимальной скорости, на которой можно выпускать головки, и номинальной.
- Блок управления позиционированием формирует импульсы управления соленоидом для перехода с цилиндра на цилиндр по команде микроконтроллера и следит за положением головки на треке по принятым сервосигналам.
- Коммутатор головок, совмещенный с предусилителем считывания и формирователем тока записи, — микросхема, смонтированная в непосредственной близости от головок (внутри гермоблока). Такое размещение позволяет улучшить отношение сигнал/шум при считывании.
- Канал чтения-записи представляет собой цепи, выделяющие из сигнала, принятого от предусилителя, импульсы синхронизации и данных и формирующие сигналы записи. В современных винчестерах в этом узле широко применяют сигнальные процессоры, реализующие метод PRML. Раньше здесь использовались аналоговые методы обработки сигналов.
- Детектор сервометок выделяет их из потока сигналов, принимаемых с головок считывания.
- Контроллер НЖМД (Hard Disk Controller, HDC) — специализированная микросхема, выполняющая основные функции, связанные с записью и считыванием данных. Она декодирует приходящий поток считываемых данных, выделяет служебные области, находит требуемые секторы, проверяет целостность данных (по алгоритму CRC или ECC), преобразует поток битов в байты и записывает их в буферную память. При записи микросхема в нужный момент (когда подходит поле данных требуемого сектора) формирует поток сигналов, требуемых для кодирования информации, преобразуя байты данных в поток битов и вычисляя контрольные последовательности. При форматировании она формирует заказанную структуру трека.

Разработчики винчестеров стремятся к сокращению числа микросхем, применяемых в блоке электроники (это с технико-экономической точки зрения почти всегда выгодно), и распределение функциональных узлов по микросхемам может варьироваться. Так, контроллер НЖМД может объединяться с управляющим процессором и некоторыми другими узлами в одной специализированной заказной ИС.

Для кодирования данных применяются методы MFM, RLL, ARLL. В современных накопителях широко распространена технология *PRML* (Partial Response Maximum Likelihood — максимальная правдоподобность при неполном отклике), пришедшая из области телекоммуникаций. В традиционной технологии считывания использовались аналоговые пиковые детекторы сигналов воспроизведения, эти сигналы декодировались схемами считывания. Однако повышение плотности хранения данных приводит и к усилению межбитных помех,

в результате задача считывания усложняется. В технологии PRML при считывании производится оцифровка аналогового сигнала с головки и запись последовательности этих выборок в буфер. Для оцифровки больше всего подходит сигнал с магниторезистивной головки, поскольку его форма повторяет форму напряженности магнитного поля (а не производную, как в индуктивных головках). Следующий этап считывания предполагает цифровую фильтрацию записанного сигнала. Принятые фрагменты трактуются как группы закодированных битов по максимальной похожести формы отклика. Современная техника цифровой обработки позволяет целиком выполнять декодирование PRML «на лету», обеспечивая скорость считывания даже выше, чем при аналоговом декодировании RLL.

Производительность и оптимизация дисков

Задача, стоящая перед винчестером, выглядит просто — как можно быстрее отработать запрос на чтение или запись данных. Время, затрачиваемое на обмен данными одного сектора, складывается из *времени поиска* (seek time) цилиндра, *времени ожидания* (latency time) подхода сектора к головке, *времени обмена данными* сектора между накопителем и контроллером и между контроллером и памятью компьютера, являющейся источником и пунктом назначения для хранимых данных. Конечно, основные факторы, определяющие эти затраты времени, — механические: достижимая скорость позиционирования и скорость вращения диска. Последний параметр определяет и время ожидания сектора (в среднем это половина периода оборота диска), и время собственно передачи данных сектора (оно примерно равно времени оборота, деленному на количество секторов на треке). Однако одиночная передача сектора встречается редко, поскольку его размер невелик. Интерес представляет оптимизация передачи блока данных, расположенных в соседних секторах. Вполне естественно, что если блок данных не умещается в одном секторе, его продолжение логично разместить на том же треке в секторе со следующим номером, поскольку для доступа к нему нужно только дождаться его подхода к головке. Когда емкость трека оказывается исчерпанной, логично перейти на следующую поверхность того же цилиндра. Для этого достаточно переключить головки, что выполняется электронным путем довольно быстро. Правда, у современных дисков с сервоинформацией, размещенной на рабочих поверхностях, требуется перенастройка системы точного позиционирования на сигнал от другой головки, а это уже требует заметного времени. Использовав все секторы цилиндра, переходят к следующему цилиндру, для чего уже необходимо переместить головки, то есть затратить заметное время на операцию поиска. Дефрагментирующие программы как раз и занимаются тем, что размещают все блоки файлов в последовательных кластерах, а следовательно, и в секторах, упорядоченных по данному способу счета. Они справедливо рассчитывают на то, что для дискового накопителя оптимален именно такой порядок запроса секторов.

Для оптимизации производительности старых винчестеров, контроллеры которых не успевали обрабатывать и передавать хосту поток данных «на лету», применяли *чередование секторов* (interleaving). В современных накопителях

чередование секторов не применяют. Для учета затрат времени на перенастройку сервосистемы при переходе на другую головку используют *послойное смещение секторов*, при котором начало следующего трека цилиндра чуть отстает от предыдущего. Для учета этой задержки при перемещении головок на соседний цилиндр вводят *радиальное смещение* (radial skew) секторов соседних цилиндров. Чередование секторов задавалось программно указанием последовательности номеров секторов в команде форматирования трека; для задания послойного и радиального смещений общепринятый программный интерфейс отсутствует. Смещение применяют только для накопителей со встроенными контроллерами, каковыми и являются все современные винчестеры.

Теперь несколько слов о скоростях вращения, плотности записи и количестве секторов на треке. Допустимое количество секторов определяется допустимой плотностью изменения магнитного потока, длиной трека и схемой кодирования данных. Длина трека определяется его диаметром, и, естественно, внешние треки дисков длиннее внутренних. В накопителях со внешним контроллером количество секторов на всех треках принимается постоянным. Поскольку приходится ограничивать число секторов по допустимой плотности изменения магнитного потока на самых коротких внутренних треках, внешние треки оказываются «недогруженными». В накопителях с интерфейсом ST-506/412 для схемы кодирования MFM на каждом треке помещали 17 секторов по 512 байтов данных, а для схемы RLL — 26 секторов. В накопителях с интерфейсом ESDI, у которого сепаратор данных с контроллера перенесли в блок накопителя, удалось увеличить число секторов до 32-80 на трек. В накопителях на гибких дисках требование совместимости накопителей, у которых могут несколько различаться скорости вращения шпинделя, привело к снижению номинального числа секторов (9 для дискет формата 360 Кбайт) по сравнению с реально возможным (на современных НГМД вместо штатных 18 секторов можно смело размещать 20).

Зная количество секторов на треке, размер сектора и частоту вращения диска, можно определить максимальную скорость передачи данных между накопителем и контроллером. Так, для традиционного диска MFM с 17 секторами на трек и скоростью вращения 3600 об./мин скорость передачи данных составляла 1 7 x 5 1 2 x 3600 = 31,3 Мбайт/мин, или около 500 Кбайт/с при чередовании 1:1. При чередовании 2:1 за один оборот будет передано только 8 или 9 секторов, то есть скорость снижается вдвое.

Снизить неравномерность линейной плотности информации на внутренних и внешних цилиндрах можно форматированием треков с различным количеством секторов. Метод форматирования, называемый *зонной записью* (Zone Bit Recording, ZBR), позволяет существенно увеличить объем хранимых данных по сравнению с фиксированным числом секторов при тех же характеристиках носителя. Суть метода в том, что с учетом различия в длине треков цилиндры разбиваются на зоны, для которых принимается одинаковое число секторов на трек. Для внешних цилиндров число секторов на треке выбирается большим, чем для внутренних (примерно в 2 раза). Естественно, при зонной записи скорость передачи информации на внешних треках выше, чем на внутренних (за

один оборот передается большее количество секторов). При этом и частота синхронизации схем записи и считывания для разных зон оказывается различной. Внешние контроллеры (ST-506/412 или ESDI) не позволяли работать с переменным числом секторов на треке, да и операционной системе и BIOS было бы неудобно брать на себя все пересчеты, специфичные для каждой модели накопителя. Контроллер, встроенный в накопитель, скрывает от системы переменное физическое число секторов на треке, а для общения с накопителем используется фиктивная *внешняя геометрия диска*. Когда говорят о собственно дисковых устройствах, эту геометрию часто называют логической. Однако в контексте с функциями BIOS ее уже называют физической. В данной книге мы остановимся на названии «внешняя геометрия», которая может быть либо *трехмерной* (с фиктивным числом цилиндров, головок и секторов на треке), либо *линейной*, где номер сектора задается одним числом.

Заметим также, что для накопителей с зонной записью вычислить скорость передачи по числу секторов на треке и скорости вращения шпинделя затруднительно, поскольку указываемое в паспорте число секторов обычно относится лишь к логической геометрии, а к физической отношения не имеет. Иногда в документации на винчестеры с зонной записью можно найти максимальное и минимальное количества секторов на трек, по которым можно оценить диапазон скоростей передачи. Повышение скорости вращения диска и увеличение числа секторов на треке обеспечивают повышение скорости обмена и ускорение позиционирования.

Производительность винчестера определяется не только возможностями гермоблока, но и свойствами его контроллера.

Интерфейс накопителя определяет возможную скорость обмена между буферной памятью винчестера и хостом. Основные интерфейсы винчестеров обеспечивают более высокую скорость передачи, чем внутренняя скорость обмена с пластинами, и благодаря этому они не тормозят обмен с дисками. Для интерфейса ATA (см. п. 19.2) эта скорость лежит в диапазоне от 3,3 (PIO Mode 0) до

66- 133 Мбайт/с (Ultra DMA); Serial ATA — 150 Мбайт/с. Для SCSI (см. 20.4) — от 5 (обычный «узкий») до 320 Мбайт/с (Ultra4 Wide); для последовательного интерфейса SAS — 150 или 300 Мбайт/с. Интерфейс Fibre Channel обеспечивает скорость 100-800 Мбайт/с. Для внешнего подключения сменных устройств используют LPT-порт (0,1-2 Мбайт/с) или шину USB (около 1,5 Мбайт/с) — здесь низкая скорость является расплатой за мобильность (в недорогом варианте). Правда, новая версия USB 2.0 обеспечивает скорость 24 Мбайт/с.

Производительность винчестера в значительной степени зависит от объема его *буферной памяти*, который у современных винчестеров составляет 2-8 Мбайт (есть тенденция роста). Часть этой памяти (до нескольких сотен килобайт) может использоваться для встроенного ПО. Естественно, чем больше объем памяти, тем лучше. Часто винчестеры с одной и той же механикой выпускаются с разными объемами памяти (это отражается в элементах названия модели), и более дорогие модели (с большей памятью) имеют заметный выигрыш в производительности. Возможности использования буферной памяти зависят от

сложности контроллера. У старых дисков буфер вмещал всего один сектор и был однопортовым — не допускал одновременного обмена данных с диском и внешним интерфейсом. Более сложные и эффективные контроллеры имеют двухпортовый буфер на множество секторов, допускающий одновременное выполнение этих операций. В современных дисках контроллер способен использовать буфер и для *кэширования*. Общепринятой технологией кэширования диска является *упреждающее считывание* (read ahead). Суть его проста: если контроллер получает запрос на чтение сектора, то помимо запрошенного сектора он автоматически считывает в буфер и следующие за ним секторы. В результате весьма вероятный запрос к следующему сектору будет обслужен из буфера без задержки, которая возможна из-за естественной асинхронности действий операционной системы и приложений с вращением диска. Более «ловкие» контроллеры идут дальше: они считывают в буфер весь трек, как только выполняется команда позиционирования, а когда приходит следующая за ней команда чтения, данные уже находятся в буфере. Такую нулевую задержку (zero latency) обеспечивает система команд интерфейсов ATA и SCSI. Кэширование применяют и для записи, но более сдержанно — здесь есть риск потери данных (например, при внезапном отключении питания).

Как и для всякой кэш-памяти, для эффективности встроенного кэша накопителя существенным фактором является алгоритм выделения памяти и замещения старых записей. Как обычно, замене подлежат самые старые записи. Вопрос
о размере областей, выделяемых для упреждающего чтения, может решаться исходя из текущей статистики обращений. Контроллер с *адаптивным кэшированием,* заметив, что последние запросы чтения являются одиночными, перестает выделять большие области под упреждающее чтение. Если характер запросов изменяется, адаптивный контроллер принимает соответствующие решения. Кроме того, отпечаток на алгоритм кэширования накладывает и многозадачный характер современных операционных систем и их дисковых запросов. Таким образом, многозадачность проникает и во встроенные контроллеры дисков.

У современных дисков возможен выбор режима работы: для серверного применения (режим Server) можно использовать динамическое распределение буфера между потоками (переменное число и размер буферов), для настольного применения (режим Desktop) — статическое распределение (например, на 32 фиксированных сегмента).

Параметры винчестеров

После рассмотрения устройства и работы дисковых накопителей должен быть понятен смысл их параметров. Далее перечислены *общие параметры диска*:

- *Форматированная емкость* (formatted capacity), гигабайт (мегабайт), представляет собой объем хранимой полезной информации, то есть сумму полей данных всех доступных секторов. *Неформатированная емкость* (unformatted capacity) представляет собой максимальное количество битов, записываемых на всех треках диска, включая служебную информацию (заголовки секторов, контрольные коды полей данных). Соотношение форматированной

и неформатированной емкостей определяется форматом трека (размером сектора), но поскольку для рядового пользователя свободы выбора формата нет, практический интерес представляет только форматированная емкость диска, которая указывается для стандартного размера сектора (512 байт). Напомним, что мегабайт и гигабайт здесь обычно означают 10^6 и 10^9 байт. Иногда указывается число доступных секторов.

- *Скорость вращения шпинделя* (spindle speed), измеряемая в оборотах в минуту (Revolutions Per Minute, RPM), позволяет косвенно судить о производительности (внутренней скорости). Для жестких дисков широкого применения значение 3600 об./мин было стандартным несколько лет назад; сейчас обычной скоростью считается 4500 и 5400, а 7200 — более высокой. Там, где производительность особо критична, используют диски со скоростью 10 000 и 15 000 об./мин.
- *Интерфейс* (interface) определяет способ подключения накопителя. Для накопителей со встроенным контроллером распространены интерфейсы ATA и SCSI (параллельные и последовательные), для устройств внешнего исполнения применяют шины USB, FireWire и Fibre Channel, а также подключение к LPT-порту.
- Объем буферной памяти, возможности кэширования (чтение, запись, многосегментность, адаптивность).

Следующая группа параметров — *параметры внутренней организации*:

- *Количество физических дисков* (disks), или рабочих поверхностей (data surfaces), используемых для хранения данных. Современные накопители с небольшой высотой имеют малое (1-3) количество дисков для облегчения блока головок. Большее число дисков (и большая высота) характерно для старых накопителей и современных накопителей большой емкости.
- *Количество физических головок чтения-записи* (read/write heads), естественно, совпадающее с числом рабочих поверхностей. Заметим, что число головок (и рабочих поверхностей) может быть и меньше удвоенного числа дисков — обычно в каждом семействе есть такого рода модели. Это делается для утилизации дисков, у которых одна из поверхностей оказывается с производственным браком, или исходя из других технологических соображений.
- *Физическое количество цилиндров* (cylinders) от нескольких сотен, характерных для первых винчестеров, возросло до десятков тысяч.
- *Размер сектора* (bytes per sector) обычно составляет 512 байт.
- *Количество зон* и *количество секторов на треке* (sectors per track) в крайних зонах.
- *Расположение сервометок*. Они могут располагаться на выделенной поверхности (dedicated servo), встраиваться в рабочие поверхности (embedded servo) или иметь гибридный вариант расположения (hybrid servo).
- *Методом кодирования* (recording method, или data encoding scheme) и *декодирования* данных может быть метод MFM (FM почти и не применяли), RLL (ARLL) или PRML. Последний является наиболее прогрессивным.

Следующими описываются *параметры внешней* («логической») *геометрии* для устройств ATA:

- Поддержка режима линейной адресации LBA (может отсутствовать у старых дисков небольшого объема).
- Поддержка трансляции CHS (для дисков большого размера согласно ATA/ ATAPI необязательна, но на практике почти всегда имеется).
- Количество цилиндров, головок и секторов на трек (при трансляции CHS) по умолчанию и возможности задания геометрии.

Быстродействие и *производительность* характеризуются следующими параметрами:

- *Время перехода на соседний трек* (track-to-track seek), измеряемое в миллисекундах, характеризует быстродействие системы позиционирования. Для современных жестких дисков характерно время перехода 0,5-2 мс, причем для записи оно несколько больше, чем для считывания (записывать лучше при более точном позиционировании).
- *Среднее время поиска* (average seek time) определяется по серии обращений к случайным цилиндрам. Для большинства современных дисков оно составляет около 8-10 мс, в самых быстрых его удается снизить до 4-5 мс. Для винчестеров со скоростью 10000 об./мин (обозначаются как 10K) время поиска в зависимости от дальности перехода составляет от 0,3 мс (на соседний трек) до 12 мс (от края до края). Чем больше объем накопителя, тем сложнее достичь малого времени поиска: большее число головок труднее быстро перемещать; большее число цилиндров либо увеличивает длину перемещения головок, либо повышает требования к точности позиционирования.
- *Максимальное*, или *полное, время поиска* (maximum seek time, full seek time) относится к самым дальним переходам между крайними цилиндрами. Оно примерно в два раза превышает среднее время поиска.
- *Среднее время ожидания* (average latency) сектора при одиночном обращении обычно составляет половину времени полного оборота (для 3600 о.б/мин — 8 мс, 7200 об./мин — 4 мс, 15 000 об./мин — 2 мс).
- *Внутренняя скорость передачи данных* (internal transfer rate) между носителем и буферной памятью контроллера задает физический предел производительности накопителя. Эта скорость выражается в разных величинах: если указывается в мегабитах в секунду (Mb/s), то сюда кроме пользовательских данных входят и «накладные расходы» — биты служебных полей. У самых высокоскоростных винчестеров (с частотой вращения 15 000 об./мин) этот параметр подбирается к 900 Мбит/с. При выражении скорости в мегабайтах в секунду (MB/s) подразумеваются только байты пользовательских данных, поэтому пересчет в мегабиты в секунду простым умножением на 8 (число битов в байте) неправомерен. У современных винчестеров с частотой вращения 5400 об./мин скорость составляет 15-25 Мбайт/с, с частотой вращения 7200 об./мин — 30-60 Мбайт/с. Для каждой модели обычно указывают минимальное и максимальное значения скорости, соответствующие внутренним и внешним трекам (вспомним о зонном формате).

- *Внешняя скорость передачи данных* (external transfer rate), измеряемая в килобайтах (мегабайтах) полезных данных в секунду, передаваемых по шине внешнего интерфейса, зависит от быстродействия электроники контроллера, типа интерфейсной шины и режима обмена. Для интерфейса ATA в режиме обмена PIO Mode 0 скорость составляет 3,3 Мбайт/с, в режиме PIO Mode 4 — 16,6 Мбайт/с, в режиме Ultra-DMA — 33, 66,100 и даже 133 Мбайт/с. Для шин SCSI ограничения скорости в зависимости от типа электрического интерфейса составляют 5, 10, 20, 40, 80, 160 и даже 320 Мбайт/с, а для Fibre Channel — 100 и 200 Мбайт/с.
- *Длительная производительность* (sustained throughput) определяется при последовательном чтении большого количества секторов. На нее влияют все составляющие: внутренняя и внешняя скорости, время позиционирования, задержка подхода сектора, количество ошибок позиционирования и чтения. Минимальное гарантированное значение этой скорости определяет возможность применения накопителя для мультимедийных приложений (записи и чтения аудио- и видеоданных) реального времени. Современные винчестеры с частотой вращения 5400 об./мин выдерживают потоки 15-50 Мбайт/с, с частотой вращения 7200 об./мин — 20-60 Мбайт/с, с частотой вращения 10 000 об./мин — 40-80 Мбайт/с, с частотой вращения 15 000 об./мин — 40-100 Мбайт/с. Эта скорость всегда ниже максимального значения внутренней скорости и предела внешней скорости. Для мультимедийных приложений в новых дисках имеются специальные варианты команд считывания и записи (*потоковое расширение ATA/ATAPI-7),* отличающиеся особой обработкой ошибок и политикой кэширования.

Надежность (reliability) устройства и *достоверность* хранения данных (data integrity) характеризуются следующими параметрами:

- *Ожидаемое время до отказа* (Mean Time Before Failure, MTBF), измеряемое в сотнях тысяч часов, является, естественно, среднестатистическим показателем для данного изделия. Реально столько часов (100 000 ч — это более 10 лет) испытания проводить, естественно, невозможно. На самом деле производится выборка из большой группы устройств, какая-то часть которых за вполне обозримое время испытаний выходит из строя. По зафиксированному потоку отказов теория вероятностей позволяет вычислить это условное ожидаемое время безотказной работы (хотя термин «поток» в данном контексте выглядит угрожающе). Значение MTBF, равное 100 000 ч, считается малым; 200 000-400 000 ч — нормальным, а 1 000 000 ч — высоким показателем надежности. Но, как говорится, «столько не живут», а возможный отказ винчестера в течение года (всего-то 8760 часов) вполне уложится в статистический показатель (если у вас за год не отказала значительная партия устройств). Иногда указывают и ожидаемое *время наработки на отказ* (Power On Hours, POH), в котором учитывается только время работы устройства (в MTBF не учитывается, включено устройство или нет).
- Более ценным для пользователя является *гарантийный срок* (limited warranty), в течение которого изготовитель (или поставщик) обеспечивает ремонт

или замену отказавшего устройства. Примечательно, что даже при MTBF = = 800 ООО ч (91 год) изготовитель дает гарантию всего 3-5 лет.

- *Вероятность неисправимых ошибок чтения* (nonrecoverable read errors per bits read) для современных винчестеров имеет порядок одной ошибки на 10^{14} считанных битов. Оценить, много это или мало, можно следующим образом. Пусть винчестер постоянно находится в работе, и к нему непрерывно идут обращения со средней производительностью чтения, которую «на глазок» можно оценить в 1 Мбайт/с (это соответствует, например, умеренно загруженному диску сервера). Тогда простая арифметика показывает, что раз в 115 дней будут возникать ошибки, не восстанавливаемые (но обнаруженные!) схемами ECC-контроля. Вполне вероятно, что повторное считывание сектора пройдет без ошибок.
- *Вероятность исправимых ошибок чтения* (recoverable read errors per bits read) имеет порядок единицы на 10^{11} считанных битов. Если бы не было ECC-контроля (или при неисправной схеме контроля, с чем автору доводилось сталкиваться на практике), этот поток ошибок сделал бы работу с накопителем просто невыносимой (в том же примере ошибки будут появляться чаще, чем раз в три часа).
- *Вероятность ошибок поиска* (seek errors per seek) характеризует качество сервосистемы. Для современных винчестеров характерна вероятность одной ошибки на 10^{8} операций поиска. Эти ошибки (при малом их числе) вполне безобидны, поскольку наличие номера цилиндра в заголовке каждого сектора не позволяет «промахнуться» при выполнении операций чтения или записи. Повторение операции поиска только слегка снижает среднее время доступа.

Уровень акустического шума характеризуется звуковой мощностью (sound power), излучаемой винчестером. На холостом ходу для винчестеров обычного применения (скорость вращения 5400 об./мин) желателен уровень до 30 дБ, при позиционировании желательно, чтобы он возрастал не более чем на 3-4 дБ. Для высокопроизводительных винчестеров (7200 об./мин), которые, естественно, шумят больше, желателен уровень до 35 дБ на холостом ходу. Для винчестеров, предназначенных для работы в устройствах бытовой электроники, желателен уровень до 25 дБ. Часто уровень шума указывают в белах, эти цифры выглядят скромнее (25 db и 2,5 b — это одно и то же). Шум винчестера сильно зависит от корпуса компьютера, в который его устанавливают, и от способа крепления.

Потребляемая мощность определяется номинальными и пиковыми токами, потребляемыми по цепям +5 В и +12, В. Пик потребления по цепи +12 В возникает при раскрутке шпиндельного двигателя. Если блок питания компьютера не выдерживает этого пика (например, при одновременном запуске нескольких винчестеров) и напряжение «проседает», то шпиндель за требуемое время не наберет номинальную скорость и контроллер его остановит. Попытки «завести» мотор могут повторяться, что слышно по характерному звуку. Для высокоскоростных винчестеров приходится учитывать тепловыделение — для них может потребоваться специальный вентилятор.

Физические параметры включают ширину (width), высоту (heigth) и глубину (depth) корпуса накопителя, измеряемые в дюймах (inches) или миллиметрах, и Массу (weight), измеряемую в фунтах (lb) или килограммах.

Условия эксплуатации и хранения определяют возможные диапазоны температур, атмосферного давления, влажности и силы допустимых ударов. Вполне понятно, что условия эксплуатации (operating conditions) несколько жестче, чем условия хранения (non-operating conditions).

Приведем основные параметры некоторых моделей винчестеров формата 3,5", дающие представления о свойствах современных гермоблоков и контроллеров.

Некогда популярные винчестеры Quantum Fireball объемом 1-3,8 Гбайт имеют 1-3 диска (2-6 рабочих поверхностей). На каждом диске по 6810 треков, разбитых на 15 зон. Количество секторов на треке во внутренней и внешней зонах — 104 и 210 или 122 и 232. При скорости вращения 4500 об./мин скорость обмена с носителем достигает 6,3 или 6,96 Мбайт/с на внешних цилиндрах и уменьшается примерно вдвое на внутренних.

Более современные (2005 г.) диски Hitachi Deskstar 180GXP, 7K250 и 7K400 со скоростью вращения шпинделя 7200 об./мин объемом 60-400 Гбайт имеют 3-5 дисков (6-10 рабочих поверхностей). Скорость обмена с носителем до 757 Мбит/с, что обеспечивает длительную скорость 29-61 Мбайт/с в зависимости от номера зоны (их около 30). Среднее время поиска — 8,5 мс. Интерфейсы ATA (100 Мбайт/с) или Serial ATA поддерживают потоковые расширения ATA/ATAPI-7. Потребляемая мощность 9,5 Вт (модель на 400 Гбайт).

Самый быстрый на середину 2000 года винчестер Cheetah X15 (Seagate) объемом 18,4 Гбайт имеет скорость вращения 15 000 об./мин, скорость обмена с носителем — 395-492 Мбит/с (включая служебную информацию). Среднее время поиска — 3,9 мс, время перехода на соседний трек — 0,5 мс, а средняя задержка данных — 2 мс. Интерфейс — SCSI Ultra3 (160 Мбайт/с), длительная производительность обмена — 38-47,4 Мбайт/с.

Винчестер Maxtor Atlas 15K (2005 г.) объемом 36-147 Гбайт (1-4 пластины) со скоростью 15 000 об./мин обеспечивает максимальную длительную скорость чтения или записи 98 Мбайт/с. Среднее время поиска — 3,0-3,8 мс, время перехода на соседний трек — 0,3/0,5 мс (чтение/запись), а средняя задержка данных — 2 мс. Интерфейс — SCSI Ultra 320, максимальная скорость обмена с хостом — 270 Мбайт/с. Как видно, существенно вырос объем (плотность записи), и следовательно, скорость обмена с носителем (длительная производительность чтения/записи). Потребляемая мощность — 9,2-14 Вт.

Фирма Samsung в 2005 году выпустила диски с объемом пластины 80 и даже 125 Гбайт, при этом для 80-гигабайтных пластин внутренняя скорость вращения достигает 741 Мбит/с на умеренной скорости — 7200 об./мин.

И в заключение раздела приведем примеры больших и маленьких винчестеров.
Внешний диск LaCie Bigger Disk Extreme (габариты 268 x 173 x 88 мм, вес 5 кг) с интерфейсом FireWire800 имеет емкость 1 или 1,6 Тбайт. Скорость вращения 7200 об./мин, время доступа 10 мс, внутренний кэш 4 x 8 Мбайт.

Винчестер для мобильных ПК (формат 2,5") Hitachi Travelstar E5K100 (HTE541040G9AT00) имеет емкость 40 Гбайт (одна пластина), скорость вращения — 5400 об./мин, скорость обмена с носителем — до 493 Мбит/с. Среднее время поиска — 12 мс. Питание — 5 В, потребление в нормальном режиме — 2 Вт, в состояниях Stndby и Sleep — 0,2 и 0,1 Вт соответственно. В таком формате выпускаются и винчестеры со скоростью вращения 10 000 об./мин.

Микровинчестеры Hitachi Microdrive 3K6 (рис. 9.8) в формате карт CFA Type II обладают большой (для CFA) емкостью (до 6 Гбайт) и высокой скоростью (до **9,4** Мбайт/с длительная скорость). В этих устройствах всего одна пластина, скорость вращения — 3600 об./мин, скорость обмена с носителем — **67-** 125 Мбит/с, длительная скорость — 4,9—9,4 Мбайт/с. Скорость передачи по интерфейсу — до 33 Мбайт/с (UltraDMA Mode 2). Среднее время поиска — 12 мс, время перехода на соседний трек — 1 мс. Питание — 3,3 (или 5) В, потребление в покое — 13 (15) мА, при записи — 230 (280) мА.

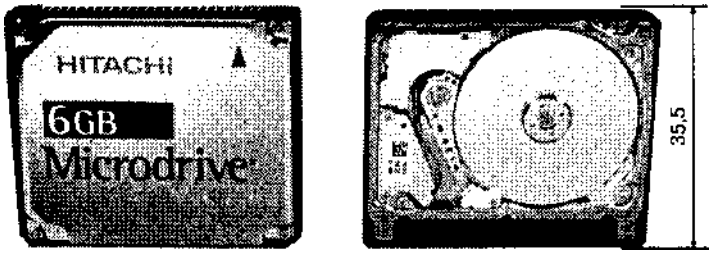


Рис. 9.8. Микровинчестер в формате карт CFA Type II

Функционирование винчестера

Как видно из предыдущего описания, современный винчестер является сложным устройством со встроенным микроконтроллером, обрабатывающим команды хост-компьютера, поступающие по внешнему интерфейсу. При подаче питания и по сигналу аппаратного сброса микроконтроллер выполняет самотестирование, сначала проверяя собственное «хозяйство» (ОЗУ, ПЗУ, регистры), а затем и остальные блоки. Далее выполняется запуск шпиндельного двигателя, и когда он наберет достаточные обороты, головки выводятся из зоны парковки и ими начинает управлять сервосистема. После этого микроконтроллер может загрузить со служебных треков диска необходимую ему информацию. На диске могут храниться таблица трансляции секторов, списки дефектных блоков, паспорт диска и даже часть программ микроконтроллера. Для повышения надежности служебная информация обычно записывается с несколькими копиями, поскольку невозможность ее считывания ведет к потере работоспособности устройства. Служебная информация может храниться и не на магнитном носителе, а в энергонезависимой электронной памяти винчестера (EEPROM или флэш). На основании служебной информации контроллер конфигурируется под конкретный гермоблок, с которым он работает, — определяет списки рабо

чих головок, число цилиндров, число секторов в треках каждой зоны и т. п. Как правило, один и тот же блок электроники используется для ряда моделей винчестеров, отличающихся числом рабочих поверхностей, причем физически у них может быть даже одно число дисков, но не все поверхности работоспособны. У таких моделей может совпадать и «прошивка» — микропрограммное обеспечение в ПЗУ или флэш-памяти. Успех конфигурирования означает готовность винчестера к исполнению команд, поступающих от хост-компьютера по внешнему интерфейсу. После этого винчестер способен предъявить *паспорт диска* (для ATA) — 512-байтный набор данных, описывающих все доступные извне свойства устройства [6].

Команды для интерфейсов ATA и SCSI, доступные в *нормальном* (регулярном) режиме работы, описаны в 19.6 и 20.2 соответственно и более подробно — в [6]. Они включают операции чтения, записи, верификации секторов, поиска и некоторые вспомогательные операции. Все эти команды работают с областями данных секторов, что подразумевает предварительное выполнение низкоуровневого форматирования диска (см. 9.12). При получении команды микроконтроллер выполняет трансляцию *внешнего адреса* запроса, поступившего по интерфейсу, в адреса реальных секторов реальных поверхностей носителя. Трансляция выполняется по таблицам, загруженным в ОЗУ микроконтроллера и учитывающим текущую внешнюю (логическую) геометрию диска ATA (у SCSI такого понятия нет), размеры зон, а также переназначение физических секторов для обхода дефектных участков поверхностей.

Низкоуровневое форматирование (формирование структуры заголовков и блоков данных секторов) в нормальном режиме доступно только в системе команд SCSI. Команда форматирования интерфейса ATA доступна лишь для старых дисков, не поддерживающих зонную запись, и при совпадении внешней геометрии с реальной. Современные диски ATA низкоуровневое форматирование выполняют исключительно в специальном *технологическом режиме*, способ перехода в который зависит от фантазии разработчика дисков. Для этого, например, могут применять нестандартные команды, причем их использование может блокироваться специальными джамперами на устройстве. Общение с винчестером в технологическом режиме может производиться и через специальный последовательный интерфейс, в качестве которого иногда применяют стандартный интерфейс RS-232C, что позволяет вести диалог с винчестером, например, через COM-порт и эмулятор терминала на ПК. Если технологические команды доступны по обычному интерфейсу (ATA или SCSI), то производитель может предлагать собственные утилиты низкоуровневого обслуживания дисков. Низкоуровневое форматирование не затрагивает сервоинформацию, которая записывается на поверхности лишь в заводских условиях. Диск с поврежденными серводорожками теряет работоспособность.

ВНИМАНИЕ ----------

Низкоуровневые утилиты предназначены для конкретных моделей или семейств устройств конкретных производителей. Использование их с «чужими» дисками, как правило, блокируется (хотя бы предупреждением). Обход этой блокировки чреват необратимым (вне заводских условий) выходом винчестера из строя.

Помимо выполнения команд, поступающих от хост-компьютера, микроконтроллер постоянно решает задачи «жизнеобеспечения» винчестера. Хранение данных на магнитном носителе всегда сопровождается появлением ошибок, причин у которых может быть множество: дефект поверхности носителя, случайное перемагничивании участка носителя, попадание посторонней частицы под головку, неточность позиционирования головки над треком, колебание головки по высоте, вызванное внешней вибрацией (ударом) корпуса накопителя, «уход» различных параметров (из-за старения, изменения температуры, давления и т. п.). Независимо от причин ошибки должны быть выявлены и по возможности исправлены. Для контроля достоверности хранения поля данных применяется CRC-код, который позволяет фиксировать ошибки некоторой кратности, а ECC- код при большей избыточности — даже исправлять ошибки. Если сектор считывается с ошибкой, контроллер автоматически выполняет повторное считывание, и при случайности ошибки велик шанс правильного считывания сектора. Однако если ошибка вызвана, например, неточностью позиционирования головки на середину трека, связанной с «уходом» параметров, повторное считывание может и не дать положительного эффекта. Накопитель с шаговым приводом головки в этой ситуации может только повторить позиционирование — вернуться на нулевой трек и снова «дошагать» до нужного — иногда это помогает. У привода с подвижной катушкой больше возможностей поиска оптимального для считывания данных положения головки. Следящая система может, например, покачать головку относительно центрального положения, заданного сервометками, и найти точку, где данные читаются верно. Если данные так и не удается считать верно, контроллер обязан сигнализировать об этом установкой бита ошибки контрольного кода в байте состояния, на что программа может отреагировать сообщением вида «CRC Data Error». Однако на практике автору однажды довелось столкнуться с винчестером, который не сообщал об ошибках данных, — возможно, у него была неисправна схема контроля CRC-кода. «Уличить» этот накопитель в недостоверности хранения оказалось нелегким делом — все тесты, естественно, проходили успешно. Только при форматировании диска обнаружились ошибки, по которым и удалось добраться до виновника нестабильной работы компьютера (эта нестабильность скорее напоминала ошибки оперативной памяти или кэш-памяти).

Если контроллеру никак не удается достоверно прочитать записанные в сектор данные, такой сектор должен быть исключен из дальнейшего использования. Если этого не сделать, бесчисленные повторные попытки обращения к нему будут отнимать массу времени, а результата все равно не будет. На уровне накопителя отметка о дефектности блока делается в заголовке сектора, запись в который, как известно, производится только во время низкоуровневого форматирования. Встроенные контроллеры современных дисков сами обнаруживают дефектные секторы и вместо них подставляют резервные, так что пользователю дефектные секторы у дисков ATA и SCSI не видны (до некоторых пор). Хотя наличие дефектов, увы, неизбежно, и их число в процессе эксплуатации винчестера потихоньку может увеличиваться, внешне диск будет выглядеть бездефектно. Это означает, что обращение по любому внешнему адресу сектора будет выполняться без ошибок. Для скрытия дефектных секторов применяют

различные стратегии использования резервных областей. Резервные секторы могут располагаться в конце каждого физического трека, но пока основные секторы исправны, резервные не задействуются. Если какой-либо сектор перестает читаться, микроконтроллер пытается перенести его данные в резервный сектор и корректирует заголовки секторов, помечая дефектный сектор и назначая резервному номер замещенного сектора. В результате сектор с данным номером снова становится нормальным, однако при линейном обращении к цепочке секторов в общем случае диску может потребоваться дополнительный оборот из-за нарушения порядка следования секторов на треке. На графике скорости линейного чтения (записи) этот скрытый дефект проявится в виде небольшого локального провала. Правда, если микроконтроллер считывает в буферную память весь трек целиком, при чтении этот дефект может оказаться незаметным. Более хитрый способ скрытия дефектов (defective sector slipping) заключается в перенумерации всех секторов трека (естественно, с перемещением данных) после замены дефектного сектора резервным, с восстановлением оптимальной для данного устройства последовательности номеров. Если же на треке оказывается слишком много дефектных секторов (местного резерва уже не хватает), то выполняется переназначение всего трека на резервную область. Резервная область, как правило, выделяется на внутренних цилиндрах — их пользователю не показывают (в паспорте диска указывается объем без учета резервных треков). Это переназначение делается уже с помощью таблиц переназначения треков. На графике линейной скорости такое переназначение оказывается гораздо более заметным, чем переназначение в пределах трека, — здесь уже требуется дополнительное время на позиционирование головок. Конечно, со временем настанет момент, когда все резервные блоки будут использованы, и тогда появление видимого дефектного блока станет сигналом к замене накопителя или попытке его «оздоровления» с потерями в емкости.

Состояние диска можно оценить по графикам линейной скорости чтения-записи, которые строятся и выводятся на экран некоторыми тестовыми программами (рис. 9.9). В нормальной ситуации эти графики представляют собой «лесенку», постепенно спускающуюся с ростом номера цилиндра. Каждая ступенька отражает зону с одним числом секторов на треке. Некоторые модели дисков имеют иной характер графика — с небольшим «горбом» в середине или волнообразный — это следствие нестандартного подхода к трансляции адреса сектора в реальные координаты. График может иметь небольшие всплески и провалы (порядка единиц процентов) — следствие асинхронности работы компонентов тестируемой цепочки, состоящей из собственно диска, его контроллера, адаптера интерфейса, системной шины и процессора, на котором исполняется тестовая программа. Более крупные зазубрины и провалы свидетельствуют о трудностях, возникающих у устройств при выполнении операций, — ошибках позиционирования и чтения, а также о скрытых дефектах (переназначенных секторах и треках).

Скрытие дефектов вызывается даже простым выполнением теста записи (неразрушающего) для всей поверхности диска (если диску не удается записать сектор, он его перемещает). Списки дефектных блоков (треков) хранятся, как

правило, в двух таблицах. Одна из них (P-list) считается постоянной и формируется при выпуске винчестера. Другая, растущая (G-list), формируется во время эксплуатации автоматически. У нового винчестера она пустая. Пользователь может получить доступ к этим таблицам лишь с помощью специальных низкоуровневых утилит обслуживания дисков. Когда штатные резервные треки тоже будут исчерпаны, диску можно продлить жизнь, выполнив его низкоуровневое переформатирование на меньшую емкость и изменив паспорт диска. Это возможно с помощью специальных утилит, работающих с внутренними таблицами дефектов и конфигурации. У диска можно исключить совсем плохие поверхности — либо чисто программно, либо выполнив еще и физическую перекоммутацию головок. Все эти действия, естественно, должны отражаться в паспорте диска. При этом диск может значительно «потерять в весе», но выжить. В тонкости процесса восстановления дисков вдаваться не будем, ограничимся лишь ссылкой на разработчика и производителя аппаратно-программного комплекса PC3000 — http://www.acelab.ru.

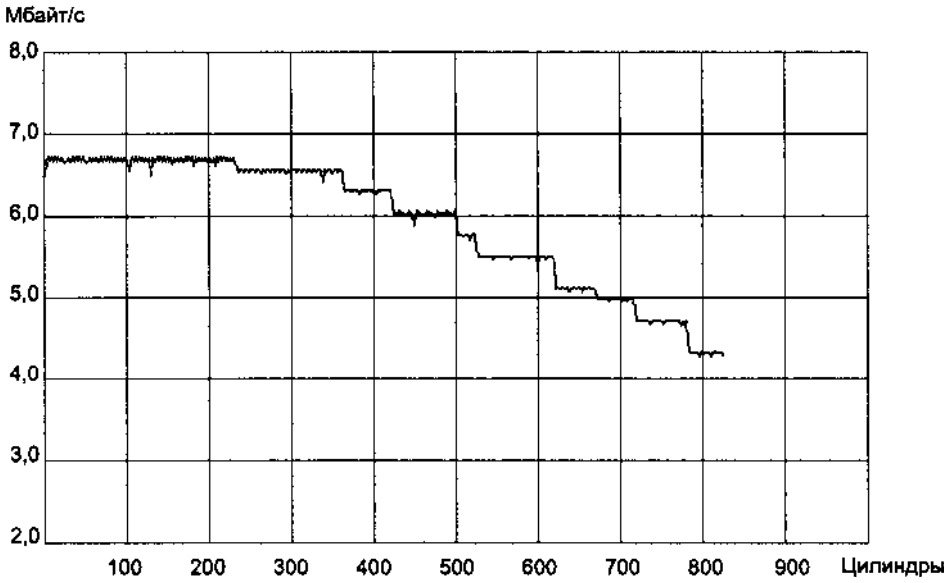


Рис. 9.9. График линейной скорости диска

Надежность считывания в значительной степени зависит от точности позиционирования головок относительно продольной оси трека. Позиционирование, обеспечиваемое сервоприводом с выделенной сервоповерхностью, может быть неоптимальным для каждой головки и требовать коррекции. Главным образом эта коррекция необходима из-за изменения рабочей температуры. «Умный» контроллер хранит карту отклонений для цилиндров и головок, которую он создает и периодически корректирует в ходе работы. Процесс автоматической *термокалибровки* (Thermal Calibration, T-Cal) накопителя инициируется встроенным контроллером и для системы случаен. Этот процесс заметен пользовате-

лю: винчестер, к которому нет обращений, вдруг «начинает жить своей жизнью», выполняя серию позиционирований (слышны характерные звуки). Во время термокалибровки доступ к данным накопителя приостанавливается, что не всегда допустимо. Накопители, предназначенные для мультимедийных целей, а также используемые как мастер-диски для записи данных на лазерный диск, должны обеспечивать довольно длительный непрерывный поток передачи данных. Их контроллеры не должны позволять себе приостановку из-за термокалибровки сеансов передачи данных.

Помимо термокалибровки есть еще один процесс, асинхронно запускаемый контроллером диска, — так называемое *свипирование* (sweeping): если к диску долгое время отсутствуют обращения, он перемещает головки в новое, случайным образом определенное положение. Этот процесс напоминает действие экранной заставки (screen saver) монитора и предназначен для выравнивания степеней износа различных частей поверхности диска.

В современных дисках начинают применять слежение за «высотой полета» головок во время записи, поскольку ее отклонение от номинала может привести к потере записываемых данных (впоследствии их не удастся достоверно считать). При обнаружении факта превышения допустимой высоты микроконтроллер повторяет операцию записи, что позволяет значительно повысить надежность хранения данных.

Дисковые накопители являются, пожалуй, той частью компьютера, отказ которой оборачивается самыми крупными убытками (если потерянные данные не имеют копий). Естественно, надежность их пытаются повышать всеми возможными способами, но отказы все-таки случаются. Отказы разделяются на предсказуемые и непредсказуемые. *Предсказуемые отказы* (predictable failure) появляются в результате постепенного ухода каких-либо параметров от номинальных значений, когда этот уход превысит некоторый порог. Если наблюдать за такими параметрами накопителей, как время разгона до заданной скорости, время позиционирования, процент ошибок позиционирования, «высота полета» головок, производительность (зависящая и от числа вынужденных повторов для успешного выполнения операций), количество задействованных резервных секторов и др., то становится возможным предсказание отказов. Сообщение об их приближении позволяет операционной системе и/или пользователю принять необходимые меры и предотвратить крупный ущерб. Для предупреждения отказов служит технология *S.M.A.R.T.* (Self-Monitoring, Analysis and Reporting Technology — технология самонаблюдения, анализа и сообщения), применяемая в современных накопителях. Эта технология, разработанная фирмой Seagate, имеет корни в технологиях IntelliSafe фирмы Compaq и PFA (Predictive Failure Analysis — анализ предсказуемых отказов) фирмы IBM.

Задачи слежения за параметрами накопителя возлагаются на контроллер, а программному обеспечению компьютера остается только периодически интересоваться, все ли в порядке в накопителе или грядет беда. Спецификации S.M.A.R.T. существуют в двух версиях — для интерфейсов ATA и для SCSI, которые различаются как по системам команд, так и по сообщениям состояния. Конечно, остаются и *непредсказуемые отказы* (non-predictable failure), которые

случаются внезапно. Чаще всего они вызываются разрушениями электронных схем под действием импульсных помех или механических частей, которые страдают от ударов. Снизить вероятность непредсказуемых отказов можно путем совершенствования технологий производства компонентов.

Технология S.M.A.R.T, позволяет следить за параметрами устройства, фиксировать критические события во внутренних журналах, расположенных в секторах служебных областей диска, считывать эти журналы, а также запускать тесты поверхности по команде от хост-компьютера. Тесты могут исполняться в разных режимах (см. 19.6), различающихся степенью отвлечения винчестера от выполнения своих «прямых обязанностей» (команд считывания и записи). Действия по восстановлению, например, плохо читаемых секторов выполняются по инициативе программы хост-компьютера, использующей результаты тестов S.M.A.R.T. Фирма Western Digital в своей технологии Data Lifeguard пошла дальше — действия по тестированию и восстановлению выполняются микроконтроллером винчестера по его инициативе. Микроконтроллер в автономном (off-line) режиме выполняет сканирование секторов при отсутствии команд после 8 часов работы шпинделя, если от хоста не поступает команд в течение 15 секунд. Секторы с исправимой ошибкой ECC проверяются на дефектность поверхности, и если дефекта нет, то сектор «освежается» путем перезаписи и в дальнейшем читается нормально. При обнаружении дефекта поверхности секторы заменяются резервными. Если подается команда от хоста, сканирование приостанавливается. Оно продолжится с того же места после 15 минут вращения и 15 секунд паузы между командами хоста. Такое фоновое сканирование и самовосстановление диска не только не замедляет обращения от хост-компьютера, но даже увеличивает производительность за счет снижения вероятности повторных считываний секторов, читающихся с неисправимой ошибкой.

Эта же фирма вводит в новые диски мониторинг температур с помощью термо- датчиков, расположенных в устройстве. О превышении первого порога устройство сообщает кодами ошибки 01/0B/01. Температура первого порога (по умолчанию — 60°C) может программироваться. Если слежение за температурой в S.M.A.R.T. разрешено, то каждые 25 минут значение температуры записывается в журнале S.M.A.R.T. (страница 2F, ее чтение вызывает немедленное обновление записи с замером температуры). По превышении порога частота обновления повышается (раз в 15 минут). По достижении второго порога (65 °C) появляется предупреждение о необходимости отключения (коды 01/0B/80). Если разрешено автоматическое отключение, то шпиндель останавливается, и для его последующего запуска требуется команда Start Unit.

Сменные магнитные диски большой емкости

Накопители на гибких магнитных дисках в классическом исполнении остановились на емкости носителя 1,44 Мбайт. Диски на 2,88 Мбайт распространения не получили, да такое повышение объема по нынешним меркам и несущественно. Для повышения емкости требуется радикальное увеличение плотности записи, в первую очередь за счет увеличения числа треков. Однако при этом не

обходимо точное позиционирование, какое на обычном шаговом двигателе (разомкнутая система) недостижимо. Решением проблемы стало совмещение технологий, а именно технологии магнитных записи-считывания и технологии оптического позиционирования. Результатом стали так называемые *гибкие магнитооптические диски* (floptical drives) — 3,5" НГМД сверхвысокой плотности. Первые модели имели емкость около 20 Мбайт (755 дорожек, 27 секторов по 512 байт), скорость вращения — 720 об./мин, интерфейс — SCSI, ATA или специальный адаптер, позволяющий использовать их как дисковод A. Накопитель совместим и с обычными дискетами емкостью 720 Кбайт и 1,44 Мбайт (2,88 Мбайт только по чтению). По удельной стоимости хранения информации при относительно небольшой емкости дискеты эти устройства не вызывали особого интереса.

Устройства *LS-120* (Laser Servo 120 Мбайт) имеют емкость дискеты 120 Мбайт — по 1736 информационных треков на каждой стороне с зонным форматом записи. У устройства интерфейс ATAPI, логическая геометрия — 960 цилиндров x x 8 головок x 32 сектора на трек. Физически на каждом треке размещается 51-92 сектора размером 512 байт. При лазерном позиционировании задействуется 900 сервотреков. Накопитель существенно дешевле конкурирующих с ним магнитооптических устройств, о которых речь пойдет далее, но удельная стоимость носителя гораздо выше. Скорость обмена достигает 200-300 Кбайт/с. Современные версии BIOS поддерживают это устройство и позволяют включить его в последовательность загрузки устройств. Устройство LS-120 дает возможность отказаться от использования стандартного дисковода (совместимо с дискетами 1,44 Мбайт по чтению и записи). Как и обычные дискеты, носители для LS-120 чувствительны к сильным магнитным полям.

Популярные в свое время накопители *Zip* фирмы Iomega имеют емкость носителя 100 и 250 Мбайт, причем устройства на 250 Мбайт совместимы с носителями на 100 Мбайт (но не наоборот). Носитель — гибкий диск, помещенный в пластмассовый картридж формата 3". Накопитель несколько дороже LS-120, зато носители дешевле, правда, совместимости с обычными дискетами нет. Интерфейс — IDE, SCSI, USB или LPT. Для устройств SCSI и USB предлагаются переходники на PC Card, есть и варианты подключения к FireWire (скорость до 2 Мбайт/с). Среднее время доступа — 29 мс. Для устройств на 100 Мбайт характерна скорость обмена до 1,4 Мбайт/с, для 250 Мбайт — до 2,4 Мбайт/с. Эта скорость достигается лишь на быстрых интерфейсах (SCSI и ATA), для LPT и USB скорость ограничивается на уровне 0,8-1 Мбайт/с. Реальная средняя скорость — 750-1000 Кбайт/с. Время короткого форматирования носителя (без верификации) — 10 с, длинного (с верификацией) — 10 мин. Накопители выдерживают 10 000 установок носителей, картридж — до 2000. Накопители Zip широко использовались как устройства архивации и переноса данных. Благодаря невысокой цене они были популярны как у корпоративных, так и у «домашних» пользователей.

С накопителями Zip связана история о так называемых «щелчках смерти» — щелчках дисководов при чтении некоторых «зараженных» дискет. Щелчки возникают, когда накопитель обнаруживает неисправимую ошибку чтения, — при

этом он выводит головки из картриджа на парковочную площадку, очищая их, и вводит повторно. Само по себе это щелканье безобидно, но является признаком плохой записи (повреждения сервометок, низкоуровневого формата, служебных областей). Носитель, вызывающий щелканье одного накопителя, скорее всего, заставит щелкать и другие — отсюда миф о «заразности» щелчков. Крайне редко, но все-таки встречаются и действительно «смертоносные» носители, у которых поврежден (порван, помят) наружный край диска. Если такой носитель вставить в дисковод, при вводе головок он может их буквально срубить. После этого дисковод будет долго пытаться «найти» (уже без головок!) данные на носителе, щелкая при повторных введениях-выведениях остатков головок. Перенос такого носителя на другой дисковод наверняка погубит и его, поскольку 3000 об./мин — скорость нешуточная.

ВНИМАНИЕ ----------

Если дисковод Zip при считывании начинает щелкать, следует внимательно осмотреть край носителя, отведя защитную шторку и проворачивая диск вручную на полный оборот. Диск с малейшими обнаруженными дефектами (зазубринами, замятиями и т. п.) должен быть сразу изъят из обращения.

Существует и карманный вариант дисковода Zip — *PocketZip Mobile Drive* с емкостью диска 40 Мбайт. Помимо внешнего интерфейса, он снабжен устройством считывания флэш-карт CompactFlash+ и SmartMedia. Устройства ZipCD к обычным картриджам Zip/Jaz отношения не имеют — это просто устройства CD-RW.

Накопители *Jaz* той же фирмы Iomega являются развитием Zip, но уже на твердом носителе: емкость — 1 Гбайт у первых моделей — в дальнейшем увеличена до 2 Гбайт. Интерфейс — IDE, SCSI, возможны переходники на USB или LPT. Пиковая скорость достигает 20 Мбайт/с, средняя — 7,5 Мбайт/с, время доступа — около 16 мс. Накопители Jaz распространены не так широко, как Zip (сказывается цена, особенно на носители), они хорошо подходят для архивации больших объемов данных.

Накопители на магнитных дисках могут иметь различные уровни сменяемости. Обычный накопитель на жестком диске (винчестер) устанавливается на компьютер надолго, и для его подключения требуется открыть корпус системного блока и отключить питание. Существуют и накопители, допускающие «горячую» замену (hot swap), для которой не требуется отключение питания и имеется специальный конструктив, позволяющий устанавливать и снимать винчестер со стороны лицевой панели, не разбирая компьютер. Однако такая замена производится в основном в сервисных целях. *Съемные винчестеры* — устройства с аналогичными свойствами, но предназначенные для переноса или безопасного (в смысле конфиденциальности) хранения данных. Выпускаются недорогие переходники Mobile Rack, позволяющие использовать в качестве съемного обычный ATA-диск. При этом накопитель 3,5" устанавливается во внешний отсек 5". Однако следует помнить, что обычные накопители не рассчитаны на вибрацию и удары, опасность которых при частых переносах устройств повышается. Винчестер же, тщательно закутанный в мягкую упаковку, занимает

много места. Поэтому больший интерес представляют накопители именно со съемными носителями.

Диски Бернулли (Bernoulli removable media drive) имеют привод размером 5", дисковод использует гибкие диски 3,5" в жесткой кассете. Объем составляет 35-150 Мбайт. При вращении диска со скоростью 3600 об./мин возникает эффект Бернулли, поддерживающий головки. По скоростным параметрам диски Бернулли близки к винчестеру. Кассета устойчива к внешним воздействиям. Интерфейсы: внутренние — IDE, SCSI; внешние — SCSI, LPT-порт. Производитель — Iomega. Встречаются редко.

Устройства с *кассетными жесткими дисками* (removable media drives), выпускаемые фирмой SyQuest, используют картриджи, в которых размещены и диски (1-2 пластины), и головки. Они имеют все атрибуты современных винчестеров — встроенную сервоинформацию, зонную запись, стандартный размер сектора 512 байт, небольшое время поиска. В устройствах серии EZ применяются картриджи PDC (Power Disk Cartridge), выпускаемые фирмами SyQuest, No- mai, Kao Infosystems, Maxell, Polaroid и Xyratex с емкостями 135, 230, 270 и 540 Мбайт. В картридже расположена одна пластина формата 3,5". Устройство *EZFlyer* на 230 Мбайт выпускается во внешнем (LPT) и внутреннем (EIDE, в отсек 3,5") исполнениях, раньше выпускались и внешние с интерфейсом SCSI. Время доступа — 13,5 мс, скорость вращения — 3600 об./мин, скорость обмена — до 2,4 Мбайт/с (IDE). Устройство совместимо с картриджами на 135 Мбайт, используемыми в ранее выпускавшихся устройствах *EZ135 с* похожими свойствами.

Устройство *SparQ* использует специальный картридж 1 Гбайт (тоже с одной пластиной 3,5"). Выпускается во внешнем (LPT) и внутреннем (EIDE, в отсек 3,5") исполнении. В устройстве *SyJet* применяется картридж емкостью 1,5 Гбайт с двумя пластинами 3,5". Во внешнем исполнении имеет интерфейсы LPT или SCSI, во внутреннем — EIDE, устанавливается в отсек 3,5". Устройства SparQ и SyJet никак не совместимы.

По параметрам производительности кассетные жесткие диски большой емкости вполне сопоставимы с винчестерами: среднее время доступа — 12 мс, время перехода на соседний трек — 1,5 мс, скорость вращения — 5400 об./мин, длительная скорость обмена — 3,7-6,9 Мбайт/с (для IDE и SCSI). Однако высокая цена картриджей ограничивает сферу их применения — они эффективны, когда выдвигаются жесткие требования к производительности (особенно при записи). Судя по отсутствию изменений в объемах носителей и ценах, кассетные диски фирмой SyQuest развиваться перестали (да и жива ли фирма?).

Магнитооптические диски

В устройствах с магнитооптическими дисками, *МОД* (Magneto-Optical Drives, MOD), в процессе записи и чтения магнитного носителя используется лазер. МОД сочетают практически неограниченное число перезаписей, свойственное магнитным носителям, с чрезвычайно надежным хранением записанной информации. Устройство привода МОД укладывается в традиционную схему — име

ются шпиндель, который вращает диск, и головка, перемещаемая над поверхностью диска системой позиционирования. Расстояние от головки до поверхности диска — около 1 мм, что исключает риск касания поверхности диска (отсюда очень высокая надежность). На головке расположены лазер и оптическая система, фокусирующая луч на магнитном слое носителя. На диске поверх магнитного слоя имеется слой субстрата, защищающего магнитный слой от механических воздействий. Луч на поверхности диска (субстрата) дает пятно размером около 1 мм, так что пылинки и микроцарапины не оказывают существенного влияния на происходящие процессы. На магнитном слое, находящемся под прозрачным слоем субстрата (толщина субстрата — около 1,2 мм), за счет фокусировки пятно уменьшается уже до микронных размеров, что и определяет достижимые (в плане уменьшения) размеры хранящих «ячеек». Расстояние между треками у дисков 128 Мбайт — 1,6 мкм, у 540/650 Мбайт — 1,1 мкм. Один бит на треке у дисков 128 Мбайт занимает 0,52 мкм, у 540/650 Мбайт — 0,31 мкм. Принципы записи и чтения МОД весьма своеобразны.

Запись осуществляется термомагнитным способом: магнитное поле головки способно перемагнитить только микроскопическую зону носителя, разогреваемую лазерным лучом до температуры выше точки Кюри (порядка 200 °С). Зона, вышедшая из-под луча, «замораживает» полученное состояние намагниченности. Традиционно в магнитооптике используют двухпроходную запись. Для того чтобы записать информацию в секторе, после позиционирования головки за первый оборот сектор стирают. Для этого головка создает постоянное магнитное поле, а лазер включается на полную мощность, когда под ним проходит требуемый сектор (секторы). В результате все «засвеченные» области данных переводятся в состояние с одинаковым направлением намагниченности. На следующем обороте выполняется собственно запись — направление магнитного поля головки меняется на противоположное, и формируются мощные импульсы лазера над теми точками, состояние которых нужно изменить, чтобы закодировать требуемую информацию. В результате именно эти ячейки примагничиваются. Для большей достоверности на третьем обороте выполняется верификация — считывание записанной информации. В принципе, возможна и однопроходная запись, если модулировать магнитное поле (переключать его направление) над каждой битовой областью. Однако в данной технологии (при большом расстоянии головки от диска, которое уменьшать не хотят принципиально) из-за явления индукции магнитная модуляция на больших скоростях записи требует непомерных затрат энергии. Модуляция магнитного поля (Magnetic Field Modulation, MFM) применяется при записи мини-дисков в аудио- технике (там не требуется высокая скорость записи). Однопроходная запись для компьютерного применения основана на технологии LIMDOW, о которой рассказывается далее.

Считывание информации с магнитного слоя носителя выполняется тоже с помощью лазера (при малой мощности излучения) и основано на эффекте Керра — изменении поляризации света под действием магнитного поля. Отраженный луч проходит через поляризационную оптику, в результате на фотоприем- ник приходит луч, интенсивность которого модулирована (по амплитуде) в со

ответствии с записью на магнитном слое. Разрешающая способность оптики и фотоприемника определяет достижимую плотность хранения информации — записать можно и более мелкие ячейки, но считать их не удастся. Повысить разрешение при считывании позволяет технология MSR (см. далее).

Магнитооптические диски организованы так же, как и магнитные, — у них имеются дорожки, разбитые на секторы, только нумерация дорожек начинается от центра диска. Размер сектора может быть стандартным (512 байт данных) или увеличенным (2048 байт данных). Больший размер сектора снижает долю накладных расходов на служебную информацию. Количество секторов на треке переменно — здесь тоже применяется зонная запись. Помимо пользовательской области, на диске в каждой зоне имеются служебные и резервные области, позволяющие переназначать дефектные секторы прозрачно для пользователя. Встроенный контроллер накопителя МОД выполняет трансляцию физической геометрии (с зонным форматом) в логическую и имеет внутренние средства для переназначения дефектных блоков.

Магнитооптические диски бывают двух размеров — 5,25" (двусторонние) емкостью 650 Мбайт, 1,3, 2,6, 4,6 Гбайт и 3,5" (односторонние) емкостью 128, 230, 540, 640 Мбайт и 1,3 Гбайт.

Магнитооптические устройства ввиду стандартизации носителя являются распространенным средством не только архивации, но и обмена данными. Носители стандартизованы ISO/IEC (для 3,5" дисков стандарт 15041:1997), что обеспечивает совместимость дисков и устройств разных производителей.

Существует ряд градаций объемов МОД, обозначаемых через кратность объему начальной модели диска (табл. 9.7). Диски различаются плотностью размещения треков; методами модуляции — PWM (Pulse Width Modulation — широтно-импульсная модуляция) для 1x и 2x, PPM (Pulse Position Modulation — позиционно-импульсная модуляция) для 4x; методами кодирования — RLL 2.7 для 1x и 2x, RLL 1.7 для 4x и 5x; а также организацией зонного формата записи. Устройства и диски имеют обратную совместимость: устройства большей емкости могут работать и с дисками меньшей емкости. Устройства 4x (5x) обычно могут полноценно работать с дисками 2x и читать диски 1x.

Таблица 9.7. Емкость магнитооптических дисков

Плотность	Диск 130 мм (5")	Диск 90 мм (3,5")
1x	560/650 Мбайт	128 Мбайт
2x	1,2/1,3 Гбайт	230 Мбайт
4x	2,3/2,6 Гбайт	–
5x	–	540/640 Мбайт
10x	4,6 Гбайт	1,3 Гбайт

Для одних и тех же устройств могут указываться два значения емкости, например 540/ 640 Мбайт. Меньшее значение соответствует форматированию со стандартным (512 байт) размером сектора, большее — с увеличенным (в данном случае до 2048 байт).

Приводы МОД выпускаются во внешнем и внутреннем исполнениях. Поскольку при записи происходит значительное тепловыделение, в ряде случаев пред-

почтительно внешнее исполнение устройств. Устройства имеют интерфейс SCSI, IDE или Fibre Channel, обеспечивающий скорость передачи, достойную МОД. Для подключения к LPT и USB обычно применяются переходные адаптеры SCSI или IDE.

Благодаря малому размеру диска наиболее популярны устройства формата 3,5", для которых уже достигнута емкость 1,3 Гбайт. Магнитооптические накопители и диски производятся множеством фирм, лидером в производстве устройств 3,5" является Fujitsu. Современные устройства имеют скорость вращения 3000-4500 об./мин, что обеспечивает малую задержку доступа (7 мс) и среднее время поиска 20-30 мс. Скорость передачи данных даже у одного устройства зависит от емкости диска — диски большей емкости работают быстрее, поскольку у них больше секторов на треке. Скорость передачи на начальных цилиндрах, которыми в МОД являются внутренние цилиндры, заметно ниже, чем на внешних — следствие зонной записи (ZBR). Для магнитооптики с традиционной технологией характерно значительное различие скоростей записи и считывания — запись происходит существенно медленнее из-за необходимости двух-трех проходов над одним и тем же треком. По скоростным характеристикам считывания приводы МОД находятся между накопителями на гибких (типа ZIP) и жестких Jaz) магнитных дисках. Время доступа (позиционирования) упирается в массу головки (инерционность).

Более прогрессивная технология записи *LIMDOW* (Light Intensity Modulation Direct OverWrite — непосредственная перезапись с модуляцией интенсивности луча) позволяет исключить проход стирания, чем повышается скорость записи в 2 раза (в 1,5 раза, если дополнительно выполняется верификация). Диски для такой технологии записи имеют более сложную структуру — несколько магнитных слоев с различающимися свойствами (с разной температурой Кюри), в которых имеет место магнитное взаимодействие элементов соседних слоев. Верхний слой (на который падает луч) является информационным, состояние намагниченности нижнего слоя задается при инициализации диска (на этапе изготовления) и остается неизменным. Головка при записи также создает постоянное магнитное поле, но мощность лазера модулируется. Поле от головки стремится перемагнитить информационный слой в одном направлении (нуль), а поле нижнего слоя — в противоположном (единица). Для записи нуля вырабатывается сильный импульс лазера, для записи единицы — более слабый. В процессе остывания результатом взаимодействия слоев является запомненное направление намагниченности информационного слоя, отвечающее мощности импульса. Для того чтобы эти процессы были не слишком критичны к точности температуры нагрева, диски имеют довольно сложную многослойную структуру, что, естественно, заметно сказывается на их стоимости. Ускоренную запись позволяют выполнять только накопители, поддерживающие эту технологию, и только для специальных дисков, имеющих пометку «OW». Диски OW, по сравнению с обычными, обладают еще и большим гарантированным сроком хранения. Технология LIMDOW применяется только к дискам 540/ 640 Мбайт и реализована на приводах DynaMO, GigaMO фирмы Fujitsu.

Форматирование МОД выполняется специальной утилитой, входящей в комплект драйверов устройства (MO Formatter). Низкоуровневое форматирование

МОД (LLF) выполняется в процессе производства, в дальнейшем пользователь может выполнить его с помощью утилиты. Поводом для выполнения LLF может быть потеря производительности диска (большое число ошибок). При низкоуровневом форматировании задается и размер сектора. В зависимости от объема диска и производительности привода низкоуровневое форматирование может занимать от нескольких минут до получаса. Форматирование верхнего уровня выполняется под конкретную файловую систему. В целях экономии времени имеет смысл приобретать предварительно форматированные диски (preformatted MO Disk), для которых выполнены обе стадии форматирования.

Форматирование магнитооптических дисков на верхнем уровне может выполняться в *стиле дискет* (super floppy) или в *стиле винчестера* (PC/AT HDD). В первом случае диск представляется в виде очень большой дискеты, в его нулевом логическом блоке содержатся загрузчик и дескриптор носителя (без таблицы разделов). Дискета, безусловно, подразумевает сменяемость, но в этом режиме МОД может работать только с драйвером, загружаемым уже операционной системой. Этот формат поддерживается и Windows, и OS/2. При форматировании в стиле винчестера диск начинается с таблицы разделов, и для системы выглядит как жесткий диск, который в случае интерфейса SCSI может обслуживаться BIOS хост-адаптера без всяких загружаемых драйверов. Это позволяет даже загружать ОС с МОД, но потом не всякая BIOS «догадается», что у этого «жесткого диска» возможна смена носителя. Загрузка с МОД ATAPI возможна далеко не во всех версиях BIOS. Некоторые версии Windows 9x не позволяют использовать разделы сменного диска, кроме первого, — для них приходится выбирать формат super floppy. Некорректная отработка смены носителя может привести к потере данных, когда после смены диска ОС не обновляет в ОЗУ дескриптор диска и таблицы размещения файлов. Заметим, что смена носителей для Macintosh отрабатывается иначе, чем для PC, поэтому накопители МОД с интерфейсом SCSI имеют переключатель «Mac-PC», который должен быть корректно установлен.

ВНИМАНИЕ ----------

При использовании сектора размером 2048 байт при загрузке с МОД могут возникнуть проблемы — в отличие от специального загружаемого драйвера, BIOS может «не догадаться» учесть нестандартный размер сектора. Кроме того, с этим размером сектора не работают утилиты DriveSpace и FDISK.

Магнитооптические диски устойчивы к сильным внешним магнитным полям, поскольку при нормальной температуре коэрцитивная сила носителя велика; они не боятся перепадов температуры, солнечного света, вибрации. Гарантированное время жизни диска — около 40 лет.

У магнитооптических дисков, в принципе, есть еще возможности повышения плотности хранения информации. Технология *магнитного суперразрешения* (Magnetic Super Resolution, MSR) позволяет увеличить плотность хранения данных по сравнению с чисто оптическими технологиями при том же размере пятна луча. При записи температура нагрева распределяется по пятну неравномерно — в центре она значительно выше. Поэтому пороговая температура достигается

не на всей площади пятна, а только в меньшей центральной области, и только в этой области происходит перемагничивание. Поскольку поверхность носителя движется относительно головки, нагретая область представляет собой эллипс, вытянутый вдоль дорожки. Процесс считывания отличается от классического — его технология напоминает LIMDOW. Диск MSR имеет несколько магнитных слоев (в упрощенной схеме — три). Для оптического считывания виден только верхний слой — *слой считывания* (reading out layer), информация записывается на нижний — *слой записи* (recording layer), а между ними расположен *промежуточный слой* (intermediate layer), который при определенной температуре способен транслировать состояние нижнего слоя в верхний, видимый. Считывание производится довольно мощным лучом, который воздействует на магнитные свойства верхних слоев, не «задевая» (по температуре) в присутствии постоянного магнитного поля свойства нижнего слоя, хранящего данные. Сверхразрешение достигается благодаря эффекту двойного маскирования: в центре пятна луча, где температура высокая, промежуточный слой теряет магнитные свойства и не передает состояние слоя хранения на выходной слой, который намагничивается внешним полем. На краю пятна работает низкотемпературная маска — здесь промежуточный слой намагничивается внешним полем перпендикулярно слою хранения, и слой считывания опять-таки оказывается намагниченным внешним полем головки. И только в узкой области умеренной температуры состояние слоя хранения транслируется на видимый слой, благодаря чему сигнал принимается от области, гораздо меньшей, чем пятно луча лазера, с хорошим отношением сигнал/шум.

Дальнейшим развитием технологии MSR являются системы MAMMOS (Magnetic AMplified Magneto Optical System) и DWDD (Domain Wall Displacement Detection), находящиеся еще в стадии лабораторных исследований.

9.8. Оптические диски — CD, DVD, PD

В оптических дисках хранение информации основано на изменении оптических свойств (в основном, степени отражения) поверхности носителя. В процессе считывания при освещении трека лазерным лучом возникает модуляция интенсивности отраженного луча, воспринимаемого фотоприемником. В модулированном луче закодирована двоичная информация, размещенная на треке. На этом принципе основаны диски CD, а также их «потомки» — DVD и стоящие особняком диски PD. Подробно внутренняя организация компакт-дисков различного назначения рассмотрена в [6], здесь же остановимся на прикладной стороне.

Диски CD - CD, CD-R, CD-RW

В компьютер компакт-диск (Compact Disk, CD) пришел из цифровой аудиозаписи. *Аудиокомпакт-диски*, называемые *Audio-CD,* были разработаны фирмами Sony и Philips в 1982 году. Диаметр компакт-диска — 120 мм, толщина — 1,2 мм. Как и грампластинки, диски имеют одну спиральную дорожку, но, в отличие от грампластинок, начинающуюся от центра диска. Эта спираль имеет 22 188 вит

ков (поперечная плотность — около 600 витков на 1 мм) и длину более 5 километров. Область с диаметром 46-50 мм является *вводной* (Lead-In Area, LIA), а область 116-117 мм — *выводной зоной* (Lead-Out Area, LOA). Область между этими зонами называется *программной* (program area). Дорожка представляет собой цепочку «ямок» (pits) в прозрачной основе диска, за которой расположен светоотражающий слой. Поперечный шаг витков спирали — 1,6 мкм, ширина дорожки (ямок) — 0,5 мкм, глубина ямок — 0,125 мкм. Края ямок соответствуют двоичным единицам канальной информации, кодирующей записанную на диске полезную информацию. Участок без изменения глубины (как ямка, так и «равнина») соответствует двоичным нулям, число нулевых битов определяется длиной этого участка. Длина ямок лежит в пределах 0,83-3,56 мкм. Для считывания применяют инфракрасный лазер с длиной волны 780 нм (в воздухе). Глубина ямок выбрана равной 1/4 длины волны луча лазера в прозрачном материале основы диска. Благодаря этому луч, отраженный от дна ямки, возвращается в приемник в противофазе с лучом, отраженным от поверхности (он в обе стороны проходит лишние 1/2 волны), — они взаимоуничтожаются, что повышает контрастность восприятия ямок. Для выравнивания продольной плотности записи диск вращается с переменной угловой скоростью, а привод обеспечивает постоянство линейной скорости носителя, проходящего под головкой. Этим обусловлено большое время доступа, поскольку дополнительное время уходит на разгон (торможение) диска при достаточно быстром перемещении головки. Диск способен хранить информацию 74 минут звучания стереосигнала с частотой квантования 44,1 КГц и 16-разрядными выборками. На диске используется только одна поверхность. Каждое музыкальное произведение (или его часть) записывается на одном треке, всего на диске может быть до 99 треков. Во вводной зоне размещена таблица содержимого (Table Of Content, TOC), в которой описаны координаты каждого трека и выводной зоны. Внутри каждого трека могут быть расставлены *индексы*, маркирующие определенные точки в записи, — их можно быстро находить и выполнять воспроизведение с заданного места. Помимо основного информационного канала, несущего звуковую информацию, на диске имеются служебные *субканалы* (каналы субкода P, Q, R, S, T, U, V, W) с пропускной способностью 1/192 от основного канала каждый. Из этих субканалов широко используются лишь P и Q, которые служат для навигации по диску, хранения краткой информации (идентификаторов) о содержимом диска и треков, а также хранения TOC.

В таком виде появились и первые компакт-диски, ориентированные на хранение данных, для считывания которых применяются приводы CD-ROM (Compact Disk Read Only Memory — компакт-диск, предназначенный только для чтения). Они базируются на тех же методах организации и кодирования данных на физическом уровне, что и аудиодиски, но отличаются способом организации и использования хранимой информации.

Спиральный трек CD-ROM, как и CD-DA, начинается с *вводной зоны* (LIA), в которой размещается низкоуровневая таблица TOC с координатами начала всех треков и выводной зоны. За ней следует *зона данных*, которая может содержать до 99 *треков.* Завершает диск *выводная зона* (LOA). Адресация инфор

мации на треке происходит от аудиодисков: адрес информационного блока состоит из номера *минуты* (0-73), номера *секунды* (0-59) и номера *фракции* (0-74). Каждая фракция (блок данных) несет 1/75 секунды аудиоданных: при частоте 44,1 кГц два 16-битных канала требуют 44100 x 2 x 2/75 = 2352 байт. Такой блок на треке записывается как последовательность 98 *кадров* (frame). В каждом кадре имеется поле синхронизации, собственно данные, контрольный код и субкод. *Контрольный код* кадра обеспечивает обнаружение и исправление ошибок с уровнем, вполне приемлемым для воспроизведения аудиозаписей, но недостаточным для надежного хранения данных. В аудиозаписи кадр, в котором ошибка обнаружена, но исправить ее не удалось, игнорируется (воспроизводятся интерполированные данные). Для хранения данных такой подход неприемлем. *Субкод* формирует *субканалы* (P, Q, W), используемые для навигации и служебных целей. *Поля данных* 98 кадров блока в сумме составляют 2352 байт, защищенных (с исправлением ошибок) на уровне кадров.

Блок (сектор) CD-ROM задействует эти 2352 байта следующим образом. В начале блока находится *заголовок* (16 байт): 12-байтное поле синхронизации, 3-байтный адрес (Min:Sec:Frac), байт с номером режима данного сектора. *Режим* (CD-ROM Mode nn) определяет использование поля данных блока: 00 — нули (нет данных), 01 — 2048 байт пользовательских данных и дополнительные контрольные коды, 02 — 2336 байт «сырых» данных (без дополнительной защиты от ошибок).

В записываемых дисках (CD-R и CD-RW) по сравнению с печатными дисками (CD-DA и CD-ROM) на треке, помимо вводной (LIA) и выводной (LOA) зон, а также зоны данных (Program Area, PA), имеются дополнительные:

- PCA (Power Calibration Area) — зона для калибровки мощности лазера (в этой зоне делаются пробные записи);
- PMA (Program Memory Area) — зона для промежуточного хранения TOC (координат начала и конца треков) сеанса записи. При закрытии сеанса эта информация переписывается в LIA данного сеанса.

Объем стандартного диска CD-DA (74 минуты) составляет 74 x 60 x 75 =

- 333 000 блоков. Если в блоке использовать по 2048 байт (с дополнительной защитой), то это составляет 681 984 000 байт; если по 2336 байт (без дополнительного ECC) — 777 888 000 байт.

Миниатюрные диски *(CD Single)* диаметром 80 мм имеют емкость около 200 Мбайт. Для этих дисков на лотках приводов CD имеется небольшое углубление, обеспечивающее возможность их использования наравне с «большими». Их не следует путать с мини-дисками (Sony MiniDisk), являющимися перезаписываемыми магнитооптическими дисками.

Скорость передачи данных стандартного аудиодиска составляет 75 блоков в секунду, в CD-ROM (а также CD-R и CD-RW) эта скорость называется *1x* и при размере 2048 байт составляет 150 Кбайт/с (1К = 1024). Для устройств хранения такая скорость маловата, и были определены скорости 2x, 4x, 8x ..... 56x. При
воды со скоростями до 12-16x обеспечивают постоянную линейную скорость

по всей рабочей поверхности[1]. Высокоскоростные приводы для режимов выше 12-16х обеспечивают постоянную угловую скорость. Более высокая кратность скорости реально обеспечивается только на внешней части спирали (напомним, что это ближе к ее концу).
Приводы CD имеют большую величину *времени доступа* — 75-500 мс; такая медлительность вызывается и необходимостью разгона и торможения диска при доступе к различным участкам носителя.
Поначалу технология записи на оптические диски была очень дорогой, и в компьютерах использовали только устройства чтения и диски CD-ROM. Впоследствии были разработаны диски CD-R (Recordable CD — записываемый компакт-диск), записываемые пользователем, и устройства записи (рекордеры) CD-Writer, CD-Recorder (естественно, способные и считывать информацию). Эти диски и устройства обеспечивали лишь *однократную запись*, их еще называли *CD-WORM* (Write Once, Read Many — однократная запись, многократное чтение) или *CD-WO* (Write Once). Устройства с возможностью *многократной записи* на оптический диск первоначально назвали *CD-E* (Erasable — стираемые). Однако с маркетинговой точки зрения такое название показалось непривлекательным (от того, что данные могут быть стерты, восторг мало кто испытывает), и его заменили на *CD-RW* (Rewritable CD — перезаписываемый компакт- диск), хотя физическая суть та же. Особую привлекательность CD-R и CD-RW придает совместимость этих дисков с обычными приводами CD-ROM и даже аудиоплеерами[2]. Существуют также и оптические диски PD (Phase change Disk), допускающие многократную перезапись, но никак не совместимые с приводами CD-ROM. Правда, имеются комбайны CD/PD, работающие с разными типами дисков. Более новые и емкие диски DVD (Digital Video Disk — цифровой видеодиск, или Digital Versatile Disk — универсальный цифровой диск) совместимости с CD не имеют. Но и здесь появились комбайны, позволяющие работать и с DVD, и с CD.

Носители информации CD

Оптические диски CD-ROM, CD-R и CD-RW имеют прозрачную поликарбонатную (пластиковую) основу, над которой расположен хранящий информацию слой, защищенный сверху лаком. На верхнюю сторону этого «пирога» может быть нанесена этикетка. Хранящий слой расположен ближе всего к верхней стороне; механические повреждения (царапины, вмятины) с верхней стороны чаще приводят к неисправимым ошибкам чтения. Царапины, как и пылинки, с нижней стороны, через которую светит лазер, не так страшны — через них проходит луч с еще довольно большим диаметром пятна (порядка 1 мм). Луч фокусируется в точку микронных размеров уже на самом хранящем слое, так что мелкие дефекты на внешней поверхности не оказывают существенного влияния на оптические процессы. Устройство хранящего слоя может быть различным:

[1] Для CD-DA (CD-ROM 1x) скорость вращения составляет примерно 200-400 об./мин.

[2] Старые модели приводов не работают с CD-RW.

- Штампованные (печатные) диски CD имеют рельефную верхнюю сторону прозрачной основы, покрытую светоотражающим напылением. Ямки (pits) и ровные участки (lands) трека дают разную интенсивность отраженного луча, которая регистрируется фотоприемником. Штампованные диски изготовляются на специальном заводском оборудовании. Исходная информация для штампа берется с записанного мастер-диска, с которого за несколько технологических этапов получают пресс-формы.
- Однократно записываемые диски (CD-R) имеют покрывающий основу слой органического красителя, поверх которого нанесено светоотражающее напыление (золото или сплав серебряного цвета). При записи выжигаются фрагменты красителя, в результате отраженный луч также будет промодулирован по интенсивности.
- Перезаписываемые диски (CD-RW, они же CD-E) под отражающим слоем имеют регистрирующий слой, который может менять свое состояние между поликристаллическим и аморфным. Прозрачность слоя зависит от его состояния. При перезаписи состояние отдельных участков изменяется: в зависимости от степени нагрева участка лучом записывающего лазера при остывании фиксируется то или иное его состояние. В отличие от печатных дисков и CD-R, отражающих около 70 % мощности падающего луча, диски CD-RW обладают существенно меньшей отражающей способностью.

Помещение информации на диски может производиться как печатью с матрицы (наподобие тиражирования грампластинок), так и непосредственной записью на носитель в устройстве CD-R или CD-RW, о котором речь пойдет далее. Диски, изготовленные матричным способом, имеют себестоимость, измеряемую единицами центов. Индивидуально записанные диски обходятся дороже — порядка 0,5-1 доллара. Независимо от способа записи оптические диски могут быть прочитаны на любом устройстве считывания, поддерживающем данный формат записи (конечно, при наличии качественного носителя и записывающей аппаратуры). Однако между штампованными и записанными дисками все же имеется некоторая разница. Перезаписываемые диски CD-RW по сравнению с CD и CD-R при считывании дают меньшую амплитуду сигнала. По этой причине приводы без автоматической регулировки чувствительности приемника (старые модели, выпущенные до 1998 года, включая ряд моделей 8x) не могут считывать диски CD-RW. На способность привода читать CD-RW указывает логотип «MultiRead».

Для упрощения записывающей аппаратуры на *болванке* (target) — чистом диске для записи — по всей поверхности при изготовлении наносится спиральная дорожка разметки (pregroove). Разметка отпечатана на верхнем слое поликарбонатного субстрата, по ней при записи наводится головка. Эта дорожка, по которой при записи диск разбивается на кадры, содержит коды разметки диска по времени (Actual Time In Pregroove, ATIP). На этой же дорожке имеется информация о требуемой мощности лазера и возможной скорости записи. Скорость записи зависит как от диска, так и от привода. При попытке записи на диск со скоростью большей, чем гарантированная, четкость изменения оптических свойств участков ухудшается и диск может оказаться нечитаемым.

Болванки для записи имеют маркировку типа:

- CD-R, или Compact Disc Recordable — диски с однократной записью, подходят для устройств CD-R и CD-RW;
- CD-RW, или Compact Disc Rewritable — диски с многократной записью, подходят только для устройств CD-RW.

На диске также указывается диапазон возможных скоростей записи (минимальная и максимальная скорости). Эта информация печатается на его этикетке (для пользователя) и заносится в служебную область разметки (для привода). Высокоскоростные болванки немного дороже низкоскоростных, но это плата за возможную экономию времени при записи. Полностью диск на скорости 1x записывается 74 минуты (плюс еще несколько минут может потребоваться на запись служебной информации). Скорости 2x, 4x, 6x,..., 24x и т. д. позволяют сократить это время, но не в указанное число раз: высокая скорость может быть только ближе к концу диска, и есть еще почти постоянные накладные расходы. Привод (или его драйвер) может выбрать скорость ниже указанной (если ему «не понравится» имя производителя, записанное в служебной области разметки).

Стандартный «размер» болванки — 74 минуты или 650 Мбайт («двоичных»), иногда в рекламных целях указывают 680 Мбайт, но уже «десятичных». Оба этих числа соответствуют 333 ООО секторам по 2048 байт пользовательских данных — итого 681 984 ООО байт. Иногда пишут даже 780 Мбайт, это те же 333 ООО секторов, но уже «сырых» данных по 2336 байт без ECC. Получили распространение и болванки большего объема (но того же размера), вмещающие до 80 минут аудио (около 700 Мбайт данных). Малогабаритные болванки диаметром 80 мм вмещают около 200 Мбайт. Есть и болванки размером с визитную карточку, в их прямоугольник вписывается диск, на котором умещается 30-60 Мбайт.

Диски бывают разных цветов, в зависимости от цвета отражающего и регистрирующего слоев:

- Серебряный цвет имеют печатные диски (прозрачная подложка, алюминиевый отражающий слой). Алюминий на диске хоть и медленно, но все-таки окисляется и меняет свои отражающие свойства, поэтому время жизни печатных дисков оценивают в 10-15 лет.
- В голубых и зеленых (если смотреть снизу) болванках CD-R в регистрирующем слое используется цианин (cyanine) — материал голубого цвета (cyan, откуда и название). Зеленый цвет болванок дает золотой отражающий слой, голубой остается при отражающем слое из серебра или сплавов алюминия. Эти болванки имеют среднюю стойкость к перепадам температуры и солнечному свету. Предполагаемое время жизни диска при нормальных условиях — 75 лет.
- Золотистые болванки CD-R имеют регистрирующий слой из фталоцианина (phtalocyanine), они более стойкие к внешним воздействиям. Предполагаемое время жизни диска — 200 лет.
- Серо-коричневый цвет имеют болванки CD-RW (цвет регистрирующего слоя).

Верхняя поверхность болванки может быть защищена довольно прочным покрытием, на котором можно даже печатать этикетки на специальном принтере для дисков. Это отражается пометкой «Scratch resistant printable surface» на упаковке диска. От руки диск можно подписывать только мягким фломастером с войлочным пером и чернилами на водной основе, но ни в коем случае не шариковой ручкой и не карандашом. Наклеивать этикетки крайне нежелательно — на высокой скорости вращения они вызовут сильную вибрацию диска со всеми неприятными последствиями (не только шум).

Диски рекомендуется хранить в специальных пластмассовых футлярах, где они фиксируются за отверстие. Для того чтобы аккуратно вынуть диск, нужно одним пальцем нажать на фиксатор, а двумя другими взять диск за торцевые поверхности. При этом диск не будет сильно изгибаться (при нажатии фиксатор отпускает диск) и пачкаться руками. Особенно чувствительны к изгибу диски CD-RW, поскольку их регистрирующий слой находится в аморфном (полужидком) состоянии. Хранить диски, которыми пользуются неоднократно, в конвертах не рекомендуется — при вынимании и укладке поверхности диска неизбежно царапаются. В такой упаковке часто приходят дистрибутивы ПО, а также диски, вложенные в книги.

Все диски следует беречь от деформаций, царапин, нагревания, действия солнечных лучей.

Болванки обычно продаются в футлярах, коробками по 10, 50, 100 и 500 штук. Для крупных потребителей есть и «шпиндельный» вариант поставки — 100 или более дисков продаются стопкой, без индивидуальных футляров. Это удешевляет поставки, поскольку сам диск в несколько раз легче и тоньше футляра; болванки везут издалека, а футляры могут быть и местного производства.

Материал регистрирующего слоя CD-R подвержен старению: запись критична к оптическим свойствам материала, которые со временем изменяются. Поэтому время жизни болванок до записи ограничено несколькими годами (изготовители обещают 5-10 лет). Этот материал также чувствителен к ультрафиолетовым лучам и солнечному свету.

Многосеансовые диски

Специальным вариантом записываемого диска является *многосеансовый диск*, в котором распознаваемое обычным считыванием содержимое записываемого диска может меняться пользователем несколько раз. Структуру многосеансового диска иллюстрирует рис. 9.10, на котором серым цветом отмечены записанные области (две закрытые сессии и третья — не закрытая).

Сессией (session) называют набор треков (от 1 до 99), которому предшествует вводная зона, содержащая TOC с указателями начала каждого из этих треков. За последним треком имеется и выводная зона, начало которой также задано в TOC. Сразу за выводной зоной может быть записана вводная зона следующей сессии.

Каждая сессия (структура, записанная за один сеанс) выглядит как обычный диск CD-ROM, но есть нюансы в записях вводной зоны. Сессия называется за

крытой, когда ее программная область обрамлена вводной и выводной зонами. Однако в ее TOC указатель на выводную зону может указывать либо на начало выводной зоны, либо на ее конец, то есть на начало вводной зоны следующей сессии. Когда указатель описывает начало выводной зоны, диск становится закр*ытым* — следующую сессию к нему уже не добавить. Когда он указывает на конец вводной зоны, на диск возможна запись последующей сессии (если хватает ресурсов: места на диске, места в PMA и номеров треков).

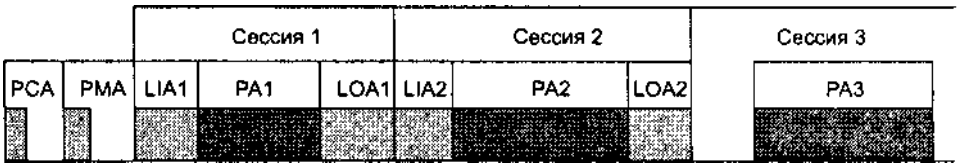


Рис. 9.10. Структура многосеансового диска: PCA — область калибровки, PMA — область сохранения координат, LIA — вводная зона, PA — область данных сессии, LOA - выводная зона

Первый трек первой сессии должен иметь номер 01, следующие треки — последовательно нарастающие номера. Номер первого трека очередной сессии должен быть следующим за номером последнего трека предыдущей сессии. Максимальный номер трека — 99.

Многосеансовые, или мультисессионные (multi session), диски содержат более одной сессии, и все сессии физически доступны для чтения. Очередная записываемая сессия может быть как полностью независимой (ее TOC содержит ссылки только на ее собственные треки), так и связанной с предыдущими сессиями (linked session). Связь может быть как на уровне треков (абсолютные координаты «старых» треков все или частично включают в TOC новой сессии), так и на уровне файлов (для CD-ROM). Связь на уровне файлов требует внесения ссылок на файлы прежних сессий в каталоги и таблицу путей, являющиеся логической частью файловой системы CD-ROM.

Возможности использования информации конкретных сессий зависят от устройства считывания и его ПО. Аудиоплееры, как правило, считывают только первую сессию — остальными они просто не интересуются. Приводы CD-ROM и их ПО могут, в принципе, читать любую сессию. Если они будут читать последнюю сессию, появляется возможность «перезаписать» диск CD-R, фактически, только дописав следующую сессию. На этих свойствах построены диски *CD Plus*, они же *CD Extra*, у которых первая сессия предназначена для аудио- плееров, а вторая (и последующие) — для приводов CD-ROM. Приводы CD- ROM и аудиоплееры способны считывать только закрытые сессии, не закрытые сессии доступны только рекордерам. Диск, в принципе, закрывать не обязательно, но может встретиться привод, не желающий (совместно со своим ПО) читать незакрытый диск.

Многосеансовая запись впервые появилась в PhotoCD, а затем и в CD-ROM XA. Сейчас на нее распространяется стандарт, описанный в «Оранжевой книге», согласно которой многосеансовая запись может производиться в физиче

ском формате Model (CD-ROM) или Mode 2 (CD-ROM XA). Все сессии одного диска должны записываться в одном из этих режимов.

Поддержка многосеансовых дисков появилась уже в ряде моделей приводов 4x, ее имеют практически все накопители 8x и более высокоскоростные. По умолчанию привод, поддерживающий многосеансовый режим считывания, должен обращаться к последней сессии. В таблице путей, записанной в этой сессии, могут содержаться и ссылки на файлы из предыдущих сессий. Таким образом, в зависимости от наличия этих ссылок через таблицу путей последней сессии оказываются доступными не только ее данные, но и любые файлы предыдущих сессий. При этом оказывается возможным и «обновление» прежних файлов, которое сводится к записи новых их версий и к включению в таблицу путей ссылок только на эти версии. «Удаление» файлов сводится к тому, что ссылка на них не включается в таблицу путей последней сессии. Обычные CD-плееры и накопители CD-ROM, не поддерживающие многосеансовый режим, читают данные TOC (и таблицу путей) только первой сессии. Некоторые программные драйверы, обнаружив носитель с физическим форматом Mode 1 CD-ROM (а не Mode 2 CD-ROM XA), ошибочно рассматривают его как односеансовый и обращаются к первой сессии. В результате остальные сессии оказываются недоступными, но эта проблема решается просто — заменой драйвера на его более корректную версию.

Какая из сессий доступна по умолчанию — первая или последняя — зависит от программного обеспечения. В MS-DOS по умолчанию доступна первая сессия, в Windows 95 — последняя. Более «ловкое» ПО позволяет выбрать номер доступной сессии.

Диски DVD

Название DVD поначалу расшифровывалось как Digital Video Disk — диск для цифровой видеозаписи, сейчас же подразумевается иное — Digital Versatile Disk — универсальный цифровой диск. В DVD нашли свое развитие принципы CD, направленные на повышение плотности хранения и скорости передачи информации. Эти диски имеют те же внешние размеры, что и CD (диаметр 120 мм и толщину 1,2 мм), однако представляют собой «бутерброд» из двух пластин. Для повышения емкости ширина трека и продольный размер битовой ячейки уменьшены примерно вдвое, снижены издержки, связанные с избыточностью кодов коррекции ошибок. Кроме того, могут использоваться две стороны диска, а на каждой стороне информация может храниться в двух слоях, таким образом, один диск может иметь уже четыре рабочих слоя. Для считывания DVD требуется лазер с длиной волны 635/650 нм (для CD используется длина волны 780 нм). Изменены системы канального (применяются 16-битные коды) и избыточного кодирования.

Каждая пластина DVD может быть как однослойной (аналогичной по конструкции диску CD), так и двухслойной. В двухслойной пластине «ямки» расположены в двух плоскостях, нижний слой сверху покрыт полупрозрачной полу- отражающей пленкой, а верхний — отражающей (рис. 9.11). Какой из слоев считывается, определяется фокусировкой луча (сигналы другого слоя из-за

расфокусировки размываются и на их фоне различим требуемый слой). В двухслойных дисках размер ячейки увеличен примерно на 10 %, за счет чего емкость каждого слоя несколько ниже, чем у однослойных.

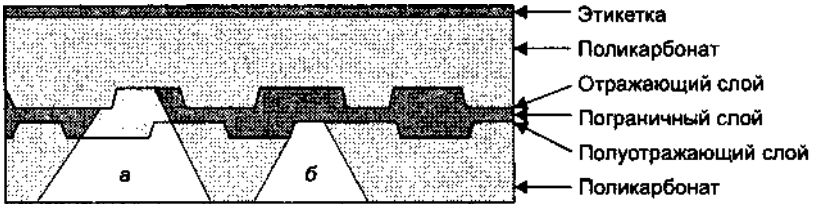


Рис. 9.11. Устройство двухслойного диска DVD: а — луч, сфокусированный на верхнем слое; *б* — луч, сфокусированный на нижнем слое

Диски DVD выпускаются в разных сочетаниях количества сторон (SS — Single Sided, односторонние; DS — Dual Sided, двусторонние) и слоев (SL — Single Layer, однослойные; DL — Dual Layer, двухслойные). В односторонних дисках вторая пластина может использоваться в качестве этикетки, маркировка двусторонних дисков проблематична, поскольку обе стороны требуются для считывания. Для считывания обеих сторон DS-дисков во многих моделях приводов приходится переворачивать диск. Производство дисков DS/DD сложно, а потому и дорого.

Двухслойные диски DVD пока что распространены только печатные, а перезаписываемые диски бывают в основном однослойные (но могут быть и двусторонними). Двухслойные записываемые диски DVD±R DL (они же DVD±R9) и приводы для них выходят на рынок только в 2005 году. Параметры дисков DVD приведены в табл. 9.8. В обозначениях, например, DVD-5, DVD-9 число указывает емкость в миллиардах байт, округленную, как правило, в большую сторону (исключение составляют DVD-1 и DVD-4). В таблице приведена емкость и в «двоичных» гигабайтах (Гбайт).

Таблица 9.8. Типы и форматы дисков DVD

Формат	Сторон/слоев	Емкость, Гбайт / млрд. байт
Диски диаметром 120 мм		
DVD-5	SS/SL	4,37 / 4,70
DVD-9	SS/DL	7,95 / 8,54
DVD-10	DS/SL	8,74 / 9,40
DVD-14	DS/ML	12,32 / 13,24
DVD-18	DS/DL	15,90 / 17,08
DVD-R 1.0	SS/SL	3,68 / 3,95
DVD±R 2.0, DVD±RW 2.0	SS/SL	4,37 / 4,70
DVD±R 2.0, DVD±RW 2.0	DS/SL	8,75 / 9,40
DVD±R9, DVD±R DL	SS/DL	7,95 / 8,54
DVD-RAM 1.0	SS/SL	2,40 / 2,58
DVD-RAM 1.0	DS/SL	4,80 / 5,16

продолжение

Таблица 9.8 (*продолжение)*

Формат	Сторон/слоев	Емкость, Гбайт / млрд. байт
DVD-RAM 2.0	SS/SL	4,37 / 4,70
DVD-RAM 2.0	DS/SL	8,75 / 9,40
HD DVD-ROM	SS/SL	15
HD DVD-ROM	SS/DL	30
HD DVD-RW	SS/SL	20
HD DVD-RW	SS/DL	32
HD DVD-R	SS/SL	15
BD-ROM (R, RE)	SS/SL	23,3–27
BD-ROM (R, RE)	SS/DL	46,6–54
Мини-диски диаметром 80 мм		
DVD-1	SS/SL	1,36 / 1,46
DVD-2	SS/DL	2,47 / 2,66
DVD-3	DS/SL	2,72 / 2,92
DVD-4	DS/DL	4,95 / 5,32
DVD-RAM 2.0	SS/SL	1,36 / 1,46
DVD-RAM 2.0	DS/SL	2,47 / 2,65

Помимо распространенных дисков DVD в таблице приведены параметры и новых разработок. Диски *DVD высокой плотности* (High Density DVD, HD DVD), считываются лазером с короткой длиной волны (405 нм). Базовая скорость чтения — 36,5 Мбайт/с. Предусматриваются варианты однослойных и двухслойных дисков HD DVD-ROM и HD DVD-RW, а также однослойных дисков HD DVD-R.

С такими же лазерами работают диски *BD* (Blu-ray Disk), название которых пошло от цвета луча (голубой-фиолетовый, в слове blue последнюю букву опустили). Этими дисками планируется заменить кассетные (ленточные) видеомагнитофоны. Для компьютеров предусматриваются варианты BD-ROM, BD-R и BD-RE. Скорость чтения и записи 1x — 36,5 Мбайт/с, возможна и скорость 2x.

Диски FMD (флюоресцентные многослойные), сулящие большие объемы (называемые и трехмерными), пока не вышли из стадии лабораторных исследований.

Базовой скоростью DVD является скорость, достаточная для считывания видеодисков, — около 1,35 Мбайт/с, что эквивалентно 9x для CD-ROM. Однако эта скорость передачи обеспечивается при скорости вращения DVD, примерно в 3 раза меньшей, чем у CD-ROM. Этим объясняется тот факт, что высокоскоростные приводы DVD-ROM, читающие и CD, обеспечивают скорость считывания CD, меньшую, чем для DVD, — сказывается ограничение скорости вращения, при которой биения еще приемлемы. Высокие скорости DVD-ROM (2x, 4x и т. д.) требуются лишь для ускорения обмена данными при работе компьютерных приложений — для видео достаточно 1x. Предельной считается скорость 16x (21,6 Мбайт/с), она уже достигнута при чтении и даже однократной записи (DVD ± R); многоразовая запись DVD ± RW пока работает со скоростями 4x-8x. Время доступа у приводов DVD такое же большое, как и у приводов CD (около 120-150 мс).

Первые *записываемые диски DVD* — DVD-R и рекордеры появились в 1997 году, но были очень дороги. Они используют ту же технологию, что и CD-R, но более тонкую геометрию и иной тип краски.

Диски DVD-R полностью совместимы с DVD-ROM, DVD-Video и DVD-Audio, так что они могут считываться на любых устройствах, подходящих по назначению (проигрывателях или приводах для компьютеров). Диск DVD-R содержит

5 областей:

- PC A (Power Calibration Area) — область калибровки мощности;
- RMA (Recording Management Area) — область управления записями;
- LIA (Lead-In Area) — вводная зона;
- data recordable area — область данных;
- LOA (Lead-Out Area) — выводная зона.

Диски DVD-R допускают инкрементную запись, аналогично многосеансовым дискам CD-R. Определены два типа инкрементной записи:

- Type 1 — допускает чтение дисков DVD-R в системах с ISO 9660, использует файловую систему UDF Bridge;
- Type 2 — для непосредственного манипулирования файлами (прямого доступа), использует UDF без возможностей ISO 9660.

Для обоих типов каждая секция данных пишется в обрамленную область (bordered area) после области border in, за которой следует область border out, а также область border in, предшествующая следующей секции данных. Каждая обрамленная область начинается с файловой системы UDF и завершается виртуальной таблицей размещения файлов (Virtual Allocation Table, VAT).

Формат *DVD-RW* (он же DVD-R/W и DVD-ER), основанный на DVD-R, предложен фирмой Pioneer. В этих дисках имеет место изменение фазы состояния слоя, что обеспечивает возможность стирания и последующей записи (до 1000 раз). Приводы DVD-RW могут записывать диски DVD-R, DVD-RW, CD-R и CD-RW. У дисков DVD-RW отражающая способность выше, чем у DVD- RAM и DVD+RW, так что они могут считываться и на существующих приводах DVD-ROM (хотя возможно их ошибочное восприятие как двухслойного диска DVD-ROM).

Диски и рекордеры *DVD-RAM* (перезаписываемые) появились вслед за DVD- R, но по более низким ценам. Вместо постоянной линейной скорости (CLV) здесь используется разделение диска на зоны, и в каждой зоне угловая скорость постоянна (Constant Angular Velocity, CAV), так что средняя линейная скорость по всему диску примерно одинакова. Постоянная угловая скорость обеспечивает быстрый произвольный доступ (в пределах зоны). Диски DVD-RAM заключены в картриджи, и для работы с ними требуются специальные приводы (Caddy-Type). Это связано с особой чувствительностью технологии к чистоте и дефектам поверхности диска. Картриджи бывают двух типов: из картриджей Type 1 диск вынуть нельзя, а из картриджей Type 2 диск, в принципе, извлекаем (после «аккуратной поломки» картриджа). Некоторые приводы не позволяют записывать DVD-RAM без картриджа, некоторые не позволяют записывать и в

картридже, если диск из него вынимался. Заметим, что все другие форматы DVD обходятся без картриджей.

Спецификация DVD+RW для перезаписываемых (до 1000 раз) дисков предложена фирмами Hewlett-Packard, Philips и Sony при поддержке Verbatim, Ricoh и Yamaha (ветеранов CD-RW). Диски появились в 2001 году, но этот формат не стандартизован DVD-форумом. Диски совместимы с DVD-ROM. *Приводы DVD+* RW считывают DVD-ROM и CD (и записывают CD-R и CD-RW), многие современные приводы DVD+RW читают (и даже записывают) DVD-R и DVD- RW, но «не понимают» DVD-RAM. При записи DVD+RW может быть отключен (опционально) механизм обхода дефектных блоков, при этом записываемый диск становится совместимым с любыми устройствами считывания DVD.

Спецификация DVD+RW позволяет использовать для видеозаписей режим *CLV* (постоянная линейная скорость для последовательного доступа) или *CAV* (постоянная угловая скорость для произвольного доступа), однако записывающая аппаратура CAV пока не поддерживает.

Спецификация DVD+R — вариант DVD+RW с однократной записью на более дешевые болванки, появился в 2002 году. Этот формат поддерживают не все модели приводов DVD+RW.

Современные приводы-комбайны позволяют работать с дисками DVD и CD, обеспечивая запись и чтение разных форматов (но не всех). Более дешевые приводы DVD только считывают, но и из них многие поддерживают запись на CD-R и CD-RW.

Диски *DVD-Video* несут видеоинформацию с разрешением 720 x 480/576, сжатие по алгоритму MPEG-2 (средняя скорость потока — 3,5 Мбит/с), аудиосигнал по схеме 5.1 (Dolby Digital Sound), 133 минуты.

Диски *DVD-Audio* несут высококачественный многоканальный (объёмный) аудиосигнал с новыми возможностями, недоступными в CD-DA. Полный диск (4,7 Гбайт) вмещает 2 часа объемного или 4 часа стереофонического воспроизведения.

Диски *DVD-ROM*, *DVD-RAM*, *DVD-RW*, *DVD+RW*, *DVD-R* и *DVD+R* используются для хранения и переноса больших объемов информации, они должны заменить CD в тех случаях, когда объема 650 Мбайт уже недостаточно (многотомные игры, энциклопедии, базы данных, архивы). Из (пере)записываемых дисков наиболее распространены DVD ± R и DVD ± RW, их поддерживает большинство современных приводов. Диски DVD-RAM поддерживаются реже; их форматированная (полезная) емкость меньше номинальной (указанной в таблице).

Для *защиты от копирования* и неавторизованного использования высококачественной видео- и аудиопродукции применяются системы скремблирования содержимого (Content Scrambling System, CSS). Видео- и аудиофайлы шифруются по ключам, уникальным для каждого диска и каждого набора произведений, записанных на диске. Ключ в зашифрованном виде хранится на диске и считывается декодером воспроизводящего устройства. Алгоритмы шифрования дер

жатся в секрете компаниями, занимающимися производством видеодисков и проигрывающей аппаратуры. Файлы данных не шифруются. На диске DVD может быть организована защита файлов от перезаписи через аналоговый выход композитного видеосигнала Macrovision APS (Analogue Protection System — система аналоговой защиты) — в сигнал вводятся специальные искажения. На аналоговые выходы RGB и YUV эта защита пока не распространяется. Для установки флага защиты требуется лицензирование у Macrovision. Разработкой новых методов защиты от пиратства занимается рабочая группа CPTWG (Copy Protection Working Group) DVD-форума.

Диски DVD-Video имеют код региона, который должен совпадать с кодом региона, установленным в проигрывателе. Весь мир поделили на 6 зон (Россия относится к зоне 5), и каждый диск предназначен для продажи (воспроизведения) лишь в своем регионе. В приводе DVD, устанавливаемом в компьютер, номер региона можно задать программно. Возможность смены кода региона определяется уровнем контроля региональной защиты (Region Protection Control, RPC), реализованным в микропрограмме привода. Привод RPC-1 позволяет менять код региона неограниченное число раз, RPC-2 — не более 5 раз. «Народные умельцы», естественно, уже нашли способ «лечения этого недуга» заменой «прошивки», однако найти версию микропрограммы (firmware) с RPC-1 можно не для каждой модели DVD.

В DVD используется файловая система микро-UDF — подмножество UDF (Universal Disk Format). Файловая система не зависит от платформы, обеспечивает эффективный файловый обмен, ориентирована на диски CD-ROM и CD-R, основана на стандарте ISO 13346. Имеется расширение UDF для поддержки перезаписываемых дисков. Комбинация UDF и ISO 9660, известная как UDF Bridge, позволяет обращаться к данным дисков из ОС как не поддерживающих UDF (например, Windows 95), так и поддерживающих UDF (Windows 98/NT /XP/200x).

Диски DVD для видео и аудио используют только файлы в системе UDF, размер файла не должен превышать 1 Гбайт. Как для компьютерных, так и для телевизионных приложений диски DVD должны иметь единую файловую систему. Видео- и аудиофайлы на дисках DVD должны находиться в каталогах VIDE0_TS и AUDI0_TS соответственно, расположенных в корневом каталоге диска.

Для воспроизведения DVD-видео в компьютере должен быть аппаратный или программный декодер MPEG-2. Аппаратный декодер может работать даже на маломощном процессоре типа Pentium 133; как правило, он имеет и выход аналогового видеосигнала для подключения телевизора. Для программного декодирования требуется процессор классом не ниже Pentium II-266 с соответствующим графическим адаптером.

Для базовой поддержки DVD в Windows требуется:

- чтение секторов данных и поддержка системы команд DVD-ROM;
- поддержка файловой системы UDF;

- поддержка потоковых данных типа MPEG-2 для видео- и Dolby Digital для аудиоданных;
- интерфейс DirectShow (ActiveMovie), заменяющий MCI (Media Control Interface) для улучшения совместимости с новым стандартным интерфейсом проигрывания VOB-файлов (Video Object);
- интерфейс DirectDraw, поддерживающий передачу декодированного видео- потока с декодера MPEG-2 на графическую карту через выделенные шины;
- поддержка защиты копирования и кодов регионов для аппаратных и программных декодеров.

Устройство приводов CD-ROM, CD-R, CD-RW и DVD

По устройству считыватели и рекордеры компакт-дисков напоминают обычные дисководы. Однако помимо приводов шпинделя и головки они имеют еще механизм загрузки диска и более сложную головку. Головка состоит из лазерного излучателя, фотоприемника и наклонного зеркала. Все это фиксируется на подвижной каретке головки. На качающейся подвеске каретки закреплена пластмассовая фокусирующая линза, с которой связана малогабаритная катушка индуктивности, помещенная в поле постоянного магнита, установленного на каретке. Катушка и магнит образуют магнитоэлектрический привод подвески линзы, обеспечивающий ее перемещение в направлении, перпендикулярном плоскости диска. Привод подвески линзы под управлением контроллера, встроенного в накопитель, обеспечивает точную фокусировку лучей оптической системы на светоотражающем слое диска, отслеживая биения поверхности диска при вращении. Понятно, что из-за инерционности системы фокусировки есть предел скорости вращения, на котором система уже не успевает отслеживать колебания. При юстировке оптической системы линзу выставляют параллельно плоскости диска с помощью регулировочных винтов на каретке. Механика привода довольно проста, но следует помнить о нежности пластмассовой линзы: неаккуратная чистка может оставить на ней микроскопические царапины, и считывание станет неустойчивым. Существуют накопители с самоочищающимися линзами (self-cleaning lenses). Для изоляции от окружающей среды (drive sealing) в накопителях могут применяться двойные пылезащитные дверки.

Способ загрузки диска в накопитель зависит от конструкции. Больше всего распространены *устройства с выдвигающимся лотком*, или подносом, так называемые накопители Tray-Type (tray — поднос). Диск кладется на лоток рабочей стороной вниз, после чего специальный привод затягивает поднос внутрь устройства и укладывает диск на шпиндель. Движением лотка можно управлять программно или с помощью кнопки на лицевой панели устройства. Лоток втягивается даже просто от легкого подталкивания (при включенном питании). Устройства с лоточной подачей самые дешевые. Их недостаток заключается в том, что диск приходится брать руками, при этом его можно испачкать, уронить или поцарапать.

Существуют так называемые *накопители Caddy-Type*, у которых CD укладывается в специальную защитную кассету (caddy), аналогичную защитному чехлу дискеты 3". Кассета просто вставляется в окно накопителя, и специального привода для загрузки диска не требуется. Такое решение предохраняет диски от случайного повреждения и позволяет их загружать в любом положении накопителя. При частой смене дисков желательно иметь несколько кассет, причем хранить диски прямо в них, но это несколько накладно.

Существуют также накопители, не очень корректно названные *CD-Changer,* в которые можно одновременно установить несколько CD (до четырех). В этих устройствах механизм загрузки иной: диск (без чехла) вставляется в щель, и специальный привод затягивает его внутрь корпуса. После этого устройству можно «скормить» следующий диск. При извлечении диск выдвигается из щели примерно наполовину, после чего его вынимают рукой. Существуют и варианты с кассетой-магазином на несколько дисков, которая «заряжается» и вставляется в устройство. Считывание в таком накопителе возможно, конечно, только с одного (активного) диска, а его смена занимает 1-5 с и производится вручную с помощью кнопок пульта или программно. Однако неудачная программная поддержка (по крайней мере, первыми версиями драйверов для Windows 95) может отбить охоту использовать этот сервис, особенно когда прослушиваешь аудиодиск — ведет себя это устройство слишком суетливо.

Накопители CD-ROM имеют внешние размеры, соответствующие формату 5" устройств половинной высоты. Считывать диски привод CD-ROM может в любом положении, но загрузка диска с лотка не в горизонтальном положении крайне затруднительна и небезопасна как для диска, так и для накопителя. Есть лотки и со специальными фиксаторами, поддерживающими диск в вертикальном положении привода. Накопители Caddy-Type и CD-Changer менее прихотливы — они могут работать и принимать диск в любом положении.

Первые накопители имели *собственные интерфейсы*, по фирмам-производителям называемые интерфейсами *Sony, Panasonic* и *Mitsumi*. Все эти интерфейсы по смыслу напоминают 8-битный вариант шины ATA, но о совместимости с ней нет и речи. Применение накопителей с этими интерфейсами в значительной степени затруднено из-за отсутствия соответствующих драйверов устройств. Современные накопители выпускаются в основном с интерфейсами SCSI и ATA (ATAPI), возможно подключение к шине USB, PC Card и к LPT-порту. Последний вариант может применяться даже в рекордерах, но он работает только в режиме порта EPP и только с довольно мощным процессором. Для блокнотных ПК имеются малогабаритные накопители, размещаемые в их корпусах.

Вслед за первыми моделями, имеющими скорость считывания 150 Кбайт/с, вскоре появились устройства с удвоенной и учетверенной скоростью считывания — Double-Speed, Quadro-Speed (QuadSpin). Для устройств с большей скоростью слов уже не нашлось, и теперь кратность скорости обозначают числами: 2x, 4x, 6x, 8x, 10x, 12x, 16x, 20x, 24x,..., 56x... Вплоть до 8-12x устройство действительно обеспечивало скорость n x 150 Кбайт/с по всему объему диска. Однако при большей кратности для считывания внутренних витков спирали по

требовалась бы слишком высокая частота вращения (чтобы добиться требуемой линейной скорости). При большой частоте вращения усиливаются биения диска из-за его несбалансированности (даже из-за этикетки), и система слежения уже не справляется с фокусировкой на прыгающем треке. Накопители с большой кратностью обеспечивают указанную скорость лишь на внешних витках спирали, а постоянная линейная скорость поддерживается не по всему радиусу — ближе к центру переходят на постоянную угловую скорость. Выдержать скорость 56x на внутренней части спирали сможет не всякий носитель, и в случае какого-то механического дефекта он разлетится внутри привода на кусочки. Есть накопители, которые держат постоянную угловую скорость на всем диске, а требуемую скорость (например, для аудиодиска) поддерживают за счет буферной памяти. Из-за возможных повторных считываний неустойчиво воспроизводимых данных реальная скорость может оказаться ниже номинальной даже на внешних витках. Чем выше кратность скорости накопителя, тем обычно меньше время доступа. Естественно, что высокоскоростные накопители часто более критичны к носителям (плохо читают «пиратские» диски из-за возможно низкого качества носителя и тиражирующей аппаратуры). Высокие скорости вращения приводят к повышенному шуму и вибрациям диска (и привода). Это особенно неприятно, когда привод CD используется для проигрывания аудиодисков, когда высокая скорость и не нужна. Для снижения скорости вращения применяют специальные утилиты и драйверы, «обуздывающие» быстроходные приводы.

Практически все приводы CD/DVD позволяют воспроизводить и аудиодиски (CD-DA), для чего они имеют встроенные цифроаналоговые преобразователи (ЦАП) и *аналоговый интерфейс* с линейным выходом стереосигнала уровня 0,2 В. На лицевую панель обычно выносят гнездо для подключения наушников и регулятор громкости. Для проигрывания аудиодисков накопители часто снабжают кнопкой, с помощью которой можно начать воспроизведение без помощи программных средств. Если при запуске аудиодиска индикатор на лицевой панели CD-ROM светится, а звука нет, причину следует искать в несовпадении разводки аналогового интерфейсного кабеля с разъемом звуковой карты. В принципе, возможно и считывание аудиодисков в цифровом виде по интерфейсу передачи данных с целью дальнейшей цифровой обработки или сохранения на другом носителе, но это позволяют не все накопители. Существуют и приводы, считывающие аудиодиски с выходом на цифровой интерфейс S/P DIF, применяемый в цифровой аудиоаппаратуре. Этот интерфейс выведен на двухштырьковый разъем (цепи D и 6), и его можно подключить к цифровому входу соответствующей звуковой карты. Однако в отличие от аналогового интерфейса, на который при проигрывании аудиодисков сигнал выводится всегда, S/P DIF иногда капризничает (например, выводит только один канал). Причина капризов может скрываться во встроенном ПО и драйвере CD-ROM. Возможна и связь этих капризов со скоростью считывания (при принудительном снижении скорости иногда удается получить оба канала).

Записывающие устройства — *CD-рекордеры* и *DVD-рекордеры* — по конструкции не слишком отличаются от устройств чтения, но их лазер в режиме записи

имеет гораздо большую мощность. Практически все современные рекордеры способны и записывать, и перезаписывать диски, в зависимости от типа болванок. Способности рекордеров (как, впрочем, и считывателей) в значительной степени зависят от версии встроенного микропрограммного обеспечения (firmware), и ряд проблем, возникающих при эксплуатации, решается обновлением встроенного ПО. Для этого используются специальные утилиты или перепрограммируется ПЗУ привода.

Стоит отметить, что время наработки на отказ у устройств CR-R/RW значительно меньше, чем у CD-ROM. Кроме того, из-за более сложной и тяжелой головки время доступа CD-R/RW в режиме считывания больше, чем у обычных приводов CD-ROM, да и скорость считывания ниже. По этим причинам использовать CD-R/RW для интенсивного чтения дисков нецелесообразно.

Применительно к устройствам чтения и записи компакт-дисков принята следующая терминология:

- *CD-ROM* — устройство считывания компакт-дисков. Обозначение 2x, 4x, ..., 52x... указывает скорость считывания (максимальную). Приводы, действительно выдерживающие указанную кратность скорости по всему объему, называют TrueX.
- *CD-R, CD-Recorder* — устройство для записи дисков CD-R, способное также считывать печатные и записываемые диски. Обозначение вида 2x/4x соответствует скорости записи/считывания.
- *CD-RW, CD-ReWriter* — устройство для записи дисков CD-R и CD-RW, способное также считывать печатные, записываемые и перезаписываемые диски. Обозначение вида 4x/2x/32x соответствует скорости записи (на CD-R)/ перезаписи (на CD-RW)/считывания. Самые быстрые (2005 г.) устройства обеспечивают скорости 52x/32x/52x.
- *Multisession CD-ROM* — эти устройства позволяют считывать данные, записанные за несколько сеансов.
- *XA-Ready CD-ROM* — эти устройства не имеют собственного ADPCM-деко- дера, и аудиоданные формата XA могут быть считаны только с помощью звуковой карты.
- *MultiRead CD-ROM* — приводы, способные считывать диски различных форматов, включая диски CD-RW, а также диски, записанные в пакетном режиме. Старые приводы (включая некоторые модели 8x) этой способностью не обладают.
- *Multimedia CD-ROM* — привод, удовлетворяющий спецификации MPC (MultiMedia PC). Должен иметь внешний аудиовыход для проигрывания CD-DA, скорость не ниже lx (MPC1), 2x (MPC2) или 4x (MPC3); начиная с MPC2 должен читать многосессионные диски, включая режим секторов Mode 2. Спецификация MPC1 (Level1) была принята в 1993 году (процессор — 386SX), MPC3 — в 1995 году (процессор Pentium-75; MPEG-1 декодер 352 x 240/288, 30/25 кадров/с). Более высоких уровней MPC не вводили.

Приводы DVD также различаются по способности к записи/перезаписи, но здесь гораздо больше разновидностей из-за обилия стандартов. Современные устройства поддерживают, как правило, несколько стандартов. Распространены устройства-*комбайны,* сочетающие в себе возможности приводов CD-R/RW и DVD-ROM (плюс возможность записи DVD). По цене эти устройства стали уже вполне доступными и получили массовое распространение.

Файловые системы для CD и DVD

Для аудиодисков не требовалось создавать какую-либо специальную логическую структуру — достаточно того, что каждой аудиозаписи соответствует собственный трек. Для хранения данных требуется организация файловой системы, которая не может просто повторять дисковую файловую систему, например, MS-DOS. Такая система при большом времени доступа с большим количеством каталогов и файлов (а емкость 650 Мбайт к этому располагает) работала бы крайне медленно. В настоящее время для компакт-дисков, используемых в PC, распространены почти эквивалентные файловые системы HSF и ISO 9660, которые иногда отождествляют. Здесь кратко рассматриваются только некоторые из стандартов (подробнее см. в [5], [11]).

ISO 9660 — первый стандарт (1988 г.) для хранения данных на CD-ROM. Структуру тома описывает таблица содержимого (TOC или VTOC), которая хранится в его логических секторах (на треке тома). Один диск (или сессия) может иметь и несколько томов, но эта возможность поддерживается не всяким ПО и практически не используется. В TOC описаны все файлы, присутствующие на диске, — имя, дата создания, атрибуты, положение всех экстентов файла (экстент — фрагмент файла, записанный в непрерывной цепочке блоков). Файлы на диске располагаются в каталогах, образующих древовидную структуру, и каждый каталог содержит список входящих в него файлов, их атрибуты и указатели на секторы, в которых располагается начало файлов или их экстентов. Для ускорения поиска файлов на диске кроме каталогов имеется дополнительная *таблица путей* (path table), содержащая список путей (в символьном формате) ко всем подкаталогам диска и адресам их начальных секторов. На имена файлов и длину путей накладывается ряд ограничений. К данному стандарту имеется несколько расширений (*Rock Ridge, Joliet, Romeo).*

Поскольку диск CD-RW допускает перезапись, структура файловой системы ISO 9660, не ориентированная на возможность модификации уже записанных данных, — для него не лучшее решение.

Файловая система UDF (Universal Data Format) с *пакетами переменной длины* построена иначе. Здесь файлы хранятся рядом со своими описаниями, допустима длина имен до 127 символов. Каждый пакет представляет отдельный файл (или его экстент), в начале пакета имеется описание файла (имя, дата, атрибуты, длина файла и данного экстента). Никаких общих таблиц размещения файлов и экстентов для UDF не требуется — последовательное чтение пакетов позволяет собрать все файлы диска. Конечно, для быстрого поиска нужного файла в памяти компьютера строится виртуальная таблица размещения файлов. Диск (сессия) с пакетами переменной длины может быть закрытым и иметь дескрип

торы тома файловой системы ISO 9660, тогда он будет читаться традиционными средствами. Иначе для его чтения нужен специальный драйвер UDF. Организация диска в виде пакетов переменной длины очень эффективна с точки зрения расходуемого дискового пространства, поскольку здесь мало «отходов» (на каждом файле теряется в среднем половина сектора размером 2 Кбайт).

Для перезаписываемых дисков появляется еще одна возможность организации. Чистый диск форматируется *пакетами фиксированной длины* (обычно 2 сектора — 4 Кбайт) и выглядит совсем как большая дискета. Правда, при этом велики накладные расходы на организацию формата — около 18 %. Обычная (650 Мбайт) чистая болванка может быть отформатирована на 494 Мбайт полезного пространства. При этом диск в обычном смысле не закрывается, и с ним может общаться (даже по чтению) только устройство-рекордер. Обращение к такому диску оказывается довольно медленным. Для формата с фиксированной длиной требуется хранение таблицы размещения файлов прямо на диске. Обновления файлов требуют внесения изменений в таблицу. Каждый раз изменения записывать нельзя, поскольку число перезаписей CD-RW все-таки ограничено (порядка 1000 раз). Поэтому в процессе работы таблица размещения файлов хранится в памяти компьютера, и только когда носитель извлекают, предварительно она записывается на CD-RW. Из этого следует уязвимость данной системы: авария во время записи до изъятия носителя приведет к несоответствию между таблицей, записанной на носителе, и реальным расположением файлов (то есть чревата потерей доступа к данным). Диск с пакетами фиксированной длины не может быть закрыт как ISO 9660 и поэтому стандартными средствами не читается. С ним можно работать лишь на приводе MultiRead или рекордере CD-RW через специальный драйвер (например, UDF Reader).

DirectCD — приложение Windows, выпущенное фирмой Adaptec для облегчения работы с CD-рекордером. При запущенном приложении DirectCD с диском в устройстве CD-RW можно работать так же, как с обычным диском: создавать, копировать, изменять, удалять, переименовывать файлы (только скорость записи ниже, чем у винчестера). Диск форматируется пакетами фиксированной длины. Приложение дозволяет работать и с CD-R, но, естественно, с дополнительными ограничениями.

Запись на оптические диски

Запись на оптический диск имеет свою специфику, связанную как с организацией диска (одна спиральная дорожка), так и с особенностями управления лазером. В отличие от магнитных и магнитооптических дисков, обеспечивающих произвольный доступ к любому сектору как по чтению, так и по записи, информация должна записываться непрерывным потоком в цепочку секторов. Поначалу рекордеры (записывающие устройства) могли записывать за одно включение записывающего лазера не менее целого трека оптического диска. Следующим заходом можно было дописать последующий трек (треки). Таким образом, по записи CD-R являются устройствами с последовательным доступом. С перезаписываемыми дисками CD-RW появился пакетный режим записи, который

позволяет снять это ограничение — правда, ценой некоторых потерь пространства на диске и увеличения времени записи.

В течение всего времени записи, когда работает прожигающий лазер, на рекордер в требуемом темпе должна поступать записываемая информация. Опустошение буфера устройства (underrun) не допускается — в режиме записи устройство не может ждать. Прерывание процесса записи (приостановка потока данных), как правило, губит болванку.

Для устройств и дисков CD-R возможны следующие режимы записи:

- Весь диск сразу (Disk At Once, DAO). В этом режиме лазер включается на время записи всего диска от начала до конца, вся информация записывается на диск, включая вводную и выводную зоны, и последующая запись на эту болванку уже невозможна (даже если остается место). Для записи в режиме DAO требуются чистые болванки. Диски, записанные в режиме DAO, читаются на любых приводах и могут быть использованы как мастер-диски для производства печатных (штампованных) CD. Режим DAO реализован не во всех рекордерах, он может не поддерживаться записывающим ПО (пакетом и драйверами).
- Сессия сразу (Session At Once, SAO). В этом режиме за одно включение лазера записываются все треки, а также вводная и выводная зоны одной сессии. Режим малораспространенный, используется для дисков CD-Extra.
- Потрековая запись (Track At Once, TAO). В этом режиме лазер включается на время записи одного трека. В начале каждого трека записывается предзазор (pre-gap) длительностью 2 секунды (150 секторов). Этот режим применяется как для односеансовой, так и для многосеансовой записи. Режим пригоден для дисков любого назначения (аудио, CD-ROM и т. п.). Нормально записанные диски читаются на любых приводах. В этом режиме сначала на диск пишутся информационные треки, а вводная зона остается свободной. Координаты начала треков, а также координаты начала свободной области, следующей за последним уже записанным треком, временно сохраняются в служебной области болванки (PMA). Вводная и выводная зоны записываются позже — при закрытии сессии. До закрытия сессии (в ISO 9660 включающей запись логической TOC и таблицы путей) записанные данные для обычных приводов CD-ROM остаются недоступными.
- Пакетная запись (packet writing). В этом режиме за одно включение лазера записывается произвольное количество блоков — пакет. Длина пакета не превышает объема буфера рекордера, благодаря чему опустошение буфера при записи не грозит порчей диска. Лазер включается на запись, только если в буфере уже имеется полный пакет. Между пакетами записывается всего 7 промежуточных блоков. Пакеты могут быть фиксированной или переменной длины. Пакетную запись ввели на CD-RW, благодаря ей появилась возможность прямого доступа по записи к отдельным блокам диска (при пакетах фиксированной длины). Пакетная запись поддерживается не всеми рекордерами. Диски, записанные в пакетном режиме, читаются не всеми приводами CD-ROM (у них возникают проблемы с чтением промежуточных блоков). Для аудиодисков пакетная запись непригодна. Для чтения диска,

записанного в этом режиме, требуется драйвер файловой системы UDF. Сессия (диск) с пакетами переменной длины может быть закрыта и в формате ISO 9660 Level 3, тогда она будет читаться и с помощью редиректора (типа MSCDEX), поддерживающего Level 3 (старые MSCDEX поддерживают только Level 1 без чередования и фрагментации файлов).

Стирание диска предполагается только для CD-RW — при стирании вся стираемая область переходит в одно состояние (фазу). Стирание может быть полным (full erase) или быстрым (quick erase). При полном стирании выполняется «зачистка» всего диска, включая и информацию TOC во вводной зоне. При быстром стирании очищаются лишь отдельные области диска. Диск со стертой информацией TOC выглядит пустым, но при этом может содержать информацию (до которой очень трудно добраться). Быстрое стирание, затрагивающее лишь структуры данных томов, используют на дисках с пакетной записью (например, в DirectCD). Диск может быть настолько испорчен, что рекордер не сможет выполнить и стирание. В этом случае может помочь стирание солнечным светом или ультрафиолетовыми лучами (в устройстве для стирания ультрафиолетовых ПЗУ).

Для дисков CD-R штатного стирания (с целью последующего использования диска) не предусмотрено, однако это не означает, что записанный диск CD-R невозможно стереть (то есть испортить), применяя стандартный рекордер (правда, может потребоваться нестандартная программа).

За один сеанс на диске должна быть сформирована стандартная структура (сессия), включающая как треки с данными (программная область), так и служебные вводную и выводную зоны. Как уже отмечалось, в Q-субканале вводной зоны содержится таблица содержимого диска TOC, в которой описаны абсолютные координаты начала всех треков, а также выводной зоны.

При записи очередной сессии многосессионного диска данные предыдущих сессий можно выборочно включать в оглавления (на уровне треков — в TOC вводной зоны, на уровне файлов — в дескрипторы тома). Тогда для считывателя многосессионный диск будет выглядеть как единое целое, а запись очередной сессии может изменить его видимое содержание. Напомним, что последняя сессия станет доступной для чтения только после ее закрытия (finalize). Закрывать диск, в принципе, не обязательно. После закрытия диска записать на него новые сессии уже невозможно, и следовательно, невозможно «изменение» его файлов. Первая сессия «съедает» 20 Мбайт, каждая новая сессия приводит к потере 13,5 Мбайт емкости диска (накладные расходы на вводную и выводную зоны), так что записывать множество мелких сессий невыгодно.

Закрытием сессии называется процесс записи вводной зоны со сформированной таблицей TOC, а также выводной зоны. До закрытия сессии стандартные устройства чтения не располагают информацией о координатах начала треков — эта информация временно сохраняется рекордером в специально отведенной зоне PMA, не входящей в стандартную область, доступную для записи (650 Мбайт). Про эту область «знает» только записывающее ПО, и читают ее только рекордеры. Незакрытая сессия не доступна никаким устройствам чтения

в «штатном» режиме. Рекордер может дописывать в незакрытую сессию треки до тех пор, пока на диске есть доступное место, пока не будет достигнуто предельное число треков в сессии (99) и пока есть место в PMA для временного хранения координат начала трека. После закрытия к сессии уже не могут быть добавлены треки, но может быть открыта новая сессия, если не закрыт диск.

Закрытием диска (финализацией) называют запись вводной и выводной зон, причем в TOC указывается начало выводной дорожки (а не начало возможной вводной для последующей сессии). После закрытия диска к нему уже не могут быть добавлены сессии (и треки). Закрывать диск, в принципе, не обязательно, достаточно закрытия сессии.

Когда закрывается сессия или диск с файловой системой ISO 9660, помимо «физической» таблицы TOC, описывающей положения треков, на диск (в программную область) записывается и «логическая» таблица TOC тома, в которой описывается положение всех записанных файлов (при желании включая и файлы предыдущих сессий). При этом в области уже записанных файлов никаких изменений не производится. После такого закрытия диск можно читать стандартным приводом CD-ROM (для CD-RW требуется MultiRead CD-ROM) со стандартными драйверами (MSCDEX для DOS, встроенные средства Windows 9x/NT). До этого закрытия логическая таблица TOC существует лишь в памяти (на жестком диске) пишущего компьютера, а физическая таблица TOC — в PMA на записываемом носителе. Если диск вынуть из рекордера до закрытия (или в случае аварии), логическая таблица TOC не попадет на диск. Данные на диске останутся, но доступа к ним не будет.

Форматирование диска UDF с пакетами фиксированной длины помимо реального форматирования обеспечивает запись всего диска и к тому же закрывает его (физически — записывая вводную и выводную зоны и делая необходимые ссылки в TOC Q-субканала вводной зоны). При этом на диск можно записывать (удалять, переименовывать...) файлы. Диск, кроме рекордера, можно будет читать в MultiRead CD-ROM с драйверами UDF, но для традиционных средств чтения он останется недоступным.

«Форматирование» диска UDF с пакетами переменной длины на самом деле только «наводит» каталог на свободную область сессии. На диск можно записывать (точнее, дописывать) файлы до тех пор, пока не закрыта сессия или диск. После закрытия сессии можно будет открыть новую и продолжать запись, после закрытия диска — уже нет. На считывающих приводах данные будут читаться лишь для закрытой сессии (диска), открытую сессию может читать только рекордер (он пользуется PMA). Если сессия закрыта в формате ISO, диск можно будет читать всеми традиционными средствами.

Хотя средства записи и позволяют смешивать форматы Mode 1 и Mode 2, при считывании таких дисков наверняка возникнут проблемы. Если есть необходимость сочетать треки с «сырыми» (без ECC) и защищенными от ошибок данными, то должен использоваться режим Mode 2, в котором возможны обе эти формы представления данных.

Чистые болванки на самом деле не совсем пустые. На их поликарбонатном субстрате отпечатана спиральная дорожка, содержащая временную разметку. Эту разметку «понимает» только рекордер. На этой же дорожке отпечатана информация о носителе, которая может не полностью соответствовать действительности (штампом могут пользоваться разные производители):

- Manufacturer — производитель матрицы (штампа), но не обязательно диска.
- Writable/Rewritable — тип болванки (CD-R или CD-RW).
- Dye type — тип краски (для CD-R), информация для настройки записывающего лазера. Однако краска может быть иной (см. выше), а для настройки все равно используется область PCA.
- Spiral length in blocks — длина спирали (количество блоков, доступных для записи). Соответствует действительности, так как определяется только штампом.
- Rated speed — допустимая скорость записи. Если не указана, то для CD-R допустима скорость 1x, для CD-RW — 2x. Превышение скорости чревато порчей диска.
- Audio — болванка может использоваться и на автономном рекордере аудиодисков (более высокое качество).

Практически все модели современных рекордеров CD позволяют работать с болванками обоих типов — и CD-R, и CD-RW. Выбор типа носителей делают с учетом назначения записи (передача информации, архивация данных с необходимостью сохранения предыстории и без, другие задачи). Перезаписываемые болванки дороже, но они дают право на «бесплатную» ошибку.

По способу подготовки данных для записи различают запись с образа CD и запись «на лету». Более надежен способ с предварительным созданием образа CD (CD Image или Virtual CD — виртуальный компакт-диск, не путать с Video CD). При этом вся информация для записываемого диска должна быть предварительно сформирована в виде файла-образа на каком-либо носителе (винчестере). При записи образ считывается и передается на рекордер с требуемой скоростью, не допуская опустошения буфера. Для хранения образа требуется свободное дисковое пространство (до 650 Мбайт при записи целого диска). В качестве носителя образа могут применяться отнюдь не все магнитные диски: если во время записи винчестер вдруг займется непрерываемой внутренней термокалибровкой, болванка будет загублена. Запись «на лету» (on-a-fly) не требует резервирования большого объема внешней памяти для хранения образа — файлы считываются с мест своего обычного хранения, но также должна быть гарантирована скорость и непрерывность считывания.

Чтобы застраховаться от опустошения буфера рекордера, компьютер, предназначенный для записи, должным образом конфигурируют:

- Компьютер должен быть достаточно мощным (быстродействующий процессор, большой объем ОЗУ, быстрые диски).
- Предпочтительный интерфейс рекордера и дисков — SCSI-2 (SCSI-3). Он имеет более высокую производительность, чем ATA, при одновременной

работе с несколькими устройствами (в данном случае по крайней мере двумя — винчестером и рекордером).

- Если используется интерфейс ATA, то винчестер с образом и рекордер следует устанавливать на разных каналах ATA.
- Предпочтительный режим работы драйверов — прямое управление шиной (bus mastering).
- Для хранения образа желательно иметь отдельный раздел жесткого диска (а то и отдельный винчестер), который следует регулярно дефрагментировать (не во время записи!).
- На время записи компьютер не должен играть роль сервера сети (его диски и принтеры не должны быть разделяемыми), поскольку неожиданный приход запроса внешнего пользователя может загрузить компьютер так, что поток данных на рекордер приостановится. Если имеется модем, для его ПО должна быть запрещена реакция на звонки.
- На компьютере на время записи должен быть запрещен автоматический запуск приложений по расписанию (ScanDisk, Defrag, антивирусные программы) — их внезапный запуск тоже может слишком загрузить компьютер.
- На компьютере на время записи должны быть отключены экранные заставки.
- На компьютере должно быть отключено автоматическое распознавание диска CD-ROM, иначе во время записи система неожиданно «увидит» новый диск и попытается его воспроизвести, что почти наверняка прервет процесс записи.
- Средства управления энергопотреблением рекомендуется отключить, чтобы компьютер случайно не «заснул» в процессе записи («заснуть» может и привод CD-ROM, с которого делают копию).
- На время записи не следует запускать лишних приложений, особенно ресурсоемких. Не стоит также прослушивать аудиодиск (если имеется и привод CD-ROM) — хотя при правильной настройке этот процесс потребляет мало ресурсов, в случае ошибки чтения (или иных нештатных ситуаций) система может на некоторое время оказаться заблокированной.
- Предпочтительны модели рекордеров с большим объемом буфера. Чем больший объем буфера для записи имеет устройство CD-R, тем оно менее чувствительно к неравномерности входного потока данных.
- Скорость записи должна выбираться, исходя из качества болванок и производительности компьютера. Чем выше скорость записи, тем выше требования к скорости подачи входного потока данных. Запись на пониженной скорости может оказаться более качественной (будет меньше проблем считывания на разных приводах).

Для защиты болванок от порчи в современных рекордерах применяют технологию BURN-Proof (Buffer UnderRuN Proof — проверка буфера на опустошение) — очередная порция записи не начинается, если в буфере недостаточно

информации. При большом размере буфера (2-4 Мбайт) эта технология работает довольно надежно.

Приложения, записывающие и перезаписывающие CD, часто имеют функцию тестирования, при которой имитируется весь процесс записи диска, но без включения лазера. Таким образом удается проверить, все ли компоненты будущего диска находятся на своих местах и доступны (это особенно важно при записи «на лету», а не с образа) и достаточна ли скорость подачи данных на рекордер. Однако прохождение теста не гарантирует успех последующей записи, если внезапно вмешается один из упомянутых факторов. Например, поскольку в тестовом режиме TOC не записывается, то не сработает и механизм автоматического распознавания диска, а при реальной записи этот механизм, скорее всего, сработает некстати (если он, как рекомендовано, не был предварительно отключен).

Оптические диски с прямым доступом

Помимо CD-ROM, CD-R и CD-RW существуют не столь широко распространенные оптические диски с прямым доступом.

PD/CD (Phase change Disk — диск с фазовым кодированием) — комбинированный накопитель, записывающий информацию на специальный носитель (и с него же считывающий) по методу изменения фазы состояния вещества. Носитель представляет собой многослойный диск в защитном картридже, у которого в одном из слоев может изменяться фаза состояния вещества (как и в CD-RW, но в PD этот принцип применили раньше). Считывание основано на изменении степени отражения участков с разной фазой состояния вещества. В отличие от CD с одним спиральным треком, PD имеет концентрические треки (как у магнитных дисков) и, следовательно, произвольный метод доступа. Шпиндель накопителя поддерживает постоянную угловую скорость для каждой зоны треков (а не постоянную линейную, как в CD). Это позволяет снизить время доступа при поиске в пределах зоны, поскольку не тратится время на разгон или торможение диска. Емкость PD, как и CD, составляет 650 Мбайт, но диски PD не могут читаться на накопителе CD-ROM. В то же время устройство PD/CD (например, модель PD 650) вполне справляется с чтением и обычных дисков CD — тип установленного носителя определяется автоматически. Большим преимуществом PD перед CD-R является возможность многократных циклов стирания-записи и, естественно, считывания при прямом доступе к данным. Недостаток — несовместимость PD и CD.

WORM (Write Once, Read Many times) — устройства с однократной записью и многократным считыванием специфического носителя. Устойчивый к внешним воздействиям картридж 5" емкостью 650 Мбайт — 1,3 Гбайт записывают по технологии, похожей на CD-WORM. Стоимость устройств высокая, стандартов нет.

Термин WARM (Write And Read Many times) подразумевает многократную запись и считывание, но стандартизованных оптических устройств данного типа пока нет.

Перечисленные устройства не выдержали конкуренции с CD и DVD.

9.9. Ленточные устройства — стримеры

Стримеры — это устройства с магнитным принципом хранения на ленточных носителях последовательного доступа, обеспечивающие возможность работы со сменными носителями — картриджами и кассетами с магнитной лентой. Значительный объем носителей современных стримеров (гигабайты) и неограниченное число носителей в запасе (на полках) позволяет использовать стримеры в больших хранилищах данных. Ряд моделей картриджей применяется в авто- загрузчиках, стекерах и библиотеках. В картриджах помимо собственно носителя могут присутствовать средства автоматической идентификации носителей (до их загрузки): штрих-коды или энергонезависимая память. Основное применение стримеров — резервное копирование и архивирование. При сугубо последовательном доступе, характерном для резервного копирования и архивирования, стримеры обеспечивают высокую скорость записи и считывания. Многие устройства обеспечивают *компрессию* «на лету» — хранят на ленте сжатую информацию. Поэтому параметры емкости и скорости для пользовательских данных часто указывают с ожидаемым двукратным сжатием. Здесь мы приводим *физические* параметры — объемы и скорости записи/считывания данных на носителе.

Во всех стримерах лента движется относительно головок; для получения высокой скорости записи относительная скорость должна быть порядка 0,7 м/с. Для высокой плотности записи лента должна содержать много узких дорожек. В старых устройствах с лентой шириной 0,5" было 9 параллельных дорожек и использовались параллельные запись/чтение байта. Эти громоздкие, дорогие и ненадежные устройства имели сложный лентопротяжный механизм (ЛПМ), следящие системы и вакуумные подсосы ленты для обеспечения больших скоростей и ускорений ленты (при тяжелых катушках). Развитие идеи ленточной записи привело к упрощению механики, повышению плотности записи и переходу на последовательную запись. Существует два основных варианта записи на ленту, различающиеся положением дорожек (рис. 9.12):

- линейная (включая серпантинную);
- наклонно-строчная (геликоидальная).

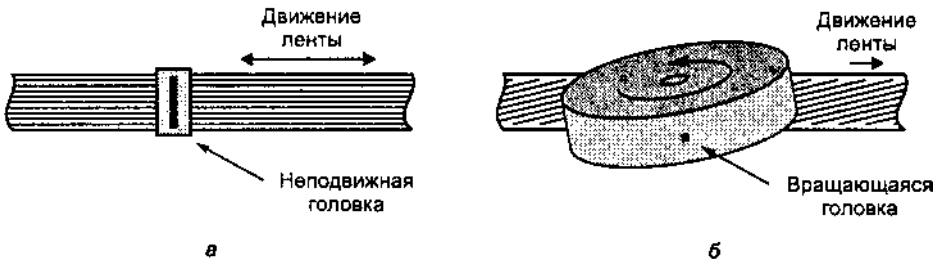


Рис. 9.12. Положение дорожек на ленте: *а* — линейная запись, *б* — наклонно-строчная запись

В устройствах с *линейной записью* требуемая скорость обеспечивается только движением ленты. Это порождает проблемы при организации стартстопного

режима: большие ускорения требуют применения более толстой ленты, соответственно, снижается емкость.

Наиболее распространены устройства *DLT* (Digital Linear Tape — цифровая лента с линейной записью): лента шириной 0,5" в однокатушечном картридже (весь объем заполнен лентой), приемная катушка находится в устройстве. ЛПМ вытаскивает ленту за лидер-петлю и заправляет в приемный барабан, ЛПМ немаленький и сложный, зато картридж простой (и дешевый). Сами устройства DLT довольно дорогие, но надежные и предназначены для интенсивной эксплуатации. DLT — это промышленный стандарт, его поддерживает множество производителей устройств (одиночных устройств и библиотек) и картриджей. Обеспечивается обратная совместимость: более емкие устройства могут работать с лентами прежних моделей (меньшего объема). Ленты DLT отличаются хорошей переносимостью с устройства на устройство. В устройствах SuperDLT (не совместимы с DLT) для получения высокой плотности записи применяют магниторезистивные головки и оптическое позиционирование. Для лент DLT и SuperDLT гарантируется надежность при числе проходов по головкам 500 000-1 000 000, срок хранения 30 лет.

В стримерах *SLR,* произведенных по фирменной технологии Tandberg, используется лента шириной 1/4", давшая название картриджу QIC (Quarter Inch Cartridge — четвертьдюймовый картридж). В картридже установлены две катушки, привод ленты осуществляется кольцевым ремнем в картридже. Лента высовывается в окошко для головок. ЛПМ для этого картриджа простой: он содержит только тонвал и ролик. Головка подвешена на катушке, обеспечивающей позиционирование по серводорожкам, — это позволяет увеличить число дорожек.

К линейной записи относится открытый стандарт *LTO* (Linear Tape Open format — открытый формат лент с линейной записью), поддерживаемый фирмами IBM, HP, Seagate. Здесь используется широкая лента (0,5") и имеются два варианта оптимизации:

- Accelis — минимальное время доступа, максимальная скорость, двухкатушечный катридж, исходное положение — середина ленты.
- Ultrium — максимальная емкость устройства (100-200 Гбайт), встроенная в картридж энергонезависимая память, ускоряющая доступ в библиотеках. Для лент Ultrim гарантируется надежность при среднем числе проходов **1** 000 000, ленту можно использовать (устанавливать-снимать) до 260 раз.

В устройствах с *наклонно-строчной записью* линейная скорость ленты низкая, а высокая относительная скорость обеспечивается геликоидальным сканированием, обеспечиваемым вращающимся блоком головок (как в видеомагнитофонах). Благодаря этому при старте и торможении снижаются потери времени и нагрузки на ленту, так что можно применять современные более тонкие ленты. Привод довольно сложный, содержит много движущихся частей, лента для работы вытягивается из картриджа.

Стримеры на кассетах DAT (Digital Audio Tape — ленты для цифровой звукозаписи) с *4-миллиметровой* (точнее, 3,81-миллиметровой) лентой — самые дешевые и компактные. Однако они не предназначены для интенсивной работы

и возможны проблемы при переносе картриджей. Устройства с *8-миллиметровой* лентой обеспечивают более высокую скорость и большую емкость. Устройства Sony AIT предлагают быстрое чтение каталогов ленты из памяти (EEPROM), установленной в кассете (Memory In Cassette, MIC). Для лент DAT гарантируется надежность при числе проходов 5000.

В настоящее время, несмотря на возрастающую популярность оптических перезаписываемых дисков, стримеры продолжают свое существование и развитие. Те емкости носителей и скорости, которыми они оперируют (табл. 9.9), для оптических устройств пока недосягаемы.

Таблица 9.9. Основные параметры стримеров

Устройство	Объем, Гбайт	Скорость, Мбайт/с	Лента
Линейная запись			
DLT4000,	20	1,5	0,5"
DLT7000,	35	5	
DLT8000	40	6	
SuperDLT I:	110	11	0,5"
SDLT220	160	16	
SDLT320			
SuperDLT II	300	36	0,5"
LTO Ultrim	100–200	80	0,5"
Travan	10	7	8 мм (устройство формата 3,5")
	20	14	
SLR40,	20	3	1/4" (QIC)
SLR60,	30	4	
SLR140	70	6	
Наклонно-строчная запись			
DDS-1,	4	0,78–2,4	DAT (3,81 мм)
DDS-3,	12	3	
DDS-4,	20	3	
DAT 72	36	3,5	
Sony AIT-1,	35	3	8 мм
AIT-2	50	6	
Exabyte	20	3	8 мм
Mammoth,	60	12	
Mammoth-2			

9.10. Твердотельные устройства хранения

Твердотельные устройства хранения используются в миниатюрных компьютерах, а также компьютеризованных цифровых устройствах бытовой электроники — фотокамерах, плеерах, приемниках глобальной системы навигации (Global Positioning System, GPS), музыкальных инструментах и пр. В большин-

стве своем эти устройства основаны на микросхемах флэш-памяти, в наиболее современных используется память со структурой NAND (см. 8.5). Этому типу флэш-памяти присуще быстрое чтение, запись и стирание небольших блоков (256 или 512 байт), что удобно для записи файлов. Правда, для этой памяти характерны довольно медленное чтение произвольного байта и отсутствие возможности побайтной записи, но в областях применения этих карт такая возможность и не требуется, поскольку они ориентированы на блочный обмен. Устройства на флэш-памяти являются энергонезависимыми (в режиме хранения не требуют питания), экономичными в плане потребления, особенно при чтении, достаточно производительными, но, увы, недешевыми. Запись на эти носители специфична: быстрее всего она выполняется в чистый (стертый) блок (сектор диска), а перезапись требует относительно длительного стирания. Кроме того, флэш-память имеет хоть и большое (порядка 10^5-10^6), но ограниченное число циклов стирания-записи — как ни странно, у устройств с подвижным носителем с этим ограничением не сталкиваются. *Устройства хранения* обычно представляют собой комбинацию собственно микросхем флэш-памяти (твердотельного носителя) и микроконтроллера, обеспечивающего внешние интерфейсные функции. Этим они отличаются от *карт памяти с линейным доступом*, например Miniature Card, на которых располагаются только микросхемы памяти (встроенный контроллер не требуется).

Есть и другие твердотельные устройства хранения, например ферроэлектрическая память (FRAM), но пока что массового применения они не имеют. Менее чем за десятилетие устройства хранения на флэш-памяти прошли большой путь от трехдюймовых электронных «дисков» («винчестеров») до современных устройств размером с почтовую марку. Интерфейсы карт внешней памяти тесно связаны с их конструктивами; основные характеристики наиболее распространенных карт приведены в табл. 9.10. Из представленных в таблице устройств хранения полноценными являются только карты CFA; карты SMC, MMC и SD — это лишь носители, а MC — это электронная память.

Таблица 9.10. Основные характеристики карт внешней памяти

Параметр	CompactFlash (CFA)	SmartMedia Card (SMC)	MultiMedia Card (MMC/RS-MMC)	Secure Digital (SD/mini-SD)	Miniature Card (MC)
Длина	36,0	45,0	32,0/18,0	32,0/18,0	33,0
Ширина	43,0	37,0	24,0	24,0/20	38,0
Высота	3,3/5,0	0,76	1,4	2,1/1,7	3,5
Коннектор	Штырьковый	Печатный	Печатный	Печатный	Эластомер
Число контактов	50	22	7, 9 или 13	9	60

Перечисленные карты можно подключать и к обычным компьютерам. Для этого существуют различные адаптеры: для слотов PC Card (к блокнотным ПК), для шины USB (для самых разных ПК) и для других внешних интерфейсов. С помощью этих адаптеров компьютер «видит» подключенную карту как обычный сменный носитель информации (диск). Конечно, компьютеры общего назначения в такой внешней памяти не нуждаются — своя и больше, и дешевле.

Главная цель подключения карты к ПК — быстрая передача прикладных данных бытового характера (фотографий, музыки и т. п.) или перенос (хранение) информации (вместо дискет).

Для подключения карт SmartMedia компания SmartDisk разработала оригинальное устройство FlashPath™: оно выполнено в виде дискеты (не дисковода!) формата 3,5" и имеет слот, в который вставляется флэш-карта SmartMedia. В устройстве имеется электронная схема, передающая информацию с карты в компьютер (и обратно) через магнитные головки дисковода. Питается эта «дискета» от батареек («таблеток»). Устройство совместимо со стандартными дисководами на 1,44 Мбайт, но для его работы, естественно, требуется специальный драйвер. Скорость обмена невысока: 4 Мбайт передаются примерно за

1 минуты.

В качестве интерфейса твердотельных устройств хранения может использоваться и самый обычный вариант ATA. Существуют устройства DOM (Disk On Module) — небольшие модули, которые вставляются в обычный 40-контактный разъем ATA, имеющийся на любой современной системной плате. Правда, этот «винчестер» не блещет выдающимися параметрами: объем — 4-256 Мбайт, средняя скорость передачи данных — 1,6 Мбайт/с, удельная стоимость хранения — около $1,5 за мегабайт. Внутри модуля находится флэш-память структуры NAND с контроллером, эмулирующим систему команд ATA. Основное применение этих модулей — хранение ПО встраиваемых компьютеров, но можно ими пользоваться и как сменными устройствами хранения.

Интересный вариант «твердотельного диска» для микрокомпьютеров и микроконтроллеров — DiskOnChip — предлагает фирма M-Systems. Это микросхема, имеющая интерфейс 8/16-битной статической памяти, легко подключаемый к шине ISA (или локальной шине). Модель Millennium Plus объемом 32 Мбайт содержит массив флэш-памяти архитектуры NAND, модуль статической памяти SRAM (1 Кбайт), интерфейсные схемы, логику защиты записи и чтения и схемы обнаружения и исправления ошибок. Эта микросхема отображается на 8-килобайтную страницу пространства памяти компьютера в области C8000- EFFFFh. По сигналу аппаратного сброса начальный блок из флэш-памяти выгружается в SRAM; если обнаруживается ошибка, то берется следующий (резервный) блок. Этот блок содержит код процедуры инициализации «диска», которая обнаруживается тестом POST как модуль расширения BIOS (см. 5.3). Процедура загружает из флэш-массива в системное ОЗУ драйвер блочного устройства, которое становится первым или последним логическим жестким диском (по выбору при конфигурировании). Далее к этому «диску» можно обращаться обычным способом (через прерывание Int 13h), с него же может загружаться ОС. Интерфейс допускает каскадирование — объединение в единый диск до 4 микросхем, увеличивая его объем до 128 Мбайт, при этом все микросхемы отображаются через общее окно памяти (используют общий сигнал выборки). Встроенное ПО поддерживает полную эмуляцию диска с прозрачным исправлением ошибок и переназначением дефектных секторов. Микросхема обеспечивает длительную скорость записи 750 Кбайт/с, считывания —

1,4 Мбайт/с. Пиковая скорость чтения-записи достигает 20 Мбайт/с. В устрой

стве имеется уникальный идентификационный номер, область для однократного программирования, возможность защиты от записи отдельных зон и возможность закрытия доступа паролем (который невозможно считывать, а можно только сравнивать).

Флэш-память USB

Устройства хранения с интерфейсом USB завоевали популярность благодаря удобству подключения (USB теперь есть практически на всех компьютерах) и малым габаритам (но не крошечным размерам, как флэш-карты, которые можно легко потерять). В этих устройствах имеются флэш-память NAND и контроллер, представляющий для USB устройство того или иного вида.

- Непосредственно *устройство хранения прямого доступа* (класс 08). Устройство хранения имеет встроенный контроллер, его драйвер только организует доставку команд чтения-записи и собственно блоков хранимых данных. Встроенный контроллер выполняет все манипуляции с блоками и страницами флэш-памяти, предоставляя функции полноценного устройства хранения. В частности, он оперирует ECC-кодами для обнаружения и исправления ошибок, а также ведает распределением физических страниц флэш-памяти. Круг этих задач описан в разделе, посвященном картам SMC.
- *Адаптер флэш-карты* (например, SMC) с подключенной картой. Такой адаптер флэш-карты (и сама карта), как правило, «интеллектом» не наделен — он предоставляет только интерфейс между флэш-памятью и хостом. Все функции, превращающие «голую» флэш-память в устройство хранения прямого доступа, должен выполнять драйвер.

Из-за этого разнообразия разным моделям флэш-памяти USB требуются разные драйверы. Современные ОС (Windows XP) имеют встроенные драйверы, «понимающие» большинство моделей и обеспечивающие с устройствами выполнение операций чтения-записи, как с обычным диском. Современные версии BIOS также «понимают» флэш-память и даже позволяют включать эти устройства в список загрузочных. Новые модели флэш-памяти имеют и новые возможности, в том числе возможность закрытия доступа к устройству паролем. Для использования этих возможностей следует устанавливать драйверы и утилиты, входящие в комплект поставки устройства. Существуют комбинированные малогабаритные устройства USB, в которых помимо памяти есть и ее пользователи, например, MP3-плеер, диктофон и радиоприемник с MP3-компрессором.

Как показывает практика, флэш-память USB нельзя рассматривать как надежное устройство хранения — оно может подвести в самый неподходящий момент. Случай из практики: с помощью флэш-памяти, логически представляющей собой адаптер и карту SMC, не удавалось (неоднократно) перенести архивный файл с компьютера на компьютер: при распаковке обнаруживалась ошибка CRC. При сравнении записанного файла с оригиналом обнаружилось, что файл на флэш-памяти содержит 16 Кбайт (это как раз размер блока SMC) сплошных нулей (для файлов-архивов такого быть не может), однако ни при записи, ни

при считывании никаких сообщений об ошибках не появлялось! К сожалению, эта ошибка была нестабильной, и предъявить претензии поставщику (за неделю до окончания гарантийного срока) не удалось. Данные на флэш-памяти можно потерять и по вине пользователя, если после записи (или в процессе) извлечь устройство без предварительной его остановки по команде пользователя (и согласию ОС на удаление оборудования). При этом нарушается целостность логической структуры, излечиваемая форматированием устройства.

Для флэш-памяти USB бывают опасны и сами порты USB, если на них неправильно подано питание, — устройство может сгореть (физически).

CompactFlash

Карты CompactFlash (рис. 9.13), поддерживаемые ассоциацией CFA (Compact Flash Association), широко используются в различных электронных приборах: цифровых фотокамерах, фотопринтерах, MP3-плейерах, цифровых диктофонах, персональных коммуникаторах и, конечно же, компьютерах — настольных, карманных, мобильных. Карты CFA Type I имеют размер 42,8 x x 36,4 x 3,3 мм (4 мм с учетом выступа) и 50-контактный разъем (розетка на карте, двухрядный штырьковый разъем с шагом 1,27 мм на слоте). У карт Type II толщина 5 мм. Назначение контактов приведено в табл. 9.11. Через переходник с 50 на 68-контактный разъем карты могут устанавливаться в слот PC Card Type II или III, имеющийся практически во всех блокнотных ПК. Объем памяти выпускаемых в настоящее время карт — от 4 Мбайт до 1 Гбайт, напряжение питания — 5 или 3,3 В. Карты могут работать в одном из трех режимов: карт памяти (Мет), карт ввода-вывода PC Card (I/O), «чистого» режима IDE (ATA). В первых двух режимах карты функционируют с теми же интерфейсными сигналами, что и PC Card. В режиме IDE электрический интерфейс и система команд полностью совместимы со спецификацией ATA (см. 19.6), правда, обмен данными возможен только в режиме PIO. Режим IDE выбирается заземлением на стороне хоста сигнала ATA_SEL#. При этом из шины адреса используются только сигналы A[2:0] (остальные заземлены хостом); шина данных при

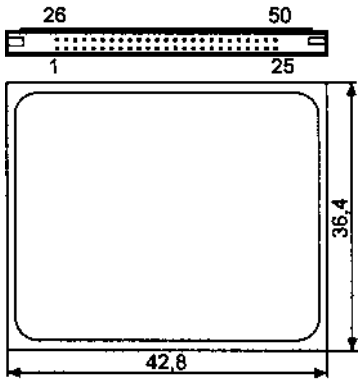


Рис. 9.13. Карта CompactFlash

обращениях к регистрам ATA имеет разрядность 8 бит, а при передаче данных — 16. Сигналы CS0# и CS1# используются для выбора блока командных и управляющих регистров соответственно. Сигналы PDIAG#, DASP#, CSEL#, RESET# и IORDY соответствуют спецификации ATA. Сигналом CSEL# выбирается роль карты: при заземленном контакте — устройство 0 (master), при разомкнутом — устройство 1 (slave); можно воспользоваться и «кабельной выборкой». Сигналы REG# и WE# должны подключаться к шине питания (V_{cc}). Сигналы CD1# и CD2# являются индикаторами установки карты (их контакты замыкаются последними, на карте они заземлены). Существуют чисто пассивные адаптеры-переходники, позволяющие подключить карту Compact Flash к обычному порту ATA (IDE), имеющемуся на любой современной системной плате.

Таблица 9.11. Назначение контактов карт CompactFlash

№	Mem	I/O	IDE	№	Mem	I/O	IDE
1	GND	GND	GND	26	CD1#	CD1#	CD1#
2	D03	D03	D03	27	D11	D11	D11
3	D04	D04	D04	28	D12	D12	D12
4	D05	D05	D05	29	D13	D13	D13
5	D06	D06	D06	30	D14	D14	D14
6	D07	D07	D07	31	D15	D15	D15
7	CE1#	CE1#	CS0#	32	CE2#	CE2#	CS1#
8	A10	A10	A10	33	VS1#	VS1#	VS1#
9	OE#	OE#	ATA_SEL#	34	IORD#	IORD#	IORD#
10	A09	A09	A09	35	IOWR#	IOWR#	IOWR#
11	A08	A08	A08	36	WE#	WE#	WE#
12	A07	A07	A07	37	RDY/BSY	IREQ	INTRQ
13	VCC	VCC	VCC	38	VCC	VCC	VCC
14	A06	A06	A06	39	CSEL#	CSEL#	CSEL#
15	A05	A05	A05	40	VS2#	VS2#	VS2#
16	A04	A04	A04	41	RESET	RESET	RESET#
17	A03	A03	A03	42	WAIT#	WAIT#	IORDY
18	A02	A02	A02	43	INPACK#	INPACK#	INPACK#
19	A01	A01	A01	44	REG#	REG#	REG#
20	A00	A00	A00	45	BVD2	SPKR#	DASP#
21	D00	D00	D00	46	BVD1	STSCHG#	PDIAG#
22	D01	D01	D01	47	D08	D08	D08
23	D02	D02	D02	48	D09	D09	D09
24	WP	IOIS16#	IOCS16#	49	D10	D10	D10
25	CD2#	CD2#	CD2#	50	GND	GND	GND

В формате CompactFlash Type II (и с тем же интерфейсом) выпускаются микровинчестеры (IBM Microdrive). Как ни странно на первый взгляд, они отличаются от твердотельной флэш-памяти существенно лучшими техническими параметрами: быстродействием, скоростью и объемом.

SmartMedia Card

Карты *SmartMedia Card* (SMC), поддерживаемые ассоциацией PCMCIA, предназначены примерно для того же круга приложений, что и CompactFlash. Они совсем тонкие, имеют менее «нежный» печатный разъем с малым числом контактов (всего 22) и не боятся не только повышенной влажности, но и воды. Карты основаны на микросхемах флэш-памяти с организацией запоминающих ячеек NAND. Имеются карты SMC с однократным программированием (на основе масочных ПЗУ), для них возможно только считывание информации. Средняя скорость передачи данных — 2 Мбайт/с, пиковая — до 10. Вид карт приведен на рис. 9.14. Карты на 5 В выпускаются объемом 16 и 32 Мбит (2 и 4 Мбайт). Карты на 3,3 В выпускаются объемом 16, 32 или 64 Мбит (2, 4, 8 Мбайт); у них контакт 17 соединен с V_{cc}. Для карт SmartMedia выпускают простые переходные адаптеры на слот PC Card Type II. Появились даже устройства FlashPath™ для считывания этих карт в обычном (!) дисководе 1,44 Мбайт.

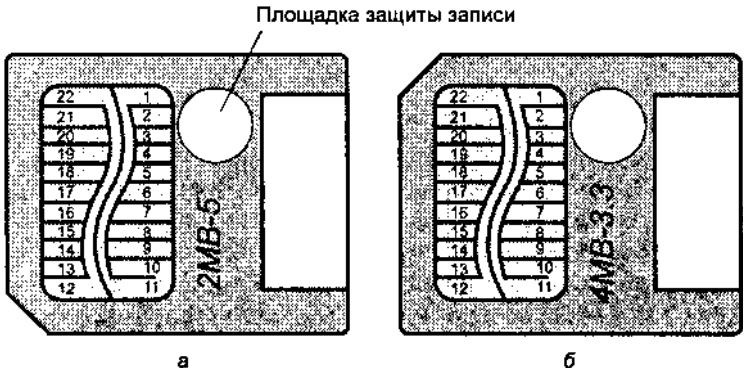


Рис. 9.14. Карты SmartMedia: а — питание 5 В, *б* — питание 3,3 В

Физическая организация карты

Физический интерфейс SMC — параллельная 8-битная шина данных в совокупности с сигналами управления, уровни сигналов — ТТЛ (5 или 3,3 В). По шине передаются команды, адреса и собственно данные. Назначение контактов и сигналов интерфейса приведено в табл. 9.12.

Таблица 9.12. Назначение контактов SmartMedia Card

Контакт	I/O	Назначение
6–9, 13–16	I/O	D[0:7] – двунаправленная шина данных
21	I	CE# – выбор карты (низким уровнем)
2	I	CLE – признак подачи команды (команда фиксируется по сигналу WE#)
3	I	ALE – признак подачи адреса (адрес фиксируется по сигналу WE#)
20	I	RE# – сигнал чтения данных или состояния (результат фиксируется по положительному перепаду)

Контакт	I/O	Назначение
4	I	WE# — сигнал ввода адреса, команды или записи данных (состояние шины фиксируется по положительному перепаду)
19	O	R/B# (Ready/Busy#) — сигнал готовности (высокий уровень) к чтению или записи данных. Низкий уровень означает занятость внутренними операциями. Выход с открытым стоком, требует подтягивающего резистора
5	I	WP# — предохранение от записи. Низкий уровень запрещает работу внутреннего источника высокого напряжения, что используется для предохранения от неокончанных операций при нестабильном питании
11	O	CD# (Card Detect) — сигнал наличия карты (заземлен на карте)
17	O	LVD (Low Voltage Detect) — признак напряжения питания, на 5-вольтовых флэш-картах свободен, на 3,3-вольтовых флэш-картах и масочных ПЗУ соединен с V_{CC}
1, 10, 18		GND — земля
12, 22		V_{CC} — питание карты (3,3 или 5 В)

Флэш-память SMC имеет блочную организацию с независимым поблочным стиранием и постраничной записью. Блок состоит из набора *страниц* (pages); каждая страница состоит из *области данных* (data area) и *служебной секции* (redundant section). Страницы карты малого объема (1-2 Мбайта) имеют размер 256 + 8 байт, новые карты (4 Мбайта и больше) имеют страницы 512 + 16 байт. Страницы (256 + 8) в плане служебной информации используют парами (четные и нечетные), и логически такая пара 2 х (256 + 8) эквивалентна большой странице 512 + 16. В служебной области каждой страницы находятся поля ECC-кода, относящиеся к данной странице, а также *атрибуты блока* и *страницы*. Атрибуты блока (одинаковые для всех страниц данного блока) указывают его адрес и состояние (свободен, назначен, действителен, негоден).

ECC-контроль выполняется внешними (по отношению к карте SMC) средствами, карта обеспечивает только место для хранения ECC-кода. 256-битные поля данных рассматриваются как 2048-битные строки, для которых имеется 22-бит- ное поле ECC-кода. Принятая схема ECC позволяет исправлять однократные и обнаруживать двукратные ошибки на каждой странице.

Размер блока (число страниц) зависит от объема карты и составляет 4, 8 или 16 Кбайт области данных. В картах, поддерживающих мультиблочную запись, блоки распределяются по независимым *районам* (district): блоки 0, 4, 8... — район 1; блоки 1, 5, 9... — район 2; блоки 2, 6, 10... — район 3; блоки 3, 7, 11 ... — район 4. Карта может состоять из нескольких чипов памяти, в каждом чипе располагаются блоки со смежными адресами (например, блоки 0-4095 — чип 1, 4096-8192 - чип 2).

Помимо NAND-массива карта имеет считываемый идентификатор (4 байта: код изготовителя, код устройства и два байта настройки), *расширенный идентификатор* (индикатор 4-канальной организации). Некоторые типы карт имеют *уникальный 128-битный идентификатору* процедура его считывания не разглашается.

Обмен с картами SMC выполняется по командам. *Команды* подаются по шине данных и сопровождаются сигналом WE# при высоком уровне CLE. Подавать ко-

манду можно только при готовности карты (высоком уровне R/B#), исключение составляют команды чтения состояния и сброса. Вслед за командой посылается адрес (если требуется) и выполняется обмен данными. После команды чтения состояния по сигналам RE# карта возвращает *байт состояния*: признаки успеш- ности/неудачи выполнения операции, готовности/занятости карты и защиты от записи. Физический обмен данными с картой выполняется побайтно последовательно, начиная с указанного адреса. Адресация данных довольно специфична. Адрес байта состоит из *номера страницы* (старшая часть номера страницы является *номером блока)* и *номера байта.*

Логическая организация карты

Для того чтобы карту SMC можно было использовать как полноценное устройство хранения, способное записывать и считывать 512-байтные секторы по их линейному адресу (LBA), определена стандартная логическая организация карты (рис. 9.15).

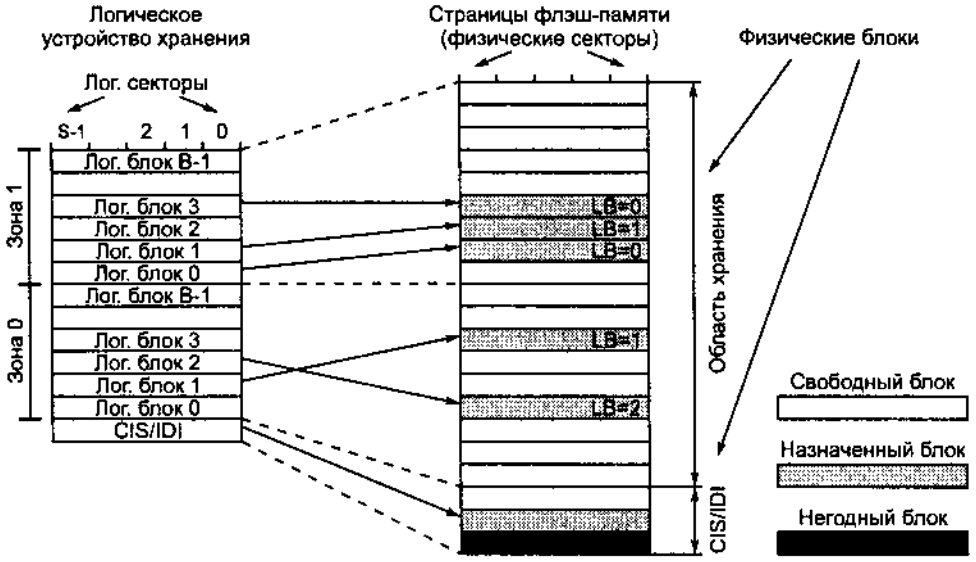


Рис. 9.15. Логическая организация карты SmartMedia

Логически карта разделена на *зоны* — наборы физических блоков; границы зон с физической организацией карт не связаны. Начальная зона предназначена для хранения идентификационной информации (CIS/IDI), остальная часть (storing area — область хранения) предназначена для записи и чтения данных. Эта часть может состоять из одной или нескольких *зон хранения* (read-write zone). Число зон хранения (максимальный номер зоны) зависит от объема карты.

Каждая *зона хранения* образует массив *логических блоков*, состоящих из 16 или 32 *секторов.* Логический блок — это физический блок памяти, *назначенный* для использования. Каждый логический блок имеет свой *логический номер*, уни

кальный в пределах зоны. Логический номер блока хранится в служебных секциях всех страниц (секторов) *физического блока*, в которых расположен данный логический блок.
Для каждой зоны хранения на карте выделяется физических блоков больше, чем число логических блоков в зоне. Это дает свободу маневров при перезаписи секторов. Кроме того, должен быть запас на случай, если блок станет негодным (для этого достаточно «сломаться» всего лишь одному его сектору).
Зона как часть логического устройства хранения представляет собой упорядоченную последовательность логических блоков с нарастающими номерами (от
0 до MAX_BLOCKNUM - 1). Все устройство хранения — это последовательность линейно адресуемых (LBA) логических секторов. По линейному логическому адресу сектора определяются номер его зоны (если зон хранения 2 и более), номер логического блока и номер сектора в блоке. Номер зоны (Zone) определяется делением логического адреса сектора на число секторов в зоне. Разделив остаток от деления на число секторов в блоке, получаем логический номер блока (Log-Block). Остаток от этого деления дает номер сектора, то есть *номер физической страницы в блоке* (Sector). *Номер физического блока* (PhyBlock) определяется по заданному номеру логического блока через просмотровую таблицу (Look Up Table, LUT), формируемую драйвером устройства в ОЗУ.
Таблица формируется по результатам чтения служебных данных всех физических блоков области хранения SMC, индексом в таблице является номер логического блока, элементы таблицы содержат номера физических блоков. Таблица обеспечивает однозначное соответствие физических и логических блоков, однако ситуация, когда не всем логическим блокам назначен номер физического блока, считается нормальной. Для каждой зоны хранения должна быть своя таблица LUT (или одна сквозная).
При первоначальной записи сектора логического блока выбирается свободный нормальный физический блок и в страницу, соответствующую номеру сектора, записываются данные, флаг их действительности, поля ECC и две копии логического номера блока. Если запись прошла успешно, номер физического блока прописывается в LUT; иначе выбирается другой блок, а плохой помечается соответствующим флагом. При записи других секторов в этот же логический блок тот же логический номер прописывается в требуемые страницы. При перезаписи любого сектора (группы секторов) логического блока все неизменяемые секторы старого блока должны быть скопированы в новый (свободный) блок. Затем в новый блок записываются обновляемые секторы (и их ECC). При успешной записи в LUT заносится номер нового физического блока, старый блок помечается как недействительный (далее его можно стереть). При неудаче записи выбирается другой свободный блок. Очевидно, что посекторные модификаци крайне невыгодны, более эффективны мультисекторные операции записи.

MultiMediaCard и Secure Digital

Карты *MultiMediaCard* (MMC) и *Secure Digital* (SD) предназначены для хранения мультимедийной информации, включая электронные книги (eBook). Одна

ко назначение этих карт различно: карты MMC предназначены для широкого распространения данных — музыки, игр, электронных книг, — и они являются довольно дешевыми носителями информации. Карты SD предназначены для безопасного (в смысле конфиденциальности) распространения информации, и они несколько дороже.

Карты MMC могут содержать постоянную (ROM) или перезаписываемую (флэш) память или использоваться как устройства ввода-вывода (как коммуникационные устройства). Доступ к данным может быть защищен паролем. Карта предоставляет только память, которая может быть и ненадежной. Контроль и исправление ошибок возлагаются на внешние (по отношению к карте) компоненты: интеллектуальный адаптер или драйвер в хосте (с примитивным адаптером, который может быть даже просто программно-управляемым интерфейсом на базе LPT-порта). В картах MMC помимо собственно памяти имеется несколько регистров. Регистр OCR (Operation Conditions Register) хранит параметры питания карты и позволяет определять состояние питания. Регистр CID (Card IDentification) содержит уникальный 128-битный идентификатор карты. Регистр CSD (Card-Specific Data) определяет тип содержимого памяти и атрибуты доступа: формат данных, метод коррекции ошибок, время доступа и т. п. В 16-битный регистр относительного адреса (Relative Card Address, RCA) можно записать адрес, используемый для адресуемых обменов хоста с картой. Адрес устанавливается на этапе идентификации, по умолчанию — 0001h.

Существует несколько вариантов карт:

- Обычные карты MMC, размер 32 x 24 x 1,4 мм (рис. 9.16, а), 1-битная шина данных при частоте до 20 МГц, питание — 3,3 В (2,7-3,6 В).
- Карта MMPplus при тех же размерах может иметь разрядность шины 1, 4 или 8 бит при частоте до 26 МГц (52 МГц в картах HS-MMC), скорость чтения — не ниже 2,4 Мбайт/с, питание — 3,3 В.
- Карты MMCmobile, они же DV-RS-MMC, — это карты уменьшенного размера (reduced size) 18 x 24 x 1,4 мм (рис. 9.16, *б),* которые с помощью механического адаптера-удлинителя (рис. 9.16, *в)* можно вставлять в обычный слот. Питание двойное (dual voltage) — 1,8 В (1,65-1,95) или 3,3 В (на выбор). Разрядность и скорость те же.
- Secure MMC — карты с развитыми функциями защиты информации (некая комбинация MMC и смарт-карты в конструктиве MMC). Питание двойное, разрядность и скорость те же.

В данное время (2005 г.) доступны полноразмерные карты до 1 Гбайт и мини-карты до 256 Мбайт.

Интерфейс карт может работать в двух режимах: шины MMC и SPI, режим выбирается при инициализации (после подключения). В *режиме шины MMC* сигнал Clk сопровождает каждый бит данных на шине CMD и все (1,4 или 8) биты

шины DAT. Тактовая частота может лежать в пределах от нуля до максимальной (20, 26 или 52 МГц). Двунаправленная линия CMD предназначена для подачи команды на карту и получения от нее ответа. При использовании данного режима возможны команды одиночного, последовательного и блочного чте- ния/записи. Передача данных контролируется CRC-кодом. К шине MMC можно подключить несколько карт, адресуя их с помощью регистра адреса.

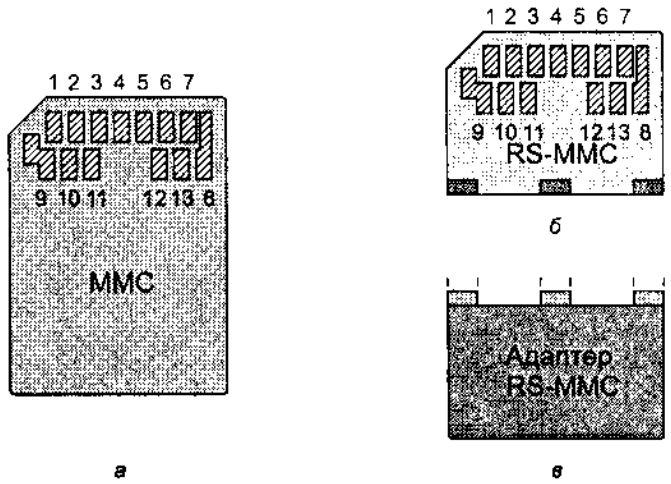


Рис. 9.16. Карты памяти MMC: *а* — обычная *(вид сзади), б* — DV-RS-MMC, *в* — адаптер

В режиме SPI используется однобитный интерфейс с раздельными линиями ввода и вывода [6]. В этом режиме CRC-контроль передач не применяется. Карты можно соединять в цепочку, для выбора карты задействуют сигнал CS.

Назначение контактов карт MMC и SD приведено в табл. 9.13. У обычных карт MMC отсутствуют контакты 9-13, у карт SD нет контактов 10-13.

Таблица 9.13. Назначение контактов карт MMC и SD

Контакт	Цепь в режиме MMC	Цепь в режиме SPI
1	DAT3*	CS
2	CMD	DI
3	Vss1	Vss1
4	Vdd	Vdd
5	CLK	SCLK
6	Vss2	Vss2
7	DAT0	DO
8	DAT1*	-
9	DAT2*	-

продолжение

Таблица 9.13 (*продолжение)*

Контакт	Цепь в режиме MMC	Цепь в режиме SPI
10	DAT4*	–
11	DAT5*	–
12	DAT6*	–
13	DAT7*	–

Карты *Secure Digital* (SD) «отпочковались» от первоначального варианта карт MMC. В картах SD используется 4-битная шина данных. В первоначальной версии частота интерфейса достигала 25 МГц, в версии SD1.1 частота поднята до 50 МГц. Конструктивно карты SD сделаны так, что устройства, работающие с ними, физически могут взаимодействовать и с картами MMC, но не наоборот (в тонкий слот MMC более толстую карту SD и не вставить). Логическая совместимость должна обеспечиваться программным обеспечением хоста (устройства, в которое вставляют карту). Карты SD разработаны альянсом трех компаний — Toshiba, Matsushita (более известной по торговой марке Panasonic) и SanDisk, которые организовали ассоциацию SDA (Secure Digital Association), выпускающую спецификации, принятые уже сотнями фирм во всем мире. Членство в ассоциации платное и дорогостоящее; по вполне понятным причинам технические детали SD широко не освещаются (иначе грош цена провозглашаемой безопасности).

Карты SD (рис. 9.17, а) имеют размер 32 x 24 x 2,1 мм; справа на рисунке виден переключатель защиты записи (WP), предохраняющий от случайного стирания. На печатной плате карты SD смонтированы флэш-память структуры NAND, SD-контроллер и вспомогательные компоненты. 9-контактный разъем карт SD и MMC совпадает (см. табл. 9.13). Карты SD допускают «горячее» подключение/отключение. Конструкция коннектора рассчитана на 10 ООО циклов вставки-изъятия. Карты выдерживают до 200 000 - 300 000 циклов записи в каждый блок флэш-памяти и падение на пол с высоты 3 м. Они устойчивы и к жаре, и к морозу. Скорость передачи данных у первых карт SD — 2 Мбайт/с, объем — 8-512 Мбайт; в 2005 году достигнута скорость чтения и записи 22,5 и 15 Мбайт/с и объем до 2 Гбайт. Начальная удельная стоимость хранения была высокой — $3 за 1 Мбайт, но в 2005 году она уже снизилась до уровня $0,1 за 1 Мбайт (что тоже немало!).

Карты MiniSD (рис. 9.17, *б)* имеют те же свойства, но уменьшенные габариты (20 x 20 x 1,27 мм) и емкость (пока до 1 Гбайт), питание — 2,7-3,6 В. Для установки в слот стандартных карт SD предназначены пассивные адаптеры (рис. 9.17, *в).*

Для карт SD разработаны (и разрабатываются) спецификации форматов данных на трех уровнях:

- физический уровень описывает перезаписываемые карты (SD-Rewritable) и постоянную память (SD-Read Only);
- уровень файловой системы использует стандарт ISO 9293;

Рис. 9.17. Карты SD: а — обычная *(вид сзади), б* — MiniSD, *в* — адаптер

- прикладной уровень учитывает специфику атрибутов различных типов содержимого — музыки (SD-Audio), изображений (SD-Picture), речи (SD-Voi- ce), видео (SD-Video) и др.

На всех уровнях действуют средства безопасности (security), являющиеся основным «коньком» SD. В SD используются технологии обеспечения безопасности CPRM (Content Protection for Recordable Media — защита содержимого записываемых носителей) — стандарта шифрования и сертификации/ аутентификации, разработанного и лицензируемого фирмами IBM, Intel, Matsushita (Panasonic) и Toshiba. Карты SD-Audio отвечают требованиям SDMI (Secure Digital Music Initiative) к портативным устройствам.

Карта SD имеет три области хранения с разными возможностями доступа: область хранения ключей шифрования и аутентификации, область секретных данных и область данных общего назначения. Секретные данные хранятся и передаются в зашифрованном виде, их кодирование-декодирование выполняется хостом (устройством, в которое устанавливается карта). Для того чтобы установить канал обмена секретными данными, требуется взаимная аутентификация хоста и карты: хост должен «признать» карту, а карта — хост. Таким образом, обмен данными с защищенной областью карты возможен только на «фирменных» устройствах (до тех пор пока алгоритмы и ключи шифрования не попадут в руки хакеров).

Карты SD продаются в форматированном виде; при необходимости они могут быть переформатированы на специальном устройстве или хосте SD, имеющем функции форматирования. Нештатное форматирование (например, на компьютере) может привести карту в негодность — защита может сработать, как дверь, захлопнувшаяся на замок, единственный ключ от которого остался внутри.

Совместимость SD и MMC весьма ограничена. Хосты карт SD совместимы с картами MMC на физическом уровне; в картах MMC и SD используется одна и та же структура таблицы размещения файлов (FAT), чем обеспечивается совместимость и на уровне записи/чтения файлов. Однако на прикладном уровне программные спецификации («начинки») этих карт могут и различаться. Так, в SD MP3-плееры из-за разницы в форматах данных нельзя вставлять карты

MMC. Данные общего назначения (несекретные) могут быть перенесены с MMC на SD, но перенос секретных данных хосты не допустят (если хост — не компьютер со взломанным ПО). Хосты для карт MMC и SD несовместимы даже чисто физически (карты SD в слот для MMC не влезут по толщине).

Помимо карт SD-памяти планируется выпуск *карт ввода-вывода* (SD I/O card). В первую очередь рассматривается интерфейс Bluetooth, позволяющий быстро и без проводов синхронизировать данные устройств, имеющих слот SD, друг с другом и с устройствами с этим интерфейсом.

Miniature Card

Карты Miniature Card предназначены для использования в недорогих устройствах бытовой электроники в качестве сменной флэш-памяти, а также расширения динамической памяти (рис. 9.18). В интерфейсе карт обеспечивается линейный доступ к произвольной ячейке памяти с адресуемым объемом до 64 Мбайт. Карты имеют немультиплексированную 16-разрядную шину данных и могут работать в пакетном режиме передачи. Также имеется возможность работы с 8-разрядным хостом (старший и младший байты шины запараллеливаются, линия BS8# заземляется). Интерфейс хоста для этих карт конфигурируется на функционирование в режиме DRAM или флэш-памяти; карты снабжаются микросхемой энергонезависимой памяти идентификации с интерфейсом I^2C. Карты имеют прорези для правильного позиционирования и коннектор, обеспечивающий «горячее» (даже для DRAM!) подключение. При установке передний край карты вводят в слот, при этом подключаются контакты «первой очереди»: GND, VCC и CINS# (контакт, замыкающийся на «землю»). Затем карта опускается вниз, прижимаются контакты основного эластомерного разъема и замыкается контакт обнаружения CD#. Карты могут работать с питанием 5 и 3,3 В; для идентификации питания служат сигналы VS1#, VS2#. Карта Miniature Card через переходный адаптер может устанавливаться в слот PC Card типа 2.

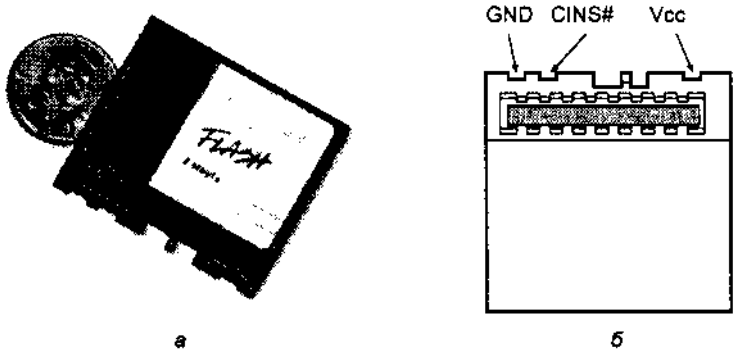


Рис. 9.18. Карта Miniature Card: а - общий вид, *б* — вид снизу

При работе по интерфейсу DRAM используются линии адреса A[12:0] и управляющие сигналы RAS#, CASH# и CASL# (стробы для старшего и младшего байтов),

а также WE#. Карты имеют отдельную линию питания V_{CCr} для регенерации при отключенном основном питании.
При работе по интерфейсу флэш-памяти используются линии адреса A[24:0] и управляющие сигналы 0E# (чтение), WE# (запись), CEH# и CEL# (выборка для старшего и младшего байтов).

9.11. Системная поддержка внешней памяти

Дисковая память имеет стандартную поддержку на уровне BIOS и операционной системы. Поддержка дисков со стороны BIOS заключается в предоставлении вышестоящим уровням программного обеспечения возможности чтения и записи секторов диска, форматирования трека и выполнения вспомогательных функций. Эти возможности предоставляются программным вызовом прерывания BIOS Int 13h — дискового сервиса. Дисковый сервис BIOS предназначен для изоляции вышестоящего ПО (ОС и приложений) от подробностей реализации дисковой системы. Дисковый сервис Int 13h работает на уровне *физических устройств*, называемых также *физическими приводами* (physical drive). При вызове требуется задать номер функции (сервиса), логический адрес устройства (не путать с именем логического диска!), адрес сектора и число передаваемых секторов. Кроме того, нужно указать, в каком месте оперативной памяти находится буфер для обмена данными с диском. Все хлопоты по взаимодействию с контроллером требуемого диска берет на себя BIOS. Драйверы традиционных устройств — НГМД и винчестеров с интерфейсом ATA — находятся в системной микросхеме BIOS. Для других устройств, в том числе и винчестеров с интерфейсом SCSI, используются специальные собственные драйверы. Для дисков SCSI они располагаются в дополнительном модуле BIOS, находящемся на карте контроллера SCSI (или в системной микросхеме BIOS, если контроллер SCSI расположен на системной плате) или в загружаемых программных модулях.
Стандартные драйверы дисковых функций BIOS (включая и расширенный сервис) имеют однозадачное происхождение. Во время выполнения функции значительное процессорное время может затрачиваться на ожидание завершения операции относительно медленным (по своей электромеханической природе) устройством. Драйверы многозадачного режима построены иначе: у них есть *вызывающая часть*, инициализирующая начало операции, и *обработчик аппаратного прерывания* от контроллера, сообщающий операционной системе о выполнении операции и результате. Таким образом, время, необходимое контроллеру и накопителю для выполнения операции, для процессора остается свободным. Естественно, что многозадачный драйвер операционной системы должен иметь точное представление об аппаратной реализации контроллера, поскольку общение с ним идет на уровне регистров, а не функций BIOS. Это является одной из причин того, что «серьезную» операционную систему нужно устанавливать на компьютер, на котором она будет работать, в конкретном

аппаратном окружении. Простое копирование операционной системы с соседнего компьютера может быть чревато серьезными проблемами и даже потерей данных.

Сервисы BIOS работают в реальном (16-разрядном) режиме процессора. Для современных ОС, работающих в 32-разрядном защищенном режиме, вызовы функций BIOS неэффективны еще и из-за необходимости согласования режимов, сопровождаемого большими накладными расходами. Своей функцией 48h расширенный дисковый сервис дает операционной системе все сведения, необходимые для реализации обеих частей дисковых драйверов, работающих в защищенном режиме процессора.

Интерфейс дискового сервиса допускает многосекторные операции чтения и записи, но для дискет запрашиваемая цепочка секторов не должна переходить за границу адресованного трека. В зависимости от того, поддерживают ли конкретный адресуемый контроллер (и привод) многосекторные передачи (и разрешена ли эта возможность в CMOS Setup), многосекторный вызов будет транслироваться на физическое устройство либо цепочкой одиночных запросов, либо как многосекторный. Заметим, что некоторые старые накопители могут некорректно отрабатывать многосекторные запросы на физическом уровне. Если имеются подозрения на эту неисправность, многосекторный режим, часто даже для каждого накопителя индивидуально, может быть запрещен настройкой CMOS Setup. Конечно, при этом теряется производительность из-за увеличения доли «накладных расходов» на формирование запросов.

Традиционный сервис BIOS INT 13h

Традиционный дисковый сервис (функции 00-3F) имеет программный интерфейс, сохранившийся еще со времен IBM PC/XT. Параметры вызова передаются через 16-разрядные регистры процессора, что накладывает ограничение на возможности адресации. В традиционном сервисе используется трехмерная адресация CHS: *номер цилиндра* (0-1023), *номер головки* (0-255) и *номер начального сектора* (1-63). Из-за несовпадения разрядности компонентов C:H:S (10:8:
6 бит) с разрядностью интерфейса ATA (16:4:8 бит) он позволяет непосредственно оперировать дисками ATA размером до 528 Мбайт (общедоступны 10:4:6 бит). Когда появились диски большего размера, в традиционный дисковый сервис ввели расширение (алгоритмы преобразования адреса), позволяющее преодолеть этот барьер и, теоретически, работать с дисками объемом до
8,4 Гбайт. Однако на практике по пути к объему 8,4 Гбайт пришлось преодолевать еще несколько барьеров, вызванных просчетами разработчиков ОС и BIOS. Трехмерный адрес CHS для дисков ATA к реальному физическому адресу отношения практически не имеет. Для дисков с интерфейсом SCSI, у которых адресация секторов линейная, BIOS контроллера SCSI эмулирует трехмерную геометрию.

Формально традиционный сервис позволяет работать с дисками, имеющими до 1024 x 256 x 63 = 16 515 072 секторов (около 8,4 Гбайт). Ряд операционных систем имеет ошибку, не позволяющую использовать полный объем, допустимый

данным сервисом. Для дисков объемом более 15 481 935 секторов следует пользоваться только функциями расширенного сервиса (см. далее).

Традиционно дисковый сервис подразделяет физические диски на *дискеты* (diskette, номера 0-7Fh) и *фиксированные диски* (fixed disk, номера 80h-FFh). Набор функций для этих классов устройств несколько различается как по составу, так и по реализации (выполняются разными модулями). Контроллеры дисковых интерфейсов, имеющие в своем составе дополнительные модули BIOS, перехватывают вектор Int 13h, беря на себя обслуживание своих устройств. Помимо функций дискового сервиса (Int 13h) с дисковыми устройствами связаны еще и векторы, обслуживающие аппаратные прерывания от контроллера НГМД — Int 0Eh (линия IRQ 6) и от контроллера жестких дисков — Int 76h (линия IRQ 14). При наличии двухканального порта ATA второй канал обычно задействует линию IRQ 15 (вектор 77h). В XT контроллер жестких дисков занимал линию IRQ 5 (вектор 0Dh). Дополнительные контроллеры дисков могут использовать и другие прерывания. Аппаратные прерывания вырабатываются контроллерами по завершении (нормальном и аварийном) внутренних операций. На эти прерывания BIOS не реагирует, а при инициализации их векторы направляются на программную заглушку (инструкцию IRET).

Традиционные сервисы пользуются таблицами параметров: *таблица параметров дискет* (Diskette Parameter Table, DPT) задается указателем в памяти по адресу 0:78h, *таблица параметров жестких дисков* (Hard Disk Parameter Table, HDPT) — по адресу 0:104h или 0:118h. Стандартная таблица HDPT морально устарела: она не позволяет различать физическую и логическую геометрию, зато в ней есть параметры предварительной компенсации записи и парковки, которые уже очень давно не используются. В расширенной версии BIOS, в которой реализована трансляция геометрии, таблица параметров имеет иной вид — в ней представлена и физическая, и логическая геометрия диска. Параметры жестких дисков заносятся в таблицы во время начального тестирования компьютера (во время теста POST). Они задаются утилитой CMOS Setup или определяются автоматически по фактически подключенным устройствам (в зависимости от настройки Setup). Таблицы параметров жестких дисков не позволяют описывать диски размером более 33,8 Гбайт, поскольку предел физической геометрии — это 65 536 x 16 x 63. Однако для современных ОС, оперирующих большими дисками, это не проблема — они попросту данными таблицами не пользуются, а работают в режиме LBA.

Расширенный сервис BIOS INT 13h

Расширенный дисковый сервис (Enhanced Disk Drive Services, EDD) BIOS, предложенный фирмой Phoenix Technologies LTD, реализуется многими разработчиками BIOS и устройств массовой памяти. Он позволяет работать с устройствами, имеющими объем до 2^{64} секторов, эффективно используя архитектуру 32- и 64-разрядных процессоров. Сервис оперирует *линейным логическим адресом* (LBA) сектора. Вместо традиционных таблиц параметров дисков в нем применяются новые, дающие исчерпывающую информацию об устройствах, их физической организации и интерфейсе. Устройства могут иметь сменные носители

и сами быть съемными (имеется в виду возможность изъятия устройства в процессе работы компьютера — например, подключенного к шине USB или IEEE- 1394), так что понятие «сменяемость носителя» несколько размывается. Такие устройства должны поддерживать механизм уведомления о смене носителя и программное блокирование смены носителя. По прогнозам, емкости данного интерфейса должно хватить на 15-20 лет.

Расширенный сервис, как и традиционный, вызывается *программным прерыванием* Int 13h с номерами функций свыше 3Fh, номер устройства допустим в диапазоне 80h - FFh. Основные параметры вызова — начальный адрес блока, число секторов для передачи и адрес буфера — передаются через *адресный пакет устройства* (device address packet). Подобная передача параметров в сравнении с передачей через регистры процессора, что характерно для традиционного сервиса, предоставляет больше возможностей, так как формат пакета довольно просторный.

Поскольку расширение (и отдельные его функции) BIOS может отсутствовать, имеется *функция проверки наличия расширения.* Расширение может действовать избирательно (не для всех устройств), так что проверку надо производить для конкретного устройства, интересующего программу. *Функции расширенных чтения, записи, верификации и поиска* по смыслу не отличаются от их аналогов для традиционного сервиса. Для работы со сменными носителями введены *функции отпирания-запирания, извлечения* и *проверки факта смены носителя.* От идеологии традиционного сервиса сильно отличается *функция получения параметров устройства.* Она возвращает в ОЗУ буфер с набором параметров и детальным описанием устройства, позволяющим ОС и приложениям работать с ним, минуя BIOS. *Функция установки аппаратной конфигурации* позволяет управлять режимом передачи (PIO, DMA), а также предварительной выборкой (поиском).

Расширения BIOS Int 13h используют ОС Windows 95, Windows 98, Windows 200x/XP. Правда, это использование ограничено лишь начальной загрузкой и процессом установки (FDISK, FORMAT), поскольку в регулярной работе применяются собственные 32-разрядные драйверы. Расширения BIOS Int 13h *не* используют DOS (все версии), Windows 3.1x, Windows NT, Novell NetWare, OS/2 Warp, Linux, Unix.

Преодоление барьера 528 Мбайт (ECHS и LBA)

При работе с устройствами ATA имеется барьер в 528 Мбайт: для них предусмотрен только 4-битный регистр номера головки и 6-битный номер сектора (в BIOS — 6 и 8 бит соответственно), но при этом используется 16-битный номер цилиндра (в BIOS — 10 бит). Выбрав самые жесткие ограничения по каждой из координат, получаем барьер на уровне около 528 миллионов байт:

(2^{10} = 1024 цилиндра) x (2^4 = 16 головок) x (2^6 = 1 - 63 сектора) x x 512 байт - 528 482 304 байт.

Чтобы преодолеть 528-мегабайтный барьер дисков ATA, не трогая программного интерфейса, в BIOS ввели расширение традиционного дискового сервиса.

Интерфейс ATA в трехмерной геометрии позволяет реализовать довольно большой (но уже не запредельный) объем диска:

$(2^{16} = 65\ 536$ цилиндров) x ($2^4 = 16$ головок) x x ($2^8 - 1 =$ 255 сектора) x 512 байт = 136,9 Гбайт.

Чтобы достичь хотя бы интерфейсного ограничения BIOS (8,4 Гбайт), стали применять *трансляцию параметров вызова* функций Int 13h, которые назовем *логическими*, в *физические*[1] параметры, передаваемые контроллерам ATA-дисков. Естественно, логический объем диска не может превышать физического:

$$(C \times H \times 5)_{лог} \le (C \times H \times 5)_{физ}.$$

Есть два основных пути трансляции: оставаться в пределах трехмерной геометрии *CHS* и перейти на логическую адресацию блоков *LBA*. Возможности выбора пути трансляции определяются конкретной версией BIOS.

При выборе в CMOS Setup параметров, называемых *Large Disk* или *ECHS* (Extended CHS), общение с накопителем производится по трехмерной схеме, но с преобразованными номерами цилиндров и головок. Способов трансляции, к сожалению, может быть несколько. Можно, например, физическое количество цилиндров диска разделить на К (К = 2, 4, 8, 16... при этом параметр преобразования К определяется так, чтобы результат не превышал 1024), а физическое число головок умножить на К (это произведение не должно превышать 255). Умножение и деление на степень двойки легко выполняются путем сдвигов, программный код трансляции получается компактным и работает быстро. Эта схема преобразования, называемая трансляцией сдвигом битов (bit shift translation), рекомендуется фирмой Phoenix для включения в расширенный дисковый сервис BIOS. Она работает на любых дисках ATA, как поддерживающих LBA, так и не поддерживающих. Количество секторов на треке при трансляции не изменяется. Существуют и другие способы трансляции геометрии, но здесь мы их рассматривать не будем. Нам важно знать лишь цель преобразования и тот факт, что *из-за несовпадения методов трансляции в режимах Large Disk (ECHS) возможна несовместимость дисков, размеченных для различных версий BIOS.*

Когда в CMOS Setup выбирается параметр *LBA*, это означает использование схемы *трансляции с помощью LBA* (LBA assisted translation).

Здесь параметры стандартных вызовов транслируются в *линейный адрес*, который вычисляется однозначно в «естественном» порядке счета секторов. Сектору с нулевым логическим адресом соответствует первый сектор нулевой головки нулевого цилиндра. Общая формула вычисления логического адреса выглядит так:

$$LBA - (CYL \times HDS + HD) \times SPT + SEC - 1.$$

Здесь CYL, HD и SEC — логические номера цилиндра, головки и сектора в пространстве CHS; HDS — количество головок; SPT — количество секторов на тре

[1] Зная устройство современных винчестеров, здесь и далее мы не будем добираться до истинно физических параметров — реальных номеров цилиндра, головки и сектора.

ке. Трехмерную геометрию в ответ на вызов функции 8 BIOS Int 13h сообщает следующим образом:

- количество секторов на треке (SPT) — всегда 63;
- количество головок (HDS) — 16, 32, 64, 128 или 255, в зависимости от объема диска;
- количество цилиндров определяется делением общего числа секторов на произведение SPT x HDS.

Число головок выбирается таким, чтобы полученное число цилиндров не превышало 1024. Недостатком данного метода являются более сложные вычисления при преобразованиях (здесь при умножении и делении уже не обойтись просто сдвигами).

При использовании данного метода вовсе не обязательна линейная адресация в обращениях к физическим устройствам ATA — если диск способен установить подходящую геометрию, то обращения могут производиться и в режиме CHS. Обращение к диску в режиме LBA, безусловно, выгодно лишь когда BIOS обслуживает вызов расширенной функции 4x, поскольку при этом не требуется никакой трансляции (экономится процессорное время). Для того чтобы определить, в каком режиме используется диск ATA, можно прочитать байт из регистра D/H (1F6 для первичного контроллера или 176 для вторичного). Значение Ex или Fx указывает на то, что в последней операции применялся режим LBA (для ведущего или ведомого устройства соответственно), Ax или Bx — режим CHS. Прочитать порт можно, например, с помощью команды I 1F6 (I 176 для вторичного контроллера) отладчика DEBUG (в среде DOS).

Итак, для дисков размером более 528 Мбайт BIOS определяет фиктивные параметры геометрии (если большие диски поддерживаются). Операционной системе и прикладным программам они будут сообщаться как реальные, а BIOS при вызовах дисковых сервисов преобразует параметры вызова так, чтобы номера цилиндров и головок принадлежали пространству «внешних координат» ATA-диска.

Еще раз подчеркнем, что 528-мегабайтный барьер для дисков ATA имеет чисто *программное происхождение.* Электрический интерфейс и регистры контроллера накопителя ограничивают объем примерно до 137 Гбайт. Средства преодоления этого барьера (и последующих) носят чисто программный характер и должны быть реализованы на уровне BIOS. Возможные варианты перечислены ниже:

- Современные версии BIOS имеют встроенный механизм трансляции, который включается для дисков объемом более 528 Мбайт. При трансляции может использоваться режим LBA, если его поддерживает диск, или ECHS (Large Disk), который доступен для любых ATA-дисков. Выбор режима трансляции выполняется при конфигурировании диска в CMOS Setup. Ряд версий BIOS не поддерживают LBA, некоторые не поддерживают ECHS. Если в компьютере применяется флэш-BIOS, но трансляция не поддерживается, имеет смысл обновить версию BIOS, чтобы проблема преодоления барьера решалась самым лаконичным способом.

- Для компьютеров со старыми версиями BIOS, не поддерживающими встроенной трансляции и не допускающими легкого обновления, выпускались платы интерфейса E-IDE, снабженные микросхемой расширения ROM BIOS. Эта микросхема хранит программный код драйвера Int 13h, реализующего тот или иной тип трансляции (обычно ECHS). В остальном плата могла не отличаться от традиционного адаптера ATA или иметь интерфейс шины VLB (реже PCI) и более сложный контроллер с буферами и управлением шиной (Bus-Master).
- Если BIOS трансляцию не поддерживает, а плата с расширением ROM BIOS не применяется, то остается программная загрузка драйвера Int 13h. Однако этот драйвер должен быть загружен до загрузки операционной системы, которая уже опирается на BIOS, например, через главный загрузчик (MBR). Код этого загрузчика исполняется в конце теста POST, и если он до загрузки загрузчика операционной системы (загрузчика из активного раздела диска) поместит в память новый драйвер Int 13h, то операционная система воспримет его как часть BIOS. На таком принципе работают драйверы типа Disk Manager и EZ-Drive. Данное решение проблемы имеет ряд неудобств. Во-первых, загружаемый драйвер использует часть области стандартной памяти, которую бывает жалко. Во-вторых, процедура конфигурирования диска с таким драйвером сложнее, чем обычного, поскольку требует установки дополнительного программного обеспечения, да еще и размещаемого в системной области диска. И в-третьих, в нештатных ситуациях (например, под действием вируса) обслуживание диска становится более тонкой задачей. Так, универсальное (хотя и не общеизвестное) «лечение» главной загрузочной записи командой FDISK /MBR заменяет загрузчик драйвера стандартным загрузчиком, и процедуру конфигурирования диска придется повторять. Конечно, в Disk Manager имеются соответствующие утилиты, но о необходимости их использования нужно помнить, а сами утилиты хранить на дискете. Однако ради увеличения доступного объема диска приходится идти и на такие издержки.

Поскольку преобразования в LBA отличаются от ECHS, диски, размеченные в режиме LBA, не работают в режиме ECHS, и наоборот. Тот же результат возможен при переносе диска, размеченного в режиме ECHS, на другой компьютер, у которого применен другой алгоритм трансляции. Если повезет, то эта неработоспособность проявится сразу в виде невозможности загрузки операционной системы (сообщение «Missing Operation System»). Значительно хуже, когда ОС загружается и, как кажется, нормально работает, а в это время, возможно, все записи на диск идут по неправильным адресам и крупная потеря данных неумолимо приближается. По той же причине *нельзя пытаться решать проблему больших дисков, форматируя их на компьютере с расширенной версией BIOS, а эксплуатируя на машине со стандартной версией, имеющей 528-мегабайтный барьер.* Сначала все будет хорошо, но когда начальная 528-мегабайт- ная область заполнится, следующая запись пойдет по нулевому цилиндру. А там находятся и главная загрузочная запись, и таблица разделов, и копии FAT. До области данных эта перекрывающая запись дойти, скорее всего, не

успеет, но легче от этого не станет — данные будут потеряны. Сказанное не очень актуально для операционных систем, не использующих BIOS при обращении к дискам, — если такую ОС удается загрузить с диска, то проблему большого диска можно считать решенной.

Сервисы операционной системы

Операционная система предоставляет прикладным программам сервисы более сложные, чем функции BIOS, обеспечивая доступ к файловой системе диска. Самый низкий уровень обращения, допускаемый операционной системой, оперирует с секторами *логических дисков*, или *логических приводов* (logical drive), ассоциированных с именами устройств (A, B, C, ..., Z). Один *физический диск* (жесткий) может содержать несколько логических дисков (см. п. 9.6). На физическом диске могут присутствовать системные области (MBR и резервированные секторы), доступ к которым операционная система не предоставляет.

Для работы утилит и приложений с дисковой памятью имеется несколько уровней сервиса:

- Обращение к контроллеру дисков на уровне регистров с заданием адреса блока и количества секторов, требуемых для передачи. Это самый производительный способ обмена данными, но он требует знания как организации разделов диска и файловой системы, так и контроллера диска (его системы команд). Доступ обеспечивается ко всем элементам, кроме блоков, закрытых командой Set Max Address (для дисков ATA). При этом обходится фаза трансляции геометрии, выполняемая, в общем, по усмотрению BIOS.
- Использование сервисов BIOS Int 13h также дает неограниченный доступ к диску, но при традиционных вызовах (функций с номерами ниже 40h) доступны лишь диски объемом до 8,4 Гбайт (при трансляции геометрии) или до 528 Мбайт (без трансляции). За абстрагирование от системы команд контроллера приходится расплачиваться некоторым снижением производительности, связанным с накладными расходами программного интерфейса.
- Используя интерфейс функций DOS, можно получить доступ (по функциям чтения-записи абсолютных секторов) к любому сектору логического диска, указав лишь его логическое имя. Но при этом первым доступным сектором (с логическим адресом 0) становится загрузочный сектор логического диска, а секторы с таблицами разделов и те, что не попадают в разделы логических дисков, оказываются недоступными. Для осмысленного доступа к данным необходимо знать структуры FAT и каталогов, и риск «наломать дров», вплоть до полной потери данных, довольно велик.
- И наконец, интерфейс файловой системы ОС позволяет довольно легко создавать, искать и удалять файлы и каталоги, читать и записывать файлы целиком или частично и выполнять некоторые другие вспомогательные функции. При этом от пользователя данного интерфейса скрываются все тонкости работы с FAT и элементами каталогов (и, конечно же, все премудрости нижних уровней).

Прикладные программы и утилиты изолируют пользователя от всех этих тонкостей, что позволяет ему сконцентрировать внимание на решении своих задач. Уровень, на котором программа обращается к дискам, выбирается программистом из соображений минимальной достаточности: чем выше уровень сервиса, тем легче достичь совместимости с остальным ПО (предоставляется меньше возможностей для ошибок). Чем ниже уровень сервиса, тем больше знаний нужно вложить в программу, чтобы она общалась с диском в соответствие с общими правилами.

Системная поддержка CD-ROM

Поскольку диск CD-ROM по организации данных (файловой системе) существенно отличается от традиционных дисков (гибких и жестких), для «прозрачного» доступа приложений к файлам на CD-ROM требуются специальные программные средства. Несмотря на возможность загрузки ОС с CD-ROM, реализованную в современных версиях BIOS, полной поддержки доступа к CD-ROM (такой, как к обычным дискам) BIOS не предоставляет. Приложениям доступ к CD-ROM обеспечивают только сервисы операционной системы, встроенные в ОС или загружаемые. Для MS-DOS имеется утилита MSCDEX.EXE (Microsoft DOS extensions for CD-ROM), загружающая резидентный драйвер эмуляции логического диска DOS с файловой системой FAT16 при чтении CD-ROM с томами формата ISO 9660 или HSF. Для ОС Windows и ряда других, изначально спроектированных с учетом поддержки CD-ROM, расширение типа MSCDEX не требуется, но, естественно, сам привод должен поддерживаться системным драйвером.

В ОС Windows доступ к параметрам и свойствам приводов CD-ROM осуществляется через Панель управления. Здесь можно задавать режимы обмена, логическое имя (букву) диска, а также автоматическое распознавание диска AIN (Auto Insert Notification — автоматическое уведомление системы об установке носителя). На автоматическом распознавании базируются автоматический запуск проигрывателя (autoplay) при загрузке аудиодиска и автоматический запуск приложения (autorun), указанного в файле AUTORUN.INF. Когда эти функции разрешены, пользователю остается только вставить диск — все остальное делается автоматически, хочет он того или нет.

Загружаемые диски CD-ROM

Для обеспечения возможности загрузки ОС с CD-ROM фирмы Phoenix и IBM в году выпустили спецификацию «E1 Torito Bootable CD-ROM Format Specification». Цель спецификации — обеспечить возможность загрузки ОС и приложений с CD-ROM средствами BIOS (на «голой» машине). При этом имеются следующие возможности:

- загрузка ОС по выбору из загрузочного каталога (boot catalog), находящегося на CD-ROM;
- предоставление выбора при конфигурировании CD-ROM (в виде жесткого диска или дискеты);

- переименование существующих приводов (если необходимо);
- использование сервисов BIOS (доступ в режиме LBA) для обращения к кодам и данным;
- совместимость с приложениями DOS и Windows, использующими функции Int 13h.

Спецификация расширяет традиционный набор функций BIOS, она ориентирована на приводы с интерфейсами ATAPI и SCSI. В пункт выбора последовательности загрузочных устройств утилиты CMOS Setup должно быть введено новое устройство — CD-ROM и должен предлагаться выбор последовательности опроса устройств, например:

- A:, CD-ROM, C:;
- CD-ROM, A:, C:;
- CD-ROM, C:, A:;
- C:, A:, CD-ROM;
- CD-ROM only.

В параметрах всегда должна быть возможность запрета загрузки с CD-ROM, поскольку формат начального сектора CD может быть произвольным (в зависимости от назначения диска), а загружаться можно только со специального диска CD-ROM. Загружаемый диск CD-ROM должен иметь загрузочную запись, загрузочный каталог и файл-образ дискеты или жесткого диска. На одном диске CD-ROM может быть несколько образов; все они должны быть описаны в загрузочном каталоге.

При загрузке образа дискеты CD-ROM представляется в виде диска A, при этом существующий первый дисковод получает имя B. Если в системе два дисковода, то второй становится недоступным. При загрузке с образа жесткого диска CD-ROM замещает диск C (номер устройства 80h), номера жестких дисков сдвигаются, они все остаются доступными. Для доступа приложений и ОС к данным, расположенным в области образа диска, могут использоваться вызовы Int 13h без загрузки каких бы то ни было драйверов. Образ диска выглядит, как обычный диск, но защищенный от записи. Для доступа к остальному содержимому CD-ROM должны загружаться драйверы, необходимые операционной системе и размещенные в образе диска.

Функции начального загрузчика (bootstrap loader) Int 19h, выполняемого в конце теста POST, расширяются. Если ему предписана попытка загрузки с CD-ROM, то он читает загрузочную запись тома и, если она присутствует, берет из нее ссылку на загрузочный каталог CD-ROM, в котором должны быть ссылки на загрузочный образ. Возможны два варианта реализации загрузчика в BIOS: с единственным образом (single-image Int 19h) или с множеством образов (multiple-image Int 19h). «Единообразный» загрузчик обращается к образу, описанному в элементе каталога, предназначенном для загрузки по умолчанию. «Многообразный» загрузчик позволяет выбрать требуемый образ из списка возможных автоматически или запрашивая ответ пользователя.

Собственно диск CD-ROM может иметь разные варианты структуры: обычный (незагрузочный), загрузочный с одним образом, загрузочный с множеством образов, а также незагрузочный, но с образами, доступными через Int 13h. После загрузки ОС доступ ко всем образам, указанным в загрузочном каталоге, возможен через Int 13h, как к обычным дискам, и если их файловая система поддерживается загруженной ОС, доступ к файлам и каталогам тоже будет обычным. Доступ к областям CD-ROM, не входящим в образы, на логическом уровне осуществляется через специальный драйвер. Доступ ко всем секторам CD-ROM возможен через функции 41-48h Int 13h в режиме LBA, размер блока 2048 байт. Спецификация совместима с дисками ISO-9660.

Загрузочный образ представляет собой посекторную копию эмулируемого диска, секторы копируются друг за другом в естественном порядке (по нарастанию номера LBA от 0 до конца диска). Эмулятор, встроенный в BIOS Int 13h, для традиционных вызовов (функции 0 - 19h) осуществляет пересчет виртуальных секторов (размером 512 байт, как у всех дисков) в 2048-байтные секторы CD- ROM. BIOS Int 13h отвечает за представление трехмерной (CHS) геометрии образа, по которой к нему могут обращаться программы, использующие этот интерфейс. Спецификация требует, чтобы образ жесткого диска содержал только один раздел, описанный обязательно в первом элементе таблицы разделов. Эмулируемый диск описывается как устройство ATAPI, что позволяет программам при необходимости распознать факт эмуляции.

При обращении к CD-ROM в процессе загрузки проверяется индикатор загружаемости выбранного образа (в загрузочном каталоге); если он отличен от 88h, то производится обращение к следующему устройству в загрузочной последовательности, заданной в CMOS Setup. Но при этом включается эмуляция дисков, и ОС, загруженная с другого устройства, получает доступ к содержимому образов через Int 13h без всяких драйверов.

К сервисам BIOS Int 13h добавляются несколько новых функций:

- *начать-завершить эмуляцию диска* (4Ah/4Bh), *начать эмуляцию диска и выполнить загрузку* (4Ch);
- *прочитать секторы загрузочного каталога* (4Dh); функции 41 - 48h позволяют обращаться к любым логическим секторам CD-ROM (в режиме LBA с размером сектора 2048 байт), когда для данного привода включена эмуляция.

9.12. Установка и обслуживание устройств

В этом разделе описываются типовые проблемы, с которыми сталкиваются пользователи, даются их объяснение на основе материала предыдущих разделов и практические советы.

Установка новых устройств

Теоретически установка нового устройства в компьютер производится довольно просто: конфигурируется аппаратный интерфейс (если требуется), устрой

ство подключается к компьютеру и при необходимости выполняется настройка BIOS. Далее может потребоваться конфигурирование (разбивка жесткого диска на разделы), форматирование и «заселение» устройства полезной информацией.

Для аппаратного конфигурирования у устройств *ATA/ATAPI* (см. 19.2) устанавливается только джампер адресации в одно из положений: *M* (Master — ведущее устройство, оно же устройство 0), *S* (Slave — ведомое устройство, оно же устройство 1) или *CS* (Cable Select — кабельная выборка).

ВНИМАНИЕ ----------

На одном канале ATA не должно быть двух ведущих или двух ведомых устройств. В старых системах (с неинтеллектуальным контроллером ATA) для работы ведомого наличие ведущего в канале было обязательным, новые контроллеры при отсутствии ведущего сами исполняют его интерфейсные обязанности.

Устройства *SATA* конфигурирования не требуют, однако в CMOS Setup появляются дополнительные параметры конфигурирования контроллера. Эти параметры позволяют совместно использовать интерфейсы ATA и SATA, а также определяют режим работы контроллера SATA, от совместимого (с парой устройств M/S) до самого современного (AHCI). Адрес устройства определяется номером порта SATA, к которому оно подключается, и соответствующими параметрам CMOS Setup. Если порт SATA сконфигурирован на работу в режиме эмуляции устройств IDE[1], то для него указываются номер канала IDE (Pri/ Sec — первичный/вторичный) и «роль» устройства (Master/Slave). В этом случае параллельный интерфейс эмулируемого канала ATA становится недоступным. Заметим, что из-за парности устройств каналов IDE назначение эмуляции для одного порта SATA отрезает доступ сразу к паре устройств параллельной шины. Правда, при этом автоматически может включиться режим эмуляции для второго порта SATA. Режим эмуляции позволяет работать с устройствами SATA обычным драйверам IDE. Если порт SATA сконфигурирован для режима SATA, то в списке IDE устройств (выводимом в POST) появляются дополнительные каналы IDE для каждого порта SATA. Можно ли с этими устройствами работать, как с обычными устройствами IDE (но с нестандартными адресами, которые получит контроллер), зависит от модели хост-контроллера SATA (см. 19.4).

У *устройств SCSI* устанавливается идентификатор *SCSI ID*, определяющий адрес устройства на шине, и некоторые другие параметры (терминаторы, контроль четности и др., см. 20.4),

Адреса устройств определяют порядок их обнаружения и нумерации. Когда в системе имеются диски обоих типов (ATA и SCSI), порядок устройств определяется настройкой CMOS Setup и SCSI Setup, а не только джамперами, установленными на дисках.

[1] Под IDE подразумевают параллельный интерфейс с парой устройств M/S.

ВНИМАНИЕ

На одной шине (точнее, в домене) SCSI не должно быть устройств с совпадающими идентификаторами. Каждая шина должна иметь включенные терминаторы на обоих концах шлейфа. Тип интерфейса устройства — линейный (SE), дифференциальный (Diff) или низковольтный дифференциальный (LVD) — должен соответствовать интерфейсу контроллера. Сочетание дифференциального интерфейса (Diff) с линейным (SE) или низковольтным дифференциальным (LVD) недопустимо. Сочетание SE и LVD неэффективно.

Далее описывается порядок действий при установке нового внутреннего устройства хранения. Для внешних устройств с «горячим подключением» к шине USB или FireWire многие из последующих пунктов неактуальны:

1. Устройство устанавливается в *выключенный* компьютер, к нему правильно (без переворотов, смещений и заламываний контактов) подключаются интерфейсный и питающий кабели.
2. Компьютер включается и выполняется встроенная утилита CMOS Setup. В разделе Standard CMOS Setup устанавливаются требуемые типы дисков: 0-47 для старых дисков небольшого объема, 47 (User Type) для дисков ATA любой геометрии, Auto (автоматическое распознавание) или None (нет диска ATA).

   Для жестких дисков ATA и *устройств ATAPI* (CD-ROM, CD-RW, CD-R, DVD-ROM...) удобно выбрать вариант Auto; тогда, если BIOS с устанавливаемым диском работает без проблем (см. 19.2), то все установки и замены устройств не будут требовать входа в CMOS Setup. Определение параметров User Type, как правило, производится через раздел IDE Autodetection (автоматическое определение подключенных устройств и выбор способа трансляции геометрии). В некоторых случаях (например, при проблемах BIOS с большими дисками) параметры диска вводят вручную: задают число цилиндров, головок, секторов на треке, способ трансляции *(LBA* или *CHS).* Выбор режима LBA наверняка означает лишь способ трансляции геометрии (см. 19.5), при этом обращения к жесткому диску могут выполняться в режиме как LBA, так и в CHS, на усмотрение BIOS. Параметры парковки (Landing Zone) и предварительной компенсации записи (Write Precomp) ни на что теперь не влияют, их можно не указывать.

   Параметры *для дисков SCSI* при настройке CMOS Setup не указывают; если эти диски единственные (дисков ATA нет), то выбирается вариант None или Auto. Если также используются диски ATA, то параметры в CMOS Setup относятся именно к ним. Параметры жестких дисков SCSI сообщаются контроллером SCSI (его BIOS). Когда POST передает управление модулю SCSI BIOS для инициализации, на экран может выводиться приглашение к входу в утилиту SCSI Setup по определенному сочетанию клавиш (Ctrl+A, Ctrl+B, F2, F10). Эта утилита может предоставить возможность выбора параметров («геометрию») подключенных дисков, а также выполнить их низкоуровневое форматирование.
3. В разделе BIOS Features Setup выбирают требуемую последовательность опроса загрузочных устройств, в которой могут присутствовать следующие устройства:

- A — стандартный НГМД;
- C, D, E, F — диски ATA по порядку их подключения (ведущее устройство первого канала, ведомое устройство первого канала, ведущее устройство второго канала, ведомое устройство второго канала); диск, с которого произошла загрузка, получает имя C; если по порядку подключения он не первый, то нумерация остальных дисков ATA сдвигается;
- SCSI — диск SCSI с минимальным значением идентификатора, при загрузке с него он получает имя C, нумерация дисков ATA, если они есть, сдвигается (для ряда версий BIOS загрузка возможна только с устройства с ID = 0);
- CD-ROM (устройство ATAPI или SCSI);
- LS-120 (устройство ATAPI).

По умолчанию обычно установлен вариант «A, C»: при наличии дискеты в дисководе загрузка будет осуществляться с нее, при отсутствии — с диска C. Для регулярной работы лучше выбирать вариант «C, A» или «C Only», тогда при случайно забытой дискете компьютер не будет пытаться с нее загрузиться (меньше шансов «поймать» загрузочный вирус). Если нужно грузиться с SCSI при наличии дисков ATA, то ставится, например, «SCSI, A».

1. Производится загрузка ОС с выбранного диска. Непосредственно перед загрузкой на экран могут выводиться списки обнаруженных устройств, их параметры, режим обмена. Согласно последним указаниям Microsoft, эта информация считается излишней, так что пользователь может не увидеть ничего, кроме заставки загружаемой ОС.
2. Если вновь установленный диск сконфигурирован (разбит на разделы), отформатирован и не является первым (и загрузочным), нужно убедиться в том, что все его разделы «видит» загруженная ОС. После этого остается лишь «обживать» свободное пространство, при необходимости подправив пути к файлам и каталогам ранее установленных приложений. Если диск не сконфигурирован или не отформатирован, следует выполнить эти операции. Если диск должен стать загрузочным (первым), то на него еще следует установить операционную систему. Если новый диск должен заменить старый, то после установки ОС на него требуется скопировать данные со старого диска.

С новыми дисками могут поставляться утилиты, облегчающие процесс установки и «переезда». Они выполняют конфигурирование и форматирование, а также копирование файлов со старого диска и даже перенос операционной системы. Попутно они могут установить «заплатки» на BIOS, если у системной BIOS проблемы с большими дисками. Однако эти утилиты и «заплатки» рассчитаны не на все случаи жизни, и ОС не всегда экономят время. Вспоминая старую поговорку о том, что «два переезда равны одному пожару», следует быть внимательным при проведении всяких «дисковых революций» — чтобы не было «мучительно больно» от потери «добра, нажитого многолетним непосильным трудом».

К сожалению, на практике встречаются проблемы разных уровней сложности. Самые простые и часто встречающиеся — неправильное физическое подключение (перепутаны джамперы, разъемы, кабели).

ВНИМАНИЕ ----------

Неправильное подключение разъемов ATA (с переворотом или смещением) может блокировать системную плату — после включения она не будет подавать признаков жизни. Правильным подключением ситуация нормализуется.

ВНИМАНИЕ ----------

Ошибочно собранный питающий разъем может вывести из строя подключаемое устройство (когда контакты цепей +5 и +12 В перепутаны).

После внесения каких-либо изменений в дисковую подсистему (добавления/ удаления физических дисков, разделов, логических дисков) возможны изменения букв, связанных с логическими дисками (см. 9.6). При этом некоторые разделы (логические диски) могут оказаться невидимыми. Это может произойти по ряду причин:

- По умолчанию DOS резервирует под диски всего пять логических имен (A, B, C, D и E). Это связано с экономией памяти (каждый диск «съедает» свою порцию). Поскольку жесткие диски начинаются с идентификатора C, то при суммарном числе логических дисков (разделов на жестких дисках и CD- ROM) более трех последние оказываются невидимыми. Для преодоления этого ограничения в файле CONFIG.SYS указывают параметр LASTDRIVE = x, где x — последняя доступная буква.
- Для CD-ROM в Windows могут резервироваться специально выбранные буквы, причем в панели управления можно задать начальную и конечную буквы. Этот диапазон не должен перекрывать буквы, предназначенные для логических дисков винчестера.
- Раздел жесткого диска может быть перекрыт сетевым диском, назначенным на его букву.
- Windows 98 позволяет скрывать логические диски, но эта возможность устанавливается по желанию пользователя (например, утилитой TweakUI).
- Сжатый логический диск может оказаться невидимым, после того как его несущий диск переместится на иную букву (DriveSpace их теряет). Чтобы их снова увидеть, требуется запустить Мастер компрессии или вручную подправить файл DBLSPACE.INI.

Пожалуй, самые сложные проблемы возникают с установкой больших (по меркам BIOS и загружаемой ОС) дисков. Им посвящен следующий раздел.

Относительно безобидны проблемы производительности — когда обмен данными с диском идет медленнее, чем рекламируется. Здесь следует помнить, что в системе «контроллер-кабель-диск» максимально возможные режимы обмена определяются самым слабым звеном. Высокие скорости обмена требуют качественных шлейфов и соблюдения правил топологии.

ВНИМАНИЕ

Интерфейс ATA в режимах PIO и DMA не предусматривает контроля достоверности передачи по самой шине. Возможные неконтролируемые ошибки в этом тракте могут быть причиной нарушения целостности данных, включая серьезные повреждения файловой системы. Контроль достоверности на шине ATA выполняется только в режиме UltraDMA. Шина SCSI имеет контрольный бит четности, но его использованием часто пренебрегают, запрещая контроль при конфигурировании устройств (а зря!).

Для дисков ATA в CMOS Setup можно задавать некоторые параметры, влияющие на производительность:

- IDE PIO (Auto, 0-4) — режим передачи IDE (PIO Mode). Ограничивает максимальный режим, предлагаемый контроллером (устройство его может ограничить своими способностями). Автоматическое согласование режимов работает не со всеми устройствами, поэтому иногда приходится его явно ограничивать. Может задаваться отдельно для каждого канала или устройства.
- IDE DMA Transfer Mode (Disabled, Type B, Standard) — режим DMA при передаче данных по интерфейсу ATA.
- IDE UltraDMA (Enable/Disable) — разрешение режимов Ultra DMA.
- IDE Multiple Sector Mode или IDE Block Mode (Enable/Disable или число) — разрешение мультисекторного режима передачи (указание максимального количества секторов).
- IDE 32-bit Transfer (Enable/Disable) — разрешение 32-битного обращения к регистру данных IDE. Ускоряет обмен с дисками, но может быть источником проблем при некорректных драйверах.

Эти параметры определяют возможности, доступные контроллеру и BIOS при обращениях к дискам. Реальное использование этих возможностей зависит от применяемой ОС. В выводимой табличке BIOS может показать, например, режим обмена UDMA 6 (со скоростью 133 Мбайт/с). Однако это лишь потенциальные возможности — в MS-DOS режим DMA вообще не поддерживается, а другие ОС могут потребовать установки специальных драйверов (и, естественно, 80-проводного шлейфа). Без дополнительных усилий пользователя обмен с диском в режиме Ultra DMA поддерживается в Windows 98/XP/200x. Стандартные драйверы для Windows 95 и NT, OS/2 Warp 4.0 используют режимы PIO; Windows 95 OSR2 — режимы DMA, а для поддержки режима Ultra DMA требуется обновление драйверов.

Проблемы использования больших дисков

Преодоление барьера 528 Мбайт, вызванного несогласованностью формата вызова BIOS Int 13h и формата регистров дисков ATA, было только началом «бега с препятствиями», начавшегося в середине 90-х годов. Проблемы больших дисков порождаются интерфейсными ограничениями устройств и вызовов дискового сервиса BIOS, недочетами в реализации алгоритмов трансляции в BIOS, форматами хранения информации о дисках, способами определения парамет

ров диска и методами работы с дисками, используемыми ОС, а также принятой системой конфигурирования дисков.

Когда пользователь запускает процедуру автоматического определения типа диска ATA (раздел IDE Autodetection в CMOS Setup), BIOS для каждого диска запрашивает параметры (командой Identify Device) и предлагает пользователю один или несколько вариантов геометрии, различающихся способами трансляции (LBA, CHS, ECHS). Среди этих вариантов один считается (по мнению BIOS) предпочтительным; он будет принят автоматически, если пользователь установит тип Auto. Выбранная геометрия — число цилиндров, головок и секторов на трек — заносится в CMOS; эти параметры можно установить и вручную. Далее при инициализации диска BIOS пытается выполнить ATA-команду Initialize Device Parameters, заказывая указанное число головок и секторов на треке. Если попытка увенчалась успехом, диск с заданными параметрами воспринимается как существующий и становится полноправным участником дальнейших событий — загрузки ОС и определения логических дисков. Если диск отказывается принять данные параметры, то выдается сообщение «Hard Disk fail» и данный диск уже не воспринимается системой. Возможен вариант, когда BIOS для автоматического определения геометрии использует данные из таблицы разделов, если таковая уже имеется на диске (по границе любого раздела можно определить число секторов на треке и число головок). Это обеспечивает переносимость дисков с машины на машину — на новом месте BIOS будет работать по геометрии, которая была принята раньше. Но если таблица разделов повреждена, BIOS будет пытаться навязать диску некорректные параметры, которые могут быть и неприемлемы для диска. Результатом станут сообщение «Hard Disk Fail» и недоступность диска. Такой диск можно «оживить», задав параметры вручную и переписав (или просто обнулив) таблицу разделов.

При установке большого диска могут проявиться какие-нибудь неожиданные ограничения, как правило, связанные с BIOS компьютера. Самый безобидный вариант — BIOS неправильно определяет размер диска (он оказывается неожиданно малым). При этом с другого диска (дискеты, CD-ROM) загрузка может выполняться нормально, и тогда для преодоления очередного барьера можно предпринять программные меры. Правильные параметры можно попытаться установить вручную (выбрать тип User и ввести значения CHS), но BIOS может и не позволить установить нужное число цилиндров. Хуже, когда с появлением нового диска компьютер «зависает» на определении параметров диска во время теста POST (если в CMOS Setup выбран тип Auto), при входе в CMOS Setup или во время начальной загрузки. В этом случае следует в CMOS Setup задать (вручную) «безопасные» параметры CHS 1024 x 16 x 63 (504 или 528 Мбайт, смотря как считать) или даже выбрать вариант Type 1 (615 x 4 x 17), что позволит выполнить загрузку ОС (с другого устройства). Далее следует либо обновить BIOS, либо прибегнуть к фирменным спасательным средствам, которые предоставляются изготовителями дисков. Если предполагается обновление BIOS, то для выполнения этой процедуры можно отключить новый диск логически (выбрав для него вариант None) или даже физически, отключив от него интерфейсный кабель.

Самый радикальный способ преодоления барьеров — обновление BIOS. Если это сделать не удается, то используют дополнительные конфигурационные джамперы, изменяющие логическую организацию диска, а также загружаемые резидентные «заплатки» для BIOS. Ниже перечислены изменения, которые можно сделать с помощью дополнительных джамперов на дисках ATA:

- Ограничение объема диска на уровне ближайшего актуального барьера (как правило, за счет уменьшения сообщаемого числа цилиндров). При этом доступным оказывается не весь объем диска.
- Изменение сочетания параметров геометрии, сообщаемой по умолчанию (слова 1, 3 и 6 паспорта диска). Тогда BIOS воспользуется более благоприятным (для данной реализации) алгоритмом трансляции и обойдет барьер. Так, например, ряд моделей дисков имеют два варианта геометрии — с 16 и 15 головками.
- Сообщение числа цилиндров в слове 1 паспорта диска менее 4092 (для преодоления барьера 2,1 Гбайт), но при этом в словах 60-61 указывается настоящая (полная) емкость. Позволяет определить реальную емкость накопителя расширенному сервису Int 13h и ряду ОС.
- Конфигурирование одного диска в виде двух устройств половинной емкости, но при этом канал ATA он занимает полностью, как устройство 0 и устройство 1 одновременно. Этот вариант встречался на некоторых дисках объемом около 1 Гбайт (так обходили первый барьер).

Наличие и назначение дополнительных джамперов зависит от конкретной модели устройства, а их восприятие — от применяемой ОС, а также версии BIOS (системной или загружаемой «заплатки», см. далее). Чаще всего современные диски имеют один дополнительный джампер AC (Alternate Capacity), который заставляет диски объемом 2,1-33,8 Гбайт сообщать геометрию по умолчанию 4092 x 16 x 63 (2,1 Гбайт), а диски большего размера — сообщать объем 33,8 Гбайт в словах 60-61 (трехмерная геометрия для них 16 383 x 15 x 63 — 7,9 Гбайт). Если приходится устанавливать джампер для ограничения емкости в угоду BIOS, то для доступности полного объема диска в ОС, пользующихся информацией BIOS (DOS, Windows, OS/2), следует применять драйвер DDO (см. далее). В инструкциях к большим дискам даются рекомендации по их конфигурированию.

Ограничения BIOS и ATA

Для начала заметим, что диски ATA в IBM PC имеют шесть (!) различных описаний геометрии, иногда противоречащих друг другу. «Букет» этих параметров сообщает, например, утилита GLCHK (от Western Digital); подробно они описаны в [6]. Теперь перечислим барьеры емкости для дисков ATA:

- *BIOS* Int 13h (без расширений) имеет теоретический предел 8,5 Гбайт (1024 x 256 x63x512 = 8 455 716 864), так что операционные системы, работающие с дисками через старый сервис BIOS Int 13h (например, MS-DOS), уже не могут использовать современные диски в полном объеме. Расширен

ный сервис оперирует 64-битным линейным адресом сектора, так что о барьере 2^{64} x 512 байт в ближайшие десятилетия можно не вспоминать.

- *Спецификация ATA* имеет предел адресации 137 Гбайт: в режиме CHS 65 536 цилиндров, 16 головок, 255 секторов на трек дают максимальное количество 267 386 880 секторов, при размере 512 байт — 136 902 082 560 байт (136,9 Гбайт). Однако, согласно спецификациям ATA/ATAPI, в своем паспорте диск сообщает трехмерную (CHS) геометрию по умолчанию только в пределах 16 383 x 1 5 x 63 = 15 481 935 секторов — 7,9 Гбайт, даже если его реальная (и адресуемая) емкость больше. В режиме LBA используется 28-битный адрес сектора, что дает 2^{28} x 512 - 137 438 953 472 байт (137,4 Гбайт). В SATA и ATA/ATAPI-6 введено расширение адреса LBA до 48 бит, что обеспечивает адресацию до 144 Пбайт.
- *Барьер 528 Мбайт* порожден несовпадением ограничений BIOS и ATA по отдельным координатам CHS. Проблема возникла в 1993 году, в 1994-м была принята идея трансляции геометрии в BIOS.

Помимо этих основополагающих препятствий есть еще ряд частных:

- *Барьер 2,1 Гбайт* (1996 г.) проявился в тех машинах, где в BIOS экономили ячейки энергонезависимой памяти, предназначенной для хранения параметров диска. Там в байте с логическим номером головки два старших бита использовали для номера цилиндра, чем ограничили разрядность номера головки величиной 6 бит. Однако для того чтобы число физических цилиндров, начиная с 4096, можно было «втиснуть» в 10 бит логического интерфейса, его требуется разделить уже на 8 и на столько же умножить число головок. Однако 4-битное число головок ATA в пределах отведенных 6 бит можно умножить лишь на 4. Отсюда и выходит предел транслируемого (в режиме ECHS) объема диска 4095 x 16 x 63 x 512 = 2 113 413 120 байт. При загрузке с большего диска происходят зависания либо объем диска усекается (остается число цилиндров по модулю 4096). Так, диск на 2,5 Гбайт воспринимался как 429-мегабайтный. Диски большего размера часто выпускаются со специальным джампером, переключающим их в режим 4092 x 16 x 63. Радикально проблема решается обновлением BIOS.
- *Барьер 3,2 Гбайт* был порожден ошибкой трансляции в ряде реализаций Phoenix BIOS версий 4.03 и 4.04, исправленной в версии 4.05 и выше.
- *Барьер 4,2 Гбайт* (1997 г.) возник по вине операционной системы DOS, «не понимающей» числа головок 256 (очевидно, где-то для его представления используется один байт, и 256 превращается в 0). При трансляции ECHS (со сдвигом битов) число цилиндров делится на степень двойки, чтобы получилось не более 1023, а число головок умножается на то же число. В результате для дисков, сообщающих геометрию с 16 головками и 63 секторами, пределом и оказывается 4,2 Гбайт (8192 x 16 x 63 транслируется в 1024 x x 128 x 63). Больший объем потребует увеличения числа цилиндров, и следующим шагом становится роковое число головок 256. Выход из положения есть — вместо геометрии с 16 головками, предлагаемой по IDE Autodetecting в CMOS Setup, вручную установить геометрию с 15 головками. При этом

число цилиндров относительно сообщенного нужно увеличить в 16/15 = = 1,06667 раз и округлить до меньшего целого. Более новые версии BIOS (Phoenix) выполняют этот пересчет для устройств с 16 головками и числом цилиндров более 8192, после чего выполняют обычную трансляцию путем сдвигов. Трансляция через LBA не имеет данного барьера.

- *Барьер 7,9 Гбайт* в режиме ECHS возникает, когда предыдущий (4,2 Гбайт) обходят, уменьшая число головок до 15 (умножением на 16 оно превращается в 240). Тогда предел логической геометрии оказывается 1024 x 240 x 63 x x 512 = 7 927 234 560 байт. В режиме LBA этого барьера нет.
- *Барьер 8,4 Гбайт* (1998 год) обусловлен отказом от конфигураций с 256 логическими головками (в угоду DOS/Windows). Достигается только при трансляции LBA: 1024 x 255 x 63 x 512 = 8 422 686 720 байт — немногим меньше теоретического предела традиционного сервиса Int 13h.

  Для дисков объемом более 8,4 Гбайт понятие логической геометрии уже не представляет интереса, поскольку в рамки трехмерного интерфейса BIOS они не вписываются. На всякий случай приняли, что они сообщают геометрию 16 383 x 16 x 63, но работают с ними в полном объеме уже в режиме линейной адресации. Однако на этом барьеры не кончились.
- *Барьер 33,8 Гбайт* (1999 г.) появился оттого, что стандартное число головок 16 и 63 сектора на трек дает число цилиндров более 65 535, и теперь оно не помещается в отведенное для этих целей 16-разрядное слово в таблице параметров жесткого диска (HDPT) в BIOS. У Award BIOS 4.51 встреча с таким диском может вызвать зависание теста POST. На дисках большего размера джампер альтернативной конфигурации уже не используют (считается, что компьютер уже готов к таким объемам). Данные диски сообщают геометрию по умолчанию как 65 536 x 16 x 63, а работа ОС со всем объемом диска уже не требует устаревших описаний дисков в BIOS.

Преодоление ограничений BIOS

Если системная микросхема BIOS компьютера имеет один из вышеперечисленных барьеров, то следует, по возможности, ее обновить. Новые версии BIOS можно найти в Сети (на сайте производителя системной платы или в других местах). Замена BIOS в современных системных платах с флэш-памятью не составляет особого труда (правда, согласно инструкции к системной плате, может потребовать временного отключения защиты записи и стирания). Если же заменить BIOS не удается, то существует ряд идеологически сходных утилит, поставляемых производителями с новыми дисками. Их основой является Disk Manager фирмы ONTRACK Data International, Inc. Название Ontrack Disk Manager ассоциируется как со способом установки, так и с размещением программных средств. Самая главная часть ПО — динамический оверлейный драйвер (Dynamic Drive Overlay, DDO) — располагается не в файле, а на фиксированном месте нулевого трека диска. *Драйвер DDO*, называемый также *XBIOS* (расширение BIOS), решает три основные задачи:

- Автоматически идентифицирует подключенные устройства ATA. В старых версиях BIOS иногда эта функция просто отсутствовала, в новых могла «ло

маться» на вышеперечисленных барьерах. При использовании джампера альтернативной емкости диска, «обманывающего» системную микросхему BIOS, драйвер определяет реальную емкость диска по словам 60-61 его паспорта.

- Позволяет поддерживать дополнительный контроллер ATA (в старых версиях BIOS предусматривался только один).
- Корректно работает с дисками, транслируя геометрию для объемов свыше 528 Мбайт и обходя последующие барьеры.

Драйвер XBIOS (или его аналог EZ-BIOS), замещающий программный код сервиса Int 13h системной микросхемы BIOS, должен загружаться в память до загрузки ОС. Для этого загрузчик данного драйвера помещают на место стандартного загрузчика MBR (цилиндр 0, головка 0, сектор 1), модифицированный загрузчик MBR — в сектор 2, а код обработчика Int 13h — в секторы 3-17. При загрузке с жесткого диска система загружает в память первый сектор диска и передает управление на его начало, благодаря чему исполняется программа загрузки драйвера (DDO) с начальных секторов диска. После загрузки драйвера управление передается главному загрузчику, который определяет активный раздел и загружает его, уже пользуясь новым кодом драйвера Int 13h. Таким образом, «заплатка» на BIOS будет загружаться всякий раз в начале загрузки с жесткого диска.

Утилита Disk Manager в процессе установки позволяет настраивать режимы, в которых драйвер должен обращаться к устройствам ATA: режимы PIO или DMA, многосекторные передачи, кэширование чтения и записи, 32-разрядный доступ к регистру данных контроллера и т. п. Эти настройки отражаются в «теле» драйвера, записываемом в начальные секторы диска.

Если в системе установлено более одного диска, то драйвер помещается, естественно, только на первый диск (с которого выполняется загрузка), а обслуживать он может и все остальные. Если этот диск потребуется изъять, тот, который станет первым, должен быть преобразован: на него должен быть установлен драйвер Disk Manager.

При обычной загрузке с дискеты загрузчик «заплатки» исполняться не будет, поэтому доступ к жесткому диску окажется некорректным, что может привести к потере данных. Чтобы обеспечить возможность загрузки с дискеты, предусмотрено следующее: перед загрузкой ОС нужно вынуть дискету, что приведет к загрузке первого сектора жесткого диска и XBIOS. Когда в системе установлена утилита Disk Manager, до загрузки ОС *(после инициализации XBIOS) на* экране появляется голубой аншлаг с заставкой ONTRACK, а для загрузки с дискеты следует нажать клавишу пробела до и после установки загрузочной дискеты. На загрузочной дискете (ее готовит утилита Disk Manager) помимо ОС располагается файл драйвера, на него имеется ссылка из файла AUTOEXEC. BAT.

Драйверы данного типа совместимы с ОС DOS 5.0 и выше, Windows 9x, Windows NT 3.51 и 4.0 (без дополнительных установок джамперов), Windows 3.1x,

OS/2 Warp 3.0 и 4.0. Они *не совместимы с* Novell NetWare, Unix, Linux. Драйверы совместимы с компрессорами Stacker, DriveSpace, DoubleSpace.

Драйверы диска, загружаемые с него самого, конечно, являются не лучшим решением проблем больших дисков, хотя и обеспечивают временное решение. Применение драйверов данного типа доставляет дополнительные сложности и в условиях вирусных атак — «лечение» загрузочных вирусов может разрушить сам драйвер диска. Кроме того, некоторые антивирусные программы могут ошибочно принять драйвер за загрузочный вирус и «вылечить» его (насмерть!). Диагностические программы могут быть несовместимы с этими драйверами.

Если удается обновить BIOS (перепрограммировав флэш-память), то следует деинсталлировать EZ-Drive с помощью штатной утилиты, которая аккуратно вернет MBR и таблицу разделов на штатное место без потери данных на диске. Если данных не жалко, то можно «снести» EZ-Drive, запустив утилиту FDISK с ключом /MBR. После этого следует запустить FDISK повторно, сконфигурировать и отформатировать диск.

Трехмерная геометрия дисков SCSI

Для дисков SCSI (самих устройств) понятие трехмерной геометрии попросту отсутствует — при операциях обмена с ними пользуются линейным адресом (LBA). В дескрипторах команд SCSI обычно используется 32-битный адрес, что при 512-байтном секторе позволяет адресовать до 2 Тбайт данных (2^{32} секторов). Есть и формат дескрипторов с 64-битной адресацией. Поскольку программный интерфейс стандартного дискового сервиса BIOS оперирует трехмерной геометрией, ее приходится *эмулировать*. Расширенный сервис, опирающийся на ту же линейную адресацию (но уже 64-битную), никаких преобразований по пути к диску не требует.

Для дисков SCSI обработку прерывания Int 13h перехватывает BIOS контроллера SCSI; следовательно, геометрию должна «придумать» именно микросхема BIOS контроллера. Сам диск сообщает только свой объем — количество (M) и размер секторов (обычно 512 байт), который требуется «нарезать» на треки (назначить число секторов на трек S) и цилиндры (назначить число головок H и определить число цилиндров C - M/(H x S)). Привлекательно сочетание H = 64, S = 32 — тогда получается «круглое» значение объема цилиндра —

1 Мбайт (двоичных, то есть 2^{20} байт). Однако оно не годится для больших дисков, когда полученное число цилиндров превышает 1023. Здесь проявляются различия взглядов разработчиков SCSI BIOS на сочетания H и S (см. [6]). Некоторые версии BIOS считывают таблицу разделов с диска и из описателей разделов берут информацию о числе секторов и головок. Какой из вариантов представления геометрии будет использован в каждом конкретном случае, зависит от модели контроллера SCSI, версии его BIOS, установок конфигурирования контроллера и, наконец, от самого диска (есть ли на нем таблица разделов). Из этого следует возможная непереносимость дисков с информацией с машины на машину или их неработоспособность при смене контроллера. Если новый контроллер не пользуется информацией из таблицы разделов и его «воззрения» на геометрию не совпадают с геометрией, полученной во время предыдущего кон

фигурирования и форматирования дисков, то диск в DOS/Windows перестанет загружаться. Такому диску придется заново (с чистого листа!) создавать таблицу разделов — естественно, с потерей прежних данных. Чтобы не ломать голову над изучением проблем совместимости, часто для дисков SCSI при установке выполняют низкоуровневое форматирование, которое «сносит» и MBR с таблицей разделов, и всю информацию с диска. После этого геометрия определяется согласно логике SCSI BIOS. Целесообразность низкоуровневого форматирования для «зачистки» таблицы разделов сомнительна — достаточно обнулить нулевой логический блок любым средством редактирования дисков. В системе команд SCSI команда низкоуровневого форматирования является стандартной и обязательной (не то что в ATA). Распространенные рекомендации по применению этой процедуры, похоже, относятся к области «древней истории» дисков. Для «средневековых» дисков низкоуровневое форматирование может оказаться даже вредным, поскольку способно испортить заводские параметры оптимизации. Современный «умный» диск не даст себя испортить стандартными интерфейсными командами. Низкоуровневое форматирование дисков обычно можно инициировать из меню конфигурирования SCSI-контроллера, в которое можно попасть по специальному приглашению после завершения теста POST. Поскольку поддержка дисков SCSI реализована в модуле BIOS контроллера, соответствующими настройками или джамперами использование этого модуля ROM BIOS должно быть разрешено. Если в системе имеется несколько однотипных контроллеров, то достаточно разрешить ROM BIOS на одном из них. Если контроллер расположен на системной плате, SCSI BIOS входит в область системной микросхемы BIOS.

Из вышеизложенного следует, что если диск SCSI кажется меньшим, чем указано изготовителем, то проблема гнездится в SCSI BIOS. Возможно, ее удастся решить, изменив настройки в Setup контроллера SCSI.

Ограничения операционных систем

Ограничения операционных систем определяются тем, как они работают с диском. Если ОС использует для работы с диском традиционные функции BIOS Int 13h, то в лучшем случае она может работать с диском до 8,4 Гбайт, в худшем — с каким-либо барьером BIOS (см. выше). Если ОС работает через расширенные функции, то барьер пока необозрим (к этим ОС относятся Windows 95B (OSR2), Windows 98, Windows 200x и XP). Если ОС обращается к дискам только через BIOS, то и объем диска она выясняет через Int 13h(8) или Int 13h(48h) со всеми вытекающими последствиями. Если ОС определяет размер диска в обход BIOS, то интересен вопрос, как она это делает. Для дисков SCSI объем (количество секторов на всем устройстве) сообщается однозначно по соответствующей команде SCSI. Для дисков ATA ситуация сложнее: согласно спецификациям ATA/ATAPI, при объеме более 7,9 Гбайт в словах 1, 3 и 6 паспорта сообщается трехмерная геометрия 16 383 x 15 x 63 = 15 481 935 секторов (около 7,9 Гбайт). Истинный объем можно узнать только из слов 60-61. Подробнее о поведении различных ОС (и разных версий драйверов) см. в [6].

Конфигурирование, форматирование и обслуживание дисков

Для использования в качестве хранилища данных диск должен быть отформатирован. Различают два уровня форматирования дисков:

- низкоуровневое форматирование (LLF) — формирование треков (запись заголовков и заполнение области данных физических секторов) на всех рабочих поверхностях дисков;
- логическое форматирование (высокоуровневое) — для DOS/Windows это создание загрузочного сектора, FAT, корневого каталога и пометка в FAT дефектных (не прошедших верификацию) кластеров.

Для жестких дисков между этими уровнями вклинивается *конфигурирование диска* — разбивка на разделы, выбор активного раздела (только на первом физическом диске) и создание логических дисков в расширенном разделе. Низкоуровневое форматирование жестких дисков (кроме SCSI) пользователями в обычных условиях не выполняется.

ВНИМАНИЕ ----------

Выполняя конфигурирование и форматирование, а также используя утилиты восстановления, выполняющие низкоуровневое форматирование, рекомендуется отключать все остальные винчестеры, дабы избежать возможных ошибок.

Логическое форматирование, или просто *форматирование*, выполняет утилита FORMAT, которой обязательно нужно указать имя логического диска (A, B, C...). Напомним, что дискета вся является логическим диском; на винчестере логическими дисками являются разделы, помеченные кодами, «понимаемыми» данной операционной системой. В DOS/Windows 9x на жестком диске можно форматировать только первичный раздел или те подразделы расширенного раздела, для которых определены логические диски (разделы и логические диски определяются утилитой FDISK). Помимо утилиты FORMAT форматирование выполняют и некоторые другие утилиты, а в Windows форматирование диска можно инициировать из меню, раскрывающегося, например, при щелчке правой кнопкой мыши на значке диска.

При форматировании выполняется инициализация загрузочного сектора — в него помещается программа-загрузчик и в нем инициализируются поля, определяющие структуру диска. Таблица FAT и область корневого каталога обнуляются, после чего диск выглядит пустым. «Быстрое» форматирование на этом останавливается, но по полной программе следует еще проверить читаемость всех секторов. Для этого выполняется верификация (холостое чтение с проверкой на наличие ошибок), и кластеры с ошибочными секторами помечаются в FAT как дефектные. Хуже, когда дефектные секторы попадают в область FAT или загрузочного сектора. Диск с дефектом первого сектора нулевой головки нулевого цилиндра для использования непригоден.

После форматирования в области данных остаются данные старых файлов (и каталогов). Для дискет полное форматирование совмещается с низкоуровне

вым форматированием. После такого форматирования старые данные затираются заполнителем. Для жестких дисков ATA низкоуровневое форматирование в обычных условиях не выполняется. Для дисков SCSI утилита низкоуровневого форматирования обычно «зашита» в BIOS контроллера SCSI.

Конфигурирование жестких дисков — формирование таблиц разделов — выполняется утилитой FDISK. Она позволяет читать существующую таблицу разделов, добавлять или удалять разделы, создавать и удалять логические диски в расширенном разделе. Один из разделов (но не расширенный) может быть помечен как активный; правда, утилита позволяет устанавливать флаг активности только у первого физического диска. Если при помощи FDISK необходимо сделать активным раздел другого диска, его придется сделать первым (аппаратным переключением или конфигурированием через CMOS Setup).

ВНИМАНИЕ ----------

Любой жесткий диск должен иметь таблицу разделов!

Конфигурирование дисков из-за различий в способах преодоления вышеописанных барьеров имеет свои тонкости, подробно описанные в [6]. Не следует пытаться старой утилитой FDISK конфигурировать большие диски — им недоступен объем более 8,4 Гбайт. Возможны и конфузы с типами создаваемых разделов: например, не все ОС и утилиты «понимают» разделы с кодами 0B, 0C, 0E и 0F.

Форматирование и конфигурирование затрагивает не все доступные секторы диска. За сектором с главной и расширенными таблицами разделов всегда остается свободное место до конца логического трека (поскольку любой раздел должен начинаться с начала трека). Это место (около 32 Кбайт у современных дисков с 63 секторами на треке) может использоваться «втайне» от операционной системы:

- в него помещают загружаемые программные «заплатки» на BIOS (см. выше);
- это «гнездовье» загрузочных вирусов (они замещают программный код главного загрузчика своим, который до загрузки ОС загружает в память свои вредоносные процедуры);
- иногда его используют для привязки ПО к конкретной машине (на самом деле, к винчестеру).

Хотя для «зачистки» диска достаточно обнулить сектор с MBR (после этого его можно конфигурировать и форматировать с «чистого листа»), искушенные пользователи обнуляют весь нулевой трек нулевого цилиндра (если там нет ничего полезного). При этом сами разделы никоим образом не затрагиваются, и к ним можно получить доступ, восстановив MBR и таблицу разделов. Заменить главный загрузчик стандартным можно командой FDISK /MBR

Перед конфигурированием диска пользователь должен спланировать желаемое разбиение дисков, учитывая ограничения файловых систем применяемых ОС. Так, если требуется доступ к разделам (логическим дискам) из MS-DOS, то размер раздела не должен превышать 2,1 Гбайт, а файловой системой должна

быть FAT 16. Кроме того, такой раздел не должен выходить за пределы первых 8 Гбайт диска, доступных через традиционные сервисы BIOS.

Под *обслуживанием дисков* подразумевается периодическое выполнение их дефрагментации, например, утилитой Defrag. Для проверки и исправления ошибок на диске служит утилита ScanDisk, обнаруживающая и исправляющая ошибки в FAT, потерянные кластеры, пересекающиеся цепочки (кластеры, «принадлежащие» более чем одному файлу), ошибки в структуре каталогов, логические ошибки на сжатых дисках. Она также позволяет сканировать поверхность дисков, выявлять и помечать сбойные кластеры и, по возможности, перемещать данные из сбойных секторов в нормальные. В Windows 9x/XP/200x утилита автоматически запускается при установке (и переустановке) ОС, а также после аварийного завершения (прекращения) работы.

Для обслуживания сжатых дисков используется, например, утилита DrvSpace (может встречаться под именем DblSpace), которая позволяет сжимать диски, форматировать и «разжимать» сжатые диски, управлять монтированием томов сжатых дисков.

Вышеперечисленные утилиты работают с жесткими дисками только на уровне чтения-записи секторов, не забираясь в область низкоуровневого форматирования. При обнаружении сбойных секторов максимум, что они могут, — пометить в FAT сбойные кластеры. Когда число сбойных кластеров становится значительным, имеет смысл изъять из обращения дефектные секторы, а то и треки и даже целые поверхности. Это можно сделать только путем низкоуровневого форматирования. Для дисков SCSI имеется специальная команда, форматирующая весь диск. Интеллектуальный встроенный контроллер современного диска SCSI скроет все физически дефектные блоки, и диск вновь станет непрерывным массивом работоспособных секторов (но с полной потерей данных). Для дисков ATA низкоуровневое форматирование не стандартизовано, и каждый производитель использует собственные алгоритмы выполнения этой процедуры, причем они могут различаться от семейства к семейству. Если для низкоуровневого форматирования не требуется специального оборудования, то изготовители дисков поставляют к своим продуктам специальные утилиты, позволяющие проверять поверхности диска и скрывать дефектные области за счет резервных секторов или уменьшения доступного объема диска. Эти утилиты можно найти на сайтах производителей дисков (но не всех).

ВНИМАНИЕ ----------

«Низкоуровневые» утилиты пригодны лишь для работы с дисками, модели которых перечислены в аннотации к конкретной версии утилиты.

Основные причины отказов дисков

Современные диски сами по себе являются весьма надежными устройствами с развитыми средствами контроля достоверности хранения информации. В то же время различные аварии в подсистеме внешней памяти — явление, увы, нередкое. Чтобы уяснить причины аварий и предупредить возможные потери данных — а это самое ценное в компьютере! — напомним основные моменты.

Диск связан с программой, исполняемой процессором компьютера, через память, контроллер интерфейса и собственно интерфейс. Даже самая правильная программа (включая ОС, драйверы, приложения и утилиты) на самом надежном диске может порождать логические ошибки, приводящие к потере данных, если в этой цепочке есть ненадежно работающее звено. Процессор, если он не разогнан, как правило, работает правильно. Источником ошибок может быть *оперативная память* совместно с кэшем и схемами управления памятью. К сожалению, эти ошибки зачастую не выявляются тестовыми программами, но проявляются в виде неожиданных «вылетов» или «зависаний» программ. «Лечатся» неисправности проверкой правильности конфигурирования памяти, соответствия модулей требованиям системной платы, настройки временных диаграмм. Возможна «психологическая несовместимость» конкретных моделей модулей памяти с конкретными системными платами — скорее всего, здесь дело в геометрии проводников, влияющих на уровень помех. «Разогнанная» память — потенциальный источник ошибок. Достоверность хранения данных оперативной памятью в массовых ПК не проверяется (память с контролем четности отжила свой век). Для серьезных серверов, хранящих особенно дорогие данные, применяют память с ECC-контролем солидных производителей (например, Kingston). Помимо оперативной памяти следует обращать внимание и на кэш-память. У современных процессоров кэш, как правило, имеет ECC- контроль, который зачастую отключают через CMOS Setup, а зря! Причиной сбоев памяти может стать и некачественный блок питания, особенно если у него отсутствует сетевой фильтр (бывает и так в погоне за дешевизной).

Следующее после памяти слабое звено этой цепи — интерфейс устройства. В шине ATA вплоть до появления режимов UltraDMA не было никакого контроля достоверности передачи информации по самой шине. В принципе, шина достаточно надежна (иначе работать с ней было бы невозможно), но только при соблюдении ограничений на длину и нагрузку кабелей. Во многих двухканальных контроллерах шины данных обоих каналов электрически соединены друг с другом (это дешевле!), но если на каждом канале использовать кабель максимально разрешенной длины, то в сумме получается перебор (по емкости проводников). Для скоростных режимов UltraDMA предназначен 80-проводный шлейф, но систему иногда можно заставить работать и на обычном шлейфе. К счастью, здесь имеется CRC-контроль (правда, не все драйверы корректно отрабатывают его ошибки).

В шине SCSI применяется контроль четности, хотя его иногда отключают — «чтоб лучше работало!». В самых «скорострельных» вариантах SCSI стали применять уже более сложные методы контроля достоверности передачи. Проблемы на шине чаще всего вызываются неправильным использованием терминаторов (см. 20.4).

Для повышения надежности хранения иногда прибегают к зеркальному отражению (дублированию) дисков. Однако от логических ошибок, порожденных неисправной памятью или программными ошибками (сбоями), эти меры не спасают.

Самая распространенная причина появления логических ошибок дисков — внезапное отключение питания, нажатие кнопки Reset или неуместное применение «комбинации из трех пальцев» (Ctrl+Alt+Del) во время работы ОС и приложений. Это, как правило, приводит к образованию потерянных или пересекающихся кластеров и, следовательно, к повреждению пользовательских данных. Утилиты типа ScanDisk лечат эти недуги довольно быстро и эффективно (но с возможной потерей данных).

ВНИМАНИЕ ----------

Питание ПК можно выключать лишь по окончании процедуры завершения работы ОС (shutdown) — после появления разрешающей надписи (и затихания винчестера). Еще лучше, когда ОС сама выключает питание компьютера (для настольных ПК это возможно только в конструктиве ATX).

Распространенные ошибки пользователей — завершение работы приложения без сохранения нужных данных, ошибочные удаления и переименования файлов и прочие небрежности.

Самые неприятные последствия, как правило, вызывает действие особо «злобных» вирусов. Они могут разрушать данные на всех уровнях: на уровне файлов, системных областей логических дисков, таблицы разделов. Предотвратить это можно только соблюдением правил «гигиены» и профилактики с помощью свежих антивирусных средств. Обновление этих средств (как и сами вирусы) можно получить по Сети, которая становится доступной все большему числу пользователей.

Однако мелкие или крупные аварии дисковой подсистемы все же могут случаться и случаются. Чтобы потом «не было мучительно больно», *регулярно сохраняйте важные данные.*

ВНИМАНИЕ ----------

Для хранения копий важных данных (резервных копий и архивов) используйте отчуждаемые носители (или съемные устройства). Только в отчужденном состоянии (носитель снят и спрятан в сейф) данные не могут быть удалены или перезаписаны по неосторожности или под действием злоумышленника (или вируса).

Отчуждаемыми носителями могут быть дискеты, оптические диски CD-R или CD-RW, магнитооптические диски, Zip, Jaz, кассеты стримеров и другие, подходящие по объему, быстродействию и стоимости.

ВНИМАНИЕ ----------

Особо важные данные следует хранить по крайней мере на двух разных носителях (съемных устройствах). Это нужно для того, чтобы в случае несчастья во время записи текущей копии вдали от компьютера имелся носитель с предыдущей более или менее свежей версией данных. В руководствах к программам резервного копирования обычно приводятся описания различных дисциплин смены носителей, обеспечивающих их круговорот в соответствии с требованиями к надежности и режиму эксплуатации носителей.

ГЛАВА 10

Видеосистема

Персональный компьютер смог стать привлекательным вычислительным средством благодаря способности интерактивного взаимодействия с пользователем. Интерактивность подразумевает наличие устройств оперативного ввода и вывода информации. Основной поток выходной информации — визуальный, причем информация представляется как в текстовом, так и в графическом виде. Визуальная информация может выводиться на экран или на принтер с получением ее «твердой копии» (hard copy) — на бумаге, пленке и т. п. Для интерактивного режима вывод на бумагу малоинтересен, хотя в далекой истории компьютеров интерактивный режим впервые был реализован именно на телетайпе (грубо говоря, электрической пишущей машинке, подключенной к компьютеру). Данная глава посвящена активным средствам вывода визуальной информации — видеосистеме PC. Активность подразумевает возможность изменения изображения без смены носителя. Пассивные средства вывода визуальной информации (принтеры, плоттеры) рассматриваются в главе И.

В первые годы существования PC его видеосистемой называли средства вывода текстовой или графической информации на какой-либо экран. В качестве оконечного устройства чаще всего использовали (и продолжают использовать) мониторы с электронно-лучевыми трубками. Адаптеры, позволяющие подключать монитор к шине компьютера, называли видеоадаптерами и подразделяли на алфавитно-цифровые и графические. Последние, естественно, помимо графической позволяли выводить и текстовую информацию. Вся выводимая информация формировалась в результате действия и под управлением системных и прикладных программ.

По мере «взросления» на PC стали взваливать и казавшуюся ранее неподъемной ношу воспроизведения и обработки движущихся телевизионных изображений — так называемого «живого видео». Так назрела необходимость корректировки терминологии. *Видеосистема* современного компьютера состоит из обязательной графической подсистемы (формирующей изображение программно) и дополнительной подсистемы обработки видеоизображений. Обе эти составляющие части обычно используют общий монитор, а соответствующие аппаратные средства системного блока могут располагаться на раздельных картах различного функционального назначения или объединяться на одном комбинированном адаптере, который можно назвать *адаптером дисплея* (display adapter).

Графический адаптер служит для программного формирования графических и текстовых изображений и является промежуточным элементом между монитором и шиной компьютера. Изображение строится по программе, исполняемой

центральным процессором, которому могут помогать графические акселераторы и сопроцессоры. В BIOS реализована поддержка функций формирования текстовых и графических изображений, по старинке называемая *видеосервисом BIOS* (Int 10h). Существует ряд классов адаптеров (MDA, CGA, EGA, VGA, SVGA...), которые будут рассмотрены далее. В *монитор* адаптер посылает сигналы управления яркостью лучей базисных цветов *RGB* (Red, Green, Blue — красный, зеленый и синий) и синхросигналы строчной и кадровой разверток. Помимо этих сигналов, относящихся только к формированию изображения, интерфейс с монитором может содержать и сигналы обмена конфигурационной информацией между монитором и компьютером. Так, PnP-мониторы при соответствующей поддержке со стороны адаптера способны сообщать системе свои параметры (модель и параметры синхронизации).

Средства работы с *видеоизображениями*, передаваемыми в стандартах PAL, SECAM или NTSC, относятся уже к мультимедийному оборудованию. От программно-управляемых графических средств они отличаются тем, что оперируют «живым» изображением, поступающим в компьютер извне (с видеокамеры, TV-тюнера) или воспроизводимым с какого-либо носителя информации (например, привода CD/DVD).

Все компоненты дисплейного адаптера могут размещаться на одной карте расширения, а зачастую они устанавливаются прямо на системной плате и используют преимущества локального подключения. Мультимедийные средства могут размещаться на отдельных картах, связанных с графическим адаптером специальным интерфейсом, а могут выполняться в виде небольшого «дочернего» модуля, устанавливаемого на графическую карту.

Стандартизацией в области видеосистем занимается международная организация *VESA* (Video Electronic Standard Association — ассоциация по стандартизации в области видеоэлектроники), доступная по адресу http://www.vesa.org. Благодаря ее усилиям обеспечивается совместимость как на уровне аппаратных средств, так и на уровне программного обеспечения.

С самого появления персонального компьютера его видеосистему стремились строить для максимального приближения к идеалу *WYSIWYG* (What You See Is What You Get) — «что видишь, то и имеешь» (или наоборот). Поскольку под словом «имеешь» чаще всего подразумевается некоторая отпечатанная продукция, то имеется и идеал *WYSIWYP* (What You See Is What You Print) — «что видишь на мониторе, то и будет напечатано». Стремление к этим идеалам, подкрепленное техническим прогрессом, приводит к неуклонному росту качественных показателей видеосистемы и проникновению компьютерных технологий в такие области, как, например, хранение копий (точных!) произведений искусства в электронном виде.

10.1. Принципы вывода изображений

Видеосистема PC ориентирована на растровый метод вывода изображения. *Растровый метод* подразумевает, что некий рисующий инструмент, способный оставлять видимый след, сканирует всю поверхность, на которую выводится

изображение. Траектория движения инструмента постоянна и не зависит от выводимого изображения, но инструмент может рисовать, а может и не рисовать отдельные точки траектории. Видимое изображение образуется оставляемыми им точками. В случае видеомонитора инструментом является модулированный луч (или три луча базисных цветов), построчно сканирующий экран и вызывающий свечение люминофора, нанесенного на внутреннюю поверхность экрана. Каждая строка растра разбивается на некоторое количество точек — *пикселов* (pixel — сокращение от picture element — элемент изображения), засветкой каждой из которых по отдельности может управлять устройство, формирующее изображение (например, графическая карта). Видеомонитор является растровым устройством вывода динамически изменяемых изображений. Его луч сканирует экран с частотой, которая не должна позволять глазу видеть мерцание изображения. Матричные дисплеи, применяемые в блокнотных ПК, также относятся к растровым устройствам. Растровыми устройствами вывода статических изображений являются принтеры, в которых сканирование листа производится однократно (хотя возможны и многократные проходы).

Альтернатива растровым устройствам — *векторные устройства вывода изображений.* В этих устройствах инструмент прорисовывает только изображаемые фигуры и его траектория движения определяется выводимым изображением. Изображение состоит из графических примитивов, которыми могут быть отрезки прямых — векторы (откуда и название метода вывода), дуги, окружности. К векторным устройствам вывода статических изображений относятся перьевые плоттеры. Существовали (а может, где-то и сейчас используются) и векторные мониторы, однако ввиду сложности построения системы управления лучом, обеспечивающей быстрое и точное движение луча по сложной траектории, эта линия угасла.

Рассмотрим растровую систему вывода изображений, подразумевая в качестве оконечного устройства монитор с электронно-лучевой трубкой — ЭЛТ (CRT, Cathode Ray Terminal, дословно — монитор на катодно-лучевой трубке). *Сканирование* экрана модулированным лучом обеспечивается генераторами *горизонтальной* и *вертикальной* разверток монитора. Луч оставляет след только во время прямого хода по строке (слева направо). Строка разбивается на некоторое количество точек, каждая из которых может иметь независимое от других состояние (яркость и цвет). Для ЭЛТ-монитора это разбиение условно, в матричных дисплеях пикселы являются физическими. На обратном ходе по строке луч принудительно гасится. Следующая строка прорисовывается параллельно предыдущей, но с некоторым вертикальным смещением (вниз), и так происходит сканирование до окончания кадра — достижения правого нижнего угла экрана. Во время обратного хода луча по вертикали, за время которого генератор горизонтальной развертки успеет сделать несколько строчных циклов, луч также принудительно гасится. В следующем кадре сканирование может производиться по-разному. В системах с *прогрессивной* (progressive), или *не чересстрочной* (Non-Interlaced, NI), разверткой луч идет по тем же самым строкам (рис. 10.1, *а)*. В системах с *чересстрочной* разверткой (Interlaced, IL) луч идет по строкам, смещенным по вертикали на половину шага строки (рис. 10.1, *б)*. Таким образом, всю поверхность экрана луч проходит за два цикла кадровой

развертки, называемых полукадрами. Чересстрочная развертка позволяет почти вдвое снизить частоту горизонтальной (строчной) развертки, а следовательно, и темп вывода точек изображения. Выгода от этого снижения будет понятна позже, а пока поясним, как определяются частоты развертки.

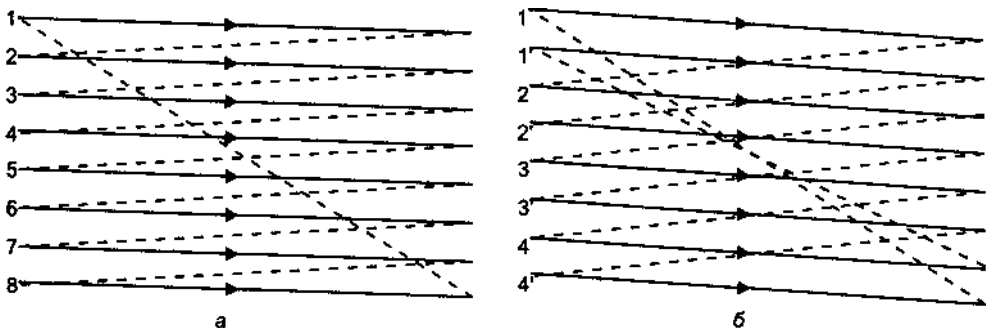


Рис. 10.1. Сканирование экрана: а — прогрессивная развертка, *б* — чересстрочная развертка

Как известно, глаз является инерционным органом зрения — он воспринимает изменение яркости или освещенности только до какой-то определенной частоты. Существует понятие *критической частоты слияния мельканий* (КЧСМ), которую измеряют так: человек смотрит неподвижно на некоторый безынерционный источник света (например, светодиод), который вспыхивает и гаснет с плавно повышаемой частотой. Сначала человек воспринимает вспышки по отдельности, с повышением частоты он видит уже только мерцание, а начиная с некоторой частоты мерцания для него сливаются в ровный свет. Эта частота и называется критической, и у разных людей она может находиться в пределах примерно 40-60 Гц. Неподвижность взгляда и источника в нашем опыте оговаривалась, поскольку при движении мелькающего объекта человек будет его воспринимать как трассу прерывистых светящихся точек (стробоскопический эффект). Наблюдение мерцающих объектов раздражает и утомляет зрительную систему, поэтому частота кадров (прорисовки экрана) должна быть по крайней мере не ниже значения КЧСМ. Таким образом, мы получили ориентировочное значение минимальной частоты кадров, равное 50 Гц (эта компромиссная частота применяется во многих телевизионных системах). Теперь посмотрим, что из этого следует. Вполне очевидно, что для получения качественного изображения экран должен иметь как можно больше точек матрицы разложения — то есть строк в кадре и точек на строке. Возьмем все еще популярный режим 800 x 600 (600 строк по 800 точек). За один период прогрессивной кадровой развертки луч должен успеть прочертить 600 видимых строк, да еще некоторое количество невидимых строк (примерно 50) он прочертит на обратном ходу по кадру. Получается, что частота строк должна составить 50 Гц x (600 + 50) = 32,5 КГц (вроде и не так уж много). Этой частоте соответствует период около 30 мкс (1/32,5), из которого на прямой ход по строке остается около 25 мкс. За это время необходимо вывести 800 точек строки, так что на каждую точку отводится 25/800 = 0,03 мкс. Это соответствует *частоте вывода точек* (DotClock) 30 МГц, а для электронных схем такая частота считается высокой. Поскольку

соседние точки выводимого изображения в принципе друг с другом не связаны, то *полоса частот* сигнала, модулирующего интенсивность луча, должна быть несколько выше этого значения (примерно на 25 %). Такую широкую полосу пропускания должен обеспечивать весь видеотракт: видеоусилители модуляторов лучей, сигнальные линии интерфейсного кабеля, и, наконец (вернее, сначала), такой широкополосный сигнал должен сформировать графический адаптер. На всех этих стадиях высокие частоты порождают технические проблемы. Если реальная полоса пропускания в этом тракте окажется уже, четкого изображения получить не удастся — переходы будут размыты. Если же частотная характеристика тракта в требуемом диапазоне будет неравномерной, появятся специфические искажения цветов или яркости около границ отображаемых объектов (справа от граничной линии). Понятно, что с технической точки зрения есть стимулы снижать требуемую верхнюю границу полосы частот видео- тракта. При чересстрочной развертке за каждый полукадр сканируется только половина строк разложения (четные в одном полукадре и нечетные в другом), следовательно, строчная частота уменьшается, а длительность прохода видимой части строки увеличивается примерно вдвое. Таким образом, при заданных условиях (разрешении экрана и ограничении минимальной кадровой частоты) чересстрочная развертка позволяет снизить требуемую полосу пропускания вдвое.

Как видно из приведенных выкладок, частотные параметры видеосистемы определяются, исходя из желаемой частоты кадров, разрешения экрана и режима развертки. Заботясь о зрении пользователя, частоту кадров стремятся повышать. При низкой частоте экран начинает мерцать, что особо заметно на больших белых полях изображения (в полном смысле слова яркий тому пример — белый фон в приложениях Windows). Разрешение экрана стремятся увеличить — чем оно выше, тем больше информации можно уместить на экране. Поскольку размер экрана постоянно увеличивается — монитор на 17-19" является уже нормой для многих видов деятельности, — потребность в разрешении, скажем, 1600 x 1200 вполне реальна. Но для этого, как явствует из сказанного ранее, уже требуется полоса 120 МГц (а кадровая частота 50 Гц — это отнюдь не идеал). В популярном для 17-дюймовых мониторов режиме 1280 x 1024 при частоте 85 Гц частота пикселов составляет 152 МГц (частота строк — 91,1 кГц). Применение чересстрочной развертки годится лишь как вынужденная мера, поскольку имеет свои специфические неприятные «видеоэффекты»: если выводится тонкая (в одну строку точек) горизонтальная линия, она заметно мерцает. Это и понятно: ведь прорисовывается она только в одном из полукадров - следовательно, с половинной кадровой частотой. Если изображение потолще (один и тот же элемент имеет точки в соседних строках), его мерцание оказывается почти незаметным.

Итак, вожделенные цели ясны: частота кадров — выше, разрешение (по вертикали и горизонтали) — больше, развертка — не чересстрочная (N1). Забегая немного вперед, заметим, что чем выше частота развертки, тем ниже производительность графической системы при построении изображений. С точки зрения пользователя привлекательность чересстрочного режима развертки заключается в цене устройств — поскольку для прогрессивной развертки требуется более высокое качество компонентов всего видеотракта, по

строчная развертка с высокой частотой кадров в режимах высокого разрешения может оказаться дорогим удовольствием. Но для профессиональной работы с текстом, графического дизайна и других видов деятельности ухудшение зрения обойдется дороже. Чересстрочная развертка широко применяется в телевидении: видеосигнал там приходится «пропихивать» через радиоканал, с шириной полосы которого всегда имеются проблемы. В современных мониторах и графических адаптерах, применяемых в PC, используются оба режима развертки с различными значениями частоты кадров. Естественно, что работать они должны в согласованных режимах.

Рассмотрев работу оконечного устройства (монитора), обсудим способы формирования изображения в графическом адаптере. Итак, у нас имеется матрица точек экрана, образованная горизонтальными строками растра (номер строки — вертикальная координата матрицы) и точками разложения строки (номер точки в строке — горизонтальная координата матрицы). Эта матрица сканируется построчным или чересстрочным образом, и во время прямого хода луча по видимым строкам графический адаптер должен формировать сигналы управления яркостью базисных цветов монитора (или одного сигнала яркости в монохромном варианте). За это время последовательно (и синхронно с ходом луча) должна выводиться информация о яркости и цвете всех точек данной строки. Синхронизация обеспечивается формированием горизонтальных и вертикальных синхроимпульсов. Таким образом, графический адаптер является задающим устройством, а монитор со своими генераторами разверток должен вписаться в заданные параметры синхронизации.

Существует два основных режима вывода информации — графический и символьный (текстовый). Первые дисплейные адаптеры из-за технических ограничений на доступный объем памяти адаптера работали в символьном режиме. Современные адаптеры в основном работают в графическом режиме, текстовый режим используется только до загрузки ОС.

Графический режим

В графическом режиме имеется возможность индивидуального управления свечением каждой точки экрана монитора независимо от состояния остальных. Этот режим обозначают как *Gr* (Graphics) или *APA* (All Points Addressable — «все точки адресуемы»). В графическом режиме каждой точке экрана — пикселу — соответствует ячейка специальной памяти, которая сканируется схемами адаптера синхронно с движением луча монитора. Точнее было бы сказать наоборот — физически движение луча вторично, так как монитор можно и не подключать, а графический адаптер все равно будет сканировать память, но логически вся конструкция строится исходя именно из поведения монитора. Эта постоянно циклически сканируемая (с кадровой частотой) память называется *видеопамятью* (video memory), или VRAM (Video RAM). Последнее сокращение можно спутать с названием специализированных микросхем динамической памяти, оптимизированной именно под данное применение. Процесс постоянного сканирования видеопамяти называется *регенерацией изображения*, и, к счастью, микросхемам динамической памяти, применяемой в этом узле, оказы

вается достаточно сканирования для регенерации информации. Для программно-управляемого построения изображений к видеопамяти также должен обеспечиваться доступ со стороны системной магистрали компьютера, причем как по записи, так и по чтению. Количество битов видеопамяти, отводимое на каждый пиксел, определяет возможное число состояний пиксела — цветов, градаций яркости или иных атрибутов (например, мерцания). Так, при одном бите на пиксел возможны лишь два состояния — светится или не светится. Два бита на пиксел доставляли немало удовольствия любителям цветных игр даже на адаптерах CGA — можно было иметь одновременно четыре цвета на экране. Четыре бита на пиксел (16 цветов), обеспечиваемые адаптером EGA, были достаточны для многих графических приложений, например графики в системах автоматического проектирования (САПР). Пределом мечтаний в свое время было 256 цветов (8 бит на пиксел) адаптера VGA — цветная фотография розы из комплекта графического редактора Paintbrush на экране монитора казалась великолепной. Сейчас остановились на режимах *High Color* (15 бит — 32 768 цветов; или 16 бит — 65 536 цветов) и *True Color* — «истинный цвет» (24 бита — 16,7 миллиона цветов), реализуемых современными адаптерами и мониторами SVGA.

Логически видеопамять может быть организована по-разному, в зависимости от количества битов на пиксел. Здесь мы опустим описание устаревших режимов с 1-4 битами на точку (адаптеры MDA, CGA и EGA, см. [1]) и остановимся на режимах VGA (8 бит), High Color и True Color (16 и 24 бита цвета). В этих режимах используется *линейная организация памяти*: последовательность байтов экранного буфера (область видеопамяти, отображаемая на экране) представляет последовательность пикселов экрана. За каждый пиксел отвечают 1-4 смежных байта. В режимах High Color и True Color содержимое этих байтов непосредственно задает цвет через двоичные коды уровней интенсивности базисных цветов RGB. Форматы байтов (младший бит — справа) выглядят следующим образом:

- 15 бит/пиксел: URRR RRGG GGGB BBBB (5-5-5);
- 16 бит/пиксел: RRRR RGGG GGGB BBBB (5-6-5), для зеленого цвета добавили лишний бит, поскольку чувствительность глаза к этому цвету выше;
- 24 бйта/пиксел: RRRR RRRR GGGG GGGG BBBB BBBB (8-8-8);
- 32 бита/пиксел: UUUU UUUU RRRR RRRR GGGG GGGG BBBB BBBB (8-8-8).

Здесь U) обозначает биты, не используемые для формирования цвета, а в цепочках битов базисных цветов (R, G, B) младший бит расположен справа.

32 бита на пиксел выгодно отводить с точки зрения повышения производительности обмена при попиксельном обращении, хотя при этом увеличивается объем видеопамяти. При 32-битном кодировании информация о пикселе передается по шине PCI/AGP быстрее всего — требуется всего одна фаза данных. При 24-битном кодировании 3/4 всех пикселов требуют для обращения двух фаз данных и только 1/4 — одной фазы. 16-битный формат тоже быстр в обращении. 8-, 16- и 32-битный форматы удобны и в плане программирования — каждый пиксел отображается байтом, словом или двойным словом, которые непо

средственно адресуются командами процессора. Для 24-битного режима пересчеты и сдвиги при адресации отнимают дополнительное время.

В режиме VGA каждый байт экранного буфера хранит 8-битный *код* цвета, определяющий номер цвета (индекс) в палитре — таблице цветов. В монитор передается информация, описывающая уровень каждого из трех базисных цветов. Если принять разрядность представления уровня каждого цвета 8 бит, то

3 x 8 = 24 бита позволяют кодировать $2^{24} \approx 16$ ООО ООО цветов. Однако в режиме VGA из них одновременно доступны лишь 256 (предел индексации 8-битным числом).

Объем видеопамяти (в битах), требуемый для хранения образа экрана, определяется как произведение количества пикселов в строке на количество строк и на количество битов на пиксел. Так, для режима 720 x 350 HGC с одним битом на точку он составляет 252 ООО бит, или около 31 Кбайт, а для режима 1280 x x 1024 True Color (32 бита) — около 5 Мбайт. Если физический объем видеопамяти превышает необходимый для отображения матрицы всего экрана, видеопамять можно разбить на страницы. *Страница* — это область видеопамяти, в которой умещается образ целого экрана. При многостраничной организации видеопамяти только одна из них может быть активной — отображаемой на экран.

Формирование битовой карты изображения в видеопамяти графического адаптера производится под управлением программы, исполняемой центральным процессором. Сама по себе задача формирования процессору вполне по силам, но при ее решении требуется пересылка большого объема информации в видеопамять, а для многих построений — еще и чтение видеопамяти процессором. Видеопамять большую часть времени занята выдачей информации схемам регенерации изображения в довольно напряженном темпе. От этого процесса она свободна только во время обратного хода луча по строке и кадру, но это — меньшая часть времени. Если обращение к активной странице видеопамяти со стороны процессора блокирует выдачу информации на монитор, на экране появляется штрих от несчитанной информации пикселов. В старых адаптерах частое обращение вызывало неприятный «снег» на экране. Дожидаться обратного хода по строке или кадру накладно: строчный период коротких (несколько микросекунд) интервалов обратного хода имеет порядок 10-25 мкс, а кадровый период длинного (миллисекунды) обратного хода — порядок 10-20 мс, в то время как цикл обращения процессора к обычной памяти не превышает сотен (у современных компьютеров — десятков) наносекунд. Так что канал связи процессора с видеопамятью представляет собой узкое «горлышко», через которое пытаются протолкнуть немалый поток данных, причем чем более высокое разрешение экрана и чем больше цветов (битов на пиксел), тем этот поток должен быть интенсивнее. Конечно, при выводе статической картинки это вроде и не страшно, но «оживить» изображение оказывается проблематично. Выходов из этого затруднения имеется несколько. Во-первых, повышают быстродействие видеопамяти. Во-вторых, расширяют разрядность шин графического адаптера, причем как внутренней (шины видеопамяти), так и интерфейсной, и применяют высокопроизводительные шины (раньше локальную шину VLB,

теперь PCI, порт AGP и PCI Express). Расширение разрядности позволяет за один цикл обращения передать больше битов данных — повысить производительность. Однако если у адаптера, к примеру VGA, разрядность интерфейсной шины составляет 16 бит, а установлен минимальный объем памяти, при котором используется только 8 бит, то эффективная разрядность интерфейса окажется всего 8 бит. То же относится и к адаптерам с 32-разрядной шиной. Этим объясняется не совсем очевидный факт, что производительность графического адаптера зависит от объема установленной видеопамяти. В-третьих, повысить скорость видеопостроений можно кэшированием видеопамяти или затенением видеопамяти, что, по сути, почти одно и то же. В этом случае при записи в область видеопамяти данные записываются как в видеопамять, так и в ОЗУ (или даже в кэш), поэтому при считывании из этой области обращение идет только к быстродействующему ОЗУ. И, в-четвертых, можно принципиально сократить объем информации, передаваемой графическому адаптеру, но для этого графический адаптер должен быть наделен «интеллектом». В современных компьютерах используются все эти решения.

Текстовый режим

В *символьном*, или *текстовому* режиме формирование изображения происходит несколько иначе. Если в графическом режиме (APA) каждой точке экрана соответствует своя ячейка видеопамяти, то в текстовом режиме ячейка видеопамяти хранит информацию о *символе*, занимающем на экране знакоместо определенного формата. *Знакоместо* представляет собой матрицу точек, в которой может быть отображен один из символов определенного набора. Здесь умышленно применяется слово «точка», а не «пиксел», поскольку пиксел является сознательно используемым элементом изображения, в то время как точки разложения символа в общем случае программиста не интересуют. В ячейке видеопамяти хранятся *код символа*, определяющий его индекс в таблице символов, и *атрибуты символа*, определяющие способ его отображения. К атрибутам относятся цвет фона, цвет символа, инверсия, мигание и подчеркивание символа. Поскольку изначально в дисплеях использовали только алфавитно-цифровые символы, такой режим работы иногда сокращенно называют *AN* (Alpha-Numerical — алфавитно-цифровой), но чаще — *TXT* (text — текстовый), что корректнее: символы псевдографики, которые широко применяются для оформления текстовой информации, к алфавитно-цифровым не отнесешь.

В текстовом режиме экран организуется в виде матрицы знакомест, образованной горизонтальными линиями (Line, LIN) и вертикальными колонками (Column, COL). Этой матрице соответствует аналогичным образом организованная видеопамять. Адаптер, работающий в текстовом режиме, имеет дополнительный блок — знакогенератор. Во время сканирования экрана выборка данных из очередной ячейки видеопамяти происходит при подходе к соответствующему знакоместу (рис. 10.2), причем одна и та же ячейка видеопамяти выбирается при проходе по всем строкам растра, образующим линию знакомест. Считанные данные попадают в знакогенератор, который вырабатывает построчную развертку соответствующего символа — его изображение на экране. *Знакогене*

ратор представляет собой запоминающее устройство — ОЗУ или ПЗУ. На его старшие адресные входы поступает код текущего символа из видеопамяти, а на младшие — номер текущей строки в отображаемой линии знакомест. Выходные данные содержат побитную развертку текущей строки разложения символа (в графическом режиме эти данные поступали из видеопамяти). Самый «скромный» знакогенератор имеет формат знакоместа 8x8 точек, причем для алфавитно-цифровых символов туда же входят и межсимвольные зазоры, необходимые для читаемости текста. Лучше всего читаются матрицы 9 x 1 4 и 9 x 1 6 символов. В адаптерах VGA память знакогенератора программно доступна, русификация (или иная локализация) адаптера выполняется программными средствами.

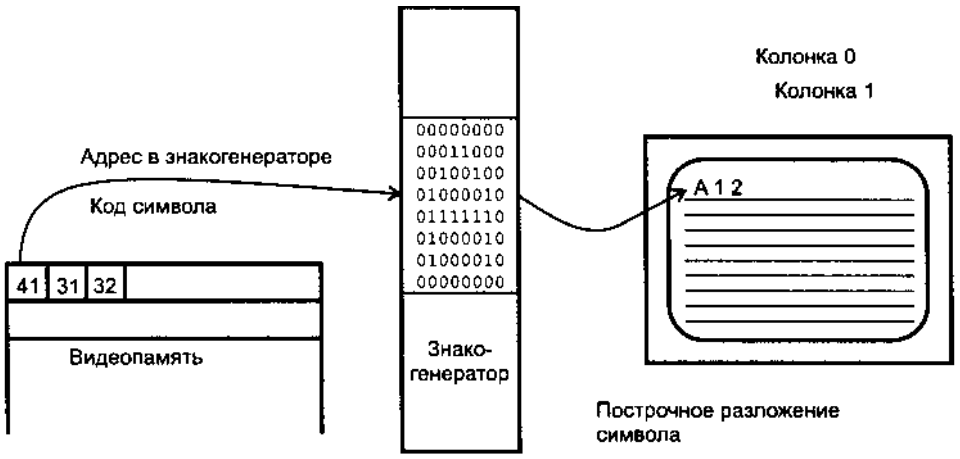


Рис. 10.2. Формирование изображения в текстовом режиме

Как уже говорилось, каждому знакоместу в видеопамяти, кроме кода символа, соответствует еще и *поле атрибутов*, обычно имеющее размер 1 байт. Этого вполне достаточно, чтобы задать цвет и интенсивность воспроизведения символа и его фона. Текстовый адаптер также имеет аппаратные средства управления *курсором.* Знакоместо, на которое указывают регистры координат курсора, оформляется особым образом. Обычно его выделяют мигающей полоской, размер и положение которой относительно знакоместа программируются. Подчеркнем, что к такому выделению байт атрибутов не имеет отношения, хотя возможен неудачный выбор атрибутов (сочетание цветов), когда курсор становится неразличимым.

Несмотря на большее количество узлов текстового адаптера по сравнению с чисто графическим, его цена была гораздо ниже. Дело в том, что в текстовом режиме с форматом 25 строк по 80 символов (максимальном для существующих чисто текстовых адаптеров) требуется всего 2 Кбайт видеопамяти для символов и 2 Кбайт для их атрибутов. При этом символы могут иметь вполне удобочитаемую матрицу разложения 9 x 14 и по 8 бит для атрибутов, определяющих цветовое оформление знакоместа. Частота считывания видеопамяти

для регенерации изображения невысока: за время прямого хода по строке должно быть считано всего 80 слов. Графический режим (720 x 350) для отображения такой же матрицы символов потребует уже около 32 Кбайт видеопамяти в монохромном варианте, а в 16-цветном — уже 128 Кбайт. Конечно, в настоящее время экономия видеопамяти в таких объемах уже не сказывается на цене адаптера, но не следует забывать и о том, что при выводе изображения эту память нужно заполнять. Поскольку в текстовом режиме в адаптер передаются только коды символов, заполнение всего экрана занимает в десятки раз меньше времени, чем построение того же изображения в графическом режиме. Программный код вывода символов в текстовом режиме проще и компактнее, чем при программном формировании его растрового изображения. По этим причинам все графические адаптеры имеют знакогенератор, дающий возможность работы и в текстовом режиме, а при переходе в графический режим знакогенератор отключается. Интеллектуальные адаптеры позволяют выводить символы (формировать их растровые изображения с заданным форматом знакомест) и в графическом режиме. При этом адаптер получает только команду с указанием координат отображаемых символов и сам поток кодов символов, после чего быстро строит их изображение, не отвлекая центральный процессор.

Обработка видеоизображений

Слово «видео» в современном толковании подразумевает привычное всем видеоизображение, которое мы видим на телевизионных экранах. Это изображение, в отличие от компьютерной графики, может получаться в результате видеосъемки естественных объектов. Чтобы подчеркнуть естественность происхождения, а также непредсказуемую подвижность изображения, ввели термин *живое видео*. Растровая система отображения информации на экране монитора PC имеет глубокие корни в телевидении, но объединить компьютерную графику с телевизионным изображением оказывается непросто. Для понимания сложностей и путей решения этой задачи кратко поясним принципы передачи цветных телевизионных изображений.

Как нам уже известно, в цветном мониторе экран сканируется одновременно тремя лучами базисных цветов и каждый луч модулируется входным сигналом. Также мы знаем, что отображение мелких элементов (большого количества пикселов в строке) требует полосы сигнала в десятки мегагерц. В телевидении широкополосные RGB-сигналы существуют только в пределах студии, передавать же их по радиоканалам в таком виде технически невозможно. Кроме того, телевизионный сигнал должен быть совместим с черно-белыми телеприемниками. В телевидении сигналы трех первичных цветов — красного, зеленого и синего — проходят через преобразователь координат, на выходе которого получают сигнал Y, несущий информацию о *яркости* (luminance) точек, и два цветоразностных сигнала U и V, несущих информацию о *цвете* (chrominance), то есть о яркости красного и синего цветов относительно зеленого. Зеленый выбран основным, поскольку зрение людей к нему наиболее чувствительно. Далее эти сигналы «путешествуют» по телевизионному тракту до телеприемника разными путями в зависимости от используемого вещательного стандарта. Сигнал

Y всегда передается на основной (несущей) частоте телевизионного канала; цветоразностные сигналы, специальным образом закодированные, передаются на поднесущей частоте канала. Потеря цветоразностного сигнала ведет к тому, что принятое изображение оказывается черно-белым. Поскольку проблема полосы пропускания видеотракта стоит остро, во всех вещательных системах принята чересстрочная развертка.

В первой системе цветного телевещания — *NTSC* (National Television Standards Committee — национальный комитет по телевизионным стандартам) принята частота кадров 30 Гц (частота полей — 60 Гц), а количество строк — 525, из которых видимых — 480. При полосе канала яркости 4,5 МГц в строке может быть различимо до 640 пикселов (вот откуда формат 640 x 480). Однако для передачи цветоразностных сигналов используется поднесущая частота 3,58 МГц, и горизонтальное разрешение снижается до 400-450 пикселов. Реально же домашний телеприемник обеспечивает примерно половину этого разрешения. Напомним, что это разрешение лишь по яркостному каналу. Цветоразностная информация (два сигнала) «втискивается» в подканал с поднесущей 3,58 МГц, да еще для экономии полосы, оставляемой яркостному каналу, после модуляции подавляют саму поднесущую и нижнюю часть спектра сигнала. Так что цветовые сигналы после таких преобразований, а также после передачи по радиоканалу и обратного восстановления в телеприемнике поступают на входы видеоусилителей R, G, B с существенно урезанной по сравнению даже с яркостным каналом полосой частот. Видеосигнал, состоящий из яркостной составляющей и поднесущей, модулированной цветоразностными сигналами, называется *композитным* (composite video). Поскольку наибольшие потери информации цветоразностного сигнала происходят при модуляции и демодуляции его поднесущей частоте, лучшее качество передачи изображения даст сигнал, взятый сразу после цветоразностного преобразователя на передающей стороне.

Помимо стандарта NTSC существуют еще два популярных в мире стандарта: PAL и SECAM.

В стандарте *PAL* (Phase Alternate Line) фаза одного из цветоразностных сигналов (R-Y) меняется от строки к строке, что и дало название этому методу. Такое решение позволило повысить стабильность декодирования. Для увеличения горизонтального разрешения поднесущая частота цветоразностного сигнала повышена до 4,43 МГц. Частота кадров — 25 Гц (при частоте полей 50 Гц), количество строк — 625. Стандарт PAL обеспечивает разрешение 800 x 600. В варианте PAL-M принят формат кадра NTSC (60 Гц и 525 строк), а в PAL-N при 625 строках (50 Гц) вернулись к поднесущей 3,58 МГц.

В нашей стране традиционно используется система французского происхождения *SECAM* (Sequence de Couleurs avec Mémoire). В этой системе вместо квадратурной модуляции поднесущей применены две поднесущие цветоразностных сигналов с частотной модуляцией. Частота кадров — 25 Гц (при частоте полей 50 Гц), количество строк — 625.

Говоря о телевизионных стандартах, не следует забывать о канале звукового сопровождения. Во всех этих системах для звука используется частотная модуляция дополнительной поднесущей частоты 6,5 МГц для SECAM; 5,5 МГц для

PAL (но иногда и 6,5 МГц); 4,5 МГц для NTSC и PAL-M; может встречаться и 6 МГц.
Все перечисленные системы в цветном режиме между собой *несовместимы*, хотя для них и существуют устройства-конверторы. Устройства обработки видеосигналов в PC могут поддерживать все системы или только некоторые из них, на что следует обращать внимание при их приобретении.
Обсудим возможные точки соприкосновения компьютерной графики и телевизионного видеосигнала.
Вывод компьютерной графики на экран обычного телевизора представляет интерес как средство презентаций: телевизоры с большим экраном применяются довольно широко и имеют умеренную цену, чего не скажешь о больших мониторах. Кроме того, транслировать один и тот же сигнал на несколько телевизоров проще, чем на несколько компьютерных мониторов. Ряд моделей современных графических карт имеют выход телевизионного сигнала, причем независимый от выхода на основной монитор. Есть и преобразователи форматов (например, *конвертор VGA-TV)* в виде отдельных внешних устройств со стандартным интерфейсом компьютерного монитора на входе и каким-либо телевизионным сигналом на выходе. В простейшем варианте конвертор только преобразует сигналы из RGB в один из интерфейсов телеприемника, но при этом требуется установка разрешения и частот синхронизации графического адаптера, совпадающих со стандартом телеприемника. Для пользователя PC эти ограничения малоприятны, а иногда и невыполнимы. Более сложные конверторы имеют собственную буферную память, которая заполняется вновь оцифрованным видеосигналом, снятым с выхода графического адаптера. На телевизионный выход информация из буфера выдается уже с телевизионной частотой. Буфер может хранить одну, несколько или все строки экрана. От этого зависят ограничения на режим разрешения и соотношения частот регенерации графического адаптера и телевизионного монитора (в последнем случае они вообще могут быть несвязанными). Естественно, эти варианты значительно различаются по сложности и цене (конвертор с полноэкранным буфером самый дорогой). Однако когда графический адаптер выводит движущееся изображение, смена которого привязана к кадровой синхронизации, при несовпадении кадровых частот на телевизионном экране движение искажается. Общей проблемой конверторов является необходимость борьбы с мерцанием (flickering): поскольку в телеприемниках используется чересстрочная развертка, то, как уже отмечалось, горизонтальная полоса шириной в пиксел воспроизводится с частотой 25 или 30 Гц, что улавливается глазом. Возможны и варианты внутренних адаптеров (карт расширения), подключаемых к шине расширения PC и внутреннему разъему графической карты (VFC или VAFC). Некоторые модели конверторов позволяют накладывать графическое изображение на внешний видеосигнал (например, для создания титров). Ввиду ограниченной горизонтальной разрешающей способности телеприемников (полоса пропускания шире 5 МГц для телевизора как такового бессмысленна) возможность замены монитора телевизором для регулярной работы сомнительна. В стандарте NTSC обеспечивается разрешение 640 x 480, в PAL и SECAM — 800 x 600. Однако такое разрешение реально

достижимо только с интерфейсом S-Video. Композитный сигнал, как было сказано ранее, не обеспечивает столь высокого разрешения. Выход телевизионного сигнала имели адаптеры CGA и EGA, с приходом VGA этот интерфейс на графических картах применять перестали. Однако на новом витке развития техники об интерфейсе с телевизионным приемником снова вспомнили. Microsoft рекомендует помимо стандартного интерфейса VGA (RGB-Analog) устанавливать на новых графических картах выход композитного сигнала S-Video. Более того, рекомендуется предусмотреть возможность одновременной работы VGA-монитора и TV-приемника, что не так-то просто обеспечить из-за различия параметров синхронизации.

Гораздо чаще используют «обратное скрещивание» — *вывод видеоизображения* на экран компьютерного монитора. Видеоизображение выводится в окно, занимающее весь экран или его часть. Выводить «живое видео» в экранный буфер (видеопамять) — задача весьма ресурсоемкая (по использованию процессора и интерфейса между ним и графическим адаптером). Более экономичный способ вывода видео — применение *видеооверлейных плат* (video overlay board). Эти платы позволяют изменять размер окна видео так же, как и размер любого окна в Windows. В оверлейной плате для видеоизображения имеется специальный «слой» видеопамяти, независимый от видеобуфера графического адаптера. В нем содержится оцифрованное растровое отображение каждого кадра видеосигнала. Поскольку для видеосигнала принято цветовое пространство в координатах Y-U-V, в рассматриваемом слое памяти пикселы также отображаются в этом пространстве, а не в R-G-B, свойственном графическим адаптерам. В такой системе движущееся видеоизображение, видимое на экране монитора, существует лишь в оверлейном буфере, но никогда не попадает в видеопамять графического адаптера и не передается ни по каким внутренним цифровым шинам компьютера. В видеопамяти графического адаптера «расчищается» окно, через которое «выглядывает» видеоизображение из оверлейного буфера. Некоторый цвет (комбинация битов RGB) принимается за прозрачный. Оверлейная логика сравнивает цвет очередного пиксела графического буфера с этим прозрачным, и если он совпадает, вместо данного пиксела выводится соответствующий пиксел видеооверлея. Если цвет не совпадает с прозрачным, то выводится пиксел из графического буфера. Таким образом, имея доступ к пикселам графического буфера, можно на видеоизображение накладывать графику для организации видеоэффектов или вывода в видеоокне «всплывающих» (popup) меню. Наложение производится на уровне потока битов сканируемых пикселов, который может передаваться в оверлейную плату через разъем Feature Connector (см. далее). Видеопоток в оверлейный буфер попадает либо с внешних видеовходов, либо от декодера (программного или аппаратного) сжатого видеопотока. Признак использования оверлея — отсутствие содержимого окна видео (пустой прямоугольник) на экранных снимках (если видео есть, значит, оно выводится через видеопамять). Оверлейная плата обычно имеет несколько входов для источников аналогового видеосигнала и программно-управляемые средства выбора одного из них. В составе такого устройства обычно есть и *фрейм- граббер* (frame grabber) — инструмент захвата видеокадра, другое название которого — *Video Capture*. По команде оператора движущееся изображение может

быть мгновенно зафиксировано в оверлейном буфере, после чего захваченный кадр может быть записан на диск в каком-либо графическом формате для последующих обработки и использования. Более совершенные устройства позволяют записывать в реальном времени последовательность видеокадров, выполняя или не выполняя их компрессию «на лету». Порты AGP 3.0 и PCI-E позволяют выполнять изохронные передачи между графической картой и системной памятью, что обеспечивает передачу даже несжатого видеопотока (в обоих направлениях).

Фрейм-граббер может быть отдельным устройством, подключаемым к источнику видеосигнала и какому-либо интерфейсу компьютера, и не иметь отношения к видеооверлею. В этом случае видеоизображение наблюдается уже не на мониторе компьютера, а на обычном телевизоре, подключенном к тому же источнику видеосигнала или фрейм-грабберу. По команде оператора требуемый кадр фиксируется в буферной памяти фрейм-граббера, откуда по интерфейсу поступает в компьютер для обработки или/и хранения.

TV-тюнер — устройство приема видеосигналов с радиочастотного входа (антенны) — в сочетании с оверлейной платой позволяет просматривать телепрограммы на обычном мониторе компьютера. Тюнер может поддерживать стандарты цветопередачи PAL, SEC AM и NTSC, но из-за несовпадения стандартов на промежуточную частоту звукового сопровождения некоторые карты не принимают звуковое сопровождение отечественных телеканалов.

Теперь посмотрим, во что «выльется» попытка *передачи видеоизображения в цифровом виде* (в формате Bitmap), естественном для графической системы компьютера. Пусть разрешение видеоэкрана составляет 640 x 480 — максимально возможное для телевизионного изображения NTSC. Поскольку аналоговый телевизионный сигнал позволяет передавать в принципе неограниченное число цветов[1], примем глубину цвета True Color — 24 бита на пиксел. Тогда одному кадру изображения будет соответствовать битовый образ объемом 640 x 480 x x 24 = 7 372 800 бит, или около 7 Мбит на кадр. В телевидении полные кадры сменяются с частотой 25 Гц (30 Гц в NTSC), так что для непосредственной передачи телевизионного изображения в формате Bitmap требуется обеспечить поток данных в 7 x 25 = 175 Мбит/с, или около 22 Мбайт/с. О том чтобы записывать такой поток данных даже на самый быстрый винчестер, раньше не было и речи, но современные диски уже могут его выдержать (см. главу 9). Но этот поток заполняет 1 Гбайт диска всего за 44 секунды. Конечно, если пожертвовать количеством цветов и «опуститься», например, до режима High Color (16 бит на пиксел), то требуемый поток уменьшится до 116 Мбит/с. Но и такой поток слишком велик. Выходом может быть только *сжатие передаваемой информации.*

Как уже отмечалось, формат Bitmap является довольно расточительным способом описания изображений. Соседние (по вертикали и горизонтали) элементы реального изображения обычно между собой сильно взаимосвязаны (коррелированы), поэтому имеются богатые возможности сжатия изображения. Иллю

[1] Конечно, он все-таки ограничен динамическим диапазоном аналогового тракта.

страция этого — очень большой коэффициент сжатия BMP-файлов любым архиватором. Если сжатие файлов данных при архивации обязательно требует возможности точного восстановления исходных данных при распаковке, то при сжатии изображений в большинстве случаев можно позволить некоторые вольности, когда восстановленное изображение не совсем точно соответствует оригиналу. И наконец, соседние кадры движущегося изображения между собой в большинстве случаев тоже сильно связаны, что наводит на мысль о применении дифференциального описания кадров. Все эти рассуждения подводят нас к пониманию возможностей сжатия видеоинформации и принципов действия *кодеков* — компрессоров-декомпрессоров видеосигнала. Как и в случае программного сжатия и восстановления данных, задача компрессии оказывается сложнее задачи восстановления (легко заметить, что распаковка файлов, например архиватором ARJ, происходит гораздо быстрее упаковки). Процедура сжатия может выполняться как одноступенчатым, так и двухступенчатым способом. В первом случае сжатие выполняется одновременно с записью в реальном масштабе времени. Во втором случае поток несжатых данных интенсивностью в несколько десятков мегабайт в секунжу записывается на специальный (очень большой и очень быстрый) диск. По окончании записи фрагмента выполняется его сжатие, которое может занимать на порядок больше времени, чем сама запись. Декомпрессия, естественно, представляет интерес лишь в том случае, если она выполняется в реальном масштабе времени (к счастью, она и реализуется проще). Ряд кодеков позволяют осуществлять декомпрессию в реальном времени чисто программными способами, используя стандартный графический адаптер SVGA. Однако программная декомпрессия значительно загружает процессор, что неблагоприятно сказывается на многозадачном использовании компьютера. Ряд современных дисплейных адаптеров имеют специальные аппаратные средства декомпрессии, разгружающие центральный процессор. На долю процессора остается лишь организация доставки сжатого потока данных к плате адаптера.

Сжатие движущихся изображений включает *внутрикадровое* (intraframe compression) и *межкадровое* (interframe compression) сжатие. Для внутрикадрового сжатия используются методы, применяемые для сжатия неподвижных изображений. В межкадровом сжатии применяется система *ключевых кадров* (key frame), содержащих полную информацию о кадре, и *дельта-кадров* (delta frame), содержащих информацию о последовательных изменениях кадров относительно ключевых. Благодаря корреляции соседних кадров дельта-кадры в общем случае несут гораздо меньше информации, чем ключевые, и, следовательно, поток их данных не так интенсивен. Периодическое вкрапление ключевых кадров позволяет избежать накопления ошибок в изображении, а также начинать прием потока в любой момент (дождавшись ближайшего ключевого кадра).

При съемке различных сюжетов межкадровая корреляция, конечно же, будет существенно варьироваться. Поэтому, чтобы оценить качество работы кодека, применяют, например, сюжеты типа «говорящие головы» (talking heads) с высокой степенью корреляции кадров и более сложные полнодвижущиеся изображения (actions), где все элементы перемещаются (например, карусель). Оценка

качества ведется как по объективным показателям, так и по субъективному восприятию. Объективными показателями являются максимальная частота кадров (frame rate), которая обеспечивается без отбрасывания кадров, и процент отбрасываемых кадров (drop frames) при обработке потока со стандартной частотой кадров. Эти показатели характеризуют производительность декомпрессора, которая может оказаться и недостаточной для обработки потока данных без потерь. Интересен также и коэффициент загрузки центрального процессора (CPU Utilization) при отработке стандартного потока, по которому можно судить о возможности исполнения других функций во время воспроизведения видео.

В процессе декомпрессии может потребоваться масштабирование кадров, для того чтобы вписать изображение в окно заданного размера. В простейшем случае декомпрессия производится в масштабе 1:1, при этом видеоизображение обычно занимает лишь часть экрана. Примитивное масштабирование достигается дублированием пикселов — один пиксел видео копируется в несколько (например, 4) смежных пикселов графического экрана. Однако при этом качество изображения заметно падает — крупные «кирпичики», из которых строится изображение, с небольшого расстояния выглядят плохо. Более тонкий механизм масштабирования подразумевает интерполяцию цветов пикселов, при этом качество изображения заметно улучшается. Однако такое масштабирование уже требует значительных затрат вычислительных ресурсов, и если их недостаточно, то вывод видеоизображения в окно большого размера сопровождается потерями кадров и, возможно, перебоями звукового сопровождения. Так что, говоря о качестве вывода видео, следует всегда оговаривать масштаб или размер видеоэкрана.

Для сжатия изображений применяются различные кодеки, сейчас остановились на кодеках стандартов MPEG (см. далее).

Видеосигнал в сжатом формате может быть сохранен на вполне рядовом носителе информации (винчестер, CD, DVD) и воспроизведен с него на мониторе компьютера. С этой цифровой записью могут выполняться любые операции нелинейного монтажа (монтажа с произвольным доступом к кадрам). Возможности такого монтажа определяются программным обеспечением и, по сути, безграничны (конечно, они определяются и производительностью компьютера — если монтаж одной минуты потребует, скажем, недели работы, то мало кто будет им пользоваться).

Для обмена видеоданными с другими устройствами сжатый поток может быть передан, например, по шине FireWire, USB или через средства телекоммуникаций (см. главы 13, 17 и 18). Если компьютер оборудован телекамерой со средствами компрессии и передачи изображений по телекоммуникационным каналам, то появляется возможность организации видеотелефона и даже видеоконференций.

Стандарты MPEG

Разработкой кодеков, предназначенных для работы (по крайней мере, для декомпрессии) в реальном масштабе времени, занимается группа *MPEG* (Moving

Picture Expert Group — группа экспертов в области движущихся изображений). Поскольку видео без звука «живым» представить трудно, MPEG занимается и аудиокодеками. Кодеки MPEG работают в пространстве Y-U-V, причем яркостная информация обрабатывается с большим разрешением, чем цветовая. В сжатом потоке данных присутствуют кадры нескольких типов:

- I-кадры (I означает intra) — ключевые кадры, кодированные без ссылок на другие (то есть содержащие полное описание статического изображения);
- P-кадры (P — predicted) содержат описание отличий текущего кадра от предыдущего;
- B-кадры (B — bi-directional) являются двунаправленными, они ссылаются и на предыдущий, и на следующий кадры.

Наличие двунаправленных кадров подразумевает, что декодер должен иметь буфер по крайней мере на три принятых кадра, а изображение должно выводиться с некоторым отставанием от входного потока. Для того чтобы кодек мог быстро включиться в работу с любого места потока, 1-кадры должны включаться в поток регулярно (в MPEG-1 — не реже, чем через 0,4 с).

MPEG-1 — стандарт ISO/IEC 11172, принятый в 1992 году. Полное название — «Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to about 1,5 MBit/s» — раскрывает его суть: кодек предназначен для записи и воспроизведения движущихся изображений и связанного с ними аудиосопровождения на цифровом носителе с потоком данных до 1,5 Мбит/с. При этом обеспечивается качество изображения на уровне кассетного видеомагнитофона *VHS* (Video Home System) со стереофоническим звуковым сопровождением. Стандарт состоит из 5 частей, описывающих систему, видео, аудио, методику тестирования на соответствие и программы кодера и декодера на языке C. Для получения столь низкой скорости потока из исходного видеопотока берется лишь одно поле (полукадр), а в каждой строке — только половина пикселов. При этом в стандарте NTSC получается формат кадра 352 x 240 при частоте 30 кадров в секунду; в стандартах PAL и SECAM — 352 x 288, 25 кадров в секунду. В обоих случаях исходный поток, подлежащий сжатию, имеет одну и ту же скорость — 352 x 240 x 30 = 352 x 288 x 25 = 2 534 400 пикселов в секунду. Увеличение размера декодированного изображения до полного экрана особого смысла не имеет, поскольку может выполняться только масштабированием (размножением пикселов); правда, в более сложных реализациях декодера можно улучшить качество изображения, применяя методы интерполяции. Аудиопоток по сравнению с исходным PCM-потоком (частота 44,1 кГц) сжимается примерно в 6 раз (типовая скорость — 256 Кбит/с). Декодер MPEG-1 содержит демультиплексор, разделяющий аудио- и видеопотоки, и раздельные декодеры для них. Скорость потока данных позволяет использовать в качестве носителя видеоинформации обыкновенный диск CD-ROM, поэтому MPEG-1 применяется в дисках CD-i и VideoCD (VCD). Однако декомпрессия требует относительно большой мощности процессора (Pentium 133, которого в 1992 году еще не было), из-за чего диски CD-i и VCD без специальной платы аппаратного декодера маломощные компьютеры воспроизводить не могут.

MPEG-2 (ISO/IEC 13818, 1995 г.) — кодек для высококачественной передачи изображений, аудиоинформации и данных при потоке 2-80 Мбит/с, обеспечивающий несколько уровней качества (табл. 10.1). Основной уровень (720 x 480, 30 кадров/с) обеспечивает качество на уровне телевещания, высокие уровни используются в профессиональной видеозаписи. Стандарт предусматривает одновременную передачу множества TV-каналов с возможностью шифрования для ограничения доступа к информации и защиты прав собственности на содержимое потоков. Первые 5 частей стандарта аналогичны MPEG-1, но с новым наполнением. Аудиокодек MPEG-2 представляет собой многоканальное расширение аудиокодека MPEG-1, что обеспечивает их совместимость по первым двум каналам. В аудиокодеке MPEG-2 имеются 2 стереоканала (фронт и тыл), обеспечивающие объемное звучание (surround), и один низкочастотный с полосой до 100 Гц. Помимо аудиокодека, совместимого с MPEG-1, в MPEG-2 входит и более совершенный аудиокодек AAC (Advanced Audio Codec), используемый в профессиональной аудиотехнике. Аудиокодек MPEG-1/MPEG-2 стал жить самостоятельной жизнью под именем MP3 (MPEG-1 Layer 3, см. 12.1), иногда по ошибке превращаемым в MPEG-3 (такого стандарта нет). То, что должно было стать MPEG-3, вылилось в высший уровень MPEG-2. Этот уровень обеспечивает качество телевидения высокой четкости, или ТВЧ (High Definition TV, HDTV).

Таблица 10.1. Уровни качества MPEG-2

Уровень	Размер изображения	Максимальный поток данных при частоте 30 кадров/с, Мбит/с
Low (низкий)	352 x 240	4
Main (основной)	720 x 480	15
High-1440 (высокий)	1440x1152	60
High (высший)	1920x1080	80

Модель взаимодействия компонентов воспроизводящих систем MPEG-1 и MPEG-2 довольно проста — данные от источника через средства доставки поступают на вход демультиплексора, где разделяются на видео- и аудиопотоки, обрабатываемые своими декодерами. Информационный поток MPEG-1 сугубо однонаправленный; в MPEG-2 добавляется двунаправленный канал взаимодействия получателя информации с источником данных (сервером вещания) через средства доставки, что обеспечивает интерактивность в смысле выбора передаваемых программ, а также адаптацию передаваемого потока к качеству канала передачи.

MPEG-4 — стандарт, ориентированный на интерактивное использование мультимедиа и сетевых коммуникаций. По сравнению с предыдущим, MPEG-4 устроен гораздо сложнее — аудио- и видеоинформация, представляемая конечному потребителю, собирается из различных *аудиовизуальных объектов* (Audio-Visual Objects, AVO). В предыдущих стандартах фигурировали лишь *потоки* (видео и аудио), которые просто воспроизводились. Объекты MPEG-4 отображаются на сцене, представляемой конечному потребителю (наблюдателю-слушателю).

Сцена представляется дисплеем и многоканальной аудиосистемой. Исходная композиция (расположение объектов на сцене), заданная создателем воспроизводимого произведения, может в широких пределах меняться пользователем. Объекты, представляемые на сцене, могут быть как естественными, так и синтетическими. Между объектами устанавливаются определенные связи. Для описания объектов, их классов и сцен введен объектно-ориентированный *язык BIFS* (сродни C++). Интерактивность позволяет управлять как представлением сцены (например, менять ракурс), так и композицией («сборкой», содержанием и поведением объектов на сцене), а также, как и в MPEG-2, взаимодействовать с источником данных через средства доставки.

Естественные аудиообъекты — это каналы передаваемого аудиосигнала, сжатого в зависимости от потребностей в качестве и доступной полосы пропускания канала передачи. Уровень качества может быть от телефонного до высокого (каналы с виртуальной скоростью 2-64 Кбит/с). Для каждого уровня качества и занимаемой полосы используется свой метод компрессии/декомпрессии. *Синтетические аудиообъекты* образуются из структурированных потоков входных данных. Декодер TTS (Text to Speech — преобразование текста в речь) синтезирует речь по потоку текстовых данных, формируя управляющие данные для анимации движения губ. Декодер имеет многонациональную языковую поддержку. Он позволяет управлять тембром и громкостью, делать паузы, выполнять «перемотку» вперед и назад. Музыкальным аналогом TTS является интерфейс MIDI, но в MPEG-4 может использоваться и более мощный и точный метод синтеза музыки — Score Driven Synthesis. Поток для синтеза музыки содержит команды, описывающие звуковые примитивы, генерируемые с помощью сигнальных процессоров. Таким образом передаются потоки для всех инструментов оркестра, позволяющие синтезировать их совместное звучание, которое может оживляться такими деталями, как шум шагов в зале или звук открываемой двери. К аудиообъектам могут быть приложены различные эффекты; аудиообъекты могут привязываться к визуальным объектам и позиционироваться в любой точке сцены (объемной).

Визуальные естественные объекты могут быть текстурами, изображениями и видео. Текстуры предназначены для наложения на каркасные двухмерные (2-Dimensional, 2D) или трехмерные (3-Dimensional, 3D) модели. Изображения могут просто помещаться в любое место сцены. Видеообъект подразумевает «живое» изображение, при этом кодированию (компрессии) может подвергаться объект не полноэкранный и не прямоугольный (например, движущийся человек на прозрачном фоне). Такой объект способен подвергаться более эффективному сжатию, правда, определение его границ (при кодировании) оказывается сложнее. *Визуальные синтетические объекты* представляют собой элементы компьютерной графики, получаемые разными способами в векторном или растровом виде. Это могут быть и двухмерные или трехмерные каркасные модели, на которые наложены текстуры (естественные и синтетические). Для повышения качества моделирования живых объектов в MPEG-4 разработана специальная система параметризованного описания человеческой головы, способная изображать различные эмоции, а также воспроизводить движение губ при раз

говоре. Движение губ может быть связано с воспроизводимым аудиосигналом, привязанным к этому объекту (голове). На модель головы может быть наложена текстура, полученная из изображения лица конкретного человека. Разрабатывается также и специализированная модель человеческого тела.

Для иллюстрации можно представить, к примеру, такую сцену. В синтетической комнате (трехмерная модель) расположен синтетический диктор (модель), лицо которого является текстурой, сделанной из портрета известного человека. Этот диктор «читает» текст (подсунутый ему пользователем или хакером через сеть), ходит по комнате, по «просьбе» пользователя может остановиться и помолчать или же повторить сказанное. Вся эта синтетика в сочетании с интерактивностью ограничивается малым объемом передаваемых данных — достаточно раз передать описание сцены, диктора и текстуру его лица, после чего требуется передача лишь текста и информации, управляющей движением моделей на сцене.

Визуальная часть MPEG-4 предоставляет инструменты и алгоритмы для эффективной компрессии видео и изображений, текстур для наложения на 2D- и 3D-сетки (каркасы), самих сеток, потоков геометрических данных, «оживляющих» сетки. Также имеются средства для произвольного доступа ко всем типам объектов, манипулирования изображениями и видеопотоками. Способы кодирования и масштабирования изображений, текстур и видео зависят от типа содержимого. Ядром средств кодирования визуальных объектов является *видео с очень низкой скоростью потока* (Very Low Bit-rate Video, VLBV) — не более 5-64 Кбит/с, позволяющей устойчиво по отношению к ошибкам передачи передавать маленькие кадры (176 x 144 пиксела) с темпом 10-15 кадров/с. Вокруг этого ядра строятся интерфейс с высокой скоростью передачи и расширения функциональности, позволяющие индивидуально кодировать объекты сцены, что и обеспечивает интерактивность. Полноэкранное изображение приемлемого качества требует скорости порядка 600 Кбит/с.

Широкое распространение MPEG-4 во все мультимедийные отрасли может сильно изменить взгляды на способы создания и использования мультимедийной продукции, но это тема отдельного обсуждения вне рамок данной книги. В MPEG-4 предусматриваются средства контроля за соблюдением прав собственности на произведения, направленные на поддержку платного распространения, защиту авторских прав и т. п. В случае необходимости допустимые изменения содержимого при воспроизведении тоже должны быть ограничены, чтобы избежать искажения смысла произведения.

Для воспроизведения сжатых видеофайлов (или потоков) требуется *декодер* — программный компонент, который может пользоваться и аппаратными ускорителями.

MPEG-плеер — декодер MPEG-1, обеспечивающий воспроизведение с компакт-дисков форматов MPEG-1 (CD-I, VideoCD). Аппаратный декодер является широко распространенным дополнением графического адаптера. В отличие от программных MPEG-декомпрессоров, он обеспечивает высокое качество воспроизведения с невысокой загрузкой процессора. В состав MPEG-плеера должен входить и аудиодекодер, при этом на графической плате с аппаратным

декодером появляется немного неожиданный дополнительный разъем аудиовыхода.
DVD-плеер — аппаратный или программный декодер MPEG-2, позволяющий воспроизводить видеозаписи с DVD-Video и Super VideoCD. Для программного декодирования требуется, как минимум, компьютер с процессором Pentium II-266, для аппаратного достаточно Pentium-133.
DivX — название программного кодека для MPEG-4 (альтернатива стандартному кодеку от Microsoft).
Для программного декодирования MPEG-4 (в реальном времени) требуется, как минимум, Pentium II-300. Однако для гладкого воспроизведения в полноэкранном режиме требуется Pentium III-733 (или AMD Duron 800). Декодирование практически полностью выполняется средствами центрального процессора, «интеллект» графического адаптера используется только для масштабирования (если требуется) выходного изображения. Для ускорения вывода может применяться видеооверлей графического адаптера, в который посылается декодированный поток кадров. Компрессия в реальном времени может быть реализована, например, на двухпроцессорном компьютере Pentium II-400. Выпускаются и PCI-карты акселераторов декодирования MPEG-4 (а заодно MPEG-1 и MPEG-2), в которых установлен специализированный сигнальный процессор и видеооверлей. Через эту карту проходит выходной поток основного графического адаптера, подключенного кабелем VGA Loop (монитор подключается к этой карте). Такая карта позволяет воспроизводить MPEG-4 на старых ПК с маломощным процессором.

10.2. Акселератор — «интеллект» графического адаптера

Под *интеллектом графического адаптера* подразумевается наличие на его плате собственного процессора, способного формировать растровое изображение в видеопамяти (bitmap) по командам, полученным от центрального процессора. Команды ориентируются на наиболее часто используемые методы описания изображений, которые строятся из отдельных графических элементов более высокого уровня, чем пикселы.
Команды рисования (drawing commands) обеспечивают построение *графических примитивов* — точек, отрезков прямой, прямоугольников, дуг, эллипсов. Примитивы такого типа в командах описываются в векторном виде, что гораздо компактнее, чем их растровый образ. Таким образом, за счет более эффективного способа описания изображений удается значительно сократить объем передаваемой графической информации. К командам рисования относится и *заливка* замкнутого контура, заданного в растровом виде, некоторым цветом или узором (pattern). Она ускоряется особенно эффективно: при программной реализации процессор должен просмотреть содержимое видеопамяти вокруг заданной точки, двигаясь по всем направлениям до обнаружения границы контура и изменяя цвет пикселов на своем пути. При этом требуются чтение боль

шого объема данных видеопамяти, их анализ и запись модифицированных данных обратно в видеопамять. Процессор интеллектуального адаптера способен выполнить эту операцию быстро и без выхода с этим потоком данных на внешнюю магистраль.

Вывод текста в графическом режиме сродни рисованию, только объекты рисования задаются кодами символов. Эти символы изображаются в растровом виде в соответствии с выбранными шрифтом, размером и цветом. В простейшем случае шрифт задается матричным разложением символа в фиксированную область (знакоместо) размером 8 x 8, 9 x 14 или 9 x 16 пикселов. Более сложные шрифты задаются контурами с переменным размером знакоместа.

Копирование блока с одного места экрана на другое требуется для «прокрутки» изображения экрана в разных направлениях. Эта команда сводится к пересылке блока битов (Bit Block Transferring, BitBlT) и может быть сильно ускорена интеллектуальным адаптером.

Для формирования курсора на графическом экране применяют команды для работы со спрайтами. *Спрайт* (sprite) — небольшой прямоугольный фрагмент изображения, который может перемещаться по экрану как единое целое. Перед использованием его программируют — определяют размер и растровое изображение для него, после этого для его перемещения по экрану достаточно только указывать его координаты.

Аппаратная поддержка окон (hardware windowing) упрощает и ускоряет работу с экраном в многозадачных (многооконных) системах. На традиционном графическом адаптере при наличии нескольких (возможно, перекрывающих друг друга) окон программе приходится отслеживать координаты обрабатываемых точек с тем, чтобы не выйти за пределы своего окна. Аппаратная поддержка окон упрощает вывод изображений: каждой задаче выделяется свое окно — область видеопамяти требуемого размера, — в котором она работает монопольно. Взаимное расположение окон сообщается интеллектуальному адаптеру, и он для регенерации изображения синхронно с движением луча по растру сканирует видеопамять не линейно, а перескакивая с области памяти одного окна на область памяти другого.

Если объем видеопамяти превышает необходимый для данного формата экрана и глубины цветов, то в ней можно строить изображение, превышающее по размеру отображаемую часть. Интеллектуальному адаптеру можно поручить *панорамирование* (panning) — отображение заданной области. При этом горизонтальная и вертикальная прокрутка изображения не требует блочных пересылок (конечно, в пределах сформированного большого изображения) — для перемещения достаточно лишь изменить указатель положения (этакий «большой спрайт»).

Вышеописанные функции интеллектуального адаптера относятся к двухмерной графике (2D). Современные графические адаптеры берут на себя и многие функции построения трехмерных изображений, о чем более подробно рассказано далее.

Ускорение построений в интеллектуальном адаптере обеспечивается несколькими факторами. Во-первых, это сокращение объема передач по магистрали. Во-вторых, во время функционирования процессора адаптера центральный процессор остается свободным, что ускоряет работу программ даже в однозадачном режиме. В-третьих, процессор адаптера, в отличие от процессора с самой сложной в мире системой команд — представителя семейства x86, ориентирован на выполнение меньшего количества инструкций, а потому способен выполнять их гораздо быстрее центрального. И, в-четвертых, скорость обмена данными внутри адаптера может повышаться за счет лучшего согласования обращений к видеопамяти для операций построения с процессом регенерации изображения, а также за счет расширения разрядности внутренней шины данных адаптера. В графических адаптерах конца 90-х годов широко применялась двухпортовая видеопамять VRAM и WRAM с разрядностью внутренней шины 64 бит (при 32-битной шине внешнего интерфейса). Современные адаптеры с 3D-акселераторами (самые критичные к производительности памяти) строятся на памяти SGRAM (SDRAM) со 128-разрядной шиной, а в самых мощных применяется память SGRAM/SDRAM с удвоенной частотой передачи (Double-Data Rate, DDR). Правда, и здесь полная разрядность шины (но уже внутренней) при малом объеме установленной видеопамяти может не использоваться. Разрядность шины пока дальше не увеличивают, но встречаются адаптеры и с двойной внутренней шиной, способной работать в полнодуплексном режиме.

По отношению к центральному процессору и оперативной памяти компьютера различают графические сопроцессоры и акселераторы. *Графический сопроцессор* представляет собой специализированный процессор с соответствующим аппаратным окружением, который подключается к шине компьютера и имеет доступ к его оперативной памяти. В процессе своей работы сопроцессор пользуется оперативной памятью, конкурируя с центральным в плане доступа и к памяти, и к шине. *Графический акселератор* работает автономно и при решении своей задачи со своим огромным объемом данных может не выходить на системную шину. Акселераторы являются традиционной составной частью практически всех современных графических адаптеров. Акселераторы для двухмерных операций (2D-accelerators), необходимых для реализации графического интерфейса пользователя (Graphic User Interface, GUI), часто называют *Win- dows-акселераторами*, поскольку их команды обычно ориентированы на функции этой популярной операционной системы. Более сложные акселераторы выполняют и трехмерные построения, их называют 3D-акселераторами.

Для построения сложных трехмерных изображений графическому акселератору должно быть явно тесно в ограниченном объеме видеопамяти. Для доступа к основной памяти компьютера он должен иметь возможность управления шиной (bus mastering). Специально для мощных графических адаптеров в 1996 году появился новый канал связи с памятью — *AGP* (Accelerated Graphic Port), шина которого описана в 14.9. Обеспечив высокую пропускную способность порта, разработчики AGP предложили технологию *DIME* (Direct Memory Execute). Согласно этой технологии, графический акселератор является мастером

шины AGP и может пользоваться основной памятью компьютера для своих нужд при трехмерных построениях. Например, в основной памяти могут храниться текстуры, которые акселератор накладывает на трехмерные поверхности. При этом снимается ограничение на размер описания текстур, которые без AGP приходится держать в ограниченном объеме видеопамяти. На дешевое решение проблемы «тесноты» нацелена и архитектура однородной памяти UMA, которая может быть реализована с помощью AGP. Однако AGP позволяет сохранить и локальную память на графическом адаптере (видеобуфер), и расширение доступной памяти не отзывается снижением производительности.

Трехмерная графика

Потребности работы с трехмерными изображениями, или 3D-графикой, имеются в широком спектре приложений — от игр, которыми увлекается масса пользователей, до систем автоматического проектирования, применяемых в архитектуре, машиностроении и других областях. Конечно же, компьютер управляет не самими трехмерными объектами, а их математическими описаниями. Трехмерное приложение оперирует объектами, описанными в некоторой глобальной системе координат (global, или world, coordinate system). Чаще всего здесь используется *ортогональная*, она же *декартова* (cartesian), система координат, в которой положение каждой точки задается ее расстоянием от начала координат по трем взаимно перпендикулярным осям *X, Y* и Z. В некоторых случаях применяют также *сферическую* систему координат, в которой положение точки задается удалением от центра и двумя углами направления. В этом «мире» находятся все объекты, которые создает и учитывает приложение, и они имеют определенное взаимное расположение. Пользователю эти объекты могут быть продемонстрированы лишь с помощью графических устройств вывода, из которых наибольший интерес пока представляет дисплей. Однако он, как и большинство устройств визуализации, имеет лишь плоский (двухмерный) экран, с помощью которого необходимо создать иллюзию трехмерного изображения. Здесь упомянем и о существовании стереоскопических систем отображения, позволяющих в значительной степени добиться эффекта присутствия наблюдателя в среде изображаемых объектов. Однако и в них тем или иным способом для каждого глаза формируется свое двухмерное изображение, так что эту иллюзию приходится создавать дважды с двух точек зрения, несколько смещенных относительно друг друга. Действительно, трехмерные устройства вывода уже существуют, но пока что они способны «выводить» лишь статические объекты. Здесь имеется в виду устройство Model Maker (фирма SPI), которое с высокой точностью «выращивает» из пластмассы объекты сложной формы, описание которых поступает из трехмерного приложения САПР.

Пока что сосредоточимся на выводе трехмерного изображения на экран графического дисплея. Как известно, в конечном счете на монитор выводится растровое изображение, сформированное в видеопамяти. На экране мы видим матрицу пикселов размерностью 800 x 600, 1024 x 768, 1280 x 1024 и больше. Каждому пикселу соответствует ячейка видеопамяти, разрядность которой определяет возможности цветопередачи. Наибольший интерес для трехмерной графики

представляют режимы, в которых цветом каждого пиксела непосредственно управляют 15-16 бит (High Color) или 24 бита (True Color) ячейки видеопамяти. Режимы с индексным определением цвета (8 бит ячейки видеопамяти определяют цвет в соответствии с программированием палитр) для реалистичного изображения трехмерных объектов малопригодны.

Трехмерное изображение должно состоять из ряда поверхностей различной формы. Эти поверхности «собираются» из отдельных элементов-полигонов, чаще треугольников, каждый из которых имеет трехмерные координаты вершин и описание поверхности (цвет, узор). Перемещение объектов (или наблюдателя) приводит к необходимости пересчета всех координат. Для создания реалистичных изображений учитываются перспектива — пространственная и атмосферная (дымка или туман), освещенность поверхностей и отражение света от них, прозрачность и многие другие факторы.

Графический конвейер

Графический конвейер (graphic pipeline) — это некоторое программно-аппаратное средство, которое преобразует описание объектов в «мире» приложения в матрицу ячеек видеопамяти растрового дисплея. Его задача — создать иллюзию, о которой говорили выше.

В глобальных координатах приложение создает объекты, состоящие из трехмерных примитивов. В этом же пространстве располагаются источники освещения, а также определяются точка зрения и направление взгляда наблюдателя. Естественно, что наблюдателю видна только часть объектов: любое тело имеет как видимую (обращенную к наблюдателю), так и невидимую (обратную) сторону. Кроме того, тела могут полностью или частично перекрывать друг друга. Взаимное расположение объектов относительно друг друга и их видимость для зафиксированного наблюдателя обрабатываются на первой стадии графического конвейера, называемой *трансформацией* (transformation). На этой стадии выполняются вращение, перемещение и масштабирование объектов, а затем и преобразование из глобального пространства в пространство наблюдения (world-to-viewspace transform), а из него и преобразование в «окно» наблюдения (viewspace-to-window transform), включая и проецирование с учетом перспективы. Попутно с преобразованием из глобального пространства в пространство наблюдения (до него или после) происходит удаление невидимых поверхностей, что значительно сокращает объем информации, участвующей в дальнейшей обработке. На следующей стадии конвейера (lighting) определяется *освещенность* (и цвет) каждой точки проекции объектов, обусловленная установленными источниками освещения и свойствами поверхностей объектов. И наконец, на стадии *растеризации* (rasterization) формируется растровый образ в видеопамяти. На этой стадии на изображения поверхностей наносятся текстуры и выполняется интерполяция интенсивности цвета точек, улучшающая восприятие сформированного изображения. Весь процесс создания растрового изображения трехмерных объектов называется *рендерингом* (rendering).

Теперь вспомним о том, что трехмерное изображение хотелось бы «оживить» движением, для чего изображения объектов в новом положении должны схо

дить с графического конвейера со скоростью хотя бы 15 кадров в секунду (современные акселераторы могут строить и 100 кадров в секунду). Насколько сложна эта задача для обычного процессора семейства x86, можно прикинуть по такому занятному примеру, который автору доводилось наблюдать воочию: безобидная, казалось бы, экранная заставка из комплекта Windows NT 4.0, на которой развевается трехмерное изображение флага известной компании, при чисто программной реализации полностью загружала один из двух процессоров Pentium Pro-166 сервера под управлением ОС Windows NT-Server. Это легко было наблюдать с помощью монитора производительности (Performance Monitor) после установки на компьютер-сервер «простой» графической карты (а зачем, казалось бы, серверу мощная графическая карта?). А если взять более сложную экранную заставку («ходилка» по трехмерному лабиринту), то четырехпроцессорный (P6-200) сервер Proliant 500 не способен создать иллюзию хождения по лабиринту — кирпичи на стенах вырисовываются слишком медленно. В то же время сравнительно маломощный компьютер Pentium-166, у которого имеется графическая карта с 3D-акселератором, позволяет «бегать» по этому лабиринту довольно быстро. Это колоссальное ускорение построений обеспечивается применением в графических картах встроенного специализированного процессора, решающего значительную часть задач графического конвейера. Конечно, со времен Pentium Pro процессоры x86 прошли значительный путь: появились блоки MMX и XMM с соответствующими расширениями команд, да и тактовая частота уже в 20 раз выше. Однако и графические задачи стали сложнее: больше объектов, выше разрядность цветопередачи, выше требуемые скорости построений. Так что и ПК с мощными процессорами без акселераторов не обходятся. О том, что происходит при трехмерных графических построениях, и пойдет речь дальше.

Графическое приложение создает *модель*, в которой объекты задаются как совокупность тел и поверхностей. Тела могут иметь разнообразную форму, описанную каким-либо математическим способом. Проще всего иметь дело с многогранниками, у которых каждая грань представляет собой часть плоскости, ограниченной многоугольником (полигоном). Описание такого тела относительно несложно — оно состоит из упорядоченного списка вершин. Сложнее дело обстоит с объектами, имеющими не плоские (криволинейные) поверхности. В этом случае в модели поверхности описываются сложными нелинейными уравнениями, однако для дальнейших построений их использование из-за громадных объемов вычислений проблематично. Для упрощения задачи криволинейные поверхности аппроксимируются многоугольниками, и, конечно же, чем мельче многоугольники, тем ближе аппроксимация к модели, но и тем более громоздким становится описание объекта, а следовательно, и больше времени требуется на его обработку. Представление криволинейной поверхности совокупностью плоских граней-многоугольников называется *тесселяцией* (tes- selation). Слово «tessera», от которого произошел этот термин, означает кубики из смальты, из которых художники собирают мозаику. Как и смальтовые кубики, многоугольники-грани должны быть простыми (не пересекающими себя на манер цифры 8), плоскими и выпуклыми — эти ограничения заметно упрощают их дальнейшую обработку.

Помимо формы объектов (описания их поверхностей) важное значение имеют их оптические свойства. Проще всего дело обстоит с непрозрачными объектами — все другие объекты, перекрытые ими для взгляда наблюдателя, просто невидимы. Эти объекты будут перекрывать и лучи от источников освещения, установленных в модели, на пути которых они встречаются. Сложнее дело обстоит с прозрачными и просвечивающимися объектами. *Прозрачность* (transparency) объекта означает возможность видеть объекты, расположенные за ним, а *просвечиваемость* (translucency) — возможность проходить через него свету от источников. Поверхность имеет некоторый *цвет*, а также характеризуется *степенью отражения* (она может быть глянцевой или матовой). Для того чтобы получить реалистичное отображение модели, приходится отслеживать прохождение лучей от установленных источников освещения, достигающих глаза воображаемого наблюдателя как при отражении от поверхностей, так и при преломлении в ходе прохождения через прозрачные и просвечиваемые объекты. При этом должны учитываться *эффекты перспективы*, как *оптической* (искажение формы), так и *атмосферной* (имитация дымки или тумана).

Рендеринг

Не претендуя на полноту объяснения всех хитростей технологии создания трехмерных изображений, обсудим некоторые ее основные моменты и поясним связанные с ними термины.

Вполне понятно, что рендеринг модели может производиться только поэлементно. Результатом тесселяции является набор многоугольников (обычно четырехугольников или треугольников, которыми манипулировать проще), аппроксимирующих поверхности объектов. Плоское растровое представление должно формироваться с учетом взаимного расположения элементов (их поверхностей) — те из них, что ближе к наблюдателю, естественно, будут перекрывать изображение более удаленных элементов. Многоугольники, оставшиеся после удаления невидимых поверхностей, сортируются по глубине: реалистичную картину удобнее получать, начиная обработку с наиболее удаленных элементов. Для учета взаимного расположения применяют так называемый Z-бу*фер*> названный по имени координаты третьего измерения *(X* и *Y* — координаты в плоскости экрана). Этот буфер представляет собой матрицу ячеек памяти, каждая из которых соответствует ячейке видеопамяти, хранящей цвет одного пиксела. В процессе рендеринга для очередного элемента формируется его растровое изображение, и для каждого пиксела этого фрагмента вычисляется параметр глубины Z (координатой его можно назвать лишь условно). В видеопамять этот фрагмент поступает с учетом результата попиксельного сравнения информации из Z-буфера с его собственными значениями. Если глубина Z данного пиксела фрагмента оказывается меньше величины Z той ячейки видеопамяти, куда должен попасть этот фрагмент, это означает, что выводимый элемент располагается ближе к наблюдателю, чем ранее обработанные элементы, отображение которых уже находится в видеопамяти. В этом случае выполняется модификация пиксела видеопамяти, а в ячейку Z-буфера видеопамяти помещается новая величина, взятая от данного фрагмента (что подразумевается под

модификацией, поясним позже). Если же результат сравнения иной, то текущий пиксел фрагмента оказывается перекрытым прежде сформированными элементами и его параметр глубины в Z-буфер не попадет. Однако цвет пиксела видеопамяти, возможно, все равно придется модифицировать, поскольку перекрывающий элемент может оказаться прозрачным. Итак, Z-буфер позволяет определить взаимное расположение текущего и ранее сформированного пикселов, которое учитывается при формировании нового значения пиксела в видеопамяти. От разрядности Z-буфера зависит разрешающая способность графического конвейера по глубине. При малой разрядности (например, 8 бит) для близко расположенных элементов рассчитанные значения Z могут совпасть, в результате картина перекрытий исказится. Большая разрядность буфера требует большого объема памяти, доступного графическому процессору. По нынешним меркам минимальная разрядность Z-буфера — 16 бит, профессиональные графические системы используют 32-битный Z-буфер.

Теперь обсудим модификацию цвета пиксела видеопамяти. В общем случае у нас есть два значения цвета — C_1 для того образа, который «ближе», и C_2 для того, что «дальше» (по Z-параметру). Результирующий цвет определяется обоими значениями и свойством «прозрачности» ближнего. Для получения нового значения цвета обычно используют так называемый *альфа-блендинг* (alpha- blending). Мерой прозрачности объекта является коэффициент a ($0 \le a \le 1$), единица соответствует полной непрозрачности. Результирующий цвет пиксела вычисляется по формуле $C = C_1 \cdot \alpha + C_2 \cdot (1 - \alpha)$, причем за этой формулой стоит в три раза больше операций, поскольку цвет определяется тремя значениями базисных цветов (R, G и B). Ну и нетрудно догадаться, что для реализации данного метода требуется и свой *альфа-буфер с* количеством ячеек, по меньшей мере равным числу пикселов на экране. Часто 8-битный коэффициент прозрачности для каждого пиксела хранят прямо в видеопамяти: при 24-битном кодировании цвета от двойного слова (32 бит), выделяемого на пиксел для упрощения адресации и ускорения обмена, как раз остается 8 бит. Такой формат видеопамяти называют RGBA.

Объекты, входящие в модель, и представляющие их элементы (тессели), не обязательно однородны по цвету: на их поверхности могут быть наложены *текстуры* — растровые картинки, исходно плоские, но как бы к ним приклеенные. Текстура состоит из элементов, называемых *текселами* (texel — texture element). Здесь уместна аналогия с созвучным термином *пиксел* (pixel — picture element), который относится к элементу изображения на экране и его образу в видеопамяти. Текстуры (в виде матриц текселов) хранятся в памяти. Для каждого многоугольника-частицы отображаемой поверхности вычисляется соответствующий ему участок текстуры — тоже многоугольник. Далее этот участок должен быть отображен в видеопамять, то есть тексель должны быть отображены в пикселы. Что должно происходить с рисунком текстуры при изменении положения плоскости, на которую она наносится, легко представить, повертев перед глазами спичечный коробок и наблюдая за этикеткой. Помимо искажения формы при поворотах учитываются изменения размера картинки текстуры при приближении и удалении объекта от наблюдателя, а также перспектива. Мае-

штабирование и повороты текстур могут приводить к различным искажениям: к примеру, увеличенное и повернутое изображение гладкого горизонтального (или вертикального) отрезка превращается в грубую ступенчатую линию. Кроме того, могут появляться «рваные» края у текстур по линиям их сопряжения. Для улучшения качества представления одной и той же текстуры в разном масштабе применяют набор нескольких версий одной и той же текстуры, выполненных с различным разрешением (обычно очередная версия имеет размер в четверть от предыдущей), — так называемый набор *MIPmap*. При рендеринге выбирается та версия, у которой масштаб ближе к требуемому. Дефекты, обусловленные растровым представлением текстуры (векторные изображения, в отличие от растровых, масштабируются и трансформируются без потери информации), могут быть устранены путем фильтрации — билинейной или более сложной, трилинейной. При *билинейной фильтрации* (bilinear filtering) цвет очередного пиксела, записываемого в видеопамять, определяется с учетом цветов прилегающих к нему четырех соседних пикселов. *Трилинейная фильтрация* (trilinear filtering) сложнее — здесь билинейная фильтрация выполняется дважды для двух соседних уровней MIPmap, ближайших к требуемому масштабу. Окончательный цвет пиксела определяется интерполяцией этих двух результатов.

Наложение текстур при всех хлопотах, связанных с его реализацией, позволяет упростить описание объектов и ускорить их рендеринг. Так, фасад кирпичного здания можно построить «по-честному», задав поверхности всех кирпичиков, оконных и дверных проемов и т. п. Но если нужно получить это изображение, например, в игре, где воображаемый наблюдатель должен приближаться и удаляться от стены, а также менять угол зрения довольно быстро, то проще представить ее одной плоскостью с «нарисованными» кирпичами и прочими деталями (вспомним каморку папы Карло). На одни и те же объекты часто накладывают несколько текстур для имитации освещенности, теней, отражений, рельефа и т. д.

И наконец, когда все объекты, расположенные на сцене, уже прорисованы, для лучшей имитации объема можно ввести эффект атмосферной перспективы — сильно удаленные объекты подернуть дымкой (туманом). Это несложно сделать, используя для попиксельного смешения цветов тумана и объектов информацию о глубине из Z-буфера: чем больше Z, тем больше на результирующий цвет влияет туман и меньше цвет исходного пиксела.

В последнее время стали использовать также трехмерные текстуры (3D textures) — трехмерные массивы пикселов. Они позволяют, например, имитировать объемный туман, динамические источники света (языки пламени).

Реализация трехмерной графики

Теперь примерим задачи трехмерной графики к возможностям рядового PC-совместимого компьютера. Реализация рендеринга требует значительного объема вычислений и оперирования большими объемами информации, причем конечная цель потока обработанных данных — видеопамять графического адаптера. Еще с 2D-графикой стало ясно, что центральный процессор x86 (даже с расши

рением MMX) не способен быстро формировать движущиеся изображения, а шина расширения (даже PCI) является «узким горлышком» для потоков данных, участвующих в видеопостроениях. Решением проблемы вывода трехмерной графики, как и раньше, явилось усиление «интеллекта» графической карты — появились 3D-акселераторы, реализующие значительную часть графического конвейера. На долю центрального процессора (возможно, с графическим сопроцессором) обычно выпадает начало конвейера (от трансформации до тесселяции), а его окончание (растеризация) выполняется акселератором графической карты.

Как ни странно, основным двигателем прогресса 3D-технологий являются игры — именно любители компьютерных игр стали главными (самыми массовыми) потребителями 3D-акселераторов. Более «серьезные» применения движущейся трехмерной графики — различные тренажеры-имитаторы полетов и езды — по сути тоже являются играми, только для серьезных людей. Трехмерная анимация, применяемая в современном телевидении и кинематографии, пока что реализуется не на массовых персональных компьютерах, а на более мощных рабочих станциях, но и там используются практически все вышеописанные элементы технологии.

Технологии построений, выполняемых 3D-акселераторами, постоянно совершенствуются, и описать все применяемые приемы в данной главе просто невозможно. Все новшества нацелены на создание фотореалистичных изображений игровых сцен с большой скоростью смены кадров (до 100 кадров/с) на экранах с большим разрешением (до 2048 x 1536) и в полноцветном режиме (True Color, 32 бита на пиксел). Конечно же, эти цели достигаются не ускорением «честных» расчетов для каждого элемента модели, а разными «обманными» приемами вроде текстур. Новинки технологий, применяемых в акселераторах, дадут видимый (в прямом смысле) эффект, если их будут знать и использовать соответствующие приложения (игры) и поддерживать драйверы, через которые приложения взаимодействуют с аппаратными средствами графической карты. Именно драйверы отвечают за распределение обязанностей между центральным процессором и графическим процессором акселератора, и их корректность определяет качественные и количественные параметры графического конвейера. Много интересных подробностей работы 3D-акселераторов можно узнать на сайте iXBT.com, где публикуются обзоры новинок, а также статьи по теории трехмерных построений.

Мощность акселератора определяется списком реализуемых функций рендеринга и качеством их выполнения, а также производительностью, измеряемой как по входу, так и по выходу. Поскольку на входе акселератор оперирует многоугольниками-тесселями, то интерес представляет их количество, обсчитываемое за единицу времени. Для определенности (и более внушительного абсолютного значения) берут параметр *Mts* (Mega Triangle per Second — миллионов треугольников в секунду). По выходу определяют скорость формирования пикселов в видеопамяти *Mps* (Mega Pixel per Second — миллионов пикселов в секунду). Интересна и скорость формирования кадров, *Fps* (Frames per second —

кадров в секунду), правда, при ее указании нужно оговаривать сложность кадров.

Трехмерные акселераторы для компьютерных игр на PC первой внедрила фирма 3dfx, до того занимавшаяся игровыми приставками, и ее название (3dfx) стало нарицательным именем 3D-акселератора. За несколько лет эта фирма выпустила серию акселераторов, большинство из которых называются Voodoo с различными номерами и добавлениями.

Сейчас (в 2005 году) основными производителями 3D-карт являются фирмы ATI (семейство карт Radeon) и NVIDIA (семейство карт GeForce). Карты выпускаются как для порта AGP (режим 8x), так и для PCI Express 16x. Порт AGP на системной плате только один, так что установить две мощные графические карты не удается. Поскольку слотов PCI Express 16x может быть несколько (уже есть системные платы с двумя такими слотами), появляется возможность установки пары карт. Это позволяет повысить производительность графического конвейера путем распараллеливания задач между двумя акселераторами. Распараллеливание может идти несколькими путями:

- Разбивка кадра на чередующиеся строки: один акселератор обрабатывает четные строки, другой — нечетные.
- Разбивка кадра на непересекающиеся зоны по вертикали, не обязательно пополам. Каждый акселератор занимается только своей областью, выполняя всю работу (и построение геометрии, и закрашивание пикселов), однако возникают сложности закраски на стыке зон. Для выравнивания нагрузки можно динамически изменять размер зон (что непросто).
- Разбивка кадра на чередующиеся полосы или области в шахматном порядке. Каждый акселератор выполняет геометрическую работу для всего кадра, а закраску — только для своих полос (клеток). Таким образом, удваивается производительность только последней части графического конвейера. Чтобы картинка не рассыпалась, акселераторы должны функционировать синхронно (быть однотипными).
- Чередование кадров — каждый акселератор строит свой кадр целиком один: — четные кадры, другой — нечетные. Смена отображаемых кадров (по мере их готовности) выполняется под управлением центрального процессора, возможна неравномерность смены кадров.

Результаты параллельной обработки должны объединяться специальным блоком и выводиться на один монитор (через один интерфейс). Фирмы ATI и NVIDIA организуют объединение по-разному.

Технология SLI (ScanLine Interleaving), применявшаяся в картах Voodoo2 фирмы 3dfx, распределяла работу построчно. Данные, поступавшие в аналоговом виде по кабелю VGA-Loop, объединялись в буфере первичной карты. Для синхронизации использовался специальный кабель-шлейф SLI.

В *технологии SLI* (Scalable Link Interface) фирмы NVIDIA имеет место обмен данными между картами по магистрали (PCI-E). В ведущей (master), или первичной (primary), карте организуется буфер для сборки итогового изображения. Для синхронизации используется специальный переходник — маленькая

печатная плата, одеваемая на объединяемые карты. Распределение работы может вестись методом разбивки на зоны или чередованием кадров. Объединяемые карты должны быть идентичными.

Технология CrossFire фирмы ATI позволяет для параллельной работы использовать специальную видеокарту в паре с любой, в которой имеется цифровой выход — интерфейс DVI. Распараллеливание может выполняться разными способами: разбивкой на зоны, чередованием полос и блоков, чередованием кадров. На специальной карте имеются блок Composing Engine, объединяющий результаты работы акселераторов, и специфический разъем DMS, к которому подключается специальный кросс-кабель. На входы объединяющего блока поступают цифровые потоки (сигналы T.M.D.S.), представляющие информацию экранных буферов обоих акселераторов. Выходной поток этого блока, являющийся результатом объединения, через интерфейс T.M.D.S. (для выхода DVI) и через интерфейс RAMDAC (для выхода VGA) поступает на разъем DMS. Кросс-кабель соединяет разъем DMS с вилкой DVI, подключаемой ко второй графической карте, и розеткой (DVI или VGA) для подключения кабеля монитора. На карте CrossFire присутствует и дополнительный выход (DVI, VGA, TV-Out), который может использоваться для вывода отдельного изображения (конечно, когда отключен режим параллельной работы). Также может быть доступен (если имеется) второй выход второго адаптера.

Память для графического акселератора

Для работы акселератору требуется довольно много памяти: пара буферов экрана (во время отображения одного буфера в другом строится новый кадр), Z-бу- фер, а-буфер (может вписываться в видеопамять) и память для хранения текстур (да еще и во многих экземплярах для MIPmap). Специализированный процессор акселератора, типовая разрядность которого 128-256 битов (а начинали с 32-64-битных), должен иметь доступ к памяти со скоростью, измеряемой уже десятками гигабайт в секунду. Наивысшая производительность достижима для локальной памяти, установленной на графическом адаптере. Самые быстрые (на весну 2005 г.) микросхемы GDRAM3 с тактовой частотой 500-800 МГц при разрядности 256 битов (32 байта!) обеспечивают пропускную способность локальной памяти 32-51,2 Гбайт/с.

Одной только локальной памятью акселератор обходиться не может (хотя ее объем уже достигает 256 Мбайт): требуется связь и с системной памятью (ОЗУ) компьютера. Для обеспечения высокопроизводительной связи с ОЗУ акселератора был разработан порт AGP (см. 14.9). Пропускная способность порта AGP повышалась от версии к версии: AGP 1.0 в режиме 2x — 533 Мбайт/с, AGP 2.0 (4x) — 1,066 Гбайт/с, AGP 3.0 (8x) — 2,132 Гбайт/с. Теперь на смену AGP приходит шина PCI-E (см. 14.10), слоты которой могут обеспечивать скорость записи или чтения от 1 Гбайт/с (слот 4x) до 4 Гбайт/с (слот 16x). Слоты 32x (8 Гбайт/с) пока что на системных платах не устанавливают. При этом на PCI-E возможен полнодуплексный режим выполнения операций, что теоретически удваивает пропускную способность интерфейса. Однако память, к которой обращается акселератор, одновременно записывать и читать данные не

может. Шина PCI для подключения 3D-акселератора уже давно не используется — мала пропускная способность и низка эффективность.

Какой бы высокопроизводительной ни была шина подключения акселератора, производительность его обмена с ОЗУ ограничивается и параметрами собственно подсистемы памяти. Ее производительность тоже стремятся повышать. Самая быстрая память DDR2 SDRAM PC2-6400 обеспечивает пропускную способность 6,4 Гбайт/с в одноканальном варианте и 12,4 Гбайт/с в двухканальном. Однако это пиковая скорость передачи, средняя всегда заметно ниже. Системной памятью пользуются еще и центральный процессор, а также контроллеры периферийных устройств (дисков, локальной сети, шин PCI и USB). Так что ее производительность делится между многими потребителями ресурсов. Кроме того, чем дальше находится память от ее «клиента» (и по длине проводников, и по количеству промежуточных узлов), тем больше задержки доступа. Так что локальная память в любом случае быстрее.

Если главная цель — не максимальная производительность, а минимальная цена, то от локальной памяти в графическом адаптере можно отказаться. Это и реализовано в архитектуре однородной памяти (Unified Memory Architecture, UMA), идея которой появилась в 90-х годах. Вариации на тему UMA — технологии HyperMemory (ATI) и TurboCache (NVIDIA). Здесь акселератор работает с системным ОЗУ как с видеопамятью, используя ее и для буфера кадра. Однако даже пропускная способность PCI-E x16 не позволяет достичь высоких скоростей при трехмерных построениях, требуемых для игр.

Одной из целей введения порта AGP в архитектуру ПК было предоставление акселератору возможности пользоваться системной памятью для хранения текстур. При этом снимается ограничение на размер описания текстур, которые без AGP приходилось держать в ограниченном объеме локальной памяти. Порт AGP позволяет акселератору работать в двух режимах — DMA и DIME (Direct Memory Execute). *В режиме DMA* акселератор при вычислениях рассматривает видеопамять как первичную, а когда ее недостаточно, подкачивает в нее данные из основной памяти. В *режиме DIME* (он же режим исполнения — Executive Mode) видеопамять и основная память для акселератора логически равнозначны и располагаются в едином адресном пространстве. При этом требуются определенные меры по отображению локальной памяти в пространство системной памяти (чтобы к ней могли обращаться программы центрального процессора) и, наоборот, участков системной памяти в пространство, доступное локальному процессору. В режиме DMA для трафика порта характерны длительные блочные передачи, а в режиме DIME трафик порта насыщен короткими произвольными запросами. Акселератору для работы в режиме DIME нужна непрерывная область доступной памяти, часть которой составляет его локальный буфер. Остальная часть адресуемой им памяти отображается на системное ОЗУ с помощью *таблицы GART* (Graphics Address Remapping Table — таблица переопределения графических адресов). В этой таблице каждой странице графической памяти ставится в соответствие своя страница системного ОЗУ (рис. 10.3); таким образом, «видение» виртуальной памяти для программы, выполняемой центральным про

цессором, согласуется с «видением» памяти программы, выполняемой процессором-акселератором. Это важно, поскольку задача графических построений решается этими процессорами совместно. Область физических адресов памяти, доступной акселератору через такое страничное преобразование, называется *апертурой AGP*. Ее размер задается при программировании регистров чипсета путем настройки параметров CMOS Setup или внешних утилит и может составлять 8, 16, 32,..., 256 Мбайт (и больше). Оптимальное значение апертуры зависит от объема памяти и используемых программ, но можно ориентироваться на половину объема системного ОЗУ.

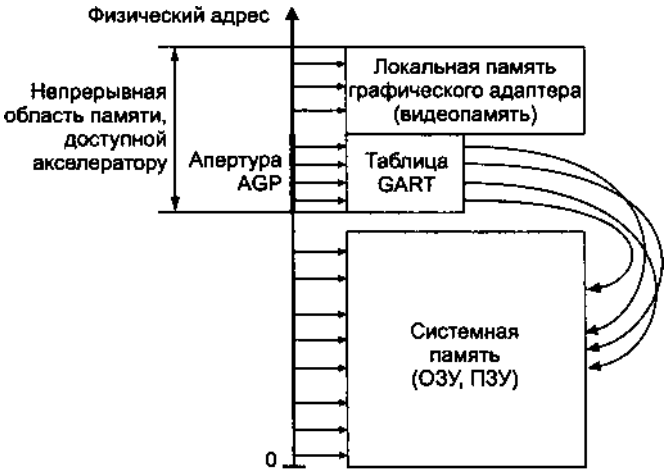


Рис. 10.3. Адресация памяти в системе с AGP

Поскольку GART ссылается на области, принадлежащие расширенной памяти, для операционных систем реального режима (MS-DOS) использование DIME может оказаться проблематичным: эти ОС не рассчитаны на нового «стороннего» потребителя расширенной памяти. В ОС защищенного режима с виртуальной адресацией памяти этой проблемы нет.

Механизм переадресации памяти через GART является частью логики порта AGP, его физическая реализация не стандартизована (ее должен «знать» драйвер порта AGP). Сама таблица GART может располагаться в системном ОЗУ, и в AGP 1.0 и 2.0 она была способна отображать лишь страницы фиксированного размера (4 Кбайт). В спецификации AGP 3.0 появилась возможность для GART работать со страницами большего размера, а также разрешать определенным страницам кэширование памяти (что усложняет логику порта). Поскольку обращения мастера AGP не отслеживают состояние кэш-памяти, область памяти, на которую указывают записи таблицы GART, объявляют некэшируемой. Обращения к некэшируемой памяти со стороны центрального процессора выполняются медленнее, так что программы центрального процессора должны избегать размещения обрабатываемых данных в этих областях. Акселератор, в принципе, может адресовать команды AGP и вне области GART; эту ситуацию

может отслеживать логика порта и сообщать об ошибке. Однако этот контроль не обязателен, и в системе, где он реализован, по умолчанию (после аппаратного сброса) он должен быть выключен. Реакция на обращения к области GART со стороны CPU или мастеров «чистой» шины PCI зависит от логики порта: либо они отвергаются, либо выполняются с той же переадресацией (как от CPU, так и от PCI, чтобы карта распределения физических адресов была единой для всех агентов).

Таблица GART (и апертура AGP) — это особое «достояние» исключительно порта AGP, специального инструмента подключения графического акселератора. С переходом на PCI-E архитектура подключения всей периферии, включая и акселератор, унифицируется, и никаких средств трансляции на системной плате не полагается. В первое переходное время для подключения к PCI-E акселераторов, ориентированных на порт AGP, приходилось использовать специальные мосты (RIALTO у ATI и HIS у NVIDIA), устанавливаемые на графической карте.

В «чистопородных» акселераторах для PCI-E проблема согласования видения памяти и более сложные задачи управления страницами памяти решаются помещением на акселератор блока управления памятью (Memory Management Unit, MMU), по смыслу напоминающего одноименный блок современных процессоров. Таким образом, структура акселератора усложняется и понемногу приближается к структуре центрального процессора.

10.3. Дисплей

Самым главным устройством вывода визуальной информации в PC является *дисплей* (display — устройство отображения). Дисплей может быть основан на различных физических принципах: здесь применимы электронно-лучевые трубки, газоплазменные матрицы, жидкокристаллические индикаторы и другие приборы. Наибольшее распространение получили дисплеи на электронно-лучевых трубках и жидкокристаллических матрицах, которым и уделим здесь основное внимание.

Электронно-лучевой дисплей

Электронно-лучевая трубка (ЭЛТ) по-английски сокращенно называется *CRT* (Cathode Ray Tube — катодно-лучевая трубка). Иногда аббревиатуру CRT расшифровывают и как Cathode Ray Terminal, что соответствует уже не самой трубке, а устройству, на ней основанному (монитору на катодно-лучевой трубке). Вместо сокращения ЭЛТ в нашем обсуждении можно использовать и название *кинескоп* — это ЭЛТ с электромагнитной системой отклонения луча, которая характерна и для телевизионных, и для компьютерных мониторов. Первые дисплеи на ЭЛТ появились еще до PC, и в них помимо ЭЛТ с окружающими ее схемами генераторов развертки и видеоусилителей находились и узлы, формирующие изображение (чаще — алфавитно-цифровое). Такие дисплеи применяются и сейчас как терминалы многопользовательских машин (например, сис-

тем UNIX). В персональных компьютерах узлы, формирующие изображение, «переехали» в системный блок, в результате дисплей функционально упростился и стал похож на монитор, применяемый в телевидении. *Монитор* содержит только ЭЛТ с видеоусилителями сигналов яркости лучей, генераторы разверток, блок питания и схемы управления этими узлами. Традиционный телевизионный *монитор* имеет низкочастотный вход композитного видеосигнала или/и раздельные входы модуляции лучей и рассчитан на работу в стандартах PAL, SECAM или NTSC, определяющих способы цветопередачи и фиксирующих частоты синхронизации. *Монитор компьютера* должен обеспечивать существенно более широкую полосу пропускания видеосигнала, поэтому композитный вход для него неприемлем. Кроме того, этому монитору приходится работать с разными параметрами синхронизации, которые зависят от выбранного режима разрешения и требований к развертке. Параметры синхронизации могут меняться в процессе работы, и компьютерный монитор должен отрабатывать эти переключения режимов.

В *монохромных мониторах* экран трубки покрыт однородным слоем мелкозернистого люминофора, который при хорошей фокусировке луча дает высокую четкость и разрешающую способность, фактически, определяемую лишь параметрами генераторов разверток. В *цветных мониторах* люминофор неоднороден — имеются три типа частиц, каждый из которых дает свечение своим базисным цветом. Соответственно, имеются три электронные пушки, каждая из которых «обстреливает» только свои частицы люминофора. Лучи всех трех пушек синхронно сканируют экран. Управляя интенсивностью каждого из лучей, получают требуемый цвет изображения каждой точки. Существует ряд технологий ЭЛТ, различающихся способом наведения лучей на свои частицы люминофора.

Классической является ЭЛТ с *теневой маской* (shadow mask). Ее экран покрывается не сплошным люминофором, а отдельными зернами-триадами, расположенными треугольником. Каждое зерно состоит из трех крупиц люминофора, которые при попадании на них потока электронов светятся базисными цветами. Крупицы триад имеют строго фиксированное относительное расположение, и сами триады наносятся на поверхность в виде равномерной матрицы. Крупицы каждого цвета «обстреливаются» из отдельной электронной пушки через теневую маску с отверстиями, соответствующими зернам матрицы. Точность попадания лучей в свои крупицы обеспечивается тщательностью изготовления кинескопа и настройкой системы сведения лучей. Шаг матрицы зерен экрана *(dot pitch)* принято измерять в миллиметрах. В первом приближении можно считать, что он совпадает с размером зерна. Однако отождествлять эти два параметра не очень корректно, и термин «dot pitch» лучше перевести как зернистость экрана, но не размер зерна. Недостатком теневой маски является ее низкая относительная прозрачность, что снижает энергию луча, достигающего люминофора. В результате изображение не очень яркое и насыщенное. Однако теневая маска обеспечивает самый «круглый» пиксел, благодаря чему изображение мелких деталей самое четкое. Поскольку электронные пушки цветов RGB располагаются треугольником (зеркально по отношению к триадам люми

нофора), мониторы с теневой маской имеют экран, выпуклый и по вертикали, и по горизонтали. Это не очень удобно (трудно избежать бликов).

В ЭЛТ со *щелевой маской* (slot mask) вместо отверстий в маске имеются вертикальные щели, а цветной люминофор наносится чередующимися полосами (тоже вертикальными). Прозрачность выше — следовательно, цвета более яркие и насыщенные. Однако пикселы получаются немного вытянутыми по вертикали. Пушки располагаются в одной горизонтальной плоскости, что позволяет сделать экран выпуклым только по горизонтали (по вертикали его поверхность прямолинейна).

В ЭЛТ с *апертурной решеткой* (aperture grilles) люминофор тоже нанесен вертикальными полосами, но в качестве маски в них используются вертикально натянутые нити, выстроенные «частоколом». Маску поддерживает одна или несколько горизонтальных проволочек, тень от которых заметна на экране (дань технологии). У 15-дюймовых мониторов проволочка одна, она расположена снизу на высоте примерно 1/3 экрана. У мониторов большего размера их может быть 2-3. Яркость и насыщенность цветов наилучшая, но четкость пикселов хуже, чем у щелевой и, тем более, у теневой маски. Экран таких трубок плоский.

Из рассмотренных трех типов трубок трубки с теневыми масками больше всего подходят для задач САПР (и обработки текста), трубки с апертурными решетками — для художественной графики и мультимедийных систем (наилучшая цветопередача). Щелевая маска — компромиссный вариант.

Параметры монитора

Мониторы подразделяются на монохромные (monochrome, или mono) и цветные (colour, или color). Монохромные мониторы практически вышли из употребления.

Цветные мониторы получили наибольшее распространение. Первые цветные мониторы, имеющие цифровой интерфейс, использовались с адаптерами CGA и EGA. Мониторы CGA работали на частотах, близких к телевизионным, и некоторые умельцы подключали вместо них цветные телевизоры. Однако по качеству изображения телевизор обычно уступает монитору. Мониторы EGA имели возможность переключения частот развертки и обеспечивали довольно высокое качество изображения. В настоящее время распространены мониторы классов VGA и SVGA, имеющие аналоговый или/и цифровой интерфейс. Мониторы VGA, допускающие работу в режиме 640 x 480, вытеснены мониторами класса SVGA, которые должны поддерживать по крайней мере режим 800 x x 600. Именно об этих мониторах в основном и пойдет речь.

Главным параметром монитора является *размер диагонали экрана* (screen size), который принято измерять в дюймах. По умолчанию считается, что ширина экрана больше его высоты и соотношение этих размеров составляет 4:3. Такую ориентацию можно назвать «пейзажной» (landscape), хотя это определение обычно опускают. Заметим, что стандартные графические режимы с высоким разрешением (640 x 480, 800 x 600 и далее) имеют то же соотношение числа то

чек в строке и числа строк. Этим достигается неискаженное изображение фигур: квадрат на экране будет иметь стороны с одинаковым числом пикселов. Существуют и мониторы с «портретной» (portrait) ориентацией, у которых высота больше ширины. Это вовсе не повернутые на бок обычные мониторы, поскольку строки развертки у них остаются горизонтальными. Данный тип монитора предназначается для издательских систем и позволяет более полно использовать площадь экрана при выводе книжных страниц. В настоящее время «портретные» мониторы встречаются редко, а в издательской деятельности чаще применяют «просто» большие мониторы (19^{м}, 21^{м} и больше). Размеры экранов приведены в табл. 10.2. Заметим, что указанный размер диагонали не является размером изображения, выводимого с гарантированным уровнем качества. По краям экрана (особенно по углам) возможны геометрические искажения, нарушение фокусировки и сведения лучей. По этим причинам изображение (видимая часть растра) выводится на меньшую площадь. Так, для экрана 15^{м} размер видимой (высококачественной) части изображения может составлять, например, 13,7^{м}. Если изготовителю монитора удается добиться почти полного использования поверхности, он не забывает упомянуть в рекламе эту особенность. Иногда случается, что производитель, добившийся лучшего использования углов, укажет завышенный размер диагонали экрана, определяющий продажную цену монитора.

Таблица 10.2. Размеры экрана мониторов

Диагональ, дюймов	Размер изображения, мм		Разрешение	
	по горизонтали	по вертикали	максимальное	рекомендуемое
14	254-264	190-200	1024x768	640 x 480
15	274-284	205-215	1280x1024	800 x 600
17	315-325	237-245	1600x 1200	1024x768
19	355-365	267-275	1600x1200	1280 x 1024
21	396-406	298-306	1600x 1200	1400 x 1050
24	436-447	328-336	1900x1200	1600x1200

Для цветных мониторов важным параметром является *размер зерна* экрана. Существуют мониторы с зернистостью 0,42, 0,39, 0,31, 0,28, 0,26 мм и меньше.

По зернистости и размеру экрана можно определить «честную» разрешающую способность экрана, поскольку зерно является мельчайшей единицей изображения. Количество зерен в строке равно ширине рабочей области, деленной на шаг зерна. Однако (может, для сокрытия реальной картины?) размер экрана задают по диагонали, а не как ширину и высоту, причем указывается внешний размер, а не размер рабочей области, к тому же в дюймах, а не в миллиметрах. Кроме того, для мониторов с теневой маской зернистость определяет шаг триад по диагонали, а для щелевой маски или апертурной решетки — по горизонтали. Так что пользователю, которого утомляют длинные пересчеты, остается поверить, что для режима 800 x 600 зернистость 0,28 мм экрана 14^{м} (с теневой маской) является приемлемой. А вот для режима 1024 x 768 при такой же зернистости только-только хватает экрана размером 15^{м}. Конечно, никто не запретит

использовать режимы с большим разрешением на небольших или/и крупнозернистых мониторах установив приемлемую для них частоту синхронизации, но качество отображения мелких элементов будет оставлять желать лучшего. В результате работа (не игра) на таком мониторе может сильно утомлять и даже вести к ухудшению зрения.

Допустимая частота развертки определяется в основном параметрами отклоняющей системы и мощностью генератора строчной развертки. В соответствии с нормами TCO 99 минимальная частота регенерации (вертикальной развертки) должна составлять 85 Гц в любом режиме, а рекомендуемая — 100 Гц. Для обеспечения прогрессивной (не чересстрочной) развертки в режимах с высоким разрешением (большим числом строк) требуется очень высокая частота строчной развертки. Так, для режима 1024 x 768 при частоте регенерации 85 Гц строчная частота должна быть порядка 70 КГц, а для режима 1600 x 1200 при частоте регенерации 100 Гц — 126 КГц.

На реальную разрешающую способность существенно влияет *полоса пропускания видеотракта* (video bandwidth). Ее связь с выбранным видеорежимом (количество точек и строк) и параметрами развертки (частота и режим) была показана выше. При недостаточно широкой полосе пропускания мелкие детали — точки и вертикальные линии толщиной в один пиксел — могут становиться нечеткими и даже незаметными. В технических данных монитора обычно указывают предельное разрешение и максимальные частоты разверток. Однако это вовсе не означает, что максимальное разрешение можно использовать на максимальной частоте, да еще и при прогрессивной развертке. Оценить предел возможностей позволяет полоса пропускания. Заботливый производитель, конечно, избавит пользователя от решения таких головоломных задач и приведет таблицы оптимальных настроек для всех режимов (если ему нечего стесняться). Грубо требуемую полосу пропускания (*BW*, Гц) можно оценить через число точек в строке (Я), число строк (V) и частоту вертикальной развертки (F, Гц):

$$BW = k \times H \times V \times F.$$

Поправочный коэффициент k = (1,3...1,4) учитывает «простои» вывода точек на обратном ходе по строке и кадру. Для чересстрочной развертки в формулу подставляется половина частоты развертки. Так, для прогрессивной развертки (NI) с частотой кадров 75 Гц в режиме 800 x 600 требуется полоса 45 МГц, для 1024 x 768 — 75 МГц, а для 1280 x 1024 — 125 МГц. Чем больше размер экрана, тем больше должна быть полоса пропускания, поскольку чем больше экран, тем большего от него требуют разрешения. Так, по самым жестким меркам высококачественный монитор 14" должен иметь полосу 65 МГц, 15" — 100 МГц, а 17" — более 135 МГц.

На некоторые мониторы с полосой пропускания более 125 МГц устанавливают BNC-разъемы для подачи видеосигналов (дополнительно с DB-15 или вместо них), или же интерфейсный кабель делают неотсоединяемым. На больших мониторах применяют и интерфейс DVI, снимающий проблемы качества разъемов.

Настройка цветопередачи

Яркость (brightness) и контрастность (contrast) изображения обычно регулируют с помощью органов управления, расположенных на лицевой панели монитора. Иногда пользователю дают возможность регулировки баланса базисных цветов, но для верного воспроизведения цвета (в режимах High Color и True Color) такая регулировка может оказаться и вредной. В высококачественных мониторах предусматривают возможность регулировки *цветовой температуры* (colour temperature) белого цвета — вручную или через канал DDC. Цветовую температуру определяют через цвет свечения раскаленного железа. Обычные лампы накаливания дают «белый» цвет с температурой около 3000 К, и этот свет нам кажется желтоватым. Люминесцентные лампы дневного света дают цветовую температуру около 10 000 К, и этот свет кажется голубоватым. «Истинно белый» цвет имеет некоторую промежуточную температуру. Kodak, например, для цветной фотопечати принимает за белый цвет с температурой 5300 К. В мониторах используют более высокие значения — 6500 и даже 9300 К. Произвольное значение температуры белого цвета можно задать балансировкой яркости двух цветов (красного и синего) относительно фиксированного уровня зеленого.

Для настройки точной цветопередачи монитора применяют *гамма-коррекцию.* У аналогового монитора на ЭЛТ передаточные характеристики цветовых каналов нелинейные и имеют вид, аппроксимируемый функцией У = ku^{y} + y, где u — входной сигнал, а y ≈ 0,75...3,75 (по умолчанию 2,2) — это и есть «гамма». Поскольку в аппроксимирующей функции всего три параметра (гамма y, крутизна *k* и смещение *y*), то калибровка должна проводиться по трем точкам: черного, белого и 50-процентного серого. Для калибровки используется специальная утилита (например, Adobe Gamma) или вкладки окон свойств графики. Для каждого шага коррекции имеется своя тестовая картинка. Сначала на мониторе устанавливаются максимальная контрастность и минимальная яркость. Далее определяют точку черного: смещение *y,* при котором небольшое приращение и относительно нулевого значения дает заметное осветление изображения. Этого добиваются плавным повышением яркости (регулировкой монитора), пока на тестовой картинке не появится серый фрагмент (цвет 1, 1, 1) на черном фоне (0, 0, 0). После этого яркость не трогают и переходят к подбору значения y. Для этого тестовая картинка содержит область, залитую 50-процентным серым (цвет 127, 127, 127), и область с чередующимися тонкими, одинаковыми по площади полосами 0 % (черный) и 100 % (белый). Изменению значения y соответствует изменение значений, загружаемых в RAMDAC графической карты (см. 10.5). В качестве значения y принимают то, при котором равномерный 50-процент- ный серый фрагмент сливается с полосатым. Гамма-коррекцию можно выполнять и для каждого базисного цвета в отдельности.

Для точной настройки цветопередачи драйверу нужно помимо гамма-коррекции передать информацию о выбранной цветовой температуре (ее выбирать надо до коррекции), а также указать хроматические характеристики люминофора (реальные значения цветов R, G, B). Эта информация содержится в файле профиля монитора, входящем в комплект драйверов монитора.

Конечно, калибровкой мониторов занимаются только пользователи, связанные с задачами точной цветопередачи (хранение и отображение репродукций картин, профессиональная цветная печать и т. п.). Рядового пользователя больше интересует *чистота цвета* (colour purity), которая может ухудшаться при намагничивании элементов кинескопа. Для размагничивания кинескопа предназначена специальная катушка, расположенная по контуру экрана. Она кратковременно включается в момент включения монитора, но некоторые мониторы позволяют выполнять размагничивание (degauss) и во время работы. Монитор чувствителен к внешним магнитным полям. Приближение динамиков с сильным магнитным полем может привести к появлению цветных пятен на экране, а работающий близко динамик даст даже «цветомузыкальный» эффект. К счастью, система размагничивания через некоторое время сотрет эти пятна, но увлекаться «исследованиями» в этом направлении не рекомендуется (кинескоп — «игрушка» дорогая).

Качество сведения лучей

Важным параметром монитора, не имеющим численного определения, является *качество сведения лучей*. При хорошем сведении тонкие белые линии (например, символы) должны быть белыми, а не радужными. Сведение лучей чаще всего «хромает» по углам экрана. Для проверки качества сведения в первом приближении подходит наблюдение за сообщениями при загрузке, выводимыми обычно белыми символами. Можно прибегнуть также к внимательному осмотру рамок окон оболочек типа Norton Commander. В тестовых программах наподобие Checklt имеются видеотесты с изображениями тонких сеток, выводимых в графических режимах относительно высокого разрешения. Пожалуй, самым удобным средством проверки качества изображения является утилита Noki Test. Регулировка сведения — занятие непростое, а плохим качеством сведения бывает трудно мотивировать претензию на гарантийный ремонт или замену монитора. Поэтому имеет смысл «покапризничать» при приобретении монитора, выбирая по возможности самый хороший из ряда предложенных — даже в пределах одной модели может оказаться большой разброс качества.

Настройка геометрии

Регулировка размеров по вертикали (V.Size) и горизонтали (H.Size) позволяет подогнать параметры генераторов развертки так, чтобы изображение попадало в заданную область. Здесь возможны два вида «перегибов»: изображение занимает меньшую область, чем можно (underscan), или, наоборот, вылезает за границы экрана (overscan). Требуется, как всегда, «золотая середина». Помимо регулировки размеров важна и юстировка — подбор смещения по вертикали (V.Shift, V-Position или V.Phase) и горизонтали (H.Shift, H-Position или H.Phase). Назвать эти регулировки смещением (shift) или позицией (position) естественно для пользователя, поскольку таково видимое на экране действие. Назвать же их фазой (phase) скорее свойственно инженеру, поскольку подразумевается фазовый сдвиг генераторов относительно синхроимпульсов. Помимо размера и положения мониторы могут обеспечивать регулировку геометрических иска

жений типа *трапеции* (trapezoid) и *бочки* (pincushion). Все эти регулировки удобнее всего производить при выводе тестового изображения в виде сетки с квадратными ячейками. Все квадраты должны выглядеть действительно квадратными. Желательно проверять одно и то же изображение с разным уровнем яркости — его размеры и форма не должны заметно изменяться. Если размер меняется (чем ярче, тем крупнее), это говорит о недостаточной мощности источника высокого напряжения кинескопа и его нестабильности при изменении яркости. Объяснение связи простое: чем ниже напряжение, тем ниже скорость электронов и больше угол отклонения луча неизменном же магнитном поле развертки. Помимо геометрических искажений наблюдение сетки может выявить такие дефекты монитора, как нестабильность генераторов развертки, которая может быть вызвана плохой фильтрацией питающих напряжений. Пульсации с частотой питающей сети приводят к волнистости вертикальной линии справа (волна обычно движется вверх или вниз) или периодическому изменению размера по вертикали. Высокочастотные пульсации приводят к дрожанию или размытости изображения опять-таки в правой части экрана (там набегает большая погрешность относительно синхроимпульсов).

Синхронизация и цифровое управление

Первые мониторы с адаптерами CGA и MDA/HGC работали на фиксированных частотах развертки. С появлением адаптера EGA «жизнь» мониторов осложнилась — новые видеорежимы требовали других частот синхронизации. При этом переключение частот приводит к необходимости подстройки геометрических параметров. Мониторы EGA имели два существенно различающихся режима синхронизации. Режим задавался относительной полярностью вертикальных синхроимпульсов. Для каждого режима (Mode 1 и Mode 2) использовались отдельные элементы подстройки, коммутируемые в зависимости от полученного указания на режим синхронизации. Поначалу поддержание работоспособности монитора на разных частотах синхронизации представляло сложную техническую проблему. Одними из первых эту проблему решили разработчики фирмы NEC, и под соответствующие мониторы фирма даже зарезервировала торговую марку MultiSync. Потом появились названия MultiScan и MultiFrequency, которые обозначают ту же возможность. Адаптеры VGA и SVGA могут использовать различные режимы разрешения без столь существенного изменения частот, но от этого легче не стало — возникла потребность выбора частот развертки. Для распознавания режима также стали применять изменение полярности синхросигналов, но теперь уже обоих — H.Sync и V.Sync. При изменении параметров синхронизации (например, при переключении задач, работающих в разных графических режимах) приходится подстраивать геометрические параметры изображения, что вручную делать не очень удобно.

Решить проблему подстройки позволило *цифровое управление* (Digital Control, DC), которое стало обычным практически для всех современных мониторов. Суть цифрового управления сводится к тому, что в монитор встраивается специализированный микроконтроллер, управляющий практически всеми параметрами монитора. Потенциометры, традиционно использовавшиеся для всех регулировок, заменили кнопками управления (пара кнопок заменяет один регу

лятор). В первых мониторах цифровое управление только повышало надежность (потенциометры подвержены износу), но вскоре пошли дальше. Поскольку микроконтроллер может хранить большое количество параметров (он для этого имеет энергонезависимую память), несложно заставить его запоминать наборы параметров каждого используемого видеорежима. Таким образом, после первоначального «обучения» контроллер позволяет быстро установить запомненные параметры текущего видеорежима. Установленный видеорежим распознается по частотам и полярности сигналов синхронизации. Цифровое управление множеством параметров потребовало бы большого количества кнопок. Чтобы не загромождать лицевую панель монитора и облегчить работу пользователя, тому же микроконтроллеру поручили на экране монитора организовать дисплей для диалогового режима настройки. Такой дисплей, встроенный в экран, сокращенно называется экранным (On Screen Display, OSD). Применение OSD позволяет всего тремя-четырьмя кнопками обеспечить неограниченное число регулировок: одна или две кнопки требуются для выбора настраиваемого параметра в меню дисплея и еще две кнопки позволяют изменять настраиваемый параметр в обе стороны. Впрочем, такая крайность не обязательна — для таких регулировок, как, например, яркость, можно выделить и специальную пару кнопок. Меню дисплея появляется на экране во время настройки, перекрывая небольшую часть выводимого изображения, и автоматически исчезает по окончании настройки. От функций OSD сделали еще один небольшой шаг — ввели в монитор режим самотестирования. В этом режиме микроконтроллер при отсутствии сигнала от компьютера сам формирует цветное графическое изображение, по которому можно произвести настройку и оценить качество монитора. Конечно, монитор должен определить причину отсутствия сигнала — это ведь может быть и команда DPMS (см. далее). Как вариант можно отсоединить сигнальный кабель от монитора, инициируя команду на включение режима самотестирования, а отсоединение этого кабеля только от компьютера может рассматриваться как команда системы DPMS на отключение.

Управление энергопотреблением

Монитор, особенно цветной с большим экраном, является одним из основных потребителей электроэнергии — современный цветной монитор 15" потребляет около 80 Вт, для большего экрана больше и мощность. Международная организация EPA (Environmental Protection Agency — агентство по охране окружающей среды) выдвинула программу энергосбережения *Energy Star*, на которую ассоциация VESA откликнулась соответствующим стандартом. Мониторы, поддерживающие режимы энергосбережения, иногда называют «зелеными» — не по цвету экрана, а по названию общественного движения. Для управления энергопотреблением разработана система *DPMS* (Display Power-Management Signaling — управление энергопотреблением дисплея). Ниже перечислены режимы энергопотребления для мониторов:

- *On* — активная (нормальная) работа. Для монитора 15" типовое потребление — 80 Вт.

- *Standby* — отключение видеосигналов и снижение яркости до минимума, при этом потребление монитора снижается примерно на 20 %. Из этого режима в нормальный *(On)* монитор переходит быстро (около секунды). Поддержка состояния Standby не является обязательной для всех мониторов. Для монитора 15" типовое потребление — 60 Вт.
- *Suspend* — отключение строчной развертки, накала и высокого напряжения кинескопа, что снижает потребление на 70 %. Переход в режим *On* занимает около 15 с. Для монитора 15" типовое потребление — менее 15 Вт.
- *Off* — отключение всех схем монитора, кроме блока DPMS, потребление снижается до единиц ватт. Переключение в нормальный режим занимает около 30 с (как включение «холодного» монитора). Если в этом режиме обесточивается и блок DPMS, то монитор можно будет включить только вручную (нажатием кнопки).

Для переключения режимов управляют активностью сигналов синхронизации. Конечно, для работы системы энергосбережения ее должны поддерживать и монитор, и дисплейный адаптер, и BIOS. Переход в режимы с пониженным потреблением и «пробуждающие» события настраиваются в *CMOS Setup* параметрами управления энергопотреблением (Power Management).

Эргономические характеристики

Теперь остановимся на некоторых *эргономических вопросах*. Оператору, работающему с монитором длительное время, небезразлично, каким воздействиям он подвергается. В первую очередь, конечно, интересует качество изображения. Естественно, изображение должно быть четким и достаточно контрастным, а цвета — чистыми. Расфокусированность и плохое качество сведения лучей приводят к напряжению глаз и, следовательно, к повышенной утомляемости со всеми вытекающими последствиями.

Важно также обеспечить правильную ориентацию экрана относительно источников освещения. Если монитор стоит напротив окна, то на экране возникают блики. Установка монитора экраном от окна, особенно с солнечной стороны, тоже не очень хороша: яркий свет, бьющий в глаза смотрящему на монитор оператору, столь же утомляет глаза. Традиционно поверхность экрана слегка выпуклая — это вызвано стремлением приблизить к прямому угол падения электронного луча ближе к краям экрана. Однако при выпуклом экране довольно трудно избавиться от бликов, вызванных внешними источниками света. Ряд новых моделей кинескопов имеет *плоский экран* (flat screen). Такой экран обеспечивает меньшие геометрические искажения изображения и отчасти избавляет от бликов. Уменьшить блики позволяет и специальное *антибликовое покрытие* (antiglare coating) экрана, а также применение стеклянных *поляризационных фильтров*. Хороший фильтр позволяет также улучшить контрастность изображения, правда, и с некоторой потерей яркости, которую можно компенсировать настройкой монитора.

Помимо видимого света — изображения на экране, - монитор является источником высокого *статического электрического потенциала*, а также *электромаг*

нитного излучения в широком спектре частот. Для снижения статического потенциала применяют антистатическое покрытие, снимающее электростатический заряд с экрана, — это отмечается аббревиатурой *AS* (Anti Static) в перечислении достоинств монитора. Потенциал снижают также многие экранные фильтры — у них даже имеется провод с зажимом «крокодил» на конце, который нужно присоединить, например, к неокрашенной металлической части заземленного корпуса компьютера. Высокий потенциал определить просто — надо поднести палец к экрану, и если с расстояния в несколько миллиметров произойдет разряд (щелчок со слабым покалыванием или щекотанием), значит, потенциал велик. Если разряд возникает с расстояния в сантиметр и больше, защитный антистатический фильтр просто необходим.

Что касается радиации, то многие ошибочно считают, что ее воздействию в наибольшей степени подвергается оператор. На самом деле большая часть излучения исходит от задней стенки (с тыльной части трубки) и достается не оператору, а его соседу при неудачной расстановке техники. Уровень радиации мониторов стремятся уменьшать, и аббревиатура *LR* (Low Radiation) указывает на заботу производителя о здоровье пользователя, но без конкретных цифр. Строгие нормы по допустимому уровню электромагнитных излучений в различных частях спектра заданы шведским стандартом *MPR* II, который, фактически, стал международным. Этот стандарт, принятый в 1990 году, определяет как магнитные, так и электрические составляющие излучения. В 1992 году был принят более жесткий стандарт, называемый *TCO 92*, описывающий еще и энергосберегающие функции монитора. Далее появился стандарт *TCO 95*, относящийся уже не только к мониторам, но и к компьютеру в целом. Более жесткие требования ко всем параметрам монитора (и компьютера) предъявляет стандарт *TCO 99*, в котором содержатся требования по эргономике (режимы разрешений и частоты регенерации), электромагнитным излучениям, экологии (выделение вредных веществ, а также способ утилизации).

Матричные дисплеи

Дисплеи на электронно-лучевых трубках применительно к портативным компьютерам имеют два принципиальных неустранимых недостатка — большие габариты (объем) и потребляемую мощность. Надо заметить, что в первом портативном компьютере IBM PC Portable был применен все-таки дисплей на ЭЛТ, но если переносить этот компьютер было нетрудно (всего-то килограммов 10-15), то об автономном питании речи не шло. В наколенных (LapTop) и блокнотных (NoteBook) ПК применяют плоские дисплейные панели (flat panel display), основанные на различных физических принципах. В последнее время плоские дисплеи стали применять и для настольных компьютеров — их цена и качество стали уже вполне приемлемыми для замены громоздких и тяжелых ЭЛТ-мониторов.

Плоские дисплеи выполняются в виде *матрицы ячеек* с какими-либо электрооптическими эффектами.

Дисплеи на жидкокристаллических панелях (Liquid Crystal Display, LCD), или ЖК-дисплеи, основаны на изменении оптической поляризации отраженного

или проходящего света под действием электрического поля. Слой жидкокристаллического вещества расположен между двумя стеклами с поляризационными решетками. Жидкокристаллическое вещество способно менять направление поляризации проходящего света в зависимости от состояния молекул. При отсутствии электрического поля направление поляризации меняется на 90°, а в дисплеях, изготовленных по технологии STN (Super Twisted Nematic), поворот достигает 270°. Под действием электрического поля молекулы «распрямляются», и угол поворота уменьшается. Таким образом, в сочетании с поляризационными решетками стекол можно управлять прозрачностью элемента, изменяя величину электрического поля. В дисплеях DSTN (Double Super Twisted Nematic) ячейки сдваиваются, что позволяет повысить контрастность изображения. Дисплейная панель представляет собой матрицу ячеек, каждая из которых находится на пересечении вертикальных и горизонтальных координатных проводников. В *пассивной матрице* (passive matrix) дисплеев на жидкие кристаллы воздействуют поля самих координатных проводников. Ячейкам пассивной матрицы свойственна большая инерционность — порядка 300-400 мс (время на «перестройку» структуры молекул жидкокристаллического вещества), из-за чего на такие дисплеи плохо выводится динамическое изображение. Специально для них применяется особый режим отображения указателя мыши — за ним тянется шлейф, без которого быстро перемещаемый указатель визуально теряется. В *активной матрице* (active matrix) каждая ячейка управляется транзистором, которым, в свою очередь, управляют через координатные шины. В любом случае панели требуют *подсветки* — либо задней (back light), либо боковой (side light) от дополнительного (чаще люминесцентного) источника освещения. Иногда используют внешнее освещение, при этом за панелью располагается зеркальная поверхность. Активные матрицы обеспечивают более высокую контрастность изображения. Цветные дисплеи имеют более сложные ячейки, состоящие из трех элементов для управления каждым из базисных цветов.

Современные плоские *TFT LCD-дисплеи* представляют собой «бутерброд» из двух стекол, между которыми расположены слои жидкокристаллического вещества и матрица тонкопленочных транзисторов (Thin Film Transistor, TFT). На переднем и заднем стеклах нанесены поляризационные решетки со взаимно перпендикулярным направлением поляризации. Жидкокристаллическая прослойка при отсутствии электрического поля поворачивает угол поляризации проходящего света на 90°, благодаря чему «бутерброд» становится прозрачным для проходящих лучей. Под действием электрического поля от напряжения, подаваемого транзистором каждой ячейки матрицы, угол поворота поляризации может быть уменьшен до нуля. Чем больше приложенное напряжение, тем меньше угол поворота и тем менее прозрачной будет ячейка. Инерционность ячеек активной матрицы у старых дисплеев составляла 20-30 мс — меньше, чем для пассивной, но все равно ощутимо. На современных дисплеях инерционность снизили до 12 мс, и на них хорошо смотрятся «живое» видео и динамические игры.

В цветных дисплеях пиксел состоит из трех ячеек, каждая из которых снабжена своим светофильтром (красным, зеленым или синим). Управляя тремя транзи

сторами пиксела, можно изменять его цвет и яркость, что, собственно, и требуется от дисплея. Разрешающая способность по цвету у ЖК-мониторов пока ниже — только 6 бит на каждый цветовой канал, так что 24-битный режим True Color они могут только эмулировать.

Размер пиксела плоского дисплея близок к зерну ЭЛТ-мониторов: у дисплея 15" с разрешением 1024 x 768 — около 0,3 мм, а у дисплея 18" с разрешением 1280 x 1024 — около 0,28 мм. Размер изображения у ЖК-дисплеев 15" больше, чем у ЭЛТ-мониторов 15[м] (но несколько меньше, чем у 17"). Есть дисплеи с очень мелкими пикселами. С одной стороны, это хорошо — при том же размере можно выводить больше информации. С другой стороны, при недостаточной остроте зрения мелкие значки и буквы меню будут читаться с трудом. При мелких пикселах, как правило, дисплей обладает меньшим углом обзора и худшей цветопередачей.

Матричная организация экрана не позволяет изменять разрешение экрана с той же легкостью, что у ЭЛТ-монитора: увеличить его просто невозможно, а уменьшить без потери качества можно только одновременно с уменьшением размера изображения. Естественное (native) разрешение определяется форматом матрицы. В графическом режиме с меньшим или большим разрешением доступно два варианта: использование не всей матрицы (или вывод не всех пикселов) или масштабирование. Масштабирование выполняется встроенными средствами монитора, выполняющими интерполяцию цвета каждого пиксела экрана, что, естественно, ухудшает качество изображения.

СОВЕТ ----------

При работе с матричным дисплеем используйте графический режим, соответствующий размеру матрицы монитора.

Поскольку элементы матрицы весьма инерционны, возникают определенные сложности в плане совместимости ЖК-дисплеев с обычными графическими адаптерами, ориентированными на ЭЛТ. Дело в том, что время, в течение которого передается информация пиксела для ЭЛТ, несоизмеримо со временем реакции ЖК-элемента. Так, даже при частоте пикселов 50 МГц (а это низкая частота) один пиксел выводится за 20 нс, а инерционность самой быстрой активной матрицы — 12 мс (в 500 000 раз выше).

В ЖК-дисплеях управление осуществляется всеми ячейками одной строки одновременно (а не последовательно, как пробегает луч ЭЛТ). Это позволяет увеличить время, в течение которого производится управление ячейкой. Для повышения контрастности часто применяют *двойное сканирование*: экран разбивается на две части, в которых сканирование происходит одновременно. Таким образом, время управления ячейкой удваивается.

В любом случае аналоговые сигналы RGB от VGA-интерфейса непосредственно использоваться для управления матрицей не могут. В ЖК-дисплеях эти сигналы оцифровываются, полученные значения (для каждого пиксела) сохраняются в буферной памяти и оттуда уже построчно выводятся на матрицу. К сожалению, в обычном VGA-интерфейсе нет сигнала синхронизации пикселов,

так что дисплею приходится формировать стробы для отсчетов пикселов самостоятельно, привязывая их к импульсам строчной синхронизации. При этом, естественно, появляется дополнительная погрешность (апертурная) оцифровки сигнала и, следовательно, качество изображения ухудшается. В интерфейсах EVC и DVI-A сигнал пиксельной синхронизации присутствует, что несколько облегчает оцифровку.

Матричная организация располагает к применению цифрового интерфейса связи с графическим адаптером. Однако большинство плоских дисплеев имеет обычный аналоговый интерфейс, совместимый с любым (S)VGA-адаптером. Более дорогие модели снабжаются цифровым интерфейсом DVI (иногда и DFP). Заметим, что из-за инерционности ячеек слишком высокой частоты развертки не требуется — даже при 60 Гц мерцания экрана нет.

К *преимуществам ЖК-дисплеев (TFT LCD)* относятся высокая яркость изображения, отсутствие геометрических искажений, четкая фокусировка, отсутствие мерцания экрана (из-за инерционности ячеек), малое энергопотребление (25-40 Вт) и тепловыделение; вдобавок они легче и занимают меньше места. Кроме того, они практически нечувствительны к внешним электромагнитным полям, от которых плавает, дергается и искажается изображение ЭЛТ-мониторов. Ряд моделей позволяет поворачивать экран на 90° (и, соответственно, менять местами координаты) — так, что он принимает «портретную» ориентацию. Вместе с тем TFT-дисплеи имеют ряд *недостатков*, обусловленных их природой, — это, в частности, низкая контрастность изображения, зависимость качества изображения от угла наблюдения (меньший угол нормального восприятия цветного изображения), инерционность ячеек, невозможность смены разрешения (кроме как малопривлекательной интерполяцией), возможность отказа ячеек (на дисплее допускается неработоспособность нескольких транзисторов) и, конечно же, пока высокая цена. Фотореалистичность изображений, характерная для современных ЭЛТ-дисплеев, для ЖК-дисплеев пока что недостижима.

Газоплазменные панели (gas plasma) основаны на свечении газа под действием электрического поля. Эти панели используются в больших плоских телевизорах. В компьютерных мониторах их пока не применяют из-за больших размеров пикселов.

Светодиодные матрицы, или *матрицы LED* (Light Emitted Diode — светоизлучающий диод), казалось бы, могли стать решением всех проблем плоских дисплеев. Однако светодиоды имеют настолько высокую потребляемую мощность по сравнению с другими типами индикаторов, что их в плоских панелях не применяют.

Трехмерный вывод изображения и виртуальная реальность

Графическая система (и видеосистема) ПК ориентирована на вывод изображения на монитор, который пользователь видит обоими глазами. Трехмерная графика (со всеми рассмотренными конвейерами) и 3D-акселераторами все равно выводится на двухмерный экран. Для того чтобы получить эффект объема, не

обходимо разделить изображения, видимые левым и правым глазами зрителя, — создать *стереопару кадров*. При выводе динамических изображений разделять приходится потоки. Для того чтобы создать полную иллюзию объема (виртуальную реальность), нужно, чтобы выводимые потоки формировались с учетом положения (поворотов, наклонов) головы наблюдателя. Разделять информацию для левого и правого глаз можно несколькими способами — используя монитор (обычный или 3D-монитор), проектор или VR-шлем. Стереопара формируется на одном выходном интерфейсе графического адаптера. Требуется решить несколько задач: совместить изображения в одном потоке, затем разделить его и довести до двух глаз.

Для вывода левого и правого кадров через один VGA-интерфейс используются различные методы:

- Чередование кадров: левые и правые кадры пары выводятся поочередно. При этом для каждого глаза формируется изображение с частотой смены кадров, равной половине кадровой частоты развертки (частоту развертки приходится повышать). Разрешение изображения для обоих глаз соответствует текущему видеорежиму графического адаптера. В видеопамяти изображения для левого и правого глаз хранятся в разных буферах.
- Чередование строк: четные и нечетные строки растра используются для своих половин стереокадра, причем они хранятся в одном буфере. Реальное разрешение по вертикали оказывается вдвое ниже, чем разрешение установленного видеорежима. Частота смены пар равна кадровой частоте.
- Разбиение кадра по горизонтали: в верхней половине формируется изображение для левого глаза, в нижнем — для правого, оба находятся в одном буфере. Разрешение по вертикали уменьшается вдвое. Частота смены пар равна кадровой частоте.
- Разбиение кадра по вертикали — здесь вдвое уменьшается разрешение по горизонтали, остальные характеристики те же (применяется редко).

При выводе через монитор или проектор наблюдатель должен пользоваться *стереоочками*, активными (связанными с компьютером) или пассивными.

В активных стереоочках для каждого глаза имеется управляемый оптический затвор, который может находиться в прозрачном или непрозрачном состоянии. Затворы открываются поочередно, синхронно с чередованием кадров (или строк) стереопары. Недостаток таких очков — высокая цена и необходимость связи зрителя с ПК. Связь может быть проводной или инфракрасной.

Самые простые (и дешевые) стереоочки — пассивные, в которых для левого и правого глаз используются поляризационные фильтры, пропускающие свет определенного направления поляризации. Сложность заключается в формировании стереопары — двух изображений с разной поляризацией. Существуют матричные 3D-дисплеи, в которых четные и нечетные строки пикселов имеют различное направление поляризации, для них используется метод чередования строк. Для ЭЛТ-мониторов различную поляризацию строк не обеспечить, поскольку строки (и пикселы) в них — «нарисованные» лучом, а не реальные, физические. Однако перед монитором можно установить управляемый светофильтр,

призванный менять направление поляризации изображения синхронно со сменой кадров или строк.

Стереоизображение можно наблюдать на специальном мониторе и без очков, если развить и использовать недостаток ЖК-мониторов — зависимость от угла наблюдения. В 3D-мониторе Sony имеет место чередование пикселов в строке и применяется специальный расщепитель изображения (дополнительный слой), благодаря которому четные пикселы видны левым глазом, нечетные — правым. Расщепитель управляемый — специальный датчик отслеживает положение головы пользователя, корректируя угол отклонения луча расщепителем (иначе голову пользователя пришлось бы фиксировать).

В *шлемах виртуальной реальности* (и в стереобиноклях) для каждого глаза формируется свое изображение, опять-таки из исходного VGA-сигнала. Здесь для каждого глаза имеется своя миниатюрная ЖК-матрица, связанная с интерфейсом через адаптер. Задача адаптера, подключенного к VGA-интерфейсу, — сформировать два композитных видеосигнала (два VGA-кабеля, подходящие к шлему, были бы слишком громоздкими). Для того чтобы из передаваемой стереопары извлечь раздельную информацию, применяют гашение кадров (при чередовании кадров), чересстрочную развертку или прогрессивную развертку с дублированием или гашением строк (при чередовании строк), дублирование кадровых импульсов (при разделении по горизонтали).

Шлем снабжается и системой виртуальной ориентации, которая посылает в компьютер информацию о положении головы наблюдателя. Эта информация управляет построением изображений в стереопарах.

10.4. Интерфейсы мониторов и видеосистем

Для подключения монитора (дисплея) к графическому адаптеру используют прямую подачу сигналов на входы видеоусилителей базовых цветов — *RGB-интерфейс.* В ходе эволюции дискретный интерфейс монохромных и первых цветных мониторов CGA и EGA с сигналами ТТЛ [1,6] сменился популярным ныне аналоговым интерфейсом VGA, обеспечивающим передачу большого количества цветов. Однако далее качество передачи аналогового сигнала перестало удовлетворять растущие потребности (с повышением частот развертки и разрешения), и появился цифровой интерфейс DVI. Для матричных дисплеев этот интерфейс более приемлем, хотя большинство матричных дисплеев и видеокарт используют противоестественный для них аналоговый интерфейс VGA.

Первые видеоадаптеры допускали подключение стандартного телевизора, что было несложно благодаря совпадению параметров синхронизации. Адаптеры и мониторы VGA работают с частотами развертки, неприемлемыми для телевизоров, и на какое-то время интерфейс телевизора на графические адаптеры ставить перестали. В современных адаптерах снова появился телевизионный выход TV-Out, который может работать независимо от видеорежимов основного

монитора. Графические адаптеры с видеооверлеем и фрейм-граббером имеют также входные интерфейсы телевизионного сигнала. Для телевизионного интерфейса поддерживается синхронизация от внешней телевизионной системы (конвертора), что важно для совмещения компьютерного видеосигнала с внешним «телевизионным окружением».

Аналоговые интерфейсы RGB

Интерфейс *RGB Analog* с аналоговой передачей сигналов яркости базисных цветов позволяет передавать формально неограниченное число оттенков. Сигналы базисных цветов в современных адаптерах формируются 8-разрядными ЦАП, что позволяет выводить 16,7 миллионов цветов *(True Color).* Для снижения перекрестных помех эти сигналы передаются по витым парам с собственными обратными линиями (Return). Для согласования с кабелем в мониторе каждая сигнальная пара нагружается резистором. Черному цвету соответствует нулевой потенциал на линиях всех цветов, полной яркости каждого цвета соответствует уровень +0,7 В (не все графические адаптеры обеспечивают полную амплитуду сигнала). Сигналы управления, состояния и синхронизации передаются сигналами ТТЛ. Обычно для горизонтальной и вертикальной синхронизации используются отдельные сигналы H_Sync и V_Sync. В адаптере PGA использовалась *совмещенная синхронизация* (composite sync) сигналом (H+V)Sync; этот режим поддерживают и многие современные мониторы.

Впервые аналоговый интерфейс был применен на адаптере PGA фирмы IBM, где для него использовался 9-контактный разъем DB-9S. В дальнейшем, начиная с адаптеров VGA, стали применять малогабаритный 15-контактный разъем с таким же внешним размером (табл. 10.3). В компьютерах Macintosh монитор, совместимый по параметрам с VGA, имеет разъем DB-15P (такой же, как и у Game-порта PC).

Таблица 10.3. Аналоговый интерфейс монитора VGA(RGB Analog)

Контакт	Видеоадаптер	Монитор	
DB-15	MCGA/VGA/SVGA/XGA	Mono	Color
1	Red	–	Red
2	Green	Video	Green
3	Blue	–	Blue
4	ID2	–	–
5	GND/DDC Return[1]	SelfTest/DDC Return	SelfTest/DDC Return
6	Red Return	Key	Red Return
7	Green Return	Video Return	Green Return
8	Blue Return	–	Blue Return
9	Ключ (нет контакта)[1]	–[1]	–[1]
10	GND (Sync Return)	GND (Sync Return)	GND (Sync Return)
11	ID0	–	GND
12	ID1/SDA[1]	– /SDA[1]	GND/SDA[1]

Контакт DB-15	Видеоадаптер MCGA/VGA/SVGA/XGA	Монитор	
		Mono	Color
13	H_Sync/(H+V)Sync[2]	H_Sync/(H+V)Sync[2]	H_Sync/(H+V)Sync[2]
14	V_Sync	V_Sync	V_Sync
15	ID3/SCL[1]	ID3/SCL[1]	ID3/SCL[1]

[1] Сигналы DDC Return, SDA и SCL задействуются только при поддержке DDC. При этом контакт 9 может использоваться для питания логики DDC (+5 В).
[2] Сигнал (H+V)Sync используется при совмещенной синхронизации.

Несмотря на наличие ключа — D-образного кожуха, — 15-контактные разъемы ухитряются вставлять в перевернутом положении, при этом один из контактов среднего ряда подгибается, а потом и ломается (штырьки этих разъемов тоньше и слабее, чем у 9-контактных). Естественно, монитор, подключенный таким образом, работать не будет.

Помимо изображения, по интерфейсу передают информацию, необходимую для автоматизации согласования параметров и режимов монитора и компьютера. «Интересы» компьютера представляет дисплейный адаптер, к которому и подключается монитор. С его помощью обеспечиваются идентификация монитора, необходимая для поддержки технологии PnP, и управление энергопотреблением монитора.

Для простейшей *параллельной идентификации монитора* в интерфейс ввели четыре логических сигнала ID0-ID3, по которым адаптер мог определить тип подключенного монитора IBM. Со стороны монитора эти линии либо подключались к шине GND, либо оставлялись неподключенными. Однако из этой системы идентификации используют лишь сигнал ID1, по которому определяют факт подключения монохромного монитора. Монохромный монитор может быть опознан адаптером и иначе — по отсутствию нагрузки на линиях Red и Blue.

Параллельную идентификацию мониторов заменила *последовательная идентификация* по каналу цифрового интерфейса *VESA DDC* (Display Data Channel). Этот канал построен на интерфейсе I^2C (DDC2B) или *ACCESS.Bus* (DDC2AB), который требует всего двух ТТЛ-сигналов — SCL и SDA. Интерфейс *DDC1* является однонаправленным — монитор посылает адаптеру блок своих параметров по линии SDA (контакт 12), которые синхронизируются сигналом V_Sync (контакт 14). На время приема блока параметров адаптер может повысить частоту V_Sync до 25 кГц (генератор кадровой развертки по такой высокой частоте синхронизироваться не будет). Интерфейс *DDC2* является двунаправленным; для синхронизации используется выделенный сигнал SCL (контакт 15). Интерфейс *DDC2AB* отличается тем, что допускает подключение ПУ, не требующих высокой скорости обмена, к компьютеру по последовательной шине *ACCESS Bus*. Блок параметров расширенной идентификации дисплея (Extended Display Identification, EDID) имеет одну и ту же структуру для любой реализации DDC.

Для *управления энергопотреблением* монитора в соответствии со стандартом VESA *DPMS* используются сигналы кадровой и строчной синхронизации V_Sync и H_Sync (табл. 10.4).

Таблица 10.4. Управление энергопотреблением монитора (VESA DPMS)

Режим	H_Sync	V_Sync
On	Активен	Активен
Standby	Неактивен	Активен
Suspend	Активен	Неактивен
Off	Неактивен	Неактивен

Интерфейс с BNC-разъемами

Разъемы, применяемые в современных адаптерах и мониторах SVGA, не предназначены для передачи высокочастотных сигналов. Пределом для них является примерно 150 МГц, что для высокого разрешения и высокой частоты регенерации недостаточно. Поэтому на больших профессиональных мониторах с высокими разрешением и частотами синхронизации имеются *BNC-разъемы* для соединения с помощью коаксиальных кабелей. Мониторы с коаксиальными входами могут быть подключены и к адаптерам с разъемом DB-15, для чего выпускаются специальные кабели-переходники. У этих переходников может быть три-пять 75-омных коаксиальных кабелей с разъемами BNC:

- 3 разъема — сигналы базисных цветов, сигнал смешанной синхронизации передается в канале зеленого цвета;
- 4 разъема — сигнал смешанной синхронизации передается по отдельному кабелю;
- 5 разъемов — сигналы вертикальной и горизонтальной синхронизации передаются по раздельным кабелям.

С помощью коаксиальных кабелей возможно удаление монитора от компьютера на расстояние до 10-15 м при хорошем изображении. Однако сигналы DDC при этом не передаются.

Комбинированный интерфейс EVC

Для расширения частотного диапазона (и учитывая тенденцию к использованию последовательных шин USB и FireWire для подключения ПУ к системному блоку компьютера) ассоциация VESA в 1995 году предложила разъем *EVC* (Enhanced Video Connector). В 1998 году была принята новая редакция, и разъем переименован в P&D-A (Plug&Display-Analog) с небольшими изменениями, касающимися резервных контактов и цепей питания зарядного устройства. Разъем имеет две секции: высокочастотную для присоединения четырех коаксиальных кабелей и низкочастотную на 30 контактов (рис. 10.4). *Высокочастотная секция* — контакты C1-C4 и C5 (экран) — требуется для передачи цветовых сигналов R, G, B и синхросигнала пикселов PX Clock. Синхросигнал пикселов «интересен» матричным дисплеям (с их цифровой природой) — он позволяет уменьшить погрешности передачи видеоинформации. Частота этого сигнала равна либо частоте сканирования пикселов, либо ее половине (на высокой частоте нужна двойная синхронизация, по фронту и спаду, что уравнивает требования к полосе пропускания для линий цветовых данных и линии синхронизации пикселов). Контакты высокочастотной секции, хотя и не являются коакси-

альными, позволяют передавать сигналы с частотами до 2 ГГц. Контактом экранов является крестообразная перегородка.

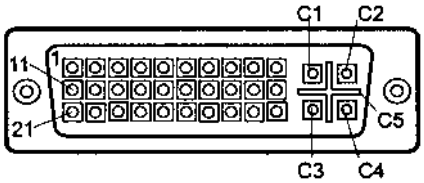


Рис. 10.4. Разъем EVC и P&D-A (розетка)

В *низкочастотной секции* (30 контактов) помимо сигналов синхронизации и канала DDC2 имеются контакты для видеовхода, входные и выходные стерео-аудиосигналы, шины USB и FireWire, а также линии питания постоянного тока для зарядки аккумуляторов портативных ПК. Разъем поделен на компактные зоны для каждой группы сигналов; правда, шины USB и 1394 используют общий контакт для экрана. Назначение контактов видеовхода (S-Video, композитного интерфейса, PAL или NTSC) может программироваться по каналу DDC2.

Стандарт определяет три уровня реализации: базовый, мультимедийный и полный. Базовый включает только видеосигналы и DDC, в мультимедийном должны быть аудиосигналы. При использовании коннектора в полном объеме монитор превращается в коммутационный центр, который соединяется с компьютером одним кабелем, а все остальные ПУ (включая клавиатуру, мышь, принтер) подключаются к монитору. Разъем может применяться для подключения портативного ПК к док-станции. EVC собирает сигналы от разных подсистем — графической, видео, аудио, последовательных шин и питания. Этот общий разъем, устанавливаемый на корпусе системного блока, может соединяться с разными платами внутренними кабелями через промежуточные разъемы. Его не следует путать с похожим по виду и названию разъемом P&D-A/D, описанным в следующем пункте. Разъемы EVC на компьютерах встречаются нечасто, и это объясняется не только их довольно высокой ценой. Устанавливать разъем EVC на графическую карту неудобно (она «обрастет» лишними интерфейсными шлейфами), а интегрированные системные платы редко имеют графические адаптеры с выдающимися параметрами, для которых он нужен.

Цифровые интерфейсы P&D, DVI и DFP

Повсеместный переход на цифровые технологии коснулся и видеомониторов. Традиционный аналоговый канал передачи видеосигналов стал узким местом видеосистемы. По пути от ЦАП к входам видеоусилителей монитора сигнал проходит через пару разъемов и кабель. Несогласованность элементов, вызывающая отражения сигналов («звон») и неравномерности частотных характеристик, приводит к искажению формы сигналов цветов, что становится особо заметным на режимах с высокими разрешением и частотой регенерации. Повысить качество изображения можно, перенеся устройства ЦАП в монитор, прямо

на плату видеоусилителей, и подав на них цифровые сигналы базисных цветов. Плоские дисплеи (матрицы TFT) строятся на основе цифровых технологий, и им приходится входные аналоговые сигналы преобразовывать обратно в цифровую форму. Все эти причины привели к необходимости разработки цифрового интерфейса для передачи информации в монитор. От этого интерфейса требуется огромная пропускная способность: к примеру, при частоте пикселов 150 МГц и кодировании каждого пиксела 24-битным числом (True Color) требуется пропускная способность 3,6 Гбит/с (450 Мбайт/с).

Для подключения плоских дисплеев был разработан специальный интерфейс PanelLink, в 1996 году его спецификация (FPDI-2) была утверждена VESA. Схема интерфейса приведена на рис. 10.5. Цифровой интерфейс имеет 3 канала передачи данных (Data[0:2]) и канал синхронизации Clock. В каналах используется дифференциальная передача сигналов с минимизацией переходов — так называемый протокол T.M.D.S. (Transition Minimized Differential Signaling). Каждый канал данных образован кодером, расположенным на видеокарте, линией связи и декодером, расположенным в дисплее. На вход кодера каждого канала поступают 8 бит кода яркости базисного цвета текущего пиксела. Кроме того, на вход канала 0 кодера поступают сигналы строчной и кадровой синхронизации, а на остальные каналы — дополнительные управляющие сигналы CTL[0:3], по паре на каждый канал. Кодеры преобразуют данные в последовательный код, для минимизации переключений 8 входных битов кодируются 10-битным символом, передаваемым по каналу последовательно. В зависимости от входного сигнала разрешения данных DE кодеры передают либо данные цветовых каналов, либо синхросигналы и управляющие биты. На приемной стороне сигналы декодируются и восстанавливаются в том же виде, в котором они поступали на входы кодеров. Частота пикселов может достигать 165 МГц, интерфейс обеспечивает максимальное разрешение 1280 x 1024 (24 бита на пиксел).

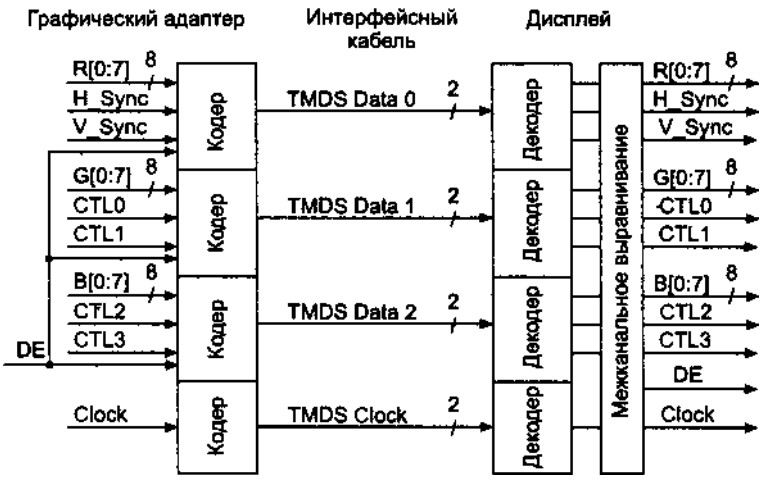


Рис. 10.5. Схема цифрового интерфейса

Физические линии реализованы экранированными витыми парами. Выбранный метод кодирования пригоден и для передачи по оптоволоконному кабелю (сигнал не имеет постоянной составляющей), но пока спецификация определяет только электрический интерфейс.

Вышеописанный протокол используется в интерфейсах P&D, DVI и DFP, из которых наибольшее распространение получил DVI (как самый мощный и универсальный). Разъем DVI можно встретить на многих графических адаптерах с двумя выходами. Почти не прижившийся дорогой разъем P&D можно рассматривать как комбинацию «усеченных» интерфейсов EVC и DVI. Интерфейс DFP (самый дешевый) также не получил широкого распространения. Благодаря стандартизованным сигналам (T.M.D.S.) при механическом несовпадении разъема монитора и графической карты возможно применение пассивных переходников-адаптеров.

В *интерфейсе VP&D* (VESA Plug-and-Display, 1997 г.), он же P&D, используется такой же разъем, как в EVC (см. рис. 10.4). Здесь нет цепей аналоговых аудиосигналов и видеовхода, а контакты, требовавшиеся для них, теперь назначены на цифровые каналы передачи сигналов. Интерфейс существует в двух вариантах: комбинированном и чисто цифровом. На комбинированный разъем P&D-A/D выведены и аналоговые сигналы (RGB и синхронизация), что обеспечивает возможность подключения как цифрового, так и традиционного аналогового монитора. В чисто цифровом варианте P&D контактов аналоговых сигналов нет; монитор с аналоговым входом (с разъемом EVC или P&D-A) с ним работать не может (конструкция разъема и не позволит его подключить). Точно так же не удастся подключить и монитор с чисто цифровым входом P&D к выходу P&D-A (EVC).

Интерфейс плоских дисплеев DFP (Digital Flat Panel, 1999 г., www.dfp-group.org) использует дешевый разъем типа MDR (mini-D ribbon) с ленточными контактами (рис. 10.6), на который выведены лишь 3 пары сигналов для цифровых каналов данных, пара для цифрового канала синхронизации, питание (+5 В), канал DDC2 и сигнал обнаружения «горячего» подключения (HPD). Частота пикселов может достигать 85 МГц (для плоских панелей не требуется слишком высокая частота развертки). Интерфейс пригоден (пока?) для режимов вплоть до 1280 x 1024 (24 бита на пиксел).

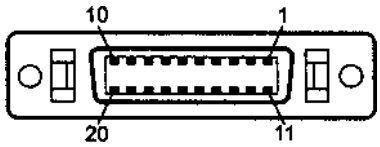


Рис. 10.6. Разъем плоского дисплея DFP

Интерфейс DVI (Digital Visual Interface) разработан группой DDWG (Digital Display Working Group — рабочая группа по цифровым дисплеям, www.ddwg. org) в 1999 году и предназначен для подключения дисплеев любого типа (ЭЛТ и матричных) к компьютеру, причем возможны два варианта коннекторов и ин

терфейса: чисто цифровой и цифровой с традиционными аналоговыми сигналами. Во втором случае к разъему DVI через пассивный переходник может быть подключен монитор с обычным аналоговым VGA-интерфейсом. Вид коннекторов DVI приведен на рис. 10.7, расположение сигнальных контактов дано в табл. 10.5. Разъем DVI устанавливается на многие современные видеокарты.

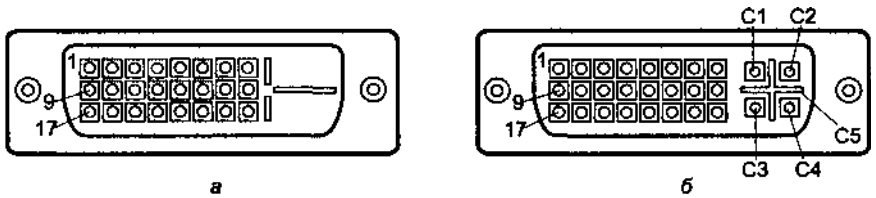


Рис. 10.7. Коннекторы DVI (розетки): а — только цифровой, б — цифровой с аналоговым Таблица 10.5.

Коннектор DVI

Контакт	Цепь	Контакт	Цепь	Контакт	Цепь
1	Data2–	9	Data1–	17	Data0–
2	Data2+	10	Data1+	18	Data0+
3	Экран 2/4	11	Экран 1/3	19	Экран 0/5
4	Data4–	12	Data3–	20	Data5–
5	Data4+	13	Data3+	21	Data5+
6	DDC Clock	14	+5 В	22	Экран Clock
7	DDC Data	15	GND (для +5 В, H_Sync и V_Sync)	23	Clock+
8	V_Sync (ТТЛ)	16	HPD	24	Clock–
C1	R (аналог.)			C3	B (аналог.)
C2	G (аналог.)	C5	GND (для R, G, B)	C4	H_Sync (ТТЛ)

В интерфейсе может использоваться 4 или 7 линий T.M.D.S. Минимальный вариант DVI позволяет передавать сигналы при частоте пикселов до 165 МГц (по трем каналам данных). Если же требуется более высокая частота, то должны быть задействованы каналы 3-5, и информационная нагрузка должна распределяться поровну между парами каналов, что позволит передавать пикселы с частотой до 330 МГц. Предусматривается и иное использование дополнительных каналов: когда 8 бит на кодирование базисного цвета покажется недостаточным(!), каналы 3, 4 и 5 могут дополнить (как младшие биты) данные каналов 0, 1 и 2 (старшие).

Помимо сигналов T.M.D.S, в интерфейс DVI входят сигналы интерфейса VESA DDC2, а также линия питания +5 В, по которой от видеокарты питаются цепи DDC, позволяя обмениваться конфигурационной информацией даже с выключенным монитором. Благодаря конфигурационной информации система определяет состояние монитора и должным образом конфигурирует имеющиеся каналы данных, согласуя возможности видеокарты и дисплея. Имеется также сигнал HPD (Hot Plug Detect), с помощью которого система может следить за подключением/отключением дисплея. «Горячее» подключение обеспечивается

также и механическими особенностями разъемов, поддерживающих требуемую последовательность соединения/рассоединения разных групп контактов. Таким образом, дисплеи с DVI обеспечивают все необходимые функции для реализации принципов PnP. Интерфейс поддерживает механизм управления энергопотреблением (DPMS).

При использовании DVI меняется местоположение ЦАП и применяется цифровой способ доставки данных. При этом гамма-коррекция возлагается на дисплей. В исходном варианте временные диаграммы сигналов аналогичны интерфейсу VGA — пикселы выводятся только во время прямого хода по строке (причем только в видимой его части, составляющей около 90 % периода строк). Однако интерфейс предусматривает способ повышения пропускной способности за счет более эффективного расходования времени. Дело в том, что традиционные ЭЛТ-мониторы имеют довольно значительное время обратного хода луча по строке и кадру, в течение которого пикселы на экран, естественно, не выводятся — в это время интерфейс простаивает. Для матричных дисплеев этих пауз не требуется, поэтому тот же объем информации о пикселах может передаваться за большее время — практически за весь период кадра. Следовательно, можно либо снизить тактовую частоту передачи пикселов (не меняя разрешения и частоты развертки), либо с той же (предельно достижимой) частотой передачи увеличить разрешение или/и частоту развертки. Спецификация DVI предполагает, что за счет внутренней буферизации возможность передачи данных в течение всего периода кадра может появиться и у цифровых дисплеев, построенных на обычных ЭЛТ. При условии буферизации экрана в дисплее можно пойти и дальше — вместо непрерывной регенерации экрана, которой озабочены традиционные видеоадаптеры, передавать данные только при изменениях изображения, но это пока лишь возможные перспективы.

Телевизионные интерфейсы

В традиционной технике цветного телевизионного вещания видеосигнал непосредственно несет информацию о мгновенном значении яркости (в нем присутствуют и синхроимпульсы отрицательной полярности), а цветовая информация передается в модулированном виде на дополнительных частотах. Таким образом обеспечивается совместимость черно-белого приемника, игнорирующего цветовую информацию, с цветным передающим каналом. Однако способы кодирования цветовой информации и частоты разверток в системах PAL, SECAM и NTS C различны.

Телевизионный интерфейс можно использовать для вывода графики низкого разрешения, в которой частоты синхронизации близки к стандартным телевизионным и не требуется особо широкая полоса пропускания видеотракта. Для вывода графической информации с высоким разрешением ни одна из традиционных вещательных систем не подходит, поскольку они имеют существенно ограниченную полосу пропускания цветовых каналов (те минимальные 35 МГц,

о которых мы говорили ранее, недостижимы). Сейчас телевизионный интерфейс TV-Out в основном используется для просмотра фильмов с компьютера, а TV-In — для ввода видео в компьютер. В видеотехнике применяют различные

низкочастотные интерфейсы, из которых здесь рассмотрены композитный интерфейс и интерфейс S-Video. Наивысшее качество передачи обеспечивает *профессиональный* (студийный) *интерфейс YUV* (professional video), использующий три сигнальные линии: здесь цветоразностные сигналы U и V передаются в немодулированном виде. При наличии дополнительного радиочастотного модулятора (Radio Frequency Modulator, RFM) видеосигнал (и звук) можно подавать и через антенный вход стандартного телевизионного приемника, но при этом еще более снижается реальное разрешение графики.

В композитном интерфейсе *(composite video)* полный стандартный видеосигнал с размахом около 1,5 В передается по коаксиальному кабелю (75 Ом). Данный интерфейс характерен для бытовых видеомагнитофонов, аналоговых телекамер, телевизоров. В ПК этот интерфейс используется как дополнительный выходной интерфейс графической карты и как входной интерфейс в устройствах захвата видеосигнала (фрейм-грабберах). Для композитного интерфейса используют коаксиальные разъемы RCA («колокольчик», рис. 10.8, а), широко применяемые в видео- и аудиотехнике. Тот же сигнал передается и через разъем SCART, популярный в европейской видеотехнике. Заметим, что аудиосигнал через этот интерфейс не передается, для него используются отдельный «колокольчик» или отдельные контакты SCART.

Интерфейс *S-Video* (Separate Video) использует раздельные 75-омные сигнальные линии: Y для канала яркости и синхронизации и C для сигнала цветности. По линии C передается поднесущая частота, модулированная цветоразностными сигналами (burst signal). Стандартный 4-контактный разъем S-Video типа mini-DIN (рис. 10.8, *а)* используется как интерфейс высококачественных видеосистем, его синонимами являются названия *S-VHS* и *Y/C*. Этот интерфейс в ПК тоже может применяться в качестве входного и дополнительного выходного; он обеспечивает более высокое качество передачи видеоизображений. Иногда задействуют и 7-контактные разъемы mini-DIN; у них внешние 4 контакта имеют то же назначение, а 3 внутренних контакта используются для разных целей (там может быть и композитный сигнал). Выход S-Video (рис. 10.8, *б)* легко преобразовать в сигнал для композитного входа (рис. 10.8, в); эта схема не обеспечивает должного согласования импедансов, но дает приемлемое качество изображения. Обратное преобразование этой схемой выполняется гораздо хуже, поскольку на яркостный сигнал воздействует помеха в виде сигнала цветности.

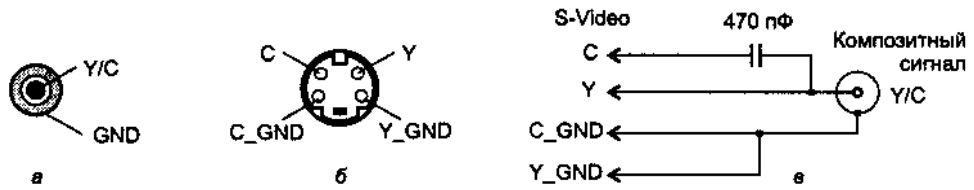


Рис. 10.8. Телевизионные интерфейсы: а — композитный RCA, б — S-Video, *в* — преобразование S-Video в композитный сигнал

10.5. Дисплейные адаптеры

Дисплейным адаптером условимся называть блок компьютера, к которому подключается дисплей — монитор на электронно-лучевой трубке или матричный монитор. В обязательный круг задач этого адаптера входит формирование изображения на экране под управлением программы компьютера, выполняемое в графическом и/или алфавитно-цифровом режиме отображения. Расширенный круг задач может включать и воспроизведение на экране того же монитора «живого» видео из потока данных, полученного от компьютера или от какого-либо источника телевизионного сигнала. Первые графические адаптеры строились на базе контроллера ЭЛТ (6845), обрамленного массой микросхем средней степени интеграции. В современных дисплейных адаптерах применяются наборы специализированных интегральных схем высокой степени интеграции — *графические чипсеты* и *видеочипсеты*. Эти микросхемы вместе с применяемыми микросхемами видеопамяти определяют основные характеристики адаптеров.

Компоненты дисплейного адаптера

Рассмотрим функциональную схему графического адаптера (рис. 10.9), которая с отдельными добавлениями или исключениями приложима практически ко всем адаптерам, применяемым в PC.

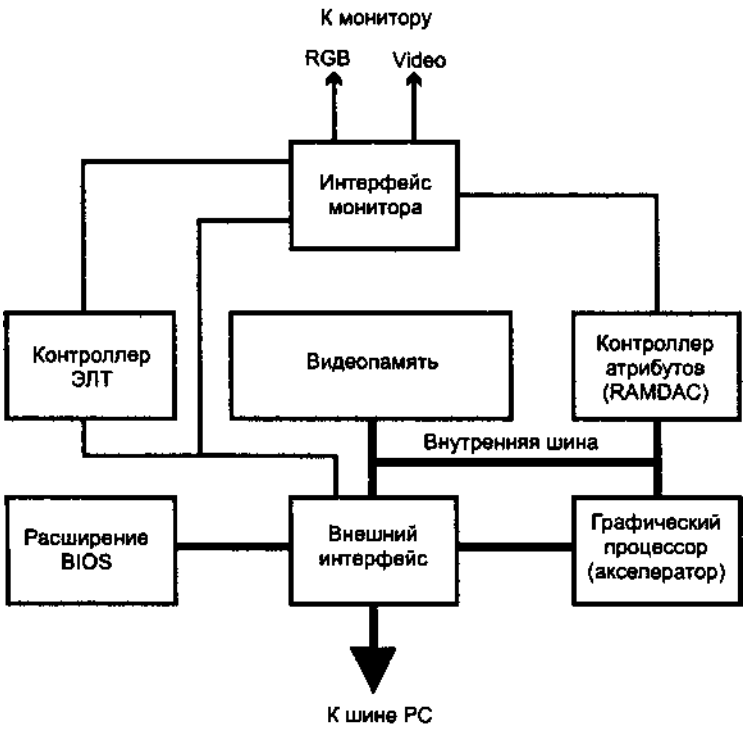


Рис. 10.9. Функциональная схема графического адаптера

Поскольку адаптер предназначен для подключения монитора, его обязательным элементом является *контроллер ЭЛТ* (CRT Controller). В задачи этого контроллера входит согласованное формирование сигналов сканирования видеопамяти (адрес и стробы чтения), а также вертикальной и горизонтальной синхронизации монитора. Контроллер ЭЛТ должен обеспечивать требуемые частоты развертки и режимы сканирования видеопамяти, которые зависят от режима отображения (графического или текстового) и организации видеопамяти, о чем говорилось в начале главы. Опорной частотой для работы контроллера является DotCLK — частота вывода пикселов в графических режимах или точек разложения символов в текстовом режиме. В самых первых моделях адаптеров в качестве контроллера ЭЛТ применялась микросхема Motorola 6845, и ее регистровая модель поддерживается современными адаптерами при эмуляции адаптеров CGA и MDA. В текстовом режиме этот же узел формирует и аппаратный курсор.

Видеопамять — это специальная область памяти, из которой контроллер ЭЛТ организует циклическое чтение для регенерации изображения. Традиционно для видеопамяти в карте распределения памяти PC была выделена область адресов A0000-BFFFFh, непосредственно доступная любому процессору x86. Первым адаптерам (MDA, CGA) этой области было более чем достаточно. Адаптеры EGA эти 128 Кбайт использовали уже полностью, а для дальнейшего увеличения объема, требующегося адаптерам VGA и SVGA, приходится применять технику переключения банков. Эта вынужденная мера осложняет программное формирование изображения, которое в режимах высокого разрешения с большим количеством цветов уже не помещается в один банк. Типовой объем видеопамяти у рядовых адаптеров без 3D-акселераторов составляет

1 Мбайт, хотя при этом разрядность шины данных памяти оказывается всего 16 бит (были популярны микросхемы памяти 4 Мбит с организацией 512K x 8). Все возможности 32-битных графических чипсетов реализовались с памятью объемом 2 Мбайт, адаптеры для профессиональной работы с цветом имели 4-8 Мбайт видеопамяти. Современные графические адаптеры имеют возможность переадресации видеопамяти в область старших адресов (выше границы 16 Мбайт), что позволяет в защищенном режиме процессоров класса 386+ (и в «большом реальном» режиме) работать с цельными образами экранов. Необходимый объем видеопамяти определяется желаемым графическим режимом (в текстовом режиме нужно всего несколько килобайт, которые погоды не делают). Требуемые объемы для одной страницы различных видеорежимов приведены в табл. 10.6. Если взять удвоенное значение этих объемов, то многие адаптеры позволят организовать двухстраничный режим с переключением буферов, что полезно для вывода динамичных изображений. У адаптеров с 3D-акселераторами потребности в памяти выше (8-256 Мбайт).

Таблица 10.6. Разрешение и требуемый объем видеопамяти

Бит/пиксел	Количество цветов	640×480	800×600	1024×768	1280×1024
4	16	150 Кбайт	234 Кбайт	384 Кбайт	640 Кбайт
8	256	300 Кбайт	469 Кбайт	768 Кбайт	1,25 Мбайт

Бит/пиксел	Количество цветов	640×480	800×600	1024×768	1280×1024
15	32 768	600 Кбайт	938 Кбайт	1,5 Мбайт	2,5 Мбайт
16	65 536	600 Кбайт	938 Кбайт	1,5 Мбайт	2,5 Мбайт
24	16 777 216	900 Кбайт	1,37 Мбайт	2,25 Мбайт	3,75 Мбайт
32[1]	16 777 216	1,172 Мбайт	1,83 Мбайт	3,0 Мбайт	5,0 Мбайт

[1] В режиме 32 бит/пиксел для цветопередачи используются только 24 младших бита.

Помимо аппаратно выделенной видеопамяти, устанавливаемой на графических адаптерах, существует и архитектура однородной памяти *(UMA).* В архитектуре UMA под видеобуфер выделяется область системного ОЗУ, что позволяет несколько удешевить компьютер. Однако эта экономия приводит к снижению производительности как графической подсистемы, так и компьютера в целом. Диаметрально противоположным подходом, нацеленным на повышение производительности, является не просто выделение видеопамяти, а еще и применение в ней микросхем динамической памяти со специальной архитектурой. Сначала применяли асинхронную память *VRAM, WRAM* и *MDRAM* [1]. В современных адаптерах используются синхронная память *RDRAM* и разновидности *SGRAM* (см. 8.2).

Трактовка данных видеопамяти зависит от используемого видеорежима. В текстовом режиме каждому знакоместу экрана соответствует слово видеопамяти, расположенное по четному адресу. При этом младший байт слова (байт с четным адресом) содержит ASCII-код символа, а старший байт — его атрибуты. Организация памяти в этом случае является линейной: цепочка слов соответствует собранной в цепь последовательности строк символов. В графическом режиме возможны разнообразные варианты организации видеопамяти; современные адаптеры (начиная с VGA) имеют линейную организацию. В старых адаптерах использовалась и многоплоскостная модель памяти [1].

Контроллер атрибутов управляет трактовкой цветовой информации, хранящейся в видеопамяти. В текстовом режиме он обрабатывает информацию из байтов атрибутов знакомест (откуда и пошло его название), в графическом — из битов текущего выводимого пиксела. Контроллер атрибутов позволяет увязать объем хранимой цветовой информации с возможностями монитора. Для монохромных (не полутоновых) мониторов часть цветовой информации может описывать такие элементы оформления, как мигание, подчеркивание и инверсия знакоместа. В состав контроллера атрибутов входят *регистры палитр*, которые служат для преобразования цветов, закодированных битами видеопамяти, в реальные цвета на экране. С появлением адаптеров VGA на плату графического адаптера из монитора «переехали» цифроаналоговые преобразователи (ЦАП) сигналов базисных цветов, и появилась возможность отображения чуть ли не бесконечного количества оттенков, кодируемых аналоговыми сигналами. Однако реальное число цветов ограничивается разрядностью ЦАП базисных цветов, которая поначалу составляла 6 бит на каждый канал, что позволяет задавать 2^{18} цветов. Для того чтобы отобразить 256 кодов цвета (8 бит на пиксел) в эти 2^{18} цветов в адаптер перед ЦАП ввели программируемые регистры, с по

мощью которых каждому из 256 кодов ставится в соответствие свой набор битов, посылаемый на схемы ЦАП базисных цветов. Функционально оказалось целесообразным объединить эти регистры, представляющие собой небольшое быстродействующее ОЗУ (RAM), с цифроаналоговыми преобразователями (Digital-to-Analog Converter, DAC). Эта функциональная сборка в настоящее время исполняется в виде микросхем *RAMDAC* (Random Access Memory Digital to Analog Converter). Схему преобразования с использованием RAMDAC иллюстрирует рис. 10.10. Номер регистра RAMDAC, из которого берется цвет текущего отображаемого пиксела, в режиме 256 цветов задают в видеопамяти 8-битным кодом цвета пиксела (рис. 10.10, *а)*.

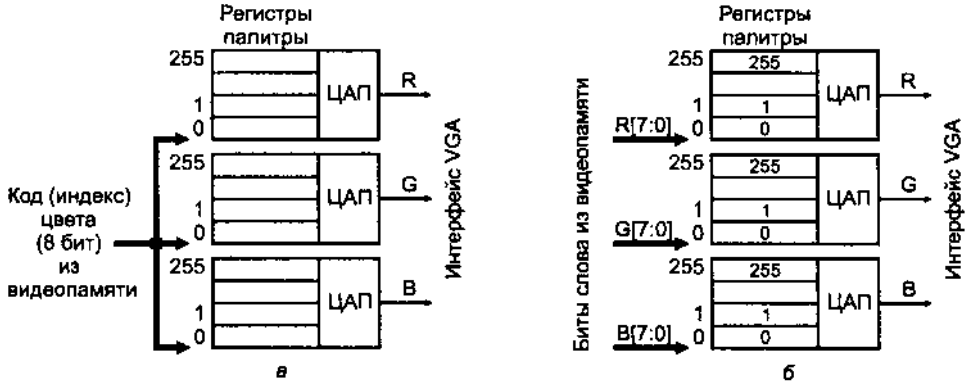


Рис. 10.10. Схема преобразования цветов через регистры палитр и RAMDAC: а — в режиме VGA, *б* — в режимах High Color и True Color

Казалось бы, для режимов High Color (15-16 бит/пиксел), а тем более True Color (24 бита) табличное преобразование цветов уже не требуется и биты каждого цвета можно подавать прямо на входы своего ЦАП. Однако если перед каждым ЦАП поставить отдельный блок регистров — ОЗУ объемом 256 x 8, адресуемое битами данного цвета, — то можно выполнять гамма-коррекцию цвета аппаратными средствами адаптера (рис. 10.10, *б)*. *Гамма-коррекция* требуется для увязки способностей цветопередачи дисплея с линейной математической моделью цветообразования графических приложений. В RAMDAC загружают таблицу, с помощь которой в выходной сигнал вводятся искажения, компенсирующие нелинейность дисплея. Однако разные типы дисплеев могут иметь разные передаточные характеристики, что для особо высоких требований к верности цветопередачи должно учитываться при программировании RAMDAC. Требование загрузки RAMDAC для всех режимов было прописано уже в спецификации PC99.

Микросхемы RAMDAC характеризуются *разрядностью* преобразователей, которая может доходит до 8 бит на цвет, и предельной *частотой* выборки точек (DotCLK), с которой они способны работать. Естественно, чем точнее должно быть преобразование, тем труднее его выполнить быстро. Трудности и договорные достижение высокого разрешения при большой частоте строчной (прогрес

сивной) развертки (эти факторы требуют максимального быстродействия RAM- DAC) с большой глубиной цвета (требующей исключительной точности преобразования) связаны и с этой причиной. Современные графические адаптеры, ориентированные на высокие разрешение и частоту развертки, имеют RAMDAC с частотой порядка 350 МГц и даже выше.

Знакогенератор предназначен для формирования растрового изображения символов в текстовом режиме экрана. Знакогенераторы адаптеров MDA/HGC и CGA программно недоступны — они выполнены в виде микросхем ПЗУ, никак не отображаемых в адресном пространстве процессора. Знакогенераторы адаптеров EGA и VGA размещаются во втором слое видеопамяти и поэтому программно доступны. При инициализации адаптера они загружаются из образов, хранящихся в ПЗУ расширения BIOS, установленных на платах графических адаптеров. Адаптер VGA позволяет одновременно хранить до восьми таблиц по 256 символов, активной (используемой для отображения) может быть либо одна из них, либо сразу две. В последнем случае набор одновременно отображаемых символов расширяется до 512, а одна из двух таблиц, требуемая для конкретного символа, определяется битом 3 его байта атрибутов.

Таблицы имеют 32-байтную развертку каждого символа в формате 16 x 16, из которой в EGA используется матрица 8 x 14, а в VGA — 9 x 16. Если таблицу знакогенератора (шрифты) для EGA загрузить в VGA, символы будут выглядеть мелковато, а в линиях, нарисованных символами псевдографики, появятся разрывы. Если же шрифты для VGA загрузить в EGA, то символы будут выглядеть усеченными (особенно снизу). Программная доступность знакогенератора снимает необходимость аппаратной русификации адаптера, но при желании можно переписать русифицированные шрифты в BIOS графического адаптера (не забыв исправить контрольную сумму в последнем байте ПЗУ). Такая процедура избавит от необходимости загрузки резидентного русификатора, занимающего место в памяти. Поскольку знакогенератор расположен в одном из слоев видеопамяти, после использования большинства графических режимов его содержимое приходится перезагружать, а встроенный драйвер BIOS по умолчанию возьмет образ, хранившийся в ПЗУ адаптера. Если туда подставить нужный шрифт, то дополнительный драйвер экрана не потребуется.

Графический контроллер является средством повышения производительности программного построения изображений (точнее — их образов) в видеопамяти. В первых графических адаптерах (CGA и HGC) этот блок, фактически, отсутствовал. Он оформился в адаптере EGA, откуда перекочевал и в VGA. В этих адаптерах его функции реализованы аппаратными средствами специализированных микросхем. Рассмотрим функции графического контроллера адаптеров EGA и VGA. Он работает с четырехслойной моделью организации видеопамяти. Адаптеры EGA и VGA имеют четыре 8-битных *регистра-защелки*, в которых фиксируются данные из соответствующих им цветовых слоев при выполнении любой операции чтения видеопамяти. В последующих операциях записи при формировании данных для каждого слоя могут принимать участие данные от процессора (1 байт) и данные из регистров-защелок соответствующих слоев (рис. 10.11). Данные от процессора могут быть предварительно циклически

сдвинуты. Над данными процессора (возможно, сдвинутыми) и регистров-защелок могут выполняться *логические операции* И, ИЛИ и ИСКЛЮЧАЮЩЕЕ ИЛИ. Кроме того, вместо результатов этих операций в некоторые слои могут быть записаны байты нулей или единиц. *Регистр битовой маски* позволяет побитно управлять источником записываемых данных: если бит регистра маски имеет нулевое значение, то в видеопамять этот бит во всех слоях будет записан из регистра-защелки. Данные от процессора (логически обработанные) будут поступать только для битов с единичным значением маски. И наконец, запись будет производиться только в разрешенные слои; правда, функция разрешения слоев относится уже к синхронизатору.

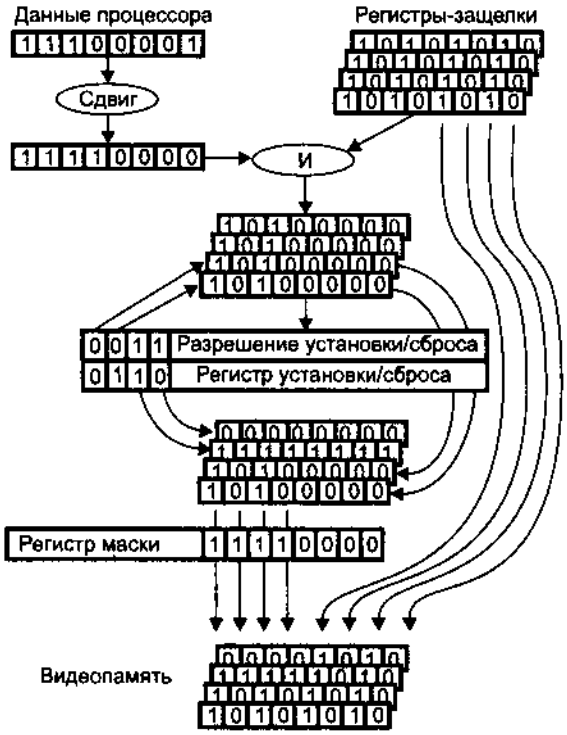


Рис. 10.11. Тракт записи графического контроллера EGA/VGA

При чтении графический контроллер может задать номер читаемого слоя. Возможно и *чтение со сравнением цветов*. В этом случае указывается код искомого цвета (значение битов для соответствующих слоев), и результатом чтения сразу всех слоев станет байт, у которого единичное значение примут биты тех пикселов, чей цвет совпадает с образцом (рис. 10.12). В сравнении цветов могут участвовать и не все слои.

Всеми функциями графического контроллера управляют через его регистры. Конечно, возможно и прямое обращение к отдельному цветовому слою как по чтению, так и по записи. Но знание возможностей графического контроллера

позволяет многие часто используемые функции возложить на его аппаратные средства. Однако если такой аппаратный графический контроллер еще приемлем для четырехслойной организации (4 бита на пиксел), то для более глубоких цветов (8 бит на пиксел и более) он оказывается уже слишком громоздким. В современных адаптерах функции графического контроллера, существенно расширенные по сравнению с EGA и VGA, выполняются встроенным микропроцессором — *графическим акселератором.* Возможные функции графического акселератора (без претензии на полноту перечисления) были рассмотрены ранее.

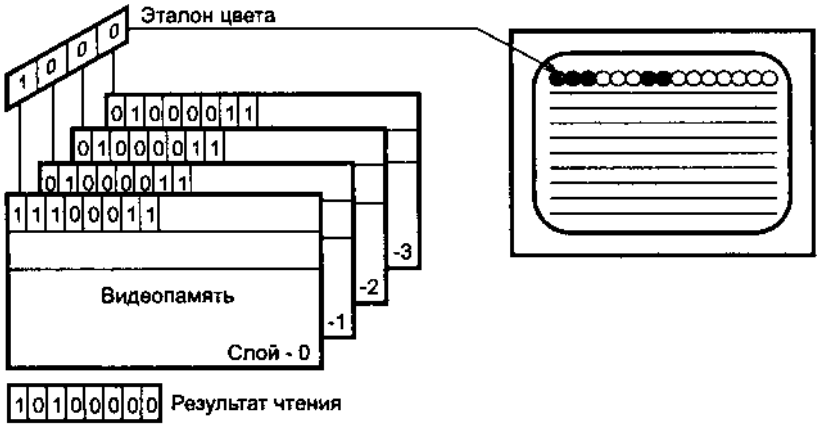


Рис. 10.12. Считывание со сравнением цвета

Синхронизатор, появившийся в адаптере EGA, позволяет синхронизировать циклы обращения процессора к видеопамяти с процессом регенерации изображения. Адаптеры имеют собственные кварцевые генераторы синхронизации (иногда несколько). От внутреннего генератора вырабатывается частота вывода пикселов *DotClock* (DotCLK), относительно которой строятся все временные последовательности сканирования видеопамяти, формирования видеосигналов и синхронизации монитора. В то же время процессор обращается к видеопамяти асинхронно относительно процесса регенерации. В задачу синхронизатора входит согласование этих асинхронных процессов. В адаптерах SVGA для шины PCI в качестве опорной для некоторых видеорежимов может использоваться частота 33 МГц прямо с шины, а циклы обращения процессора и так уже привязаны к этому синхросигналу. Таким образом, задача синхронизатора упрощается. Не стоит удивляться, когда смена процессора (при которой иногда приходится менять системную частоту) ведет к изменению геометрии изображения на экране монитора. Но это довольно редкий случай — большинство карт синхронизируется от собственного генератора. Правда, монитор с хорошей автоматической подстройкой генераторов может и скомпенсировать изменения частот синхронизации.

Внутренняя шина адаптера предназначена для высокопроизводительного обмена данными между видеопамятью, графическим акселератором и внешним ин

терфейсом. Типовая разрядность канала данных у этой шины сейчас составляет 64, 128 и даже 256 бит. Однако реально используемая разрядность может оказаться меньше, если установлены не все предусмотренные микросхемы видеопамяти.

Блок внешнего интерфейса связывает адаптер с одной из шин компьютера. Если раньше для графических адаптеров предназначалась шина ISA (8 или 16 бит), то современные графические адаптеры используют в основном высокопроизводительные шины. Локальная шина VLB довольно быстро сошла со сцены вместе с процессорами класса 486. В настоящее время для этих целей перспективна шина PCI-E, сменяющая на этом поприще порт AGP. Графический контроллер встраивается и в некоторые чипсеты системных плат.

Блок интерфейса монитора формирует выходные сигналы соответствующего типа (RGB-TTL, RGB-Analog, композитный видеосигнал или S-Video). Этот же блок отвечает за диалог с монитором: в простейшем случае — чтение битов идентификации (для VGA-мониторов), а в более сложном — обмен данными по каналу DDC. Идентификация типа подключенного монитора VGA может производиться и по уровню видеосигнала на выходах красного или синего цвета: монитор имеет терминаторы (75 Ом) на каждом из аналоговых входов. Такая нагрузка при подключении снижает напряжение выходного сигнала. У монохромного монитора используется только канал зеленого цвета — линии красного и синего остаются без нагрузки. Этот факт может зафиксировать интерфейсный блок и сообщить системе об обнаружении монохромного монитора. Правда, бывают и конфузы: если у цветного монитора отключить терминаторы (некоторые большие мониторы позволяют это сделать), то система примет его за монохромный.

Видеокомпоненты могут включать аппаратную поддержку различных кодеков (чаще всего — MPEG-декодер для воспроизведения фильмов с CD и DVD), средства поддержки видеооверлеев, фрейм-граббер, TV-тюнер.

Адаптеры VGA появились в то время, когда развитие элементной базы уже позволило задуматься над обработкой видеосигналов в PC. Для реализации внешнего видеооверлея потребовалось перехватывать поток двоичных данных сканируемых пикселов, что и обеспечивает подключение к дополнительному разъему видеосигнала *VGA Auxiliary Video Connector.* Этим краевым 26-контакт - ным разъемом снабжались почти все модели адаптеров VGA. С помощью этого интерфейса осуществляют связь графического адаптера с видеооверлейными платами (видеобластерами). Впоследствии был стандартизован разъем *VESA Feature Connector* (VFC), у которого назначение сигналов практически сохранилось, но вместо ламельного краевого стали применять двухрядный штырьковый разъем. Этот разъем графического адаптера VGA и SVGA позволяет получать поток байтов данных сканируемых пикселов при работе адаптера в режиме до 640 x 480 пикселов x 256 цветов. Разъем графического адаптера связывается с таким же разъемом видеоплаты плоским кабелем-шлейфом. Для режимов до 1024 x 768 с глубиной цвета High Color и True Color предназначен разъем VAFC (*VESA Advanced Feature Connector).* Он имеет разрядность 16/32 бит и при максимальной частоте точек 37,5 МГц обеспечивает скорость потока дан

ных 150 Мбайт/с. В 16-битной версии VAFC используются первые 56 контактов, а в 32-битной — все 80 контактов разъема, у которого плотность размещения контактов больше, чем у VFC.

Кроме этих стандартов существует и специальная внутренняя 32-битная шина для обмена данными между мультимедийными устройствами — *VESA Media Channel* (VM Channel). Эта шина (канал), в отличие от рассмотренных ранее двухточечных интерфейсов, ориентирована на широковещательную передачу данных между несколькими абонентами. Если монитор подключается не к графическому адаптеру, а к дополнительной видеооверлейной карте, то в ценообразований участвует регистр палитр не графического адаптера, а этой дополнительной карты. При этом приходится применять специальные меры для обеспечения одновременной записи в регистры палитр карт, расположенных на разных шинах. Для этого в PCI существует механизм VGA Palette Snoop (см. 14.6).

Программные модели стандартных адаптеров

Графические адаптеры PC прошли путь развития от MDA до SVGA [1]; на последнем названии, похоже, произошла длительная остановка, поскольку оно довольно всеобъемлюще. Взаимодействие программ с графическими адаптерами осуществляется путем обращения к их регистрам и видеопамяти. Поскольку новые модели адаптеров должны были обеспечивать совместимость со своими предшественниками на уровне непосредственного общения с регистрами и памятью, регистры первых адаптеров MDA/HGC, CGA и EGA входят в состав регистров современных адаптеров VGA и SVGA. В первых адаптерах многие управляющие регистры были доступны только для записи, что вызывало неудобства при переключении режимов, связанных со сменой задач. В адаптерах VGA и SVGA, появившихся, когда персональные компьютеры «повзрослели» и доросли до многозадачности, эти же регистры доступны и по чтению. Первоначально адреса регистров были жестко фиксированными, причем адаптер EGA мог иметь блок регистров в одной из двух областей: EGA#1 — 3C0-3DF, EGA#2 — 2C0-2DF, что позволяло устанавливать в систему два адаптера. Адаптерам SVGA для шины PCI, которой свойственна поддержка программного конфигурирования ресурсов (PnP), доступна возможность перемещения блока своих регистров в некотором диапазоне адресов. При этом положение регистров относительно базового адреса, естественно, сохраняется. Таким образом, в систему бесконфликтно могут быть установлены несколько адаптеров. Назначение регистров адаптеров приведено в [1].

Видеопамять графических адаптеров традиционно приписывается к области A0000h-BFFFFh, относящейся к верхней памяти (UMA). При этом может быть занята и не вся область (табл. 10.7), а при объеме видеопамяти свыше 128 Кбайт доступ к полному объему реализуется посредством переключения доступных страниц. Страница отображается на окно размером 64 Кбайт, через которое процессор получает доступ к видеопамяти. Для переключения страниц имеется специальная функция (4F05h) видеосервиса BIOS. Некоторые графические адаптеры с шиной ISA позволяют отображать свою видеопамять в об

ласть под границей 16 Мбайт, а адаптеры PCI могут ее перемещать в любую область в пределах адресуемого шиной пространства (4 Гбайт при 32-битной адресации, 64 Гбайт при 36-битном физическом адресе). Целью этого перемещения является обеспечение непрерывности доступа к любой ячейке видеопамяти без переключения страниц. Адаптеры, начиная с EGA, имеют *модули расширения BIOS*.

Таблица 10.7. Видеопамять и расширение BIOS графических адаптеров

Адаптер	Видеопамять (Video RAM)		Расширение BIOS (Video ROM)	
	Объем, Кбайт	Область в UMA	Объем, Кбайт	Область в UMA
CGA	16	B8000h–BBFFFh	–	–
MDA	4	B0000h–B0FFFh	–	–
HGC	16	B0000h–B3FFFh	–	–
EGA	128	A0000h–BFFFFh	16	C0000h–C3FFFh
VGA, SVGA	128 Кбайт – 64 Мбайт и более	A0000h–BFFFFh	32/48/64	C0000h–C7FFF/ CAFFFh/CFFFFh

При обращении к видеопамяти простейших адаптеров (CGA, MDA и HGC) со стороны шины во время видимой части прямого хода развертки на экране возникал «снег», поскольку видеопамять этих адаптеров «на два фронта» работать не могла. Без «снега» взаимодействие было возможно только во время обратного хода. Для синхронизации программы с разверткой в начале обратного хода по кадру адаптеры могли вырабатывать запросы аппаратного *прерывания* по линии IRQ2/9. Начиная с адаптера EGA необходимость в синхронизации практически отпала, поскольку в структуру адаптера ввели специальный узел — синхронизатор, примиряющий интересы процессора и контроллера ЭЛТ. Тем не менее, возможность генерации прерывания сохранилась, но карты PCI могут этот запрос посылать на любую доступную им линию. В будущем прерывания от графического адаптера планируется изжить, что указывалось еще в спецификации PC97 компании Microsoft.

Адаптеры с интерфейсами PCI, AGP и PCI-E

Адаптеры для шин PCI, AGP и *PCI-E* имеют следующие отличия от своих предшественников для шины ISA:

- высокая скорость передачи данных;
- возможность перемещения занимаемых ресурсов (адресов портов и буферной памяти) во всем диапазоне 64К портов и 4 Гбайт памяти;
- возможность использования полноразмерного окна для доступа ко всей буферной памяти (видеопамяти) без переключения страниц.

Для программной совместимости эти адаптеры по умолчанию инициализируются так, что занимают стандартные ресурсы адаптеров VGA, поэтому в работе они от них практически не отличаются. Однако есть нюанс в установке адаптера, связанный с механизмом распределения линий запросов прерываний на шине PCI (см. 14.5). Может оказаться, что адаптер разделяет линию прерыва

ния с другим устройством PCI, и при некорректной программной (а иногда и аппаратной) поддержке будут возникать конфликты. Разрешить эти конфликты поможет перестановка видеокарты (или карты, конфликтующей с ней) в иной слот PCI (для карт ISA такая перестановка была бы бессмысленной). Конфликта может и не быть, особенно если графическая карта прерываниями не пользуется.

Благодаря 32-разрядной (а можно и 64-разрядной) адресации локальная память может быть отображена целиком в одну область памяти, не занятую системным ОЗУ, ПЗУ и специальной памятью других устройств. Естественно, это должна быть область выше первого мегабайта. Однако эта область в реальном режиме (и режиме V86) процессору x86 недоступна. Для совместимости с традиционными графическими адаптерами часть локальной памяти (экранный буфер) отображается в UMA, на традиционное место видеопамяти, и для стандартных видеорежимов сохраняются традиционные механизмы переключения страниц VGA.

Если в компьютер помимо графического адаптера установлена отдельная карта для работы с видеоизображениями, в CMOS Setup включают параметр VGA PaLLette Snooping. При этом программные обращения к палитрам VGA (записи) будут достигать не только графического адаптера, но и дополнительной карты видеообработки. Для карт-комбайнов, в которых графические функции и видеофункции совмещены, этого не требуется.

Карты с *интерфейсом AGP с* точки зрения конфигурирования выглядят как устройства PCI, но с дополнительными возможностями. Порт AGP был введен исходя из потребностей ЗО-акселераторов в большом объеме доступной им высокопроизводительной памяти, участвующей в трехмерных построениях. Адаптер содержит специализированный графический процессор; локальную память, используемую как видеопамять, и локальное ОЗУ графического процессора; управляющие и конфигурационные регистры, доступные как локальному, так и центральному процессорам. Графический процессор может обращаться и к локальной памяти, и к системному ОЗУ, в котором для него могут храниться наборы данных, не умещающиеся в локальной памяти (как правило, текстуры большого объема). Основная идея AGP заключается в предоставлении графическому процессору максимально быстрого доступа к системной памяти (локальная ему и так близка), более приоритетного, чем доступ к ОЗУ со стороны других устройств. Графический процессор является мастером (ведущим устройством) порта AGP, свои запросы он может выполнять как в режиме AGP, так и в режиме PCI (см. 14.9), поддерживая или не поддерживая такие свойства AGP, как внеполосная адресация (SBA), режимы обмена 2x/4x/8x. Кроме того, адаптер является ведомым устройством PCI, для которого может поддерживаться (или не поддерживаться) быстрая запись (fast write) со стороны процессора.

Устройство AGP, как правило, реализует функции адаптера VGA и в начале теста POST конфигурируется и инициализируется для работы в этом режиме (по крайней мере, на время загрузки ОС). Для сосуществования с адаптером MDA устройство AGP не должно занимать память в адресах 0B0000-0B7FFFh

и порты 3B4h, 3B5h, 3B8h, 3B9h, 3Bah и 3BFh, включая их адресные псевдонимы на шине ISA (со всеми возможными значениями A[15:10]). Если на этапе инициализации адаптер MDA не обнаружен (его можно найти попыткой записи и последующего чтения памяти в области B0000-B7FFFh), то устройство AGP может использовать и вышеуказанные ресурсы. В зависимости от обнаруженных адаптеров MDA (на шине ISA) и VGA (на шине PCI) чипсет системной платы (главный мост) может быть настроен так, чтобы направлять обращения к стандартным ресурсам этих адаптеров либо на соответствующую шину (ISA или PCI), либо на порт AGP.

Поскольку в AGP используются довольно сложные механизмы взаимодействия с системной платой, работающие на высоких скоростях, здесь довольно часто встречаются разного рода проблемы несовместимости. Иногда для стабильности приходится отключать такие фирменные свойства AGP, как SBA, 2x/4x/8x, Fast Writes. Приобретая карту AGP, имеет смысл поинтересоваться ее совместимостью с конкретной системной платой, особенно это актуально для плат с сокетом Super 7. Вопрос совместимости может встать и при замене процессора на системной плате. Частично или полностью вопросы совместимости могут быть решены прошивкой новой версии BIOS и графической карты, а также настройкой параметров Setup (иногда ценой снижения производительности). Современные графические карты стали одним из объектов разгона (overclocking), поскольку позволяют изменять частоты ядра акселератора и шины локальной памяти. Это делают для того, чтобы на одну и ту же плату можно было устанавливать компоненты с разным быстродействием. Частота шины памяти SD- RAM/SGRAM должна соответствовать спецификации быстродействия установленных микросхем памяти. Иногда может оказаться полезным и снижение этих частот.

С картами AGP связаны перечисленные ниже параметры CMOS Setup, влияющие на производительность и стабильность работы:

- Video Memory (атрибуты видеопамяти) — UC (некэшируемая) или UCWC (некэшируемая с комбинированием записей). Тип UCWC несколько повышает производительность.
- Fast Writes — возможность быстрой записи в устройство AGP.
- AGP Mode — максимально доступный режим обмена (1x/2x/4x/8x).
- AGP Aperture (апертура AGP) — объем памяти, доступной акселератору в DIME (см. выше).
- Assign IRQ to VGA — назначение линии прерывания графической карте.
- Video BIOS Shadow, Video BIOS Cacheable, C8000-CFFFF Shadow — теневая память и кэшируемость области, занимаемой VideoBIOS. Эти области должны быть запрещены.
- Peer Concurrency, Concurrent PCI Host, PCI Streaming — одновременность выполнения обменов на разных шинах и потоковый режим PCI. Разрешение этих режимов повышает производительность.

Для мощных графических карт характерна *высокая потребляемая мощность*. Акселератор, а также быстродействующая память требуют охлаждения. Для

этого их снабжают пассивными (радиатор) или активными (радиатор с вентилятором) средствами охлаждения. При недостаточном охлаждении возможен перегрев, приводящий к сбоям и даже выходу устройств из строя. Из-за вентилятора и радиатора может оказаться недоступным соседний слот PCI (а то и не один). Вентилятор является дополнительным источником шума и периодически требует чистки от пыли и смазки.

Мощные (по производительности и потреблению) акселераторы могут вызывать проблемы с питанием, для их работы может оказаться недостаточной мощность рядового блока питания на 150-200 Вт. Требуемая мощность блока определяется всеми компонентами ПК — процессором, памятью, дисками, акселератором, и чем более они быстродействующие, тем и более «прожорливые». Недостаток мощности может проявляться в виде сбоев. Проверить, хватает ли мощности, можно замером напряжений +12 и +3,3 В на включенном адаптере — основная мощность потребляется именно по этим шинам. Подозрения в нехватке мощности можно проверить, отключив какие-либо устройства (например, «лишний» быстроходный винчестер).

Мультидисплейные системы

В большинстве случаев компьютер имеет один дисплей (монитор), подключенный к единственному выходу единственного адаптера. Однако есть возможность подключения и использования (даже одновременного) двух и даже более дисплеев, подключенных к одному или нескольким адаптерам. Архитектура первых PC с шиной ISA допускала установку до *двух графических адаптеров* с раздельными мониторами. Аппаратного конфликта ресурсов не будет, если один из адаптеров — MDA или HGC, а другой — CGA, EGA или VGA. При этом оба адаптера поддерживаются функциями видеосервиса. Консоль DOS, работающая через BIOS в телетайпном режиме, выходит на адаптер, являющийся текущим. Видеосервис BIOS работает с графическим адаптером, обслуживающим текущий видеорежим. В отличие, скажем, от дискового сервиса, здесь нет нумерации устройств с возможностью обращения к требуемому. Текущий адаптер определяется по значению битов 4 и 5 флага конфигурации в BDA (0000:0410h). Команды MODE MONO и MODE C080, выданные с консоли DOS, делают текущим адаптер MDA/HGC или CGA/EGA/VGA соответственно. Пакеты САПР и другие приложения, для которых такое разделение может быть эффективным, работают с адаптерами через собственные драйверы.

Графические адаптеры класса SVGA, поддерживающие технологию PnP, могут обеспечивать перемещение адресов требуемых системных ресурсов, и, следовательно, в системе может присутствовать несколько таких адаптеров. Для дисплейных адаптеров PCI и AGP проблема конфликтов решается благодаря способности к перемещению всех ресурсов (хотя ряд старых графических адаптеров в полной мере эту способность не поддерживает). Существуют также и многоканальные графические карты, которые содержат несколько независимых однотипных графических адаптеров. Такие карты применяют в многотерминальных компьютерах на базе PC.

В современном ПК может присутствовать несколько графических адаптеров, расположенных на картах расширения ISA, PCI, AGP или/и на системной плате. Тест POST определяет первичное устройство вывода, выполняя поиск в следующем порядке: карты ISA, карты PCI (включая AGP), устройства системной платы (включая и встроенные AGP). Первичным дисплеем (загрузочным устройством вывода) станет первое из обнаруженных в этом поиске. Такой порядок позволяет при необходимости сменить первичное устройство вывода, например в диагностических и отладочных целях. Однако инициализируются все обнаруженные устройства, и всем им по возможности выделяются системные ресурсы. Так что если адаптер устанавливается с целью замещения интегрированного в системную плату, последний имеет смысл отключить настройками CMOS Setup. Однако этот прием не всегда срабатывает, и якобы отключенный адаптер может конфликтовать с адаптером, установленным в слот расширения.

На дешевых адаптерах имеется лишь один разъем для подключения монитора — чаще всего DB-15 с аналоговым интерфейсом VGA. На многих современных графических адаптерах предусмотрена возможность подключения двух мониторов, но они обладают двумя интерфейсами. Один из них, как правило, VGA, другой может быть тоже VGA, но чаще — DVI, интерфейс плоского дисплея или один из стандартных телевизионных (композитный или S-Video), Интерфейсы могут быть независимыми, полностью или частично (иметь разные частоты синхронизации, что актуально, например, для одновременного использования VGA-монитора и телевизора). Под полной независимостью подразумевается и независимость отображаемых видеобуферов. Для этого в адаптере должно быть два блока контроллера ЭЛТ, две микросхемы RAMDAC. Для поддержки этих возможностей требуется специальный драйвер, «понимающий» особенности адаптера и умеющий донести эти возможности до ОС и приложений. Доступны, например, такие варианты использования:

- DualHead Clone, DualHead TVOut, Monitor Clone — дублирование изображения основного монитора на другом;
- DualHead Multi-Display, Extended Desktop — расширение «рабочего стола» на два монитора;
- DualHead ZOOM — вывод на второй монитор выделенной части изображения;
- DualHead DVDMax — воспроизведение DVD-фильма на втором мониторе независимо от вывода на основной монитор.

Существуют графические адаптеры с двумя выходными интерфейсными разъемами — VGA и дисплейной панели (DFP), но с одним контроллером ЭЛТ (узлом, обеспечивающим сканирование видеопамяти для регенерации изображения). У них одновременно оба интерфейса работать не могут, и при смене типа дисплея приходится программно или аппаратно переключать выходной интерфейс. Программное переключение типа интерфейса чревато казусами: ошибочное переключение на интерфейс DFP (он может называться и TFT) при наличии лишь обычного монитора приведет к тому, что последующие попытки перезагрузки будут происходить со «слепым» экраном. Если это произойдет

с интегрированным адаптером, положение спасет временная установка любого графического адаптера в слот расширения — микросхема BIOS должна воспринять его как загрузочное устройство вывода, с помощью которого настройкой CMOS Setup можно восстановить нормальное положение вещей.

10.6. Видеосервис BIOS

Дисплейный адаптер как обязательный компонент PC имеет поддержку основных функций в BIOS. Эти функции выполняются через вызов программного прерывания Int 10h — *видеосервиса BIOS.* Видеосервис позволяет устанавливать и переключать видеорежим, выполнять вывод символов и пикселов, очищать и прокручивать экран универсальными способами без особой оглядки на установленный видеорежим и выполнять некоторые другие функции. Видео- сервис BIOS необходим по крайней мере до загрузки операционной системы, которая в дальнейшем может работать с графическим адаптером напрямую; через собственные загружаемые драйверы, специфические для конкретного адаптера. Драйверы для адаптеров MDA и CGA по традиции встроены в системную микросхему BIOS (по крайней мере, у всех компьютеров, имеющих шину ISA). Программная поддержка графических адаптеров, интегрированных в системную плату, также встроена в системную микросхему BIOS. Все остальные адаптеры имеют собственный *модуль расширения BIOS* (Video BIOS), в котором хранятся коды драйверов видеосервиса (INT 10h) и таблицы знакогенераторов. Этот модуль появился с адаптерами EGA и VGA и обеспечивает возможность установки любой карты без размышлений о проблемах программной совместимости. Модуль расширения получает управление для инициализации графического адаптера почти в самом начале теста POST (до тестирования основной памяти), и его заставка появляется на экране до заставки системной микросхемы BIOS. Модуль имеет начальный адрес C0000h, его размер зависит от модели адаптера: конечный адрес EGA BIOS — C3FFFh, VGA BIOS — C7FFF/ CFFFFh. Поскольку для расширения BIOS применяют 8-разрядные микросхемы ПЗУ, время доступа к которым существенно превышает время доступа микросхем ОЗУ, для повышения производительности видеопостроений применяют затенение (Video BIOS Shadowing) или кэширование (Video BIOS Caching). В новых адаптерах BIOS оформляется в соответствии с DDIM (см. 5.3), и в рабочем режиме код BIOS располагается в ОЗУ, так что затенение и кэширование Video BIOS должно быть отключено в CMOS Setup.

У современных адаптеров программируется множество параметров — частоты и фазовые сдвиги разверток, частоты работы ядра акселератора и его шины памяти (эти частоты могут быть относительно независимыми) и другие аппаратные параметры. Для адаптеров, интегрированных в системную плату, ряд параметров можно настраивать через CMOS Setup. Для некоторых адаптеров, выполненных в виде карт расширения, существуют интерактивные версии BIOS — у карты имеется своя программа Setup, с помощью которой можно настраивать параметры BIOS. В эту программу можно войти по «горячим» клавишам. Параметрами можно управлять и с помощью специальных утилит (например, зада

вать частоту развертки, управлять параметрами AGP-адаптера), загружаемых явно или вызываемых через диалоговые окна свойств адаптеров в ОС Windows.

Современные адаптеры имеют карту флэш-BIOS, что позволяет пользователю заменять ее версию прямо в компьютере. Для этого требуются только утилита программирования, соответствующая данной модели карты, и файл с образом требуемой версии BIOS. Утилита программирования может работать только в среде «чистой» ОС DOS (при загрузке Windows 9x в этот режим можно попасть, нажав клавиши Shift+F5 или F8), причем затенение видео-BIOS должно быть запрещено в Setup (Video BIOS Shadowing — Disabled). Во время программирования экран может погаснуть (это нормально). Сама по себе процедура обновления несложна, но есть ряд опасностей:

- Во время программирования нельзя нажимать кнопку Reset и выключать компьютер — это приведет к потере работоспособности BIOS, а следовательно, и адаптера.
- Внезапное пропадание питания, а также случайный сбой ведет к тем же последствиям.
- Загрузка некорректного образа (непригодного для данного адаптера или поврежденного) ведет к тем же последствиям.

Поводом к смене BIOS могут быть: выход новой версии с исправлением ошибок, мешающих работе с существующей версией BIOS; проблемы совместимости с новым ПО или даже новым процессором и т. п. Без особой необходимости прошивать новую версию, пожалуй, не стоит: во-первых, есть некоторый риск; во-вторых, смена прошивки может повлечь за собой новые ошибки, а также отказ поставщика карты от гарантийных обязательств.

Перед повторной прошивкой следует сохранить копию старого образа BIOS, для чего существуют специальные утилиты. Если при программировании все-таки произошла одна из вышеперечисленных неприятностей и компьютер грузится с черным экраном (неоднократно и после повторного включения питания), стоит попытаться снова переписать ту же новую версию или же сохраненную старую. Для этого можно попробовать установить дополнительную графическую карту попроще (PCI или ISA в дополнение к карте AGP или любой интегрированной; ISA для обхода карты PCI) — есть шанс, что она станет загрузочным устройством вывода (см. далее), — и снова запустить утилиту. Если это не удается, но компьютер загружается, то придется делать попытку запуска утилиты вслепую.

Вместо повторной прошивки новую версию BIOS можно опробовать, загружая ее резидентно в память (до загрузки Windows, из файла AUTOEXEC.BAT), например, утилитой VGABIOS (от NVIDIA). При запуске утилите требуется указать имя файла-образа загружаемой версии BIOS, для экономии стандартной памяти можно использовать команду LOADHIGH.

Первичной задачей BIOS графического адаптера является управление видеорежимом (BIOS Video Mode), определяющим формат экрана. Микросхема BIOS адаптера должна выполнять программирование всех стандартных и специфических управляющих регистров для установки (смены) требуемого видеорежима

и выбранных параметров развертки — о способах этих переключений остальное ПО может не знать.

Большинство видеорежимов стандартизованы, и каждому присвоен свой номер. Первоначально для задания номера режима отводился один байт, и режим устанавливался параметром функции 0 Int 10h (AH = 0, AL = Mode). Режимы 0— 13h являются стандартными для адаптеров MDA, CGA, EGA, VGA. Режимы 14h - 7Fh используются с нестандартными VGA- или SVGA-расширениями BIOS, они специфичны для конкретных моделей графических адаптеров. Позже появилось стандартизованное расширение функций видеосервиса *VBE* (VESA BIOS Extensions) для адаптеров VGA и SVGA и были определены новые видеорежимы с двухбайтными номерами выше 100h. Эти режимы устанавливаются параметром функции 4F02h Int 10h (AX = 4F02h, BX = VMode).

В пределах возможностей установленного видеорежима видеосервис предоставляет средства отображения информации на различных уровнях. Простейший для программиста телетайпный режим позволяет посылать поток символов, которые построчно отображаются на экране с отработкой символов возврата каретки, перевода строки, обеспечивая «прокрутку» изображения при заполнении экрана. Есть функции и для полноэкранной работы с текстом, при которой доступны и атрибуты символа. В графическом режиме имеется возможность чтения и записи пиксела с указанными координатами. Однако видео- сервисом Int 10h программисты пользуются далеко не всегда, поскольку работает он довольно медленно. Существенно ускорить функционирование видео- сервиса позволяет затенение области ROM BIOS, хранящей программный код драйверов. Однако самым быстрым способом видеопостроений, конечно же, является непосредственная работа с видеопамятью или непосредственное общение с акселератором графического контроллера, если таковой имеется. Так что у программиста (и пользователя) обычно есть выбор: работать, не задумываясь об архитектуре конкретного дисплейного адаптера, или работать медленно с конкретной моделью адаптера. Во втором варианте расплатой за скорость является необходимость заботиться о разработке драйверов для всех (или хотя бы популярных) моделей графических адаптеров. В современных операционных системах имеются программные интерфейсы, сочетающие унификацию с высокой производительностью (например, DirectX в Windows).

Подробно рассматривать интерфейс этих функций не будем (этому посвящены отдельные книги), отметим особо лишь *функцию телетайпного вывода* Int 10h (0Eh). При вызове AH = 0Eh в AL хранится код выводимого символа, в BL — цвет (только для графического режима). Символ выводится в текущую позицию курсора, и курсор сдвигается на следующую, переходя на новую строку после конца предыдущей и прокручивая экран при его заполнении. Специальные символы вызывают возврат на начало строки (CR, код 0Dh), перевод строки (LF, 0Ah) и короткий гудок (BEL, 07h). Этой функцией часто пользуются для вывода сообщений программами, работающими на нижнем уровне (например, модулями инициализации ПЗУ расширений BIOS, загрузчиками и другими программами, не имеющими доступа к сервисам операционных систем). Программа вывода получается простейшей, работает на всех адаптерах и во всех режимах, но довольно медленно.

10.7. Параметры видеосистемы

Рассмотрев работу видеосистемы, можно сформулировать и объяснить ее основные параметры, определяемые используемым дисплейным адаптером, дисплеем (монитором) и интерфейсом, их связывающим.

Разрешение (resolution), или разрешающая способность, в графическом режиме определяется количеством точек в строке по горизонтали и числом строк на экране (например, 800 x 600 — 800 точек, 600 строк). Чем больше разрешение, тем больше информации можно вывести на экран с приемлемым качеством изображения. Достижимое разрешение определяется монитором, графическим адаптером и средствами их соединения.

Максимальное разрешение ЭЛТ-монитора определяется размерами экрана и зерна, а также полосой пропускания его видеотракта. Для ЖК-мониторов рекомендованное разрешение — его размер матрицы; все остальные варианты (если поддерживаются) реализуются путем интерполяции с потерей качества.

Максимальное разрешение со стороны графического адаптера определяется в основном объемом установленной памяти и желаемым количеством цветов.

Разрешение, с которым нормально работает вся графическая подсистема, и допустимая частота развертки взаимосвязаны. Слишком высокая частота при высоком разрешении требует столь высокой частоты обработки пикселов, что с ней могут не справиться видеопамять, RAMDAC, интерфейсный кабель, ви- деотракт монитора и/или генератор строчной развертки. Эти ограничения, происхождение которых объяснено в данной главе, приводят к ухудшению четкости и появлению эхо-сигналов.

Частота регенерации (refresh rate), или *сканирования* (scan frequency), и режим сканирования определяют качество (устойчивость) выводимого изображения. Частота регенерации является частотой кадровой (вертикальной) развертки. При частоте кадровой частоте развертки 60 Гц и ниже изображение на ЭЛТ-мониторе мерцает, что особенно заметно на больших светлых изображениях. *Режим развертки*, или *сканирования* (scan mode): *NI* (Non-Interlaced) — прогрессивный, или построчный, *I* (Interlaced) — чересстрочный. В современных системах чересстрочная развертка применяется только в крайних случаях, когда требуется слишком высокое разрешение. Однако включение чересстрочной развертки — это уже намек на запредельное (для данной системы) разрешение. При чересстрочной развертке мерцают мелкие элементы изображений (особенно заметно на тонких горизонтальных линиях). В тех случаях, когда нет цели «выжать» из машины максимум производительности, для ЭЛТ-мониторов стремятся к построчной развертке с возможно более высокой частотой кадров (75 Гц и выше), но с оглядкой на полосу пропускания видеотракта: при чрезмерно высокой частоте четкость и насыщенность изображения снижаются. Для ЖК-мониторов слишком высокая частота не требуется — мелькания сглаживаются инерционностью матрицы.

В графическом адаптере предельное сочетание разрешения экрана и частоты регенерации определяется допустимой *частотой работы RAMDAC.* «Разгон» RAMDAC ведет к ухудшению качества изображения (четкости и насыщенности цветов). Лучшие модели адаптеров, обеспечивающие режимы до 2048 x x 1536, имеют RAMDAC с частотой до 350 МГц. Хорошие адаптеры имеют RAMDAC на 250 МГц.

Возможности цветопередачи характеризуются максимальным числом одновременно присутствующих цветов на экране и цветовой гаммой — количеством возможных цветов, отображаемых монитором.

Количество одновременно присутствующих цветов определяется количеством битов видеопамяти на элемент изображения в выбранном видеорежиме: CGA — **2** бита/4 цвета, EGA — 4 бита/16 цветов, VGA — 8 битов/256 цветов, SVGA — 15-16 битов (High Color, 32-64 К цветов) и 24-32 бита (True Color, 16 М цветов).

Цветовая гамма — дискретность оттенков выводимого цвета — для систем с ЭЛТ-монитором определяется *разрядностью RAMDAC* графической карты: 18-битный преобразователь RAMDAC имеет для каждого цвета 6-битные схемы ЦАП R, G и B, не позволяющие вывести все оттенки True Color; современные адаптеры имеют 24-битные RAMDAC (по 8 бит R, G и B). Применение 24-битных преобразователей RAMDAC позволяет выполнять в адаптере гамма-коррекцию — исправлять нелинейность монитора. Для правильной коррекции требуется выполнять калибровку цвета. Для ЖК-мониторов гамма ограничивается и его внутренними (обычно 6-битными) устройствами ЦАП, их калибровка весьма условна.

В высокочастотных режимах может оказаться неприемлемым использование удлинителя аналогового интерфейсного кабеля, а также применение консольных коммутаторов. Дешевые коммутаторы часто имеют отнюдь не высокочастотные галетные переключатели, длинные «бороды» расплетенных пар сигнальных проводов и «вольности» в соединениях обратных проводов цветовых сигналов. Правда, иногда умелыми руками эти погрешности удается исправить. Отдельные контакты в разъеме VGA для обратных проводов выделены не зря — качественно передать аналоговый сигнал с полосой в сотни мегагерц на расстояние более метра можно только при непрерывности, индивидуальности обратного провода и его «перевитости» со своим сигнальным собратом по всей длине кабеля. Все неоднородности (нарушения импеданса) проявятся на экране монитора в виде эхо-сигналов, хорошо видимых правее вертикальных границ контрастных объектов (прямоугольников, вертикальных линий, букв). На концах линий (в мониторе) должны быть установлены терминаторы (резисторы 75 Ом), без которых изображение будет ярче, но с явным эхом.

Объем видеопамяти (локального буфера) в старых графических адаптерах определял соотношение разрешения, количества одновременно доступных цветов и страниц (экранных буферов). Объем памяти современных графических карт значительно превышает требования к размеру экранного буфера. У карт с 3D-aK- селератором объем локального буфера существенно влияет на производительность акселератора (чем больше, тем выше), и объем 16 Мбайт и более нужен

не столько для размещения страниц видеопамяти (см. табл. 10.6), сколько для хранения текстур и иной информации, используемой в построениях. На мощных игровых картах устанавливают до 256 Мбайт памяти.

Тип видеопамяти, ее *тактовая частота* и *разрядность* (8, 16, 32, 64, 128 и даже 256 бит) определяют производительность; в старых адаптерах эти параметры ограничивали и предельную частоту регенерации. В современных адаптерах используют разновидности синхронной динамической памяти (DDR SDRAM, SGRAM, RDRAM) гораздо большего быстродействия, чем ОЗУ.

Тип и количество *дисплейных интерфейсов* определяют возможности подключения мониторов. Большинство адаптеров имеют разъем DB-15 с аналоговым интерфейсом VGA, к которому подключаются обычные ЭЛТ-мониторы и плоские панели. В качестве второго интерфейса (если есть) используется разъем DVI с цифровым и, возможно, аналоговым интерфейсом. При наличии двух интерфейсов желательна их независимость.

Наличие и тип *телевизионных интерфейсов* (композитный, S-Video) и их функции: выход (TV-Out) и/или вход (TV-In) определяют возможности взаимодействия с видеосистемой. Выход позволяет просматривать фильмы на телевизоре (и записывать видео на видеомагнитофон). Вход позволяет вводить информацию для просмотра на мониторе, а при наличии средств видеозахвата записывать видео на диск.

Для графических *адаптеров с 3D-акселераторами* имеются специфические параметры:

- Тип (модель) графического процессора (чипсета) — их разрабатывает и производит ряд фирм, у каждой фирмы есть ряд моделей со своими особенностями, достоинствами и недостатками. Процессоры могут иметь более одного конвейера для рендеринга, и у каждого конвейера может быть несколько блоков текстуризации, позволяющих за один такт накладывать несколько текстур. При экстремальных требованиях к производительности интересна способность к совместной работе в паре с другим акселератором (технологии SLI, CrossFire).
- Частота работы графического ядра — 125-250 МГц.
- Тип интерфейса —AGP или PCI-E.
- Тип, разрядность шины и частота работы локальной памяти, определяющие ее производительность. Частота шины памяти может отличаться от частоты графического ядра. Некоторые чипсеты, имеющие несколько параллельных конвейеров, используют для каждого конвейера отдельную локальную память, что увеличивает суммарную производительность.
- Поддерживаемый объем локальной памяти — 8-256 Мбайт. Определяет количества элементов и текстур, хранимых локально, и, следовательно, производительность их обработки.
- Разрешение экрана, с которым работает акселератор.
- Разрядность цвета — 16-32 бит/пиксел. 16-битный цвет экономит ресурсы, требуемые для построения изображения, но при этом качество изображения

не удовлетворяет взыскательного пользователя. Применяют и компромиссный вариант — оптимизированный 16-битный цвет.

- Разрядность Z-буфера — 16/24/32 бит.
- Скорость обработки треугольников — характеристика производительности геометрической части конвейера.
- Скорость формирования пикселов — 100-1000 млн. пикселов/с.
- Поддерживаемая размерность текстур: 64 x 64, 128 x 128, 256 x 256 — текстуры малого размера; современные адаптеры поддерживают текстуры 1024 x 1024, 2048 x 2048 с 32-битными текселами.
- Поддержка мультитекстурирования (возможности наложения нескольких текстур).

Этот список можно продолжить по реализации всех технологических приемов — алгоритмов фильтрации (билинейной, бикубической, анизотропной...), сглаживания (anti-aliasing), установки источников освещения, эффектов тумана, применению объемных текстур, методов оптимизации расчетов, снижающих объемы вычислений, и т. п.

Возможность работы с видеоизображениями характеризуют следующие параметры:

- Аппаратная поддержка декодирования MPEG-1 (для Video CD), MPEG-2 для DVD и SVCD, а также HDTV.
- Наличие и тип входов и выходов видеосигналов.
- Поддержка смешивания видео и графики с использованием 8-битного α-па- раметра (таким образом можно, например, вводить субтитры).

Производительность дисплейного адаптера характеризуется многими показателями, среди которых можно выделить несколько групп:

- DOS performance — производительность программного вывода символов или пикселов. Использование высокопроизводительных шин PCI, AGP и PCI-E, применение специальной видеопамяти большой разрядности, теневой видеопамяти и BIOS при высокопроизводительном процессоре (CPU) обеспечивают существенное повышение производительности видеосистемы.
- GUI (Graphic User Interface) performance (2D, или Windows performance) — производительность при выводе примитивов Windows GUI. Помимо выше-приведенных факторов зависит и от эффективности 20-акселератора (Video chipset).
- 3D perfomance — производительность трехмерных построений: скорость обработки многоугольников, из которых собираются трехмерные поверхности, скорость вывода результирующих пикселов в видеопамять и ряд параметров, характерных для выполняемых специальных функций.

  - Video Display performance — производительность вывода «живого» видео, повышается применением аппаратных кодеков (MPEG и др.). Повышение производительности выражается в улучшении качества декодирования,

уменьшении числа пропущенных кадров и снижении загрузки процессора (актуально для многозадачных ОС).

Производительность конкретного адаптера зависит от выбранного разрешения, количества цветов, частоты и режима развертки. Влияние параметров развертки на производительность может показаться неочевидным, но вспомним, что видеопамять сильно загружена постоянным считыванием данных для регенерации изображения. Доступ к ней для построения изображений происходит в свободное от регенерации время, и чем выше частота сканирования, тем меньше у видеопамяти этого свободного времени и тем ниже производительность. Современные графические чипсеты в сочетании с применением специализированной видеопамяти позволяют уменьшить это влияние на производительность компьютера.

ЭЛТ-дисплей (монитор) характеризуется размером экрана, зернистостью, обеспечиваемыми частотами разверток и полосой пропускания видеотракта; поддерживаемые режимы разрешения являются производными от этих параметров. Помимо этих параметров, которые поддаются численному выражению, имеются и показатели качества изображения на экране:

- четкость и контрастность изображения;
- яркость и насыщенность цветов;
- хорошее сведение лучей по всему полю экрана;
- устойчивость изображения — отсутствие колебания и дрожания, ровная граница краев экрана, особенно справа снизу. Отсутствие мерцания относится к количественным параметрам — частоте и режиму регенерации;
- отсутствие эхо-сигналов — слабых повторов элементов изображения в негативном и позитивном виде чуть правее их оригиналов. Это не «глюки» в глазах наблюдателя, а следы несогласованности аналогового интерфейса (см. выше).

Для *ЖК-дисплея* важны размерность матрицы, непосредственно задающая используемое разрешение, и ее физический размер, определяющий размер самих пикселов. Дисплей со слишком мелкими пикселами (хоть их и много!) может оказаться некомфортным при не слишком остром зрении, а изменение разрешения относительно размера матрицы в ЖК-мониторах всегда сопровождается потерей качества. К ним применимы вышеприведенные показатели качества (кроме сведения лучей, которых в ЖК-мониторах просто нет). Дополнительные количественные показатели — угол обзора и число проблемных пикселов на экране (последнее, к сожалению, может расти).

Качество определяется визуально и, увы, довольно субъективно, так что предъявление претензий бывает затруднительным. Пользователю остается лишь возможность выбора из множества моделей мониторов и графических адаптеров. Рекомендаций здесь приводить не будем, заметим лишь, что при почти одинаковой цене в одной и той же «весовой категории» (тип монитора и размер экрана) зачастую дорогие модели не очень знатных фирм выглядят лучше дешевых моделей под престижными марками.

Напомним, что в режимах с высоким разрешением и высокой частотой регенерации (то есть при очень высокой частоте вывода пикселов) на качество изображения сильно влияет качество исполнения RAMDAC и разводки выходных цепей графического адаптера, входных цепей монитора и соединяющего их кабеля. Для придирчивого выбора компонентов графической системы (адаптера и монитора) надо смотреть изображение на мониторе, подключенном именно к желаемой модели графического адаптера, чтобы не было разочарований (ряд моделей адаптеров грешит некачественным выходом при иных технических преимуществах).

ГЛАВА 11

Устройства ввода-вывода и их интерфейсы

В этой главе рассматриваются наиболее распространенные устройства ввода и вывода. Для многих из них в IBM PC были введены специальные аппаратные интерфейсы подключения, которые также рассмотрены в данной главе. Интерфейсам общего назначения, которые используются в современных устройствах, посвящены главы последней части книги.

11.1. Клавиатура

Традиционная клавиатура PC представляет собой унифицированное устройство со стандартным разъемом и последовательным интерфейсом связи с системной платой. В качестве датчиков нажатия клавиш применяют механические контакты (открытые или герконовые), кнопки на основе токопроводящей резины, емкостные датчики и датчики на эффекте Холла. Типы клавишных датчиков влияют на надежность, долговечность и, конечно же, цену клавиатуры. Последние два типа являются самыми долговечными, поскольку в них исключены механические контактные системы. Независимо от типов применяемых датчиков нажатия клавиш все они объединяются в матрицу. Клавиатура содержит внутренний контроллер, выполненный обычно на микросхеме из семейства MCS-48 фирмы Intel, осуществляющий сканирование матрицы клавиш, управление индикаторами, внутреннюю диагностику и связь с системной платой последовательным интерфейсом по линиям KB-Data и KB-Clock.

Среди обычных (стандартных) исполнений существуют 3 основных типа клавиатур:

- Клавиатура XT на 83 клавиши, в оригинале без индикаторов. Впоследствии к ним добавили индикаторы состояния Num Lock и Caps Lock, управляемые внутренним контроллером по нажатию соответствующих клавиш, однако состояние этих индикаторов может быть не синхронизированным с флагами в ОЗУ, которыми пользуются драйверы.
- Клавиатура AT на 84 клавиши отличается от XT появлением дополнительной клавиши SysReq загадочного назначения и программно-управляемых индикаторов Num Lock, Caps Lock, Scroll Lock. Двунаправленный интерфейс с сис

темной платой позволяет программе корректно управлять индикаторами, а также программировать некоторые параметры клавиатуры и производить диагностику.

- Расширенная (enhanced) клавиатура на 101/102 клавиши, ставшая современным стандартом, применяется в большинстве моделей AT и PS/2. Некоторые расширенные клавиатуры (например, «Microsoft Natural») имеют 104 или 105 клавиш, имеются и 122-клавишные модели.

Клавиши расширенной клавиатуры разделены на несколько групп:

- основная клавиатура;
- функциональная клавиатура;
- цифровая клавиатура (numeric keypad) — при выключенном индикаторе Num Lock (или включенном индикаторе Num Lock и нажатии клавиши Shift) используется для управления курсором и экраном;
- выделенные клавиши управления курсором и экраном (дублируют эти функции цифровой клавиатуры);
- клавиши управления питанием;
- клавиши быстрого доступа к приложениям.

Дополнительные клавиши управления питанием и другими системными функциями появились сравнительно недавно. Эти клавиши — как правило, бледнолилового цвета и расположенные около клавиш управления курсором и экраном, — досаждают пользователям, привыкшим к обычной расширенной клавиатуре и работающим с ней не глядя. Клавиши быстрого доступа, больше похожие на кнопки, чем на привычные нам клавиши, — это тоже нововведение «фирменных» клавиатур. Эти клавиши предназначены для быстрого вызова некоторых приложений, например интернет-браузера и медиаплеера, и работы с ними.

По электрическому интерфейсу клавиатуры XT и AT совпадают, за исключением того, что двунаправленный интерфейс позволяет клавиатуре AT принимать команды от системной платы. Однако по логическому интерфейсу они несовместимы (клавиатура AT иногда имеет переключатель режима XT/AT). Клавиатура PS/2 отличается от AT только исполнением разъема, при необходимости можно использовать переходник.

Внутренний контроллер клавиатуры способен определить факты нажатия и отпускания клавиш, при этом можно нажимать очередную клавишу, даже удерживая несколько ранее нажатых. При нажатии клавиши клавиатура передает идентифицирующий ее *скан-код*. При удержании клавиши в нажатом положении через некоторое время клавиатура начинает автоповтор передачи скан-кода нажатия этой клавиши. Задержка автоповтора (typematic delay) и скорость автоповтора (typematic rate) для клавиатур AT программируются. Расширенная клавиатура позволяет выбирать один из трех наборов скан-кодов.

Помимо традиционного стандартного исполнения существуют и другие варианты клавиатур. Малогабаритные клавиатуры портативных компьютеров интегрированы в общий корпус, но часто эти компьютеры имеют разъем для подключения обычной внешней клавиатуры, работать с которой все-таки удобнее.

Некоторые производители оригинальных PC-совместимых компьютеров применяют собственные конструкции клавиатур, разъемов и даже интерфейса (например, Olivetti), что затрудняет их замену. При отсутствии поддержки производителем или поставщиком это может привести к тому, что в один момент компьютер может стать «чемоданом без ручки», поскольку механика клавиш подвержена износу. Кроме того, на клавиатуры иногда проливают чай (или хуже того — сладкий кофе), что выдерживают далеко не все конструкции. Существует множество вариантов клавиатур, различающихся используемыми датчиками, ощущениями от нажатия и расположением клавиш. Имеются разные эргономические варианты: клавиатуры, «разламывающиеся» на две половины, имеющие подкладки для рук и т. п. При большом объеме клавиатурного ввода на эти нюансы есть смысл обратить внимание, поскольку неправильное положение рук оператора может приводить и к профессиональным заболеваниям. По ощущению от нажатия различают клавиатуры с «кликом» и без него. «Клик» — это щелчок, раздающийся при срабатывании нажатой клавиши. Щелчок может быть акустическим (это сильно раздражает соседей по помещению) и механическим, ощущаемым пальцами как преодоление некоторого предела упругости, после которого нажимаемая клавиша проваливается. В клавиатурах без клика срабатывание датчика почувствовать не удается, и, если оператор не привык смотреть на экран, возможны пропуски символов или их ложные повторы. Конечно же, выбор клавиатуры — дело вкуса пользователя, но он определяется и финансовыми возможностями — цены клавиатур могут различаться на порядок.

Интерфейс клавиатуры

Для подключения клавиатуры предназначен последовательный синхронный интерфейс, состоящий из двух обязательных сигналов KB-Data и KB-Clock. Обе линии на системной плате подтягиваются резисторами к шине +5 В. На обеих сторонах интерфейса выходные сигналы низкого уровня формируются выходами элементов с открытым коллектором (стоком), а состояние линий может быть прочитано через входные линии контроллеров. На некоторых старых системных платах имеется дополнительный сигнал KB-Reset, сбрасывающий клавиатуру низким уровнем. Интерфейс клавиатуры AT двунаправленный: от клавиатуры передается информация о нажатии/отпускании клавиш, а передача информации к клавиатуре требуется для управления индикаторами ее состояния и программирования параметров (автоповтор, набор скан-кодов). *Контроллер интерфейса клавиатуры* и его разъем всегда располагаются на системной плате. Конструктивно возможны два варианта разъема, их вид со стороны задней панели и назначение контактов приведены на рис. 11.1. На системных платах ATX устанавливается малогабаритная розетка mini-DIN для клавиатуры PS/2 (рис. 11.1, а), иногда на этот же разъем выводится и интерфейс мыши (сигналы показаны в скобках). На старых платах использовалась 5-контактная розетка DIN (рис. 11.1, *б)*. На разъем клавиатуры через плавкий предохранитель поступает напряжение питания клавиатуры +5 В. Электрически и логически интерфейс клавиатуры PS/2 повторяет AT, поэтому для согласования типа

разъема применяют специальные переходники. Предпочтительнее использовать переходники, выполненные в виде мягкого кабеля с разъемами. Монолитный переходник, особенно с клавиатуры AT на разъем PS/2, хуже тем, что малейшее движение кабеля вызывает большой момент силы, выламывающей переходник из маленького гнезда PS/2.

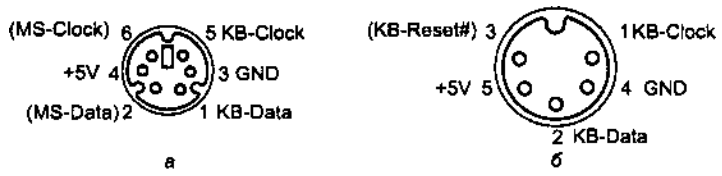


Рис. 11.1. Разъемы подключения клавиатур (вид со стороны контактов): а — PS/2, б — AT

ВНИМАНИЕ

Питание от разъема клавиатуры часто используется такими устройствами, как внешние накопители или адаптеры локальных сетей, подключаемые к параллельному порту. Плавкий предохранитель, установленный на системной плате, может не выдержать броска тока, потребляемого этими устройствами. При этом откажется работать и клавиатура — ее индикаторы даже не мигнут при включении.

Процессор общается с клавиатурой через контроллер интерфейса клавиатуры (микроконтроллер 8042 или программносовместимый с ним), в основном используя порт 60h. Прием скан-кодов осуществляется чтением этого порта.

О необходимости чтения скан-кода контроллер сигнализирует процессору через аппаратное прерывание IRQ1, сигнал которого вырабатывается по каждому событию клавиатуры (нажатию и отпусканию клавиши). Задание параметров автоповтора, выбор таблиц скан-кодов, управление светодиодными индикаторами, а также управление режимом сканирования матрицы клавиш и запуск диагностического теста осуществляются командами, посылаемыми в этот порт. Контроллер транслирует команды в посылки, направляемые к клавиатуре.

На системной плате PC/XT контроллера 8042 не было, и интерфейс клавиатуры (однонаправленный) был реализован аппаратной логикой — регистром сдвига, параллельный выход которого подключается ко входам порта A системного интерфейса i8255. По приеме байта от клавиатуры вырабатывается аппаратное прерывание IRQ1, обработчик которого может прочитать принятый байт из порта 60h. С помощью битов 7 и 6 порта 61h возможны программная блокировка и сброс клавиатуры соответственно. Сброс клавиатуры XT осуществляется обнулением линии KB-Clock.

С распространением шины USB появились клавиатуры и с этим интерфейсом, иногда они имеют и встроенный хаб, например для подключения мыши USB. Клавиатура USB питается от шины. Для клавиатуры USB требуется специальная поддержка со стороны BIOS, она имеется в современных системных платах. На таких платах в CMOS Setup имеется параметр Legacy Keyboard Support, предназначенный для включения поддержки устаревшей клавиатуры.

Контроллер интерфейса клавиатуры и мыши 8042/8242

Программируемый *микроконтроллер интерфейса клавиатуры* (KeyBoard Controller, KBC) *i8042* является посредником между клавиатурой, подключенной к нему по вышеописанному последовательному интерфейсу, и центральным процессором, с которым он связан через параллельный интерфейс. В микроконтроллере постоянно исполняется внутренняя микропрограмма, реагирующая на сигналы интерфейса клавиатуры и команды, поступающие от процессора. Эта микропрограмма (KBC BIOS) хранится во внутреннем масочном ПЗУ контроллера; внешне она недоступна, и контроллер можно рассматривать как устройство с заданными свойствами. Поскольку логика работы контроллера реализована программой, его реакция на команды процессора и сигналы интерфейса относительно медленная — время отклика измеряется десятками микросекунд. Помимо сигналов управления клавиатурой через программно-управляемые и программно-читаемые линии внешних портов контроллера формируются сигналы управления вентилем *Gate A20* и аппаратного системного сброса, а также считываются сигналы от конфигурационных джамперов системной платы. Контроллер i8242 помимо интерфейса клавиатуры поддерживает аналогичный интерфейс дополнительного устройства — PS/2-Mouse. При инициализации (по аппаратному сбросу) контроллер устанавливается в режим PS/2 или AT в зависимости от состояния определенного вывода микросхемы. В режиме AT контроллер не выполняет функций интерфейса мыши и игнорирует все команды, относящиеся к мыши. Режим выбирается включением соответствующего параметра CMOS Setup.

Контроллер расположен в пространстве ввода-вывода CPU по адресам 60h (регистр данных) и 64h (регистр состояния и команд), назначение регистров приведено в табл. 11.1. *Из регистра данных* считываются данные, принимаемые по интерфейсам от клавиатуры и мыши, а также данные, возвращаемые контроллером в ответ на адресованные ему команды. *Запись в регистр данных* обеспечивает подачу команд и данных, адресованных клавиатуре и мыши, а также данных для команд, адресованных контроллеру. В *регистр команд* записываются команды, адресованные контроллеру. Режим работы контроллера (разрешение работы интерфейсов клавиатуры и мыши и прерываний от них, трансляция скан-кодов и другие параметры) задается *командным байтом*, посылаемым в контроллер по специальной команде. Перед любой записью в контроллер необходимо убедиться в его готовности. Признаком готовности/занятости контроллера является значение бита 1 регистра состояния (порт 064h).

Таблица 11.1. Назначение регистров контроллера клавиатуры

Порт, R/W	Назначение
060 RW	*Порт данных 8042*
064 R	*Регистр состояния 8042* (R/O):
	- бит 7 — ошибка четности при последнем обмене с клавиатурой; - бит 6 — тайм

Порт, R/W	Назначение
	- бит 5 — тайм-аут передатчика/выходной буфер интерфейса мыши полон (Mouse_OBF)[1];
	- бит 4 : 0 - клавиатура на замке;
	- бит 3 : 1 — последняя запись была командой, 0 — данными;
	- бит 2: системный флаг, устанавливается в 0 по включении питания, в 1 — программно (что означает состояние завершения системного сброса ResetOK);
	- бит 1 : 1 — входной буфер контроллера полон, 0 — готовность к приему команды/данных;
	- бит 0:1 - выходной буфер интерфейса клавиатуры полон (OBF)
064 W	*Регистр команд 8042*

[1] Второе назначение бита относится к контроллеру i8242B, имеющему дополнительный интерфейс для подключения PS/2-Mouse.

Контроллер имеет два внешних порта, с помощью которых и реализуются последовательные интерфейсы, а также управление вентилем GateA20, управление сигналом сброса процессора и чтение сигналов от джамперов системной платы. Эти порты не имеют непосредственного отображения в пространстве адресов ввода-вывода PC, доступ к ним осуществляется через команды контроллера.

Приняв посылку от клавиатуры, контроллер выполняет внутреннюю трансляцию скан-кода (если это не запрещено командным байтом) и устанавливает в регистре состояния `0BF = 1`, что приводит к генерации запроса прерывания IRQ1 (если это не запрещено командным байтом). В ответ на это хост должен считать данные (транслированный скан-код, префиксы и т. п.) из порта данных (60h). Трансляция обеспечивает программную совместимость клавиатур XT и AT (см. далее) по чтению скан-кодов из порта 60h. При получении посылки от мыши контроллер не выполняет никаких преобразований и устанавливает в регистре состояния `Mouse__0BF = 1`, что приводит к генерации запроса прерывания IRQ12 (если это не запрещено командным байтом). По этому сигналу данные от мыши должны быть считаны хостом из того же порта 60h. Тот же самое происходит и при программной записи байта в выходной буфер клавиатуры (код D2) или мыши (код D3h) с установкой соответствующих битов состояния и генерацией запросов IRQ1 или IRQ12.

После получения команды, по которой контроллер должен возвращать данные, он устанавливает в регистре состояния `0BF = 1`, что приводит к генерации запроса прерывания IRQ1 (если это не запрещено командным байтом). После этого данные должны быть считаны из порта данных (по адресу 60h). Если команда возвращает несколько байтов данных, прерывание генерируется для каждого байта.

Контроллеры интерфейса клавиатуры различаются версиями встроенного ПО; из-за этого замена контроллера другим, отличающимся версией ПО (KBC BIOS), может оказаться проблематичной — BIOS системной платы должна знать особенности контроллера клавиатуры.

Скан-коды

Скан-коды передаются от клавиатуры в компьютер по фактам нажатия и отпускания клавиш. При нажатии клавиши передается ее скан-код — номер, идентифицирующий ее расположение на клавиатуре, некоторые клавиши передают цепочку кодов, начинающихся с префикса E0 или E1, за которыми следуют байты расширенного скан-кода. Современные клавиатуры могут работать в одном из 2-3 наборов (таблиц) скан-кодов, различающихся назначением кодов и способами сообщения об отпускании клавиш.
Набор Set#0 — соответствует первым клавиатурам XT и AT-84. При отпускании клавиши клавиатура передает ее скан-код с инвертированным битом 7, что для клавиш, передающих единичный скан-код, соответствует сложению с 80h. При отпускании клавиш, передающих цепочки скан-кодов, префиксы передаются без изменений, а в байтах расширенного скан-кода бит 7 инвертируется, причем эти модифицированные расширенные скан-коды передаются в порядке, обратном порядку передачи при нажатии. Принятый способ сигнализации отпускания не позволяет использовать скан-коды 60h, 61h, 5A и 6E, а также скан-коды большие, чем 79h: для них коды отпускания совпали бы с префиксами E0 и E1 и специальными ответами клавиатуры AT. Кроме того, недопустим код 00 — это значение используется в прикладном интерфейсе для сообщения об альтернативном способе набора кода.
Набор Set#1 — появился в расширенных клавиатурах (101/102 и более клавиш). Здесь при отпускании передается 2 байта: в первом содержится *признак отпускания F0, во втором* — скан-код (не модифицированный). Для клавиш, генерирующих пару кодов (префикс и расширенный код), при отпускании сначала передается префикс (E0 или E1), затем признак F0, после чего передается расширенный скан-код. Если клавиша генерирует цепочку пар кодов при нажатии, то при отпускании каждая пара дает тройку кодов, причем порядок следования расширенных кодов — обратный порядку их выдачи при нажатии.
Набор Set#2 — отличающийся от первых двух, не прижился и практически не используется (он объявлен не обязательным).
Скан-коды нажатия для трех возможных наборов приведены в [9]. Клавиши управления курсором и экраном расширенных клавиатур вызывают посылку цепочек скан-кодов, зависящую от состояния клавиши Shift и индикатора Num Lock. Также меняются коды, генерируемые клавишами Print Screen, Pause и Numeric. Цепочка кодов клавиши Pause при нажатии не укладывается в данную схему, а при отпускании этой клавиши никаких кодов не генерируется.
Номер набора задается клавиатуре и ее контроллеру центральным процессором (команда F0), по включении питания у современных клавиатур должен устанавливаться набор Set#l. Контроллер 8042/8242 по умолчанию осуществляет трансляцию кодов, посылаемых клавиатурой, в набор Set#0, коды которого и доступны процессору при чтении из порта 60h. Записью командного байта в контроллер трансляцию можно и отменить, тогда будут доступны коды, принятые от клавиатуры, в их «натуральном» виде. Считыванием (из порта 060h)

и интерпретацией скан-кодов занимается программа центрального процессора, выполняемая обработчиком прерывания IRQ1.

Автоповтор с точки зрения центрального процессора работает следующим образом. Если нажать клавишу, контроллер выработает прерывания и выдаст скан-код нажатия. Если клавишу удерживать нажатой, то через некоторое время задержки (typematic delay) клавиатура начнет генерировать серию посылок скан-кода, и они будут вызывать серию прерываний IRQ1 с передачей этого кода до тех пор, пока не будет отпущена клавиша. Если, не отпуская одной клавиши, нажать другую, то автоповторы первой клавиши прекратятся, будет передан скан-код второй клавиши, и если она тоже будет удержана, автоповтор начнется уже для ее скан-кода. Для клавиш и их комбинаций, вызывающих передачу цепочек скан-кодов, автоповтором передаются только последние пары байтов из этих цепочек.

Системная поддержка и программный интерфейс

Клавиатура имеет системную поддержку на уровне BIOS — обработку фактов нажатия и отпускания клавиш и обеспечение сервисов ввода символов с клавиатуры, а также управление ее параметрами (задержка и частота автоповтора) и индикаторами. Коды, принятые от клавиатуры ее контроллером, считываются и обрабатываются обработчиком аппаратного прерывания IRQ1 (вектор 09h). Результат обработки помещается в *клавиатурный буфер*, из которого по программному прерыванию `Int 16h` этот результат для дальнейшей обработки может быть извлечен значительно позже. Прикладной программе, для которой требуется нестандартное использование клавиатуры (например, в качестве музыкальной), придется самой заниматься обработкой аппаратного прерывания IRQ1, перехватывая вектор `Int 9h`. Перехват этого вектора также обеспечивает возможность вызова тех или иных функций резидентных программ по «горячим» клавишам.

При начальном тестировании процедура POST инициализирует клавиатуру (и ее контроллер) и запускает диагностический тест. Во время этого теста клавиатура должна мигнуть всеми индикаторами, после чего может остаться включенным только индикатор `Num Lock` (зависит от установки в CMOS Setup). В случае обнаружения ошибки клавиатуры на консоль выводится сообщение с возможным указанием скан-кода залипшей клавиши и предложением нажать клавишу F1 для продолжения. То же самое произойдет, если тест не обнаружит клавиатуру (например, из-за вывалившегося разъема или перегоревшего предохранителя), но в этом случае нажатия клавиши F1 будет уже недостаточно. Ошибку диагностики даст и подключение к компьютеру AT клавиатуры от XT, обратное «скрещивание» тоже неработоспособно. Чтобы начальная загрузка не останавливалась по ошибке (отсутствию) клавиатуры (POST будет дожидаться получения кода клавиши F1), тестирование клавиатуры может быть отменено специальным параметром CMOS Setup (см. 6.6).

Русификация клавиатуры на «аппаратном» уровне сводится к нанесению на клавиши символов русских букв, а также некоторых спецсимволов и знаков препинания, которые в русской раскладке располагаются иначе, чем в английской (это связано с разным количеством букв в алфавитах). Более того, русская раскладка для DOS и Windows по знакам препинания также не сходятся. С этими неудобствами уже давно смирились, и сейчас подавляющее большинство русифицированных клавиатур имеет одинаково подписанные клавиши (чаще под DOS). На программном уровне русификация осуществляется загружаемым драйвером, который, как правило, замещает обработчик IRQ1 драйвера BIOS — перехватывает вектор прерывания Int 9h. Для работы в DOS драйвер загружается в виде резидентной программы, которая может содержать и русификатор дисплея. Для Windows русификация осуществляется выбором требуемого драйвера в меню при начальной установке или изменении параметров системы. Для переключения языка используются клавиши Ctrl, Alt, Shift (левые и правые) в различных сочетаниях, единого подхода нет (некоторые драйверы позволяют пользователю определить собственную комбинацию клавиш).

Прерывания, вызванные приходом кодов *нажатия и отпускания клавиш*, обрабатывает *BIOS* Int 9h. Каждый принятый скан-код (или цепочка) обрабатывается с учетом состояния *клавиатурных флагов.* Результат обработки (как правило, ASCII-символ в младшем байте и скан-код в старшем) помещается в *клавиатурный буфер*, расположенный в ОЗУ. По приеме каждого символа указатель головы буфера увеличивается. Буфер организован в виде кольца, после достижения конца области буфера указатель головы установится на начало области. В случае переполнения буфера (указатель головы «догнал» указатель хвоста) очередное слово не записывается и подается звуковой сигнал. Размер буфера позволяет хранить описание шестнадцати фактов нажатий клавиш. Нажатие клавиш Ctrl, Shift, Alt и некоторых комбинаций в буфере не отмечается, но приводит к модификации битов ячеек флагов клавиатуры. Нажатие «системной» комбинации Ctrl+Alt+Del, клавиши PrintScreen (SysRq) и некоторых других к записи в клавиатурный буфер не приводит, а вызывает специальные процедуры.

Для обслуживания клавиатуры используются ячейки ОЗУ из области данных BIOS (BIOS Data Area), в которых находятся флаги клавиатуры, аккумулятор кода Alt-набора, указатели головы и хвоста буфера и собственно буфер (на 16 элементов).

Обработчик аппаратного прерывания до обработки принятого скан-кода вызывает прерывание BIOS INT15h с AH = 4Fh, а в AL находится принятый скан-код. Стандартный обработчик Int 15h(4Fh) просто выполняет возврат с CF = 0, но его можно заменить специальным обработчиком, который будет при необходимости подменять принятые скан-коды какими-либо иными (оставляя их в AL), что должно отмечаться установкой CF = 1. В старых версиях BIOS такой возможности перехвата не было, ее наличие можно определить вызовом Int 15h(C0h).

Для *клавиатуры USB* или иного устройства ввода, заменяющего клавиатуру в качестве консоли, прерывание Int 9h должно вызываться программно при обработке каждого клавиатурного события. Обработчик этого прерывания выпол

няет те же действия: пропускает скан-код через Int 15h(4Fh) и помещает в клавиатурный буфер, а также модифицирует флаги клавиатуры.

Интерфейс прикладного уровня для клавиатуры представляет *BIOS* Int 16h. Его основное назначение — извлечение слов из клавиатурного буфера. Функция задается в регистре AH при вызове, результат помещается в регистр AX:

- *AH* = 00h — чтение (с ожиданием готовности) и выборка слова из буфера (меняется указатель хвоста). Индикаторы клавиатуры обновляются в соответствии с состоянием флагов. Если буфер пуст, то на AT выполняется прерывание Int 15h (подфункция 90), что может использоваться ОС, например, для переключения задач. Чтобы программа не «зависала» на ожидании символа, предварительно стоит проверить готовность функцией 01h. Символы расширенной клавиатуры фильтруются: преобразуются в их аналоги 83-кла- вишной клавиатуры.
- *AH* = 01h — проверка готовности, чтение без выборки (указатели не изменяются). Признак наличия символа в буфере — установленный флаг ZF.
- *AH* = 02h — чтение состояния флагов (в AL — байт 0:417h, см. выше).
- *AH* = 03h — установка задержки и частоты автоповтора: BL — код задержки (00 = 250, 01 = 500, 02 = 750, 03 = 1000 мс), BH — код частоты.
- *AH* = 05h — запись слова из регистра CX в буфер (меняется указатель головы). Признак успешной записи — AL = 0, если в буфере нет места, то AL = 1.
- *AH* = 10h и *AH* = 11h — функции, аналогичные 00h и 01h, но предназначенные специально для 101/102-клавишных клавиатур — в них не выполняется фильтрация символов расширенной клавиатуры. Для ряда клавиш, отсутствующих в клавиатуре AT-84, эти функции дают результаты, отличающиеся от результатов вызовов 00h и 01h.
- *AH* = 12h — чтение расширенного состояния флагов.

Функции чтения буфера (00 и 10h) в регистре AL возвращают *ASCII-код* символа, в AH — *скан-код.* Символы, полученные нестандартным способом (в русском регистре или Alt-набором), сопровождаются нулевым скан-кодом. Alt-набор позволяет ввести в буфер любой символ — для этого его код в десятичной системе набирается на цифровой клавиатуре при нажатой клавише Alt, результат заносится в буфер при отпускании клавиши Alt.

При AL = 0 регистр AH содержит *расширенный ASCII-код* (Extended ASCII Keystroke). Дополнительные клавиши 101/102-клавишных клавиатур при использовании функций 10h-12h генерируют код E0h в младшем байте и скан-код, соответствующий аналогичным управляющим клавишам 83/84-клавишных клавиатур.

Функция записи (05h), несколько неожиданная для клавиатуры, позволяет легко имитировать работу оператора в различных демонстрационных программах. Если прикладная программа не перехватывает обслуживание клавиатуры на уровне аппаратного прерывания (Int 9h), то резидентная программа может «подбрасывать» ей в буфер слова, которые будут восприниматься как нажатие клавиш.

ASCII-коды буфера, соответствующие нажатию клавиш, приведены в [1], [7]. При русификации (или другой локализации) клавиатуры отслеживание переключения регистров (языков) ложится на обработчик аппаратного прерывания клавиатуры.

11.2. Манипуляторы-указатели — мышь, трекбол

Мышь и трекбол называют указательными устройствами (pointing devices), поскольку с их помощью пользователь может задать компьютеру местоположение курсора и подать одну из нескольких команд. Способ управления координатным курсором для системы безразличен и определяется его сферой применения и личными пристрастиями пользователя.

Устройство ввода *мышь* (mouse) передает в систему информацию о своем перемещении по плоскости и нажатии кнопок (двух или трех, а в современных моделях и больше). Обычная конструкция имеет свободно вращающийся массивный обрезиненный шарик в днище корпуса, передающий вращение на два координатных диска с фотоэлектрическими датчиками. Датчики для каждой координаты представляют собой две открытые оптопары (светодиод-фотодиод), в оптический канал которых входит вращающийся диск с прорезями. Оптопары датчиков могут оформляться в виде монолитных конструкций или быть просто отдельными элементами, установленными на печатной плате.

В мыши имеется *микроконтроллер*, который обрабатывает сигналы с датчиков и посылает в ПК информацию о перемещениях и состоянии кнопок. Один из первых вариантов мыши — *Bus Mouse* (шинная мышь) — содержит только датчики и кнопки, а обработка их сигналов производится на специализированной плате адаптера, устанавливаемого в слот шины ISA (откуда и название «шинная мышь»).

Манипулятор *трекбол* (TrackBall — дословно «следящий шар»), по сути, представляет собой перевернутую мышь, шарик которой вращают пальцами. Иногда он встраивается в клавиатуру (чаще на портативных компьютерах). Преимущество шара в том, что он не требует для работы свободной плоской поверхности, а может закрепляться зажимом на краю стола. Однако вращать шар пальцами нравится не всем (хотя при этом и можно добиться большей точности позиционирования).

Оптическая мышь (optical mouse) не имеет механических частей, подверженных загрязнению и износу. Первые модели оптических мышей ориентировались по лучам, отраженным от специального коврика с сетчатым рисунком. Теоретически это надежнее, но загрязнения и царапины на коврике приводят к неожиданным «прыжкам», наклон оси мыши относительно оси коврика сильно искажает отображение траектории движения. Современные оптические мыши имеют встроенную видеокамеру с процессором, обрабатывающим полученное изображение. Эта мышь не требует специального коврика и может функциони

ровать на любой поверхности — оптические «неровности», за которые она может «зацепиться взглядом», есть практически везде. Первые оптические мыши из-за слабости графического процессора обладали низким быстродействием — слишком быстрые перемещения сбивали их с толку. Современные оптические мыши имеют хорошие параметры: разрешение — 800 dpi, скорость — до 1 м/с, ускорение — до 10g.

3D-мышь помимо двух обычных координат перемещения позволяет задавать и третью — с дополнительного колесика, вращаемого пальцем. Это колесико, как правило, приводит в движение трещотки, нажимающие кнопки-датчики.

По *интерфейсу с компьютером* различают несколько видов мышей, из которых в современных PC-совместимых компьютерах используются следующие:

- Serial Mouse — мышь с интерфейсом RS-232C, подключаемая к COM-порту ПК. Интерфейс однонаправленный: данные передаются только от мыши, параметрами самой мыши управлять невозможно.
- PS/2-Mouse — мышь с двунаправленным интерфейсом, подключаемая к специальному интерфейсному порту системной платы.
- USB Mouse — мышь с интерфейсом USB, низкоскоростное (Low Speed, LS) устройство USB, с которым устанавливается двунаправленная связь. Мышь с интерфейсом USB удобно подключать к клавиатуре USB или монитору со встроенным USB-хабом (меньше проводов идет к системному блоку).
- Bluetooth Mouse — мышь с двунаправленным радиоинтерфейсом.

Для ряда применений (в основном игр) важна частота опроса мыши. Для интерфейса USB она может достигать 125 опросов в секунду (но не больше!), интерфейс PS/2 допускает и большие частоты (до 200 опросов в секунду). Мыши Serial Mouse по возможной частоте посылок самые медленные (ниже 30 посылок в секунду).

С интерфейсами Serial Mouse и PS/2-Mouse иногда возникают недоразумения. Хотя оба они последовательные, эти интерфейсы имеют существенные принципиальные различия в уровнях сигналов, способе синхронизации, частоте и формате посылок:

- В интерфейсе PS/2 используется однополярный сигнал с уровнями ТТЛ, питание мыши — однополярное с напряжением +5 В относительно шины GND. Сигнал в интерфейсе RS-232C мыши Serial Mouse двуполярный (см. 16.1) с уровнями срабатывания +3 В и -3 В, и для него требуется двуполярное (относительно шины GND) питание мыши.
- В интерфейсе PS/2 применяются две раздельные сигнальные линии, одна для передачи данных, другая для передачи сигналов синхронизации. В Serial Mouse реализован асинхронный способ передачи данных всего по одной линии.

Даже если не рассматривать частоты и форматы посылок, ясно, что прямой совместимости между этими интерфейсами быть не может. Тем не менее выпускаются и продаются переходники (пассивные!), позволяющие выбрать способ подключения мыши. Эти переходники предназначены только для универсаль

ных мышей, у которых встроенный контроллер по напряжению питания способен распознать, к какому интерфейсу его подключили, и установить соответствующий тип своего выходного интерфейса. Универсальные мыши, поддерживающие интерфесы и Serial Mouse, и PS/2, не особо распространены, поэтому часто приходится слышать о неудачных попытках применения таких переходников к обычным устройствам Serial Mouse или PS/2-Mouse. Сейчас выпускаются универсальные мыши PS/2-USB (интерфейсы тоже разные!) с соответствующими пассивными переходниками.

Дополнительную путаницу вносят мыши для компьютера Macintosh, которые имеют разъем, с виду напоминающий разъем PS/2. Однако при ближайшем рассмотрении и неудачной попытке включения его в PC становится ясно, что разъемы эти разные, да и интерфейс совершенно иной.

Мышь стала практически незаменимым устройством ввода для современных программ, и при интенсивном использовании она требует к себе внимания. Опыт показывает, что мышь (особенно дешевая) изнашивается гораздо быстрее, чем клавиатура. Нестабильно работающей мышью пользоваться тяжело, поскольку при отсутствии видимой обратной связи через перемещение курсора на экране возникает ощущение парализованной руки. Плохо работающие кнопки тоже доставляют немало неприятностей: если работать, например, с документом в MS Word, перемещение мыши с дребезжащей левой кнопкой может буквально рвать текст в клочья. У современных мышей настраивается много параметров, доступных при установке «родного» драйвера. Основные параметры, отвечающие темпераменту пользователя, — время двойного щелчка и чувствительность.

В блокнотных ПК в качестве манипулятора может использоваться сенсорная панель (touch pad), чувствительная к прикосновению и перемещению пальца. Применяются и миниатюрные манипуляторы в виде кнопки, чувствительной к давлению в разных направлениях. К компьютеру может быть одновременно подключено несколько устройств-указателей (например, сенсорная панель и мышь или пара мышей). Операционные системы без специальной программной поддержки сообщения от всех указателей собирают в единый поток для управления единственным курсором на экране.

Чаще всего неисправности мыши связаны с внутренним переломом проводов около корпуса, что легко исправить, вырезав износившийся кусочек провода. Также часто мышь плохо работает из-за загрязнения шарика или валиков датчиков. Если резиновый шарик или валики датчиков загрязняются, мышь перестает распознавать движение. Для сохранения чистоты желательно пользоваться ковриком для мыши (это, в отличие от «тапочек для тараканов», — не роскошь). Кроме того, шарик и валики периодически следует чистить — лучше всего протирать их тампоном, смоченным спиртом. Для чистки механики у мыши обычно имеется съемное «брюшко», снять которое можно его поворотом или сдвигом согласно указующей стрелке. Надежность распознавания движения мыши зависит и от расположения датчиков — взаимного и относительно колесика с прорезями. Некачественные оптопары чувствительны и к внешнему освещению: бывает, что разобранная мышь работает нормально, а с установленной

крышкой — нет. Иногда достаточно слегка сблизить или, наоборот, раздвинуть светодиод и фотодиод, и мышь начинает работать нормально. «Дребезжащие» кнопки ремонту практически не подлежат — их приходится заменять.

Последовательные мыши — MS Mouse и PC Mouse

Serial Mouse — мышь с последовательным интерфейсом, подключаемая через 9- или 25-контактный разъем COM-порта (табл. 11.2). Эта мышь имеет встроенный микроконтроллер, который обрабатывает сигналы от координатных датчиков и кнопок. Каждое событие — перемещение мыши или нажатие-отпускание кнопки — кодируется двоичной посылкой по интерфейсу RS-232C. Для передачи информации применяется асинхронная передача, а двуполярное питание, требуемое по протоколу RS-232, обеспечивается от управляющих линий интерфейса. Недостатком мыши Serial Mouse является то, что она занимает COM- порт и требует монопольного использования его штатной линии прерывания (IRQ4 для COM1 и IRQ3 для COM2). Конечно, то, что для работы мыши порту COM1 требуется именно прерывание IRQ4, является недостатком не самой мыши, а ее программного драйвера, но для пользователя, не увлекающегося написанием «мышиных» драйверов, важен сам факт этого ограничения. Две основные разновидности последовательных мышей — *MS Mouse* (Microsoft Mouse) и *PC Mouse* (Mouse Systems Mouse) — требуют разных драйверов, многие мыши имеют переключатель MS/PC. Эти два типа мышей при одинаковой скорости 1200 бит/с, одном стоп-бите и отсутствии контроля четности используют разные форматы посылок, приведенные в [7]:

- *MS Mouse*: 7 битов данных, трехбайтный пакет (в «классическом» варианте), положительным значениям соответствует перемещение по координате *X* вправо, а по координате *Y* вниз. Для трехкнопочных мышей добавляется четвертый байт, передаваемый только при изменении состояния средней кнопки. Для 3D-мыши четвертый байт имеет иное назначение.
- *PC Mouse*: 8 битов данных, пятибайтный пакет, положительным значениям соответствует перемещение по координате *X* вправо, а по координате *Y* вверх.

Несовпадение форматов объясняет беспорядочные перемещения курсора на экране при несоответствии драйвера типу мыши.

Таблица 11.2. Разъемы Serial Mouse

Сигнал	Контакт DB9	Контакт DB25
Data	2	3
GND	5	7
+V(питание)	7(4)	4(20)
-V(питание)	3	9

Системная поддержка последовательной мыши осуществляется только на уровне ОС (сервисы вызываются через Int 33h), драйвер мыши — загружаемый или встроенный в ОС. BIOS мышь не поддерживает, даже если и пользуется ею для

навигации в CMOS Setup. Еще раз подчеркнем, что для работы мыши обязательно требуется линия аппаратного прерывания — IRQ4 или IRQ3 для последовательных мышей на портах COM1 или COM2 соответственно.

Мышь PS/2

Мышь *PS/2-Mouse* появилась с компьютерами PS/2. Ее интерфейс и разъем 6-pin mini-DIN аналогичен клавиатурному (рис. 11.2) и, как правило, реализуется тем же контроллером клавиатуры 8242 (см. выше). Адаптер и разъем PS/2-Mouse устанавливаются на многих современных системных платах. Контроллер мыши PS/2 может также располагаться на карте расширения (ISA) и занимать дополнительные адреса в пространстве ввода-вывода. С мышью PS/2 связь двусторонняя: процессор может посылать контроллеру 8242 специальные команды через порт 60h, но, в отличие от интерфейса клавиатуры, перед записью каждого байта (и команды, и ее параметра) в порт 64h должен записываться код D4h.

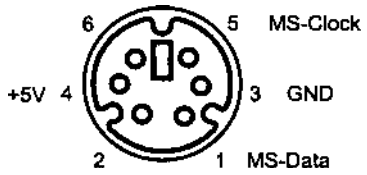


Рис. 11.2. Разъем PS/2-Mouse

Мышь может работать в одном из двух режимов. В *потоковом режиме* (stream mode) мышь посылает данные по любом изменении состояния; в *режиме опроса* (remote mode) мышь передает данные только по запросу процессора. Есть еще *диагностический режим* (wrap mode), в котором мышь возвращает эхом данные, посылаемые ей контроллером. По приеме пакета от мыши контроллер устанавливает флаг Mouse_0BF и вырабатывает прерывание IRQ12, если оно не запрещено командным байтом 8242. Формат пакета приведен в [9].

Устройства-указатели с интерфейсом PS/2 (мышь PS/2) имеют *поддержку BIOS*, обеспечивающую настройку мыши (посылку вышеперечисленных команд). Собственно драйвер мыши (обработчик прерывания по вектору 74h от запроса IRQ12), обрабатывающий ее информационные посылки, входит лишь в состав ОС или загружается отдельно. Поддержка мыши вызывается через BIOS Int 15h с кодом функции в регистре AX.

Беспроводные мыши и клавиатуры

Существуют варианты *беспроводных мышей* (cordless mouse): мышь с аккумуляторным питанием связана с интерфейсным блоком по инфракрасному каналу или по радиоканалу. Аналогично может подключаться и клавиатура. Интерфейсный блок подключается к компьютеру одним из стандартных «мышиных» интерфейсов: PS/2 или USB (возможно и через COM-порт). Беспроводная связь удобна, например, при проведении презентаций — докладчик не привязан

к компьютеру (и видеопроектору) проводами. Более элегантное решение — мышь (и клавиатура) с интерфейсом Bluetooth при наличии встроенного интерфейса Bluetooth в ПК; при этом лишних проводов и блоков нет.

ВНИМАНИЕ ----------

Беспроводные клавиатуры с радиоинтерфейсом — «находка для шпиона»: весь текст, набираемый пользователем, а также имена и пароли могут быть легко перехвачены через радиоэфир. Поток данных от мыши особой ценности не представляет. Канал инфракрасной связи в плане конфиденциальности контролируется легче («шпион» должен быть виден), но менее удобен.

Некоторые беспроводные мыши имеют специфический дефект: при работе с ресурсоемкими приложениями они теряют точность наведения курсора (не удается с первого раза попасть в нужную точку). Это можно объяснить упрощенной организацией беспроводного канала связи: мышь рапортует об изменении своего состояния с определенным периодом. Эти рапорты получает интерфейсный блок и передает их в ПК по опросу, организованному интерфейсом подключения. Если беспроводной интерфейс однонаправленный (что, естественно, проще и дешевле), то в канале беспроводной связи невозможно организовать управление потоком (или квитирование). В этом случае, если из-за повышенной загрузки центрального процессора рапорт от мыши не будет своевременно передан драйверу, очередной рапорт (с указанием очередного изменения координат) потеряется. Чтобы таких неприятностей не было, нужен либо двунаправленный беспроводной канал связи, либо более сложный алгоритм работы контроллера в интерфейсном блоке. В последнем варианте контроллер должен подсчитывать текущие координаты мыши по приходящим рапортам и сообщать их приращения по опросу от компьютера. С передачей состояния кнопок таких проблем не возникает: время нажатия относительно периода опроса достаточно велико.

11.3. Планшеты

Дигитайзер (планшет) — устройство, позволяющее вводить графическую информацию от руки пользователя. Дигитайзеры позволяют вводить *абсолютные координаты точек*, привязанные к системе координат планшета (манипуляторы сообщают только относительные перемещения). Первые модели дигитайзеров предназначались для ввода координат точек чертежей, закрепленных на планшете. Для этого они снабжались манипулятором с «оптическим прицелом» и несколькими кнопками. По нажатии кнопки дигитайзер передает текущие координаты, в графическом приложении (векторном) эти координаты используются для рисования графических примитивов. Современные дигитайзеры снабжаются и *пером*, с помощью которого можно рисовать (или писать) и ретушировать изображения. Для художественных работ перо с планшетом гораздо удобнее, точнее и производительнее, чем мышь. Перо может быть чувствительным к нажатию и/или наклону — информация о нажатии может управлять параметрами рисующего инструмента в графическом приложении. Переворот пера

может превращать «карандаш» в «ластик». В состав планшета может входить специальная мышь, которой можно работать, как обычная. Распространены планшеты форматов А3-А6; крупноформатные планшеты дороги, но для рисования и ретуширования удобны и маленькие планшеты (к тому же их легче разместить на столе). Для «сколки» чертежей существуют планшеты формата А0 (и даже больше). Планшеты обеспечивают высокое разрешение (1000, 2540 dpi и выше), погрешность 0,1-0,2 мм. Число различимых градаций нажатия пера может быть 256 и более. Интерфейс подключения — COM-порт или шина USB.

Внутреннее устройство планшетов может быть различным. Если планшет содержит матрицу принимающих антенн, то указатель представляет собой генератор электромагнитных сигналов, питающийся от батарейки или через кабель питания. Возможна и обратная конструкция: передающие антенны — в планшете, а приемник — в пере. Планшеты с сенсорными панелями воспринимают механическое воздействие «карандаша». В карманных ПК миниатюрные сенсорные панели, объединенные с жидкокристаллическим дисплеем, являются основными устройствами ввода графической информации и рукописного текста. Сенсорные панели блокнотных ПК на роль планшета не годятся: они воспринимают только прикосновение пальца (реагируют на электрическую емкость), так что точность низкая (передают только относительные перемещения и не различают степень нажатия).

11.4. Сканеры

Сканеры являются растровыми устройствами ввода изображения с оригинала — изображения на бумаге или пленке. При сканировании фрагмент оригинала освещается белым светом, отраженный свет фокусируется на фотоприемнике — ПЗС-линейке (ПЗС — прибор с зарядовой связью, английский термин — Couple-Charged Device, CCD). В линейке свет преобразуется в накопленный заряд, его «профиль» (разложение по строке) сдвигается по линейке и последовательно выводится в ЦАП. Таким образом получается цифровой поток, отображающий яркость элементов (пикселов) строки. Оцифрованное изображение запоминается во внутренней памяти сканера, каретка с лампой и линейкой сдвигается и сканируется следующая строка. В памяти делается предварительная обработка изображения, и данные выводятся через внешний интерфейс в компьютер. Объем передаваемых данных определяется разрешением, глубиной цвета и размером сканируемой области. Поток данных может быть большим, так что интерфейс может стать узким местом, определяющим производительность ввода изображения. На производительность работы со сканером влияют и параметры собственно сканера, и параметры компьютера, к которому он подключен (желательны большой объем ОЗУ и быстрая дисковая подсистема).

Существуют сканеры разнообразных конструкций, различающиеся по назначению, параметрам и цене.

В *ручных сканерах* головка с лампой прокатывается вручную. Ручные сканеры довольно дешевые (в них отсутствует сложная механика), однако геометриче

ская точность низкая (зависит и от твердости руки оператора). Ручные сканеры позволяют сканировать любые поверхности, в том числе внутренние стенки коробок и углов стен.

В самых распространенных *планшетных сканерах* оригинал кладется на стеклянный стол (как в копировальном аппарате) и под ним автоматически продвигается каретка с лампой и линейкой. Эти сканеры при умеренной цене обеспечивают высокую точность, но размер оригинала ограничен (А4, А3).

В *листопротяжных* (рулонных) *сканерах ЛИСТ* протягивается над неподвижной кареткой (как в факсе) вручную или приводом. Преимущество — неограниченная длина оригинала (можно сканировать рулоны показаний самописцев).

В *барабанных сканерах* оригинал вкладывается в барабан; вращение барабана и перемещение головки (лампы с фотоприемником) дают возможность последовательного поточечного сканирования на одном фотоприемнике. При этом обеспечивается очень высокое качество цветопередачи (точность и динамический диапазон), поскольку все точки изображения воспринимаются (последовательно) одним фотоприемником. В других типах сканеров всегда имеется погрешность от неидентичности элементов ПЗС-линейки. Барабанные сканеры очень дорогие.

Для цветного сканирования изображение должно быть разложено на базисные цвета (RGB). В трехпроходных сканерах используется одна линейка, и на каждом проходе устанавливается свой светофильтр. В однопроходных сканерах свет разделяется на 3 потока, каждый через свой светофильтр попадает на свою линейку.

Основные параметры сканера относятся к его оптике и механике.
Оптическим разрешением (измеряется в dpi) считается разрешение по горизонтали, оно определяется разрешением ПЗС-матрицы. Разрешение по вертикали называется *механическим*, оно определяется шагом мотора привода. Механическое разрешение проще повышать, и оно может быть выше оптического. Внутренней обработкой изображения в сканере разрешение по обеим осям выравнивается. Это разрешение, называемое *интерполяционным*, может быть выше оптического. Можно установить и меньшее интерполяционное разрешение (объединением пикселов), при этом уменьшается поток данных, передаваемых при сканировании в компьютер.

Глубина цвета определяется разрядностью АЦП. В большинстве случаев приложениям достаточно 24 бита на пиксел. Внутренняя разрядность сканера может быть выше (30-36 бит), что позволяет выполнять цветокоррекцию без потерь. Правда, младшие биты могут оказаться шумом.

Динамический диапазон сканирования определяется как разность максимальной и минимальной оптических плотностей[1] оригинала, воспринимаемых сканером. В основном оптическая плотность зависит от качества АЦП. Для различных

[1] Оптическая плотность (Density, *D)* определяется в логарифмических единицах: *D* = lg(Iпад/Iотр), $/_{|1ад}$ — интенсивность падающего света, $/_{отр}$ — интенсивность отраженного (для прозрачных оригиналов — прошедшего) света. 0,0 *D* — идеально белый цвет, 4,0 *D* — идеально черный цвет.

оригиналов характерен различный динамический диапазон: газетная бумага обеспечивает диапазон в пределах 0,9; мелованная — 1,5-1,9; фотоснимки — 2,3; фотопленки, слайды — 2,8-4,0. Дешевые сканеры имеют динамический диапазон 1,8-2,5; цветные планшетные — 2,5-3,8; барабанные — 3,4-4,0.

Скорость (время) сканирования: может сильно зависеть от компьютера и интерфейса (в идеале сканирование идет непрерывно, чаще — частями).

Сканер может иметь дополнительные навесные элементы: автоподатчик документов (вместо крышки, оригиналы протягиваются, каретка не передвигается), слайд-модуль (вместо крышки, заменяет лампу на каретке сканера). Сканирование слайдов имеет свои особенности: сканер должен работать на просвет (а не на отражение) и иметь большой динамический диапазон.

Для подключения сканеров используются различные интерфейсы:

- Интерфейс SCSI обеспечивает высокую скорость передачи. Иногда в комплекте со сканером поставляется простой SCSI-адаптер, служащий для его подключения, но снижающий скорость работы.
- Интерфейс USB 2.0 (желательно) обеспечивает скорость до 24 Мбайт/с, USB 1.0 - до 1,2 Мбайт/с.
- LPT-порт должен работать в режиме ECP/EPP, иначе скорость будет крайне низкой.
- Интерфейс 1394 (FireWire) обеспечивает высокую скорость, не везде есть.
- В случае нестандартных интерфейсов к сканеру требуется специальная плата.

Для работы со сканерами используют стандартное приложение (драйвер и графический интерфейс) *TWAIN* (Tool Without An Interesting Name), обеспечивающее взаимодействия сканера с прикладными пакетами ПО. Его функции — установка параметров и области сканирования, предварительное сканирование и просмотр, цветокоррекция и постобработка изображения, передача данных в приложение.

Для профессиональной работы со сканером требуется его *калибровка*: установка параметров коррекции цветопередачи по специальному тестовому изображению.

11.5. Принтеры и плоттеры

Принтер — это устройство, способное выводить изображение (печатать, откуда и название) на бумагу или пленку. Плоттер (графопостроитель) тоже выводит изображение, но он его не печатает, а вычерчивает[1]. Принтеры и плоттеры создают так называемые *твердые копии* (hard copy) документов; твердость означает невозможность их последующей произвольной модификации (стирания и подчистки в расчет не берутся). По этому признаку принтеры и плоттеры от

[1] Современные плоттеры выводят изображение тем же растровым способом, что и принтеры, но, в отличие от принтеров, изображение для плоттеров описывается векторным способом.

носятся к *пассивным устройствам графического вывода*, их противоположность — активные устройства вывода (дисплеи).
По способу печати принтеры разделяются на буквопечатающие и знакосинтезирующие (что аналогично текстовому и графическому режимам дисплея), а также последовательные и параллельные. В последовательных принтерах печать осуществляется поэлементно с продвижением по строке, и после прохода строки переходят к печати следующей строки. В параллельных принтерах строка печатается целиком. *Буквопечатающие принтеры* способны печатать только строчки символов из фиксированного набора, что ограничивает область их применения печатью текстовых документов без возможности использования привычного уже разнообразия шрифтов. Вместе с тем, у них есть преимущество в качестве печатаемых символов, а в ряде случаев — и в скорости печати. Таких принтеров существовало (и поныне существует) несколько типов. *Знакосинтезирующие,* они же матричные, принтеры позволяют печатать произвольные изображения. По способу нанесения красителя они делятся на ударные (игольчатые), термические, струйные и лазерные, хотя под матричными, как правило, подразумевают именно игольчатые.

Матричные игольчатые принтеры

Игольчатые принтеры — «ветераны» печатающих устройств ПК. По нынешним меркам их разрешение низкое (до 360 dpi), скорость и качество невысокие, цветопередача плохая, и они самые шумные. Однако они обеспечивают самую дешевую печать (дешевы и принтер, и расходные материалы); они неприхотливы к бумаге, могут работать под копирку (и без ленты) и дают механический оттиск. Последние два свойства обусловливают ряд их специфических применений (например, оттиск реализует одну из степеней защиты в новых паспортах).
Игольчатые принтеры (dot matrix printer) имеют печатающую головку, на которой расположена матрица игольчатых молоточков, управляемых электромагнитами. Иголки ударяют по бумаге через красящую ленту, бумага лежит на валике, перемещаясь только продольно (перевод строк выполняется поворотом валика), но в обоих направлениях. Перемещение по строке выполняет сама печатающая головка — она довольно легкая, поэтому ее можно двигать быстро. Управляет всей механикой встроенный микроконтроллер принтера. В его ведении находятся шаговые двигатели подачи бумаги и перемещения головки по строке, а также приводы иголок, которых может быть от 8 до 24. На принтере имеются механические или оптоэлектронные датчики крайних положений каретки, а также датчик конца бумаги. Управляя этими механизмами и пользуясь датчиками, можно вывести любое изображение. Во время печати головка движется по строке слева направо, и ударами иголок отпечатываются требуемые точки. После того как строка отпечатана, передвигается бумага и выполняется печать следующей строки. Если бумагу не перемещать, то можно повторно пропечатывать отдельные элементы (символы), и они будут выглядеть ярче. У «умных» принтеров печать может выполняться и на обратном ходе головки (это экономит время), хотя из-за люфтов механики возможно не очень точное совмещение точек, отпечатанных на прямом и обратном ходе.

Матричные принтеры могут работать как в графическом, так и в символьном режиме. Развертку символов в точечное изображение выполняет встроенный процессор (микроконтроллер) принтера, у которого есть ПЗУ с таблицами знакогенераторов. Обычно принтеры имеют несколько таблиц (для разных языков и шрифтов), переключаемых программно (по командам от компьютера), аппаратно (переключателями на принтере) или с помощью кнопок панели управления принтером.

Контроллер принтера по интерфейсу принимает от компьютера поток байтов, содержащий данные для печати и управляющие команды. Данные принимаются в буферное ОЗУ, откуда извлекаются и интерпретируются в соответствии с возможностями механики. Принтер обеспечивает обратную связь с компьютером: управляет потоком (останавливает по заполнении буфера) и сообщает свое состояние — готовность (On-Line), конец бумаги (Paper End), ошибка (Error). Это позволяет программе работать с принтером не вслепую и сообщать пользователю о необходимости вмешательства. Принтер способен печатать поступающие к нему данные, когда он включен, у него есть бумага и он подготовлен к работе. В этом состоянии (On-Line) принтер готов к приему данных от компьютера (если у него есть место в буферной памяти). Заметим, что принтер печатает строку только после того, как «поймет», что у него в буферной памяти собран окончательный образ для этой строки. В символьном режиме строка будет отпечатана в следующих случаях:

- принято столько символов, сколько умещается в строке, и еще хотя бы один (принтеру полагается воспринимать код «забоя», по которому он должен аннулировать предыдущий символ);
- принят символ возврата каретки (CR), перевода строки (LF) или формата (FF);
- оператор нажал кнопку перевода строки или формата (для их срабатывания принтер должен быть переведен в состояние Off-Line, печать строки может быть вызвана и переводом в это состояние).

Таким образом, матричный принтер является *устройством построчного вывода.* В графическом режиме идея печати та же — строка печатается целиком, когда для нее готовы данные (для всех используемых иголок). При переводе принтера в состояние Off-Line печать и прием данных приостанавливаются, но оставшиеся в буфере данные сохраняются. Буфер очищается по включении питания, аппаратному сбросу по сигналу интерфейса и по приеме специальной команды.

По включении питания, аппаратному или программному сбросу контроллер выполняет самотестирование и приводит механику в исходное состояние. Для этого он перемещает головку до срабатывания датчика левого положения, чтобы откалибровать систему позиционирования. Некоторые принтеры после этого немного прогоняют головку вправо, чтобы она не мешала заправке бумаги.

Разрешающая способность матричного принтера определяется размером матрицы иголок, но и не только им. Точки можно пропечатывать, смещая головку (влево-вправо) и бумагу (вверх-вниз) даже на долю шага, так что точки сольются в почти гладкую линию. Правда, для этого требуется довольно точная меха

ника. Разрешающая способность печати связана со скоростью: поскольку иголки все-таки инерционны, предельная частота их срабатывания ограничена. Поэтому для высокого разрешения скорость перемещения головки и бумаги невысока. Первые модели матричных принтеров были довольно грубыми, последующие позволяют достигать разрешения вплоть до 360 dpi (точек на дюйм) по обеим координатам. Принтеры, как правило, могут работать в режимах с различным разрешением: от малого разрешения для быстрой печати черновиков (draft) до разрешения NLQ (Near Line Quality — качество печати, близкое к гладким буквам пишущих машинок), считающегося высоким.

Цветные матричные принтеры работают с многоцветной (обычно трехцветной) красящей лентой. Каждая строка печатается за несколько проходов головки, и на каждый проход устанавливается лента определенного цвета. Конечно, эта цветная печать происходит небыстро, да и качество цветопередачи невысокое.

Матричные принтеры весьма неприхотливы: могут печатать практически на любой бумаге — листовой, рулонной, фальцованной. Листовая бумага подается фрикционным механизмом — валиком, к которому она прижимается обрезиненным роликом. Листы могут заправляться вручную, в более дорогих моделях имеются специальные лотки для автоматической подачи бумаги из пачки. Для печати из рулона или стопки фальцованной бумаги с перфорацией по краям механизм подачи бумаги имеет траки — резиновые или пластмассовые «гусеницы» с зубчиками. Траки расположены на общей оси и обеспечивают подачу бумаги без перекосов, неизбежных (пусть и в небольшой степени) при фрикционной подаче. Узкие принтеры позволяют печатать на бумаге шириной до формата А4 (вертикально заправленный лист), широкие — до формата А3 (горизонтально заправленный лист). Принтеры имеют направляющие, регулируемые по ширине листа, а у моделей с траками направляющие двигаются вместе с траками. Существуют специальные приспособления для печати этикеток.

Параллельные матричные принтеры (например, Mannesmann Tally) не имеют подвижной печатающей головки — у них иголки расположены вдоль всей печатаемой строки. За счет этого печать происходит очень быстро (с той же скоростью, что и у барабанных буквопечатающих принтеров). Горизонтальное разрешение у этих принтеров не обязательно определяется числом иголок: печатающий блок может немного перемещаться вдоль строки, и каждая строка может быть отпечатана за несколько ударов со смещением на каждом ударе точек друг относительно друга на доли шага иголок. От этих принтеров в основном требуется высокая скорость печати символов, так что механизм повышения разрешения, безусловно снижающий скорость, может включаться лишь для графической печати и «экзотических» шрифтов. Эти принтеры, как правило, широкие и работают с рулонной и фальцованной бумагой с перфорацией по краям (фрикционная подача на большой длине всегда будет уводить бумагу в сторону). Принтеры очень дорогие, но при большом объеме текстовой печати весьма эффективные. Расходный материал — красящая лента.

Управление принтером интуитивно понятно большинству пользователей. На лицевой панели принтера имеется несколько кнопок управления и индикато

ров режима. Для заправки и прогона бумаги следует пользоваться кнопками панели управления — вращать рукоятку валика при включенном принтере обычно не рекомендуют (при этом приходится «бороться» с шаговым двигателем подачи бумаги).
Включение питания при нажатой кнопке FF или LF обычно переводит принтер в режим тестовой распечатки, из которого можно выйти выключением питания (иногда и нажатием кнопки On Line). Включение питания с нажатием определенной комбинации кнопок (каких именно — указывается в инструкции) переводит современные принтеры в диалоговый режим настройки параметров.
Матричные принтеры обеспечивают недорогую печать — расходным материалом является красящая лента, заправленная в картридж. Проще всего заменять картридж, но дешевле — ленту (то и другое продается по отдельности). Однако следует учитывать, что картриджи специфичны для каждой модели или семейства моделей (это видно невооруженным глазом). Ленты тоже различаются — шириной и длиной, а также конфигурацией (обычное кольцо или лента Мебиуса, то есть лента с одной поверхностью). Каждый картридж рассчитан именно на свою ленту. Слишком длинная лента не уместится в картридже, слишком короткая будет быстро изнашиваться. Лента не той ширины или конфигурации (Мебиуса вместо кольца и наоборот) будет выбиваться из картриджа. Заправка картриджа лентой — довольно «грязное» дело; для заправки картридж приходится вскрывать. Есть картриджи, в которых имеется губчатый (поролоновый) валик, подкрашивающий ленту, и предусматривается возможность заправки этого валика специальными чернилами. Однако лента периодически изнашивается механически — пробивается до дыр, и ее (или картридж) все равно приходится менять. Матричные принтеры позволяют печатать несколько экземпляров сразу через копирку, а ряд принтеров имеет механизм регулировки положения головки (прижима к валику) в зависимости от числа копий (или толщины бумаги). Когда кончается красящая лента, в крайнем случае можно печатать и без нее — второй экземпляр из-под копирки оказывается вполне читаемым.
Красящая лента в картридже при перемещении головки постоянно перематывается — для этого имеется специальная механическая передача. У изношенного картриджа ответная часть передачи может прокручиваться относительно приводного вала, тогда лента в картридже останавливается. При этом лента очень быстро пробивается насквозь, поскольку иголки постоянно бьют по одному и тому же месту.

Термопринтеры

Термопринтеры по конструкции напоминают игольчатые, но вместо ударов иголок по красящей ленте их головки нагревают отдельные точки специальной термочувствительной бумаги. Эти принтеры отличаются практически бесшумной работой, правда, скорость печати невысока. Главный недостаток — требуется специальная бумага, изображение на которой получается не очень устойчивым (на солнечном свету и при нагревании бумага темнеет). В настоящее время

термопринтеры практически не выпускаются, и бумага для них может быть дефицитом. Термопечать используется в факс-машинах.

Струйные принтеры

Струйные принтеры InkJet (ink — чернила, jet — струя), как и термопринтеры, конструктивно аналогичны матричным игольчатым принтерам, но вместо удара по бумаге через красящую ленту они «стреляют» по бумаге капельками специальных (быстросохнущих) чернил из микроскопических сопел. Для формирования капли используется несколько способов:

- В электростатических принтерах из сопла выбрасывается непрерывная серия капель — технология называется CIJ (Continuous InkJet). С помощью управляющего электрода часть капель отклоняется в сборник (на рециркуляцию), часть летит на бумагу. Этой технологии свойственна высокая скорость печати. Есть вариант технологии и с «каплями по требованию» (Drop On Demand, DOD), без рециркуляции; эта печать происходит медленнее.
- В пьезоэлектрических принтерах (основная технология фирмы Epson) капли выстреливаются механическими микронасосами на пьезоэлементах. Эти принтеры чувствительны к пузырькам воздуха в чернилах. Управляемость размером капли и отсутствие «сателлитов» (мелких брызг вокруг основной капли) — свойства, полезные при полутоновой печати. Головки дорогие, но долговечные (при «правильных» чернилах), цена печати ниже.
- В пузырьковых принтерах (bubble-jet), выпускаемых фирмами HP, Lexmark, Canon, Xerox, капля выталкивается пузырьком пара (от микроскопического нагревательного элемента). Взрыв плохо управляем, вокруг капли присутствуют мелкие «сателлиты». Ресурс головок ограничен, но они дешевые и легко меняются. Разрешение — до 1200-2400 dpi.

Число сопел в головке измеряется десятками и сотнями. По конструкции они бывают с отдельными сменными чернильницами и с чернильницами, совмещенными с головкой. В совмещенном варианте предусматривается дозаправка чернильниц. Струйные принтеры работают тихо, скорость печати определяется режимом: черновая — быстро, качественная, особенно цветная печать — довольно медленно. Однако на хорошей бумаге достигается высокое качество печати. На плохой бумаге чернила растекаются, правда, против этого применяют разные ухищрения (например, подогрев бумаги для ускорения высыхания). Большинство струйных принтеров печатают только на листовой бумаге (в основном А4, но есть и А3), есть модели и для рулонной бумаги. Из-за довольно высокой цены картриджей с чернилами стоимость печати на струйном принтере, особенно цветной, оказывается довольно высокой, в то время как сами принтеры относительно недороги. Иногда у принтеров пересыхают чернила в соплах, и это, как правило, приводит к необходимости замены довольно дорогой головки. По включении питания принтер выполняет серию манипуляций с головкой и чернильницами, подготавливая их к работе. Чтобы сопла не высыхали, головка паркуется в специальном месте. Нештатное отключение питания во время рабо

ты не позволяет принтеру «припарковать» головку, и чернила могут засохнуть в соплах. Число органов управления у струйных принтеров сведено к 1-2 кнопкам, одна из которых является выключателем питания. Одной кнопкой и переключают режим On-Line/Off-Line, и выводят недопечатанную страницу, и загружают новую страницу. Перевод строки, смена шрифтов и т. д. кнопками уже не выполняются — всеми этими функциями управляет компьютер. Это вполне закономерно, поскольку в струйном принтере то место листа, в котором производится печать, скрыто от глаз (поэтому ручной перевод строки не имеет смысла), а шрифтовые возможности настолько богаты (благодаря высокому разрешению), что кнопочное управление тут просто неуместно.

Твердокрасочные и сублимационные принтеры

Твердокрасочные (Solidlnk) *принтеры* можно рассматривать как варианты струйных. В этих принтерах восковая краска расплавляется и через сопла головки наносится на вал переноса, с которого потом накатывается на бумагу. Головка принтера неподвижная, печатается вся строка целиком, что обеспечивает высокую скорость печати. Разрешение — 1200 dpi, отпечатки качественные, глянцевые, но боятся нагрева. По расходным материалам печать дешевая. Во время разогрева принтер потребляет большую мощность. Принтеры «не любят» отключений (слишком долго запускаются).

Сублимационные (термодиффузионные) принтеры обеспечивают самую высококачественную цветную печать (цвета смешиваются на бумаге). Здесь краска испаряется с ленты и переходит на бумагу (впитывается), степень нагрева испарителя регулируется (256 ступеней), за счет чего меняется интенсивность каждой точки. Для каждого базисного цвета (краски) выполняется свой проход, поэтому печать медленная. Принтеры имеют разрешение 1400 dpi (цветных точек!) и выше. Печать дорогая — дороги и принтер, и расходные материалы.

Лазерные и светодиодные принтеры

В лазерных и светодиодных (LED) принтерах используется электрографическая печать — технология переноса изображения на бумагу, издавна применяемая в копировальных аппаратах. В *лазерном принтере* имеется барабан, покрытый фоточувствительным полупроводником. Поверхность барабана электризуется, после чего модулированный лазерный луч сканирует всю поверхность барабана, разряжая засвеченные участки. Сканирование осуществляется с помощью вращающегося зеркала, направляющего луч на поверхность барабана, и вращения самого барабана. К разряженным точкам поверхности притягивается тонер — очень мелкий красящий порошок, формируя на барабане изображение полного листа. Далее синхронно с вращением барабана по нему прокатывается наэлектризованный лист бумаги, и частички тонера переходят на лист. Затем бумага с тонером прокатывается через горячие валки, и тонер припекается к бумаге, после чего лист выводится из принтера. Таким образом, лазерный принтер является постраничным печатающим устройством — он может печа

тать страницу только целиком, не имея возможности остановиться посреди строки (как последовательный) или листа (как построчный). Цветная печать осуществляется в несколько проходов — каждый раз со своим цветом тонера. Лазерные принтеры обеспечивают высокое качество печати, они обладают самым высоким разрешением (600, 1200, 2400 dpi) и работают только с листовой бумагой высокого качества, пачка которой загружается в лоток. Принтеры печатают и на пленку, используемую в полиграфии в качестве оригинал-макетов. Специально для печати на пленку принтеры имеют возможность зеркальной печати изображения (именно так печатают макеты книг). Принтеры чувствительны к механическим свойствам бумаги — плохую и мятую бумагу они заминают, и для извлечения остатков листа приходится открывать принтер (правда, это делается довольно просто). Скорость черно-белой печати достигает десятков листов в минуту, цветная печать выполняется медленнее.

В светодиодных принтерах используется тот же электрографический принцип, но вместо лазерной головки (с вращающимся зеркалом и т. п.) имеется линейка светодиодов. Количество светодиодов (размер линейки) определяет разрешение принтера. Конструкция получается проще и компактнее, что позволяет в цветных принтерах устанавливать по ходу бумаги четыре картриджа с барабанами, тонером и LED-линейками. При этом цветная печать выполняется за один проход. В лазерных принтерах такая компоновка затруднительна.

Лазерные принтеры выпускаются в широком ассортименте — от маломощных персональных до мощных. Большие принтеры имеют несколько лотков для бумаги и возможность программного выбора лотка. Для каждой модели принтера имеется оптимальная нагрузка — количество отпечатанных листов за единицу времени, а также ресурс барабана. Превышение нагрузки ведет к ускорению износа, и принтер может не успеть выработать свой официальный ресурс. Слишком малая нагрузка невыгодна — мощные принтеры стоят дорого, поэтому удельная стоимость печати окажется слишком высокой.

Расходным материалом для лазерного принтера являются картриджи с тонером; иногда имеется возможность дозаправки картриджа порошком. Стоимость печати по расходным материалам у лазерного принтера невысока, но сами принтеры дороже всех других типов (правда, и качественнее).

Лазерные принтеры имеют мощные встроенные процессоры и большой объем буферной памяти, поскольку они должны хранить изображение целой страницы с высоким разрешением. Объемом буферной памяти определяется максимальное разрешение. Особенно много памяти требует цветная печать. Если памяти недостаточно, то принтер печатает только часть изображения на листе — данные для оставшейся части изображения он принимает позже и печатает ее на следующем листе. Можно, конечно, вручную подать лист с первой частью изображения — продолжение будет допечатано, правда, точность совмещения невысока (из-за люфтов бумаги).

Память лазерного принтера может быть расширена установкой дополнительных модулей динамической памяти, однако ряд моделей довольно капризны по отношению к типам устанавливаемых модулей. Внутреннее ПО принтера, хра

нящееся в его ПЗУ, может также быть расширено путем установки дополнительных модулей (PostScript, см. далее) — как правило, флэш-памяти.

Органы управления «персональными» лазерными принтерами минимизированы (как у струйных). Мощные принтеры с несколькими лотками для подачи бумаги и различными вариантами настройки часто имеют небольшой жидкокристаллический дисплей и кнопки, позволяющие управлять принтером с помощью меню.

Лазерные (и светодиодные) принтеры потребляют большую мощность и требуют вентиляции помещения, поскольку вырабатывают озон.

Цветная печать и фотопринтеры

Цветная и полутоновая печать имеет свои сложности, поскольку большинство типов принтеров могут точку либо печатать, либо не печатать. Непосредственно управлять яркостью и цветом (яркостью базисных цветов) пиксела (как у мониторов) большинство технологий печати не позволяет. Цветной (полутоновой) пиксел в принтерах образуется из группы точек; число отпечатанных точек в группе определяет насыщенность пиксела. Для цветных принтеров различают два основных параметра:

- *Линиатура* — шаг полноцветных (полутоновых) пикселов печати, измеряется в lpi (lines per inch — число линий на дюйм). Газеты печатают с линиатурой 80-90 lpi, журналы — 133-150 lpi, высококачественная печать требует более 150 lpi (бывает и 300 lpi).
- *Разрешение принтера* — шаг точек печати, измеряется в dpi (dot per inch — число точек на дюйм). Заметим, что 1200 dpi соответствует шагу точек 21 мкм!

Линиатура связана с разрешением, со способностью принтера управлять яркостью точек и с желаемым количеством градаций полутонов. В большинстве случаев 256 градаций (как это позволяют мониторы) не требуется (мы видим не более 150).

В большинстве принтеров для получения полутонов применяется *растрирование* — представление пиксела группой соседних точек (растром). Если каждый пиксел собирается из матрицы 10x10 черно-белых точек, то, меняя число отпечатываемых точек, можно получить 101 градацию насыщенности. При этом очевидно, что линиатура (lpi) будет в 10 раз меньше разрешения (dpi). В качестве примера можно привести принтер HP LJ1200 — при разрешении 1200 dpi он обеспечивает линиатуру 180-212 lpi. Из этих рассуждений становится понятно, почему печать полутоновых изображений (черно-белых фотографий) на принтере с разрешением 600 dpi дает неудовлетворительное качество.

Для того чтобы повысить линиатуру, нужно повысить разрешение принтера, попытаться управлять яркостью элементарной точки или снизить число градаций (пожертвовать передачей полутонов). В случае цветной полутоновой печати вышеприведенные рассуждения относятся к каждому базисному цвету. В цветной печати применяют базисные цвета модели CMYK (Cyan, Magenta, Yellow, black — бирюзовый, пурпурный, желтый, черный), для каждого цвета

используется свой растр (с разными наклонами, чтобы избежать интерференции). Последний цвет (черный) требуется постольку, поскольку получить действительно черный цвет смесью трех базисных цветов (как в модели RGB) проблематично.

Фотопринтеры отличаются от обычных принтеров улучшенной передачей полутонов. Здесь прибегают к различным ухищрениям, чтобы растр был незаметным. Особенно неприятны бледные цвета — точки в растре редки. В фотопринтерах применяются различные технологии: управление яркостью точек пусть даже с небольшим числом градаций (например, путем изменения размеров капли), применение дополнительных бледных чернил (фактически, это дополнительная градация яркости), повышение разрешения. Для высококачественной печати требуются специальная бумага, а также калибровка цветопередачи. Фотопринтеры выпускают и с уменьшенным форматом листа (например, A5 или 10 x 15 см).

Фотопринтеры могут иметь и другие особенности. В ряде моделей возможно подключение флэш-карт цифровых фотоаппаратов для непосредственной печати изображения. В этом случае принтер снабжается дисплеем, с помощью которого можно выбрать нужное изображение и даже выполнить какую-то предварительную обработку (обрезать края, поменять контрастность, откорректировать цвет и т. п.).

Плоттеры

Плоттеры, они же графопостроители, предназначены для вывода чертежей. Плоттеры являются векторными устройствами (по крайней мере, по входным данным). В плоттерах первых поколений пишущий инструмент перемещался по траектории, заданной отображаемой в данный момент фигурой. Плоттер способен рисовать графические примитивы: точку, отрезок прямой, дугу, эллипс (окружность как его разновидность), прямоугольник. Поток данных, получаемый плоттером, содержит команды рисования этих примитивов и параметры. Многие плоттеры «понимают» и команды написания текста: каждую букву они внутренне интерпретируют как набор отрезков и дуг; для этого они должны иметь соответствующие таблицы знакогенераторов. Плоттеры позволяют выводить изображения на листы разного формата — от A4 у настольных устройств до A1 и A0 у крупных напольных устройств. Для принтеров такие большие размеры недоступны. По способу обеспечения движения пишущего инструмента относительно бумаги различают планшетные и рулонные плоттеры.

В *планшетном плоттере* лист бумаги укладывается на плоский стол и неподвижно закрепляется. На небольших устройствах лист по краям прижимается металлическими полосками к магнитному столу. На устройствах большого формата листы иногда присасываются воздухом через специальные отверстия в столе. Над столом в одном направлении перемещается каретка, вдоль которой перемещается пишущая головка. Вся эта конструкция, напоминающая мостовой кран, приводится в движение двумя шаговыми двигателями, обеспечивающими перемещение пишущей головки по всей поверхности листа. Точность

позиционирования измеряется десятыми и даже сотыми долями миллиметра. Головка перьевого плоттера снабжена пишущим пером. На головке имеется соленоид, который прижимает перо к бумаге в нужных местах. У струйного плоттера головка такого же типа, как и у струйного принтера (черно-белая или цветная). Приводы позиционирования и пишущего узла управляются встроенным микроконтроллером в соответствии с принимаемым потоком команд.

В *рулонном плоттере* имеется горизонтальный барабан, на который кладется лист бумаги и прижимается к барабану валиками. Края листа свободно свисают вниз (это напольные конструкции). Пишущая головка перемещается по направляющей только вдоль оси барабана. Вращение барабана (в обоих направлениях) и перемещение головки совместно обеспечивают взаимно перпендикулярные перемещения пишущего инструмента относительно бумаги. Рулонные плоттеры позволяют выводить чертежи крупного формата, не занимая при этом огромной площади (как планшетные). Здесь жестко ограничена лишь ширина рулона (A1 или AO). Есть устройства, у которых края листа не свисают, а наматываются на специальные барабаны — такие плоттеры могут «создавать» полотна длиной несколько метров. Однако в рулонном плоттере при повторных прогонах довольно трудно обеспечивать точное позиционирование бумаги, которая катается по барабану вперед-назад во время вывода чертежа огромное количество раз. Из-за этого требуется очень высокоточная (а потому и дорогая) механика.

Современные струйные рулонные плоттеры сделаны несколько иначе. По сути дела они являются растровыми струйными принтерами, головка которых имеет ряд (и не один) сопел. При выводе бумага в них по барабану прокатывается всего один раз, в одном направлении, и за этот проход растровым способом выводится все изображение. Растеризация изображения производится во внутреннем ОЗУ огромного размера, но на данном этапе развития технологии это проще, чем делать сложную механику.

Перьевой плоттер способен выбирать перья (по цвету чернил, типу и толщине) из имеющихся у него в распоряжении. Перья бывают разные — типа шариковой ручки (ball tip pen), фломастера (fiber tip pen) или керамического типа (ceramic tip pen) — каждый тип имеет свою нишу применения. Для выбора пера используют разные механизмы. В револьверном механизме перья устанавливаются в ячейки барабана, размещенного у края рабочего стола плоттера. Отдельный привод поворачивает барабан на нужный угол, предоставляя для доступа требуемую ячейку. Головка подводится к барабану и определенным движением вынимает из него перо (предварительно поставив прежнее в свободную ячейку). У других плоттеров перья устанавливаются в ряд держателей, и головка для обмена подводится к одному из них.

Внешний интерфейс плоттера — параллельный или последовательный. В отличие от принтеров, для плоттеров интерфейс не является узким местом — передача графических команд даже по последовательному интерфейсу происходит гораздо быстрее их механического исполнения. Параллельный интерфейс плоттера ничем не отличается от принтерного. С последовательным интерфейсом на старых плоттерах иногда бывают сложности. Некоторые плоттеры с последова

тельным интерфейсом управляют потоком программно, но посылают не стандартные символы XON/XOFF, а *слова* (ASCII-строки). Такой протокол обмена на уровне системы практически не поддерживается (плоттеры сами «разговаривают» с прикладной программой). Это осложняет подключение плоттера к компьютерной сети (например, через принт-сервер).

У плоттеров имеется ряд специфических параметров:

- формат бумаги (максимальный и минимальный размеры листа);
- линейная скорость движения пера при рисовании и холостых перемещениях;
- максимальное ускорение головки;
- точность позиционирования;
- повторяемость позиционирования (способность многократно попадать в заданную точку после длительных «путешествий»);
- количество цветов;
- поддерживаемые языки.

Помимо рисующих плоттеров существуют и режущие плоттеры (cutter), в них вместо пишущей имеется режущая головка с механическим или лазерным резаком.

Форматы данных

Современные принтеры способны работать в любом режиме — графическом или текстовом. После включения питания и аппаратного или программного сброса принтер готов к получению *текстовых данных и команд.* Принтеры, как правило, работают в расширенной (8-битной) таблице ASCII-кодов. Первые 32 кода (0-1Fh) используются для управляющих символов, непосредственно не отображаемых принтером. Далее следуют коды специальных символов, цифр, прописных (uppercase — верхний регистр) и строчных (lowercase — нижний регистр) букв латинского алфавита. Коды 80-FFh требуются для знаков национальных алфавитов (в частности, русского) и символов псевдографики. Из управляющих кодов, используемых при печати в символьном режиме, особо отметим коды возврата каретки (CR, 0Dh), перевода строки (LF, 0Ah) и формата (FF, 0Ch). Если принтеру задан режим AutoLF, то по коду возврата каретки принтер будет автоматически выполнять и перевод строки. Этот режим может быть задан конфигурированием принтера, а также специальным сигналом интерфейса Centronics. Файлы для печати в конце каждой строки, как правило, содержат пару кодов — CR и LF (последовательность байтов 0D, 0A), и при их распечатке в режиме AutoLF будут пропускаться пустые строки. Обычно режим AutoLF не используют. По трактовке управляющих кодов среди матричных принтеров распространены две основные системы команд: IBM (для принтера IBM Pro- Printer) и Epson. Практически все команды изменения режимов печати (переключение шрифтов, изменение размера, эффекты печати и т. п.), а также переключения в графический режим, начинаются с кода Escape (Esc, 1Bh). Далее следует один или более байтов кода команды; формат последовательности оп

ределяется первым байтом (командой), следующим за кодом Esc. Вся эта конструкция называется Escape-последовательностью.

Для графической печати существует множество языков со своими системами команд.

В матричных принтерах использовались два режима печати — битовый образ и растровый режим, довольно подробно описанные в [7]. *Битовый образ* был вполне естественным для первых 8-9-игольчатых принтеров. В этом режиме блок графических данных несет байты, отвечающие за печать одной колонки всех иголок головки принтера. Для 9-игольчатых принтеров было удобно печатать колонки из 8 точек (чтобы колонка уместилась в байт), младшему биту байта соответствовала верхняя иголка. Байты задавали соседние колонки, слева направо. Escape-последовательность графического элемента строки состоит из команды печати, кода режима (разрешения), числа колонок в строке (2 байта), за которыми следует требуемое число байтов данных для каждой колонки. Графический принтер интерпретирует эту последовательность как блок графических данных, а следующие байты — как новую команду или символ текста. Для 24-игольчатых принтеров каждую колонку задают три байта графических данных. Строка будет напечатана после подачи символов CR, LF. В строке может быть несколько графических блоков, расположенных друг за другом, и они даже могут чередоваться (или совмещаться) с текстовыми символами, но использовать эту возможность программно неудобно. Для графической печати нужно отдельно программировать и вертикальный шаг перемещения бумаги (межстрочное расстояние). Управляя шагом и графическим режимом, можно выбирать требуемое разрешение по вертикали и горизонтали. Битовый образ пригоден только для черно-белой печати; он неудобен тем, что формат блока данных зависит от числа иголок принтера (бывают и 24-, и 48-игольчатые принтеры).

В *растровом режиме* черно-белой печати каждый байт графических данных несет информацию о горизонтальной группе из восьми точек линии; старший бит соответствует левой точке, следующие друг за другом байты отображаются слева направо. После байтов, описывающих одну линию, следуют байты следующей линии (сверху вниз), и так до конца страницы (аналогично образу экрана в графическом режиме). Формат цветной печати несколько сложнее, но общая идея сохраняется. Растровый режим естественен для лазерных принтеров — он соответствует способу формирования изображения на барабане. Этот режим поддерживают и многие современные струйные принтеры. Логически этот формат удобнее, поскольку он не зависит от числа сопел; правда, требует довольно большой буферной памяти принтера, но на современном этапе развития техники это уже не проблема. Растровый режим позволяет представить любое изображение. Однако здесь (как и в случае битового образа) объем передаваемых данных растет пропорционально произведению вертикального и горизонтального разрешений (dpi) на размеры изображения (в дюймах) и число битов на пиксел для цветной печати.

Для лазерных принтеров фирма Hewlett-Packard разработала специальный *язык PCL* (Printer Control Language), в котором помимо управляющих команд, ана

логичных Escape-последовательностям матричных принтеров, имеются и графические, описывающие рисование геометрических примитивов. В языке имеются и средства работы со встроенными шрифтами принтера, обеспечивающие масштабирование и повороты букв. Язык PCL поддерживают ряд струйных принтеров. Использование языка PCL позволяет сократить объем данных, передаваемых принтеру для печати сложных изображений, состоящих из текста и графики, по сравнению с растровым форматом. Особенно эта экономия существенна для высокого разрешения и цветной печати — для PCL объем передаваемой информации не так сильно зависит от разрешения и цветности. Однако для доступа к этим возможностям язык PCL должно «понимать» и приложение, осуществляющее графический вывод. Поддержка PCL вполне естественна для приложений с векторной графикой (включая текстовые процессоры и издательские системы). Сугубо растровые системы, естественно, генерируют команды растровой печати.

Язык PostScript также предназначен для лазерных принтеров. В этом языке вся страница описывается в векторном виде. Шрифты задаются контурами (линиями Безье), и их растеризацией (в нужном цвете) занимается встроенный процессор принтера в соответствии с возможностями принтера и выбранным разрешением печати. Векторное описание всех объектов (символов и геометрических фигур) обеспечивает возможность точного выполнения трансформаций (масштабирования, позиционирования, поворотов, зеркальных отражений). При этом файл печати не зависит от типа принтера (или иного устройства) — требуется только поддержка версии языка, на которой создан файл. Шрифты, используемые для отображения страницы, передаются в файле печати в компактном векторном виде. Кроме того, в принтер PostScript встроено большое количество стандартных шрифтов, которые позволяют еще больше экономить память. Реализация PostScript требует наличия у принтера мощного встроенного процессора, ОЗУ и ПЗУ большого объема.

Для *плоттеров*, которые получают исключительно векторные команды рисования, существует несколько различных языков. Общепринятым является *язык HP-GL*, его понимают все плоттеры и практически все прикладные программы, выполняющие графический вывод на плоттер. Для плоттеров, особенно перьевых, актуальна оптимизация входных данных. Например, в многоцветных изображениях гораздо выгоднее рисовать сначала все элементы одного цвета, затем — другого. Программы, генерирующие данные для рисования, обычно поступают иначе: они «отрабатывают» изображения по объектам. Серия мелких многоцветных объектов порождает частую смену перьев, за каждым из которых головка должна «сбегать» к магазину. Иногда имеет смысл использовать дополнительные программы-оптимизаторы, входные данные для которых предоставляет выходной файл графического приложения.

Поскольку между печатающим (чертящим) приложением и принтером (плоттером) всегда находится программный драйвер, при несоответствии их языков почти всегда требуется драйвер-транслятор. Так, матричный принтер, не русифицированный на аппаратном уровне, можно русифицировать программно. Предпочтительно использовать загружаемый знакогенератор принтера — для

этого компьютер должен послать в принтер блок данных определенного формата, содержащий команды загрузки и собственно содержимое знакогенератора. Однако такая загрузка должна выполняться каждый раз после включения принтера, чтобы драйвер мог отслеживать состояние принтера (по сигналам интерфейса) и своевременно подгружать знакогенератор. Однако не все принтеры имеют такую возможность. Проще обстоит дело, когда у принтера есть знакогенератор русских букв, но они расположены в ином порядке, чем требуется. В этом случае драйвер-русификатор должен просто перекодировать символы по таблице. Правда, для этого ему требуется «понимать» графические команды принтера и прозрачно (без преобразования) пропускать графические данные. Если принтер вообще не имеет нужного алфавита и загружаемого знакогенератора, приходится печатать текст в графическом режиме. Для этого драйвер должен выполнять растеризацию символов, не известных принтеру или всех подряд (для однородности), и выводить их на принтер в графическом режиме. При этом более чем на порядок возрастает объем передаваемой информации, что снижает скорость печати, особенно при маломощном процессоре (время расходуется и на растеризацию, и на собственно вывод данных). Аппаратная или программная русификация принтеров актуальна лишь для печати текстовых файлов средствами DOS. Приложения Windows используют графические режимы принтеров, и вопросы русификации уже переходят в чисто программную область (драйверы и системные шрифты). Однако печать в графическом режиме на матричных игольчатых принтерах по нынешним меркам слишком медленна и шумна, хоть и возможна. Для такой печати больше подходят струйные, а еще лучше — лазерные принтеры.

Программный драйвер может реализовывать графический язык, не поддерживаемый принтером, — например, есть программные реализации языка PostScript. Однако при этом центральный процессор компьютера нагружается объемной задачей растеризации, причем в ОЗУ должен умещаться весь растровый образ выводимой страницы. Кроме того, на принтер при этом выводится огромный объем данных, что особенно неприятно для сетевого принтера. Так что при больших объемах печати лучше использовать настоящий «железный» (аппаратный) принтер PostScript, а не его программную эмуляцию.

Из вышесказанного вполне понятно, что драйвер принтера должен соответствовать типу принтера и его языковым возможностям. Так, при использовании принтера PostScript об этом должен «знать» и драйвер, иначе графический вывод будет производиться всегда в растровом режиме и никаких преимуществ аппаратной поддержки PostScript пользователь не получит.

Интерфейсы принтеров и плоттеров

Современные принтеры, печатающие графические изображения (в том числе текст в графическом режиме) с высоким разрешением, требуют высокоскоростной передачи данных по внешнему интерфейсу. У них интерфейс может стать узким местом, и фаза передачи данных будет занимать значительное время, расходуемое на вывод изображения. Напомним, что лазерный принтер не начинает печатать страницу до тех пор, пока она целиком не загружена в его буфер

ную память. Параллельный интерфейс для этого уже работает на пределе возможностей, обеспечивая скорость передачи до 2 Мбайт/с в режиме ECP или EPP (см. 15.3). Обычный последовательный интерфейс RS-232C с его пределом около 15 Кбайт/с здесь, конечно же, неприемлем. Тем не менее, автору лично доводилось пользоваться лазерным принтером (HP), подключенным к COM- порту, — лист на печать передавался около получаса (зато печатался быстро). В качестве внешнего интерфейса в последнее время стали чаще применять шину USB с ее удобным кабелем; в версии 1.0 она обеспечивает скорость до
1,5 Мбайт/с, а версия 2.0 дает скорость уже до 24 Мбайт/с. В принтерах может применяться и интерфейс SCSI, но широкого распространения он не получил. Также пока очень сдержанно применяется шина FireWire.

Принтеры, особенно мощные, часто применяют для совместной работы в сети — задания на печать могут посылать пользователи с разных компьютеров. Разделяемый принтер может соединяться с сетью разными способами:

- Принтер может подключаться обычным интерфейсом (параллельным или USB) к компьютеру, включенному в сеть. Этот компьютер становится принт-сервером, для чего у него должно быть запущено специальное ПО. В сетях Windows для этого достаточно запустить в сетевом окружении службу доступа к файлам и принтерам, разрешить совместный доступ к ресурсам компьютера и конкретно — к данному принтеру.
- Принтер может подключаться параллельным (или последовательным) интерфейсом к *аппаратному принт-серверу* — небольшому устройству, по виду напоминающему малогабаритный хаб и подключенному к сети. Программные (протокольные) функции принт-сервера выполняет встроенное ПО (firmware) данного устройства. Принт-сервер обычно имеет несколько внешних интерфейсных портов, параллельных, а иногда и последовательных, и к нему может быть подключено несколько принтеров (плоттеров). ПО принт- сервера обычно рассчитано на один из сетевых протоколов, и принт-сервер для Novell NetWare не подходит для сетей Windows (и наоборот). Бывают и мультипротокольные принт-серверы.
- Принтер может непосредственно подключаться к сети, как правило, по интерфейсу Ethernet, разъемом BNC (10Base2) к коаксиальному кабелю (шине) или RJ-45 (10BaseT или 100BaseTX) витой парой к сетевому концентратору. Сетевой интерфейс имеют мощные лазерные принтеры; для них предпочтительнее интерфейс 100BaseTX (Fast Ethernet), обеспечивающий скорость до 10 Мбайт/с. Протокольные функции принт-сервера в данном случае выполняются встроенным ПО принтера, и здесь также поддерживаемый протокол (протоколы) должен соответствовать используемому в сети. Сетевые принтеры (принтер с аппаратным и программным интерфейсом локальной сети), как правило, имеют и альтернативный обычный интерфейс Centronics.

Сетевой принтер (или принт-сервер, к которому он подключен), должен быть по возможности привилегированным узлом сети. Его желательно подключать к порту коммутатора или непосредственно в сегмент, в который входят его пользователи. Сетевая печать из приложений Windows очень нагружает 10-ме-

габитную сеть Ethernet, заставляя применять коммутаторы или переходить на Fast Ethernet. Подробнее об организации сетевой печати см. в [6].

Интерфейс Centronics

Большинство принтеров и плоттеров имеют внешний параллельный интерфейс *Centronics* (ИРПР-М) для непосредственного подключения к LPT-порту. Понятие «Centronics» относится как к набору сигналов и протоколу взаимодействия, так и к 36-контактному разъему на принтерах. Интерфейс ориентирован на передачу потока байтов данных к принтеру и прием сигналов состояния принтера. Интерфейс Centronics поддерживается всеми принтерами с параллельным интерфейсом. Его отечественным аналогом является интерфейс *ИРПР-М*[1]. Назначение сигналов интерфейса приведено в табл. 11.3, а временные диаграммы обмена с принтером — на рис. 11.3. Передача данных начинается с проверки готовности принтера — состояния линии Busy. Строб данных может быть коротким — доли микросекунды, и порт заканчивает его формирование, не обращая внимания на сигнал Busy. Во время строба данные должны быть действительными. Подтверждением приема байта (символа) является сигнал Аск#, который вырабатывается через неопределенное время после приема строба (за это время принтер может выполнять какую-либо длительную операцию, например прогон бумаги). Импульс Ack# является запросом принтера на прием следующего байта, его используют для формирования сигнала прерывания от порта принтера. Если прерывания не задействуют, то сигнал Ack# игнорируется и весь обмен управляется парой сигналов Strobe# и Busy. Свое состояние принтер может сообщить порту по линиям Select, Error#, PaperEnd — по ним можно определить, включен ли принтер, исправен ли он и есть ли бумага. Формированием импульса на линии Init# принтер можно инициализировать (при этом он очищает весь свой буфер данных). Режимом автоматического перевода строки, как правило, не пользуются, и сигнал AutoLF# имеет высокий уровень. Сигнал Selectln# позволяет логически отключать принтер от интерфейса.

Таблица 11.3. Сигналы интерфейса Centronics

Сигнал	I/O[1]	Контакт	Назначение
Strobe#	I	1	Строб данных. Данные фиксируются по низкому уровню сигнала
Data [0:7]	I	2–9	Линии данных. Data 0 (контакт 2) — младший бит
Ack#	O	10	Acknowledge — импульс подтверждения приема байта (запрос на прием следующего). Может использоваться для формирования запроса прерывания
Busy	O	11	Занято. Прием данных возможен только при низком уровне сигнала
PaperEnd	O	12	Высокий уровень сигнализирует о конце бумаги

[1] Не путать с интерфейсом *ИРПР* (он же *IFSP)* — интерфейсом с инверсной шиной данных, все входные линии которого сильно нагружены (недопустимо для LPT-порта), протокол квитирования отличается, сигналы ошибки и конца бумаги отсутствуют.

Сигнал	I/O[1]	Контакт	Назначение
Select	O	13	Сигнализирует о включении принтера (обычно в принтере соединяется резистором с цепью + 5 В)
Auto LF#	I	14	Автоматический перевод строки. При низком уровне принтер, получив символ CR (Carriage Return — возврат каретки), автоматически выполняет и функцию LF (Line Feed — перевод строки)
Error#	O	32	Ошибка: конец бумаги, состояние OFF-Line или внутренняя ошибка принтера
Init#	I	31	Инициализация (сброс в режим параметров умолчания, возврат к началу строки)
Select In#	I	36	Выбор принтера (низким уровнем). При высоком уровне принтер не воспринимает остальные сигналы интерфейса
GND	--	19–30, 33	Общий провод интерфейса

[1] Символы I и O задают направление (вход-выход) применительно к принтеру.

Рис. 11.3. Передача данных по протоколу Centronics

Параллельный порт (LPT) современных компьютеров может работать в разных режимах — как в стандартном SPP (его реализуют все порты), так и в расширенных (см. главу 15). Практически все принтеры могут работать с портом в режиме *SPP*, но применение расширенных режимов дает свои преимущества:

- Двунаправленный режим (*Bi-Di*) не повышает производительность, но обеспечивает доставку сообщений о состоянии и параметрах принтера.
- Скоростные режимы *(Fast Centronics)* существенно повышают производительность принтера, но могут потребовать качественного кабеля (см. далее). От принтера не требуется каких-либо дополнительных «интеллектуальных» способностей.
- Режим *ECP* — потенциально самый эффективный, имеет системную поддержку во всех версиях Windows. На некоторых принтерах он реализован не полностью (может отсутствовать аппаратная компрессия). Режим *ECP* поддерживают принтеры HP DeskJet моделей 6xx, LaserJet 4 и далее, современные модели фирмы Lexmark. Требует применения кабеля, по частотным свойствам соответствующего IEEE 1284.

Простейший вариант *кабеля подключения принтера* — 18-проводный кабель с неперевитыми проводами. Он используется для работы в режиме *SPP*. При длине более 2 м желательно, чтобы хотя бы линии Strobe# и Busy были перевиты

с отдельными общими проводами. Для скоростных режимов данный кабель может оказаться непригодным, причем сбои могут происходить нерегулярно и лишь при определенных последовательностях передаваемых кодов. Встречаются кабели *Centronics*, у которых отсутствует связь контакта 17 разъема PC с контактом 36 разъема принтера. При попытке подключения таким кабелем принтера, работающего в стандарте 1284, появится сообщение о необходимости применения двунаправленного кабеля. Принтер не может сообщить системе о поддержке расширенных режимов, на что рассчитывают драйверы принтера. Другое проявление отсутствующей связи — «зависание» принтера по окончании печати задания из Windows. Эту связь можно организовать подпайкой дополнительного провода или же просто заменой кабеля.

Неплохие электрические свойства имеют ленточные кабели, у которых сигнальные цепи (цепи управляющих сигналов) чередуются с общими проводами. Но их применение в качестве внешнего интерфейса непрактично (нет второго защитного слоя изоляции, высока уязвимость) и не эстетично (круглые кабели смотрятся лучше).

Идеальным вариантом являются кабели, в которых все сигнальные линии перевиты с общими проводами и заключены в общий экран, — то, что требует IEEE 1248. Такие кабели гарантированно работают на скоростях до 2 Мбайт/с и могут достигать длины 10 м.

В табл. 11.4 описана распайка *кабеля подключения принтера с* разъемом XI типа A (DB25-P) со стороны PC и X2 типа *B (Centronics-36)* или типа *C* (миниатюрный) со стороны принтера. Использование общих проводов (GND) зависит от качества кабеля (см. выше). В простейшем случае (18-проводный кабель) все сигналы GND объединяются в один провод. Качественные кабели требуют отдельного обратного провода для каждой сигнальной линии, однако в разъемах типа A и B для этого недостаточно контактов (в таблице в скобках указаны номера контактов разъема PC типа A, которым соответствуют обратные провода). В разъеме типа C обратный провод (GND) имеется для каждой сигнальной цепи; сигнальным контактам 1-17 этого разъема соответствуют контакты GND 19-35.

Таблица 11.4. Кабель подключения принтера

X1, разъем PC типа A	Сигнал	X2, разъем PRN типа B	X2, разъем PRN типа C
1	Strobe#	1	15
2	Data0	2	6
3	Data1	3	7
4	Data2	4	8
5	Data3	5	9
6	Data4	6	10
7	Data5	7	11
8	Data6	8	12
9	Data7	9	13
10	Ack#	10	3
11	Busy	11	1

X1, разъем PC типа A	Сигнал	X2, разъем PRN типа B	X2, разъем PRN типа C
12	PaperEnd	12	5
13	Select	13	2
14	Auto LF#	14	17
15	Error#	32	4
16	Init#	31	14
17	Select In#	36	16
18	GND (1)	19	33
19	GND (2, 3)	20, 21	24, 25
20	GND (4, 5)	22, 23	26, 27
21	GND (6, 7)	24, 25	28, 29
22	GND (8, 9)	26, 27	30, 31
23	GND (11, 15)	29	19, 22
24	GND (10, 12, 13)	28	20, 21, 23
25	GND (14, 16, 17)	30	32, 34, 35

Последовательные интерфейсы

Из последовательных интерфейсов в принтерах чаще всего используется RS- 232C для подключения к COM-порту. Встречаются принтеры с последовательными интерфейсами «токовая петля», или RS-422, которые подключаются к COM-порту через специальные переходники. Принтеры работают всегда по асинхронному протоколу передачи и, как правило, позволяют настраивать конфигурацию последовательного интерфейса. Задаются частота передачи, формат посылки (число информационных, старт- и стоп-битов, контроль четности) и протокол управления потоком: программный XON/XOFF или аппаратный RTS/CTS. Подключение принтеров и плоттеров к COM-порту требует кабеля, соответствующего выбранному протоколу, схемы кабелей приведены на рис. 11.4 и 11.5. Аппаратный протокол предпочтительнее — стандартный драйвер COM-порта пользуется именно им. Естественно, параметры интерфейса принтера должны соответствовать параметрам, заданным для COM-порта. Порт конфигурируется, например, DOS-командой MODE (см. 15.4). Заметим, что при печати средствами DOS (командой COPY или PRINT) прерывания от порта не используются.

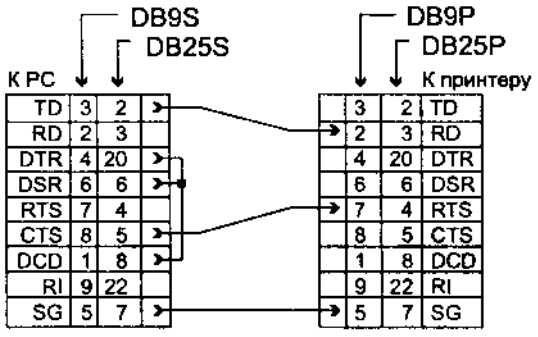


Рис. 11.4. Кабель подключения принтера по протоколу RTS/CTS

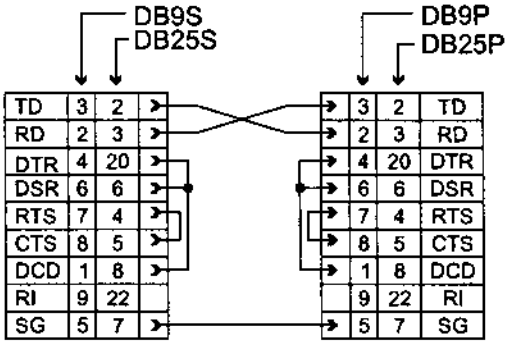


Рис. 11.5. Кабель подключения принтера по протоколу XON/XOFF

Если принтер имеет интерфейс «токовая петля», то для него потребуется преобразователь сигналов, простейшая схема которого приведена на рис. 11.6. Здесь принтер подключается по «токовой петле» к COM-порту с аппаратным управлением потоком. Для получения двуполярного сигнала, требуемого для входов COM-порта, применяется питание от интерфейса.

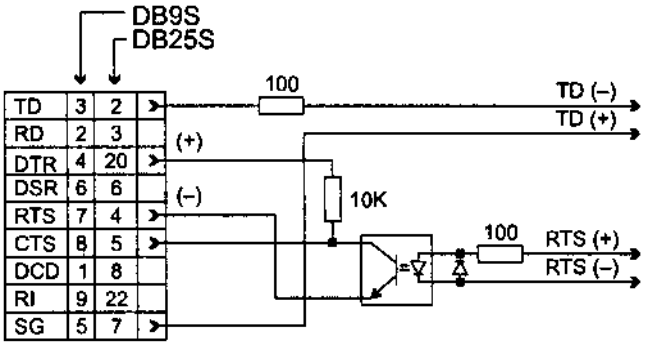


Рис. 11.6. Подключение принтера с интерфейсом «токовая петля 20 мА» к COM-порту

Системная поддержка принтера

Вывод на принтер через порт LPT в стандартном режиме (SPP) *по интерфейсу Centronics* имеет поддержку на уровне BIOS. Поддержка всех других режимов работы порта (Fast Centronics, ECP) осуществляется только дополнительными драйверами или средствами ОС. Сервис *BIOS* Int 17h обеспечивает инициализацию, вывод байта данных и опрос состояния принтера.

Перехват прерывания Int 17h является удобным способом внедрения собственных драйверов принтера. Потребность в них может возникать при подключении к порту принтера с интерфейсом *ИРПР* или необходимости перекодировки символов. Если разрабатываемый драйвер предназначен не только для перекодировки, но и для изменения протокола (через Int 17h можно организовать вывод через LPT-порт по протоколу ИРПР и даже через COM-порт), следует вни

мательно отнестись к битам возвращаемого байта состояния. При их неправильном формировании попытки вывода на печать могут приводить к сообщениям об ошибках.

Печать содержимого экрана (print screen) поддерживается прерыванием BIOS Int 05. Обработчик этого прерывания посимвольно выводит содержимое видеопамяти (в текстовом режиме) на порт LPT1. Прерывание Int 05 вызывается обработчиком аппаратного прерывания от клавиатуры (Int 09), когда обнаруживается нажатие клавиши Print Screen (PrtSc). В ОС Windows нажатие клавиши PrtSc печать не вызывает — содержимое экрана копируется в буфер обмена, откуда его можно вставить в любое приложение, использующее графику.

11.6. Игровые устройства — джойстик, руль, педали

Джойстик является одним из первых чисто развлекательных устройств IBM PC, и его название (joystick) можно буквально перевести как «палочка для удовольствия». Джойстик позволяет вводить в компьютер информацию о двух координатах ручки управления и о состоянии двух кнопок. Вместо двухкоординатной ручки может быть сделан руль автомобиля с педалью газа или что-либо иное, были и простейшие игровые устройства (paddle) с парой ручек потенциометров и парой кнопок. Джойстик используют в играх, где благодаря возможности пропорционального управления (сигнал вырабатывается пропорционально отклонению ручки) он гораздо привлекательнее, чем клавиатура.

Для устройств связи с оператором (одним из таких устройств и является джойстик) в спецификации USB выделен специальный класс — HID (Human Interface Device). Эти устройства могут предоставлять набор специальных физических дескрипторов, описывающих, какой частью тела человек воздействует на тот или иной орган управления (сообщаемый параметр). К примеру, киберперчатка имеет множество датчиков, связанных с разными частями кисти оператора. Физические дескрипторы позволяют связывать передаваемые параметры с действиями оператора.

С самых первых моделей IBM PC был введен и, фактически, стандартизован интерфейс игрового адаптера — *игровой порт* (game port), к которому можно подключить до двух джойстиков или иных устройств. Суммарно на порте доступны 4 координатных датчика (X1, X2, Y1 и Y2) и 4 кнопки. Назначение координатных датчиков зависит от игры и конструкции манипулятора. Для авиасимуляторов X1 может соответствовать перемещению рукоятки вверх-вниз, Y1 — влево-вправо, X2 — нажатию левой и правой педалей, Y2 — рукоятке сектора газа. Для автомобильных рулей X1 — руль, Y1 — газ, X2 — тормоз (газ и тормоз могут быть совмещены в координате Y1). Помимо игровых целей порт может применяться и для подключения более «серьезных» датчиков.

Ввод дискретных сигналов от кнопок пояснений не требует. Упрощенная схема одного аналогового канала приведена на рис. 11.7. Конденсатор заряжается через переменное сопротивление датчика и разряжается через ключ; компаратор

сравнивает напряжение на конденсаторе с некоторым порогом. Выходы компараторов всех четырех каналов преобразования, как и дискретные входы, собираются в регистр порта, который может быть программно считан. Любая запись в порт приводит к открытию ключей и разряду конденсаторов, при этом биты 0-3 устанавливаются в 1. Эти биты сбросятся в 0, когда конденсаторы их каналов зарядятся до порога срабатывания компаратора; время заряда до срабатывания определяется текущим значением сопротивления каждого датчика. Замыканию кнопок соответствуют нули в битах 5-7.

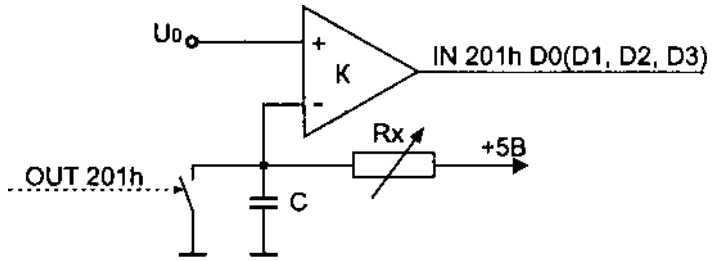


Рис. 11.7. Канал аналогового ввода

Преобразование выполняется чисто программно и инициируется выводом любого байта в регистр адаптера. Далее программа циклически выполняет чтение регистра адаптера и измеряет время до возврата в нулевое состояние битов 0-3, соответствующих четырем аналоговым каналам. Если аналоговый вход закорочен на шину GND или цепь измеряемого сопротивления разорвана, соответствующий бит не обнуляется. Поэтому в программе преобразования должен быть предусмотрен тайм-аут. Для измеряемых сопротивлений в диапазоне R = 0-100 кОм время [мкс] определяется по формуле

$$T = 24{,}2 + 11 \times R.$$

Точность и линейность преобразования невысоки, выполняется оно небыстро (до 1,12 мс) и сильно загружает процессор. Однако в отличие от «настоящих» аналого-цифровых преобразователей этот достается даром — игровой адаптер входит в состав практически всех комбинированных плат последовательных и параллельных портов и звуковых карт.

Порт имеет *разъем-розетку DB-15S*. Назначение выводов и соответствие сигналов битам регистра иллюстрирует табл. 11.5. Кнопки подключаются к шине GND, переменные резисторы — к шине питания +5 В (рис. 11.8). Вместо переменных резисторов к порту можно подключать источники измеряемых токов. Аналоговые каналы можно использовать для дискретного ввода, если их входы подключить к кнопкам, замыкающим их на шину GND, и к резисторам, «подтягивающим» их к уровню + 5 В. Два джойстика (A и B) подключаются через Y-образный переходник-разветвитель. На звуковых картах вместе с джойстиками могут подключаться и внешние MIDI-устройства через разъем «Game», и специальный кабель-адаптер, обеспечивающий гальваническую развязку входного сигнала и ограничение выходного тока. Для интерфейса MIDI ис

пользуются контакты 12 и 15, ранее предназначавшиеся для шин GND и +5V. Такое назначение делает безопасным подключение адаптера MIDI к «чистому» игровому порту и обычного джойстика — к игровому порту с сигналами MIDI.

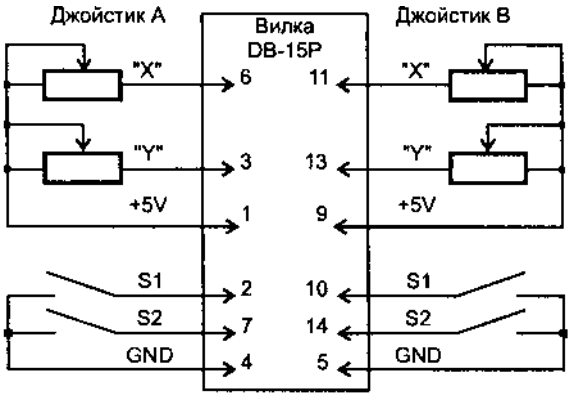


Рис. 11.8. Подключение датчиков к игровому адаптеру

Таблица 11.5. Интерфейс игрового адаптера и MIDI

Бит	Назначение	Контакт
7	Джойстик B кнопка #2	14
6	Джойстик B кнопка #1	10
5	Джойстик A кнопка #2	7
4	Джойстик A кнопка #1	2
3	Джойстик B Y-координата (Y2)	13
2	Джойстик B X-координата (X2)	11
1	Джойстик A Y-координата (Y1)	6
0	Джойстик A X-координата (X1)	3
–	GND	4, 5, (12)
–	+5 B	1, 8, 9, (15)
–	MIDI In (Rx) – вход (на звуковой карте)	15
–	MIDI Out (Tx) – выход (на звуковой карте)	12

Системную поддержку джойстика обеспечивает BIOS Int 15h при AH = 84h. Этот сервис реализует опрос кнопок и чтение координат джойстиков. Стандартный джойстик поддерживается ОС Windows.

Из-за расточительного использования процессорного времени на преобразования (а их приходится выполнять регулярно, чтобы отслеживать динамику движения) традиционный джойстик и игровой адаптер из современных компьютеров изживают. Их место должны занять устройства с интерфейсом шины USB, в которых преобразования с точки зрения центрального процессора выполняются аппаратно. Такие джойстики называют цифровыми (в отличие от вышеописанного аналогового). Есть и цифровые джойстики с интерфейсом COM- порта.

Помимо джойстиков с электрической обратной связью существуют джойстики с механической обратной связью — в них на органы управления (рычаг, руль) воздействуют моторы привода, получающие управляющие сигналы от компьютера. Таким образом, например, могут имитироваться сопротивление повороту руля автомобиля, удар по рулю при наезде на препятствие или, наоборот, ослабление сопротивления руля при заносе. Для подачи управляющих сигналов интерфейс игрового порта не приспособлен, поэтому для таких сигналов используется дополнительный интерфейсный кабель (от COM-порта). Джойстик с интерфейсом USB, естественно, по одному кабелю передает информацию в обе стороны. Цифровые джойстики требуют установки специальных драйверов.

11.7. Коммутаторы устройств ввода-вывода

Иногда возникает потребность в поочередном использовании устройств ввода-вывода несколькими близко расположенными компьютерами. Конечно, для этого можно просто каждый раз заново соединять интерфейсные кабели, но при частых переключениях это неудобно и чревато выходом устройств из строя при переключениях «на ходу».

Для *принтеров* эта задача сейчас чаще всего решается путем объединения компьютеров в локальную сеть, однако существуют и специальные устройства — коммутаторы параллельного интерфейса. В простейшем случае это просто коробка с галетным переключателем, несколькими (обычно четырьмя) входными разъемами Centronics (как у принтеров) и одним выходным DB-25S (как у LPT-порта). Входные разъемы соединяют с LPT-портами обслуживаемых компьютеров, выходной разъем — с принтером; все соединения выполняются обычными принтерными кабелями. Принтер в каждый момент работает лишь с одним компьютером в зависимости от положения переключателя, коммутирующего все интерфейсные сигналы. Есть и автоматические электронные коммутаторы, соединяющие принтер с портом, проявившим активность. Установленное соединение удерживается на время активной передачи данных. При паузе в несколько секунд коммутатор считает, что очередное задание уже выведено на печать, и полагает принтер свободным. Такая простая логика коммутации в большинстве случаев работает нормально, но для некоторых приложений и нештатных ситуаций возможна «чересполосица» вывода заданий от нескольких компьютеров.

Возможна и организация принтерной «мини-сети» нестандартными средствами (рис. 11.9). Здесь используются специальные интерфейсные адаптеры, преобразующие параллельный интерфейс LPT-порта в особый последовательный. Специальные адаптеры ETHERPass устанавливаются на разъем LPT-порта (T — передатчик) и параллельный порт принтера (R — приемник). Между собой эти устройства связываются четырехпарным кабелем UTP с разъемами RJ-45 длиной до 100 м. Питаются приемники и передатчики непосредственно от интерфейса, но при большой длине шлейфа требуют внешних блоков питания. Адап

теры позволяют объединять несколько компьютеров и несколько принтеров: передатчики и приемники имеют по паре гнезд RJ-45, и их можно соединить в цепочку (длиной до 100 м). На каждом устройстве переключателями устанавливается адрес (0-3), приемник каждого принтера должен иметь уникальный адрес. На передатчиках выставляется адрес принтера, на который должна направляться печать, причем на один принтер может направляться печать от нескольких компьютеров. Разделение заданий осуществляется по паузам между посылками символов в порты компьютеров. Сеть работает «прозрачно» для операционной системы и прикладных программ, оперируя сигналами «занято» и корректно сообщая об отсутствии бумаги. Применение этих адаптеров значительно «облегчает» кабельное хозяйство, когда требуется организация мощной системы печати. В этой сети легко обеспечить резервирование принтеров без переключения кабелей. Если передатчики подключаются к принт-серверам, механизм разделения заданий может давать сбои. В этом случае можно отказаться от обслуживания одним принтером нескольких компьютеров этой принтерной сети, а разделяемость принтера обеспечивать штатными средствами сетевой ОС.

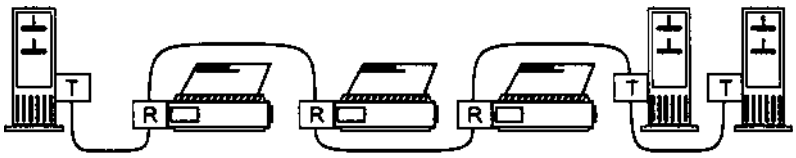


Рис. 11.9. Принтерная «мини-сеть»

Коммутация консольных устройств (клавиатуры, дисплея, а иногда и мыши) позволяет использовать одну консоль для нескольких рядом стоящих компьютеров, что актуально для серверных центров. Коммутатор подключается кабелями ко всем обслуживаемым компьютерам и к клавиатуре, монитору и мыши рабочего места оператора, которое получается удобным и компактным. Для того чтобы при начальной загрузке «голый» системный блок не потребовал подключения клавиатуры и дисплея, нужно сделать специальные настройки в CMOS Setup (см. 6.6).

Простейший коммутатор консоли — это механический переключатель, как правило, галетный, коммутирующий все сигнальные цепи интерфейсов клавиатуры, монитора и мыши. *Коммутация клавиатуры*, как правило, проблем не вызывает: интерфейс дискретный, работает на низкой частоте сигнала, кратковременное отключение питания клавиатуры на момент перекоммутации проходит незаметно. Отключение и подключение клавиатуры в процессе работы компьютером отрабатываются безболезненно и не приводят к побочным эффектам. С *коммутацией мыши* электрических проблем тоже нет, но компьютер отключенную мышь может «потерять», а в момент подключения возможны побочные эффекты (это особенность драйверов). По этой причине зачастую мышь не коммутируют — подключают к каждому компьютеру свою, благо она гораздо компактнее, чем клавиатура, а тем более монитор. Безболезненную коммутацию мыши выполняют более дорогие активные (интеллектуальные) коммута

торы, создающие иллюзию присутствия мыши для всех подключенных к такому коммутатору системных блоков. *Коммутация мониторов*, на первый взгляд, тоже не так уж сложна. Однако в режимах высокого разрешения с высокой частотой развертки некачественное выполнение монтажа и упрощение схемы приводит к появлению эхо-отражений на экране монитора (см. главу 10). Кроме того, возможны ошибки в определении типа монитора (когда он отключен) на этапе инициализации устройств. POST узнает у адаптера тип монитора (адаптер его может чувствовать по наличию терминаторов на линиях цветовых сигналов), чтобы установить видеорежим (цветной или черно-белый). ОС «разговаривает» с PnP-монитором по каналу DDC, чтобы установить режим разрешения. Если компьютер должен загружаться «вслепую» (когда он отключен от консоли), то во избежание недоразумений следует устанавливать фиксированные, а не автоматически определяемые параметры монитора в CMOS Setup (возможность их настройки есть не всегда) и в конфигурационных файлах ОС.

ГЛАВА 12

Аудиосистема ПК

С самого рождения компьютер IBM PC имел голос — PC Speaker, превращающий компьютер в простейший синтезатор. Пройдя путь от программно-управляемого динамика до современных цифровых аудиокодеков, синтезаторов и сигнальных процессоров, современные ПК стали полноправными участниками процесса создания, записи, редактирования и воспроизведения аудиоинформации высокого качества. Представление о наборе аудиосредств современного мультимедийного компьютера дает рис. 12.1.

Рис. 12.1. Аудиосистема PC

Типовая звуковая карта в своем составе имеет цифровой канал записи-воспроизведения, микшер, синтезатор и MIDI-порт.

Цифровой аудиоканал, он же *аудиокодек*, обеспечивает возможность моно- или стереофонической записи и воспроизведения аудиофайлов с уровнем качества начиная от уровня кассетного магнитофона и заканчивая уровнем аудио-CD и даже выше. *Запись* (recording) производится оцифровкой (аналого-цифровым преобразованием) выборок мгновенного значения сигнала; современные карты

позволяют принимать и цифровые аудиоданные. Оцифрованный звук может храниться в файлах, для которых обычно используется расширение .WAV[1] (сокращенно от wave — волна). Размер файла зависит от длительности записи, разрядности преобразования, частоты квантования и количества каналов (моно- или стереозапись). Эти «волновые файлы» могут редактироваться программными средствами, которые обычно позволяют вывести на экран подобие осциллограмм записанных сигналов. При *воспроизведении* (playback) поток цифровых данных выводится на внешний интерфейс, аналоговый (линейный выход или выход усилителей на колонки или наушники) или цифровой.

Микшер с программным управлением обеспечивает регулировку входных и выходных сигналов, позволяя смешивать входные сигналы от нескольких источников (микрофона, CD, внешнего входа и синтезатора). В стереокарте (а моно- карты уже давно не используют) каждый источник должен иметь раздельные регуляторы уровня для каждого канала. Внешне (в графической оболочке программного интерфейса) это может выглядеть и как общий регулятор уровня и регулятор баланса. Для монофонических источников (например, микрофона) помимо регулятора уровня имеется регулятор панорамы, позволяющий балансировать уровни сигналов, посылаемых от данного источника в левый и правый стереоканалы. Физически это опять-таки могут быть просто два регулятора уровня для одного и того же сигнала. Дополнительно к микшеру карта обычно допускает регулировку тембра по низким и высоким частотам или даже имеет эквалайзер — многополосный регулятор тембра.

Синтезатор обеспечивает имитацию звучания музыкальных инструментов и воспроизведение различных звуков. Из множества методов синтеза в звуковых картах в основном используют два — частотный и волновой:

- *FM Music Synthesizer* — синтезатор с частотной модуляцией (аббревиатура FM означает Frequency Modulation — частотная модуляция) — обеспечивает невысокое качество синтеза.
- *WT Music Synthesizer* — синтезатор с табличным синтезом (аббревиатура WT означает Wave Table — волновая таблица) хранит в своей памяти образцы сигналов натуральных инструментов. Волновые синтезаторы обеспечивают высокое качество синтеза, но поначалу они были заметно дороже.

Встроенный *усилитель* имеет мощность до 4 Вт на канал, хотя многие адаптеры обеспечивают мощность, достаточную только для наушников.

Колонки (speakers) для PC несколько отличаются от обычных бытовых акустических систем. Они, как правило, малогабаритные, поскольку предназначены для установки на столе по бокам от монитора. Малые габариты, конечно же, отражаются на качестве и выходной мощности. Хорошие колонки имеют специальный магнитный экран или улучшенную конструкцию магнитной системы динамиков, чтобы предотвратить воздействие магнитного поля на ЭЛТ-монитор. Сильное магнитное поле нарушает линейность развертки и сведение лучей на экране монитора. Ряд моделей «мультимедийных» мониторов оборудован

[1] В файлах MP3 (MPEG-2 Layer 3) сжатый аудиопоток представляется в ином виде.

встроенными акустическими системами. *Активные колонки* (active speakers) имеют встроенный усилитель, требующий внешнего (или батарейного) питания. Они могут иметь регуляторы громкости и тембра. *Пассивные колонки* встроенного усилителя не имеют, их мощность невелика. Есть модели колонок, режим работы которых (активный или пассивный) выбирается переключателем. Полоса частот и мощность обычных малогабаритных колонок недостаточны для воспроизведения в режиме Hi-Fi (High Fidelity — высокая достоверность звуковоспроизведения). Более качественные системы имеют две колонки для средних и высоких частот и одну (большую) для низких. Для высококачественного воспроизведения лучше использовать внешний стереоусилитель с собственными акустическими системами или стереонаушники.

Наушники или усилитель можно подключать и к аудиовыходу привода CD/ DVD, что позволит прослушивать аудио-CD (но не CD-ROM с файлами .MP3) независимо от наличия звуковой карты. Регулятор уровня этого выхода (диск потенциометра или кнопки) расположен на лицевой панели привода. Там же в ряде моделей приводов имеются кнопки воспроизведения и выбора трека, позволяющие управлять проигрыванием без привлечения каких-либо программных средств.

Существуют звуковые устройства для шин USB и FireWire — колонки, микрофоны и другие приемники и источники сигналов. Они передают аудиопоток в *цифровом виде* (изохронная передача) и к обычным звуковым картам непосредственно не подключаются. Доставка аудиоданных к ним и от них осуществляется программно через контроллер соответствующей шины, имеющийся на современных системных платах. При наличии достаточно мощного процессора такие устройства позволяют обходиться и без звуковой карты, реализуя все перечисленные функции чисто программными средствами. Однако применение звуковой карты расширяет возможности аудиосистемы и снижает загрузку процессора.

Для подключения электромузыкальных инструментов звуковые карты имеют *порт MIDI* (см. 12.3). Устройством ввода могут служить специальные *MIDI- клавиатуры* (как на клавишных музыкальных инструментах), устройством вывода — синтезатор звуковой карты или внешний синтезатор, подключаемый к порту MIDI. Компьютер в такой системе используется как мощное средство создания, редактирования и хранения музыкальных произведений. MIDI-ин- терфейс имеют многие профессиональные и полупрофессиональные клавишные синтезаторы.

Желание «обучить» PC звукозаписи возникло задолго до появления звуковых карт с АЦП/ЦАП и было реализовано даже через динамик AT-286. Для преобразования аналогового сигнала в дискретную форму использовался принцип широтно-импульсной модуляции (ШИМ). Для оцифровки звука умельцы применяли несложную схему, состоящую из микрофонного усилителя-фильтра на операционном усилителе и компаратора. С выхода этой схемы двуполярный дискретный сигнал (1 бит) поступал на один из управляющих входов COM- порта. Программа записи замеряла интервалы времени между переключениями этого бита и записывала эти значения в файл. Воспроизводящая программа

считывала данные из файла и с теми же интервалами переключала бит 1 порта динамика (061h). Динамик выступал в роли фильтра нижних частот этого дискретно-аналогового преобразователя. На машинах с большим динамиком (он лучше фильтровал) удавалось добиваться внятного воспроизведения речи, правда, процессор класса 286 был загружен этим полностью. Заметим, что метод ШИМ недавно снова «всплыл из небытия» в применении к новому способу сверхвысококачественной аудиозаписи на диски SACD (Super-Audio CD).

Звуковоспроизведение «подручными средствами» реализовывалось проще и качественнее — достаточно было к выходному порту данных подключить матрицу резисторов R-2R, чтобы получить простейший 8-битный ЦАП. Такая приставка называлась Covox. Если к порту подсоединить пару регистров-защелок, к каждому из которых подключить матрицу резисторов, то можно организовать и стереофонию. Такими простейшими устройствами можно было пользоваться для озвучивания некоторых игр или улучшения качества звучания музыкальной программы Screem Tracker на PC/AT-286. С появлением звуковых карт нужда в этих устройствах сошла «на нет».

12.1. Краткий экскурс в прикладную звукотехнику

Как известно, слышимые звуки представляют собой механические колебания, достигающие ушей слушателя обычно по воздуху. Диапазон частот, воспринимаемых человеческим ухом, простирается от 20 Гц до 20 кГц, причем наибольшая чувствительность приходится на частоты 2-5 кГц. В этой области ухо воспринимает сигналы в динамическом диапазоне около 140 дБ[1] (отношение звукового давления болевого порога к порогу слышимости 10^7). На краях частотного диапазона динамический диапазон сужается до 50 дБ (чувствительность уха существенно снижается, а давление болевого порога уменьшается). Разговорная речь в спектре занимает область примерно 200 Гц — 4 кГц при динамическом диапазоне около 40 дБ. Музыка может занимать практически весь слышимый диапазон частот и требовать динамического диапазона 70-90 дБ. Важной особенностью слуха является способность к локализации источника звука, обеспечиваемая его бинауральным восприятием. Дело в том, что звуковые волны воспринимаются обоими ушами, которые пространственно разнесены. Колебания от одного источника достигают ушей с разными амплитудой и фазой, что позволяет мозгу оценить направление (азимут) источника звука. Сигналы с частотами ниже 300 Гц локализуются плохо, поскольку длина волны относительно размера головы велика. Наибольшее значение для локализации имеют

[1] Децибел (дБ) — логарифмическая мера измерения мощности *P* относительно условно принятого нулевого уровня P_0, определяется как 10 log(P/Po). Когда речь идет об усилении/затухании напря жения сигнала, используют формулу 20 log*(U/Uo)*. Усилению в 10 раз соответствует +20 дБ, ос лаблению в 2 раза соответствует -6 дБ. В звукотехнике логарифмические шкалы используются «вдоль» — по оси частот и «поперек» — по оси уровней мощности, что перекликается со звуковос-приятием.

частоты от 1 до 3,2 кГц. Бинауральное восприятие позволяет не только локализовывать, но и выделять отдельные источники (например, отдельные инструменты в оркестре).

Для передачи, хранения, воспроизведения и синтеза звуков традиционно акустические колебания преобразуют в электрические (микрофон) и обратно (динамик). Первоначально вся промежуточная обработка (усиление, преобразования) сигналов производилась в аналоговой форме, естественной для оконечных электромеханических преобразователей. Хранение, опять-таки в аналоговой форме, выполнялось на механических (грампластинки) или магнитных (магнитофонные ленты) носителях. Для повышения достоверности звукопередачи, включая пространственное расположение источников звука, применяется двухканальная передача и хранение — стереофония. Упрощенно ее идея заключается в разделении трактов сигналов, предназначенных для левого и правого ушей слушателя. Такая система позволяет создать иллюзию звуковой панорамы — кажущиеся источники звука (КИЗ) располагаются на воображаемой сцене, расположенной перед слушателем. Однако пара колонок не позволяет добиться большой ширины зоны стереоэффекта. Прослушивание через головные телефоны не всегда удобно и тоже не дает полной иллюзии присутствия — поворот головы в сторону КИЗ ведет к его уходу в ту же сторону. В более сложных системах используют большее число каналов: например, 4 в квадрофонии и 6 в системе АС-3. Здесь колонки располагаются вокруг (спереди и сзади) слушателя, что позволяет получить эффект присутствия внутри некоторого озвученного объема. Есть и промежуточные варианты между дорогой квадрофонией и фактически стандартной стереофонией — квази- и псевдоквадрофония.

Аналоговое представление сигналов для обработки (фильтрации, создания различных эффектов) и хранения имеет массу недостатков. Во-первых, все устройства в той или иной степени обладают нелинейными передаточными характеристиками: проходящий через них гармонический (чисто синусоидальный) сигнал «обрастает» гармониками — составляющими с частотами, кратными основной. Мерой искажений, вносимых нелинейностью, является коэффициент гармоник, он же коэффициент нелинейных искажений (КНИ), который определяется как отношение мощности гармоник выходного сигнала к мощности основного тона. Эти искажения вносят все элементы тракта, так что их всюду стремятся минимизировать. Для современных высококачественных усилителей считается хорошим значение КНИ в десятые и сотые доли процента, для электромеханических преобразователей (особенно динамиков) значения гораздо выше.

Следующая беда — шумы и помехи, характерные для любой аналоговой техники. Они сужают динамический диапазон устройства. Отношение сигнал/шум порядка 90-100 дБ для аналоговых устройств удалось получить сравнительно недавно.

Что касается хранения информации, то и здесь аналоговая форма наиболее уязвима — грампластинки «запиливаются», магнитные ленты осыпаются и размагничиваются, в результате ранее записанный сигнал при воспроизведении силь

но искажается. Потери происходят и при тиражировании — каждая перезапись или перепечатка вносит свою долю искажений.

С развитием электроники появилась возможность большую часть «путешествия» электрического сигнала производить в цифровой форме. Теперь входной сигнал (от микрофона) после предварительного усиления оцифровывается. В цифровой форме он может передаваться, храниться (долго и без накопления ошибок), подвергаться различным искусственным преобразованиям. При воспроизведении выполняются обратное преобразование в аналоговую форму, оконечное усиление и преобразование в акустические колебания. Для цифрового хранения акустической информации стали применять лазерные компакт-диски (Audio-CD) и магнитные ленты для цифровой звукозаписи (Digital Audio Tape, DAT), которые долгое время считались эталонами качества. По мере развития средств вычислительной техники возможности обычных PC доросли до того, чтобы пропускать через себя поток цифровых аудиоданных (или создавать собственный).

Оцифровка звуковых сигналов

Для оцифровки аналогового сигнала применяются дискретизация по времени и квантование по уровню. Это означает, что регулярно (с постоянным периодом) производятся *выборки* (samples) мгновенных значений аналогового сигнала (рис. 12.2). Эти выборки *квантуются* при помощи аналого-цифрового преобразователя (Analog-to-Digital Converter, ADC), или *АЦП*. На выходе АЦП информация представляется в виде двоичного кода — то есть числа, которое может принимать одно из множества дискретных значений, определяемых разрядностью преобразователя. Очевидно, чем выше разрядность, тем точнее это число может представлять мгновенное значение аналогового сигнала. «Может» потому, что для точности характеристика преобразователя должна быть еще монотонной и линейной. В идеале передаточная характеристика преобразователя выглядит ровной «лесенкой» с одинаковыми ступеньками (линейность) и без провалов (монотонность). Поскольку мгновенные значения сигнала «не обязаны» попадать на ступеньки этой лесенки, при преобразовании возникают *шумы квантования* — отклонения квантованного значения от реального, в среднем половина кванта. Для высококачественной передачи музыки разрядность преобразователя должна составлять по крайней мере 16 бит — что мы и имеем в лазерных компакт-дисках, на качество которых будем ориентироваться для определенности.

Выбор частоты дискретизации определяется *теоремой Котельникова*. для адекватного восстановления частота дискретизации должна быть больше (лучше — с запасом) удвоенной частоты высших спектральных составляющих входного сигнала. Чтобы не интересующие нас более высокие частоты не искажали оцифровку, они должны быть тщательно отфильтрованы. В том же компакт- диске частота 44,1 кГц позволяет воспроизводить сигнал в полосе до 20 кГц — весь слышимый спектр.

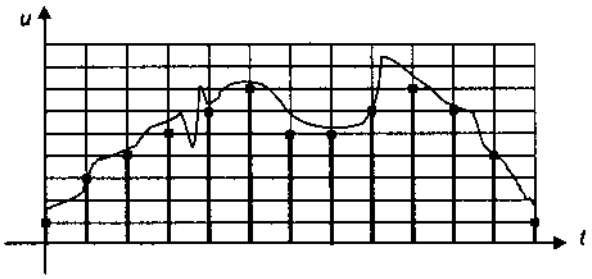


Рис. 12.2. «Классическая» оцифровка аналогового сигнала

Обратное преобразование выполняется с помощью цифроаналогового преобразователя (Digital-to-Analog Converted, DAC), или *ЦАП,* на вход которого поступает цифровой поток с той же частотой. Аналоговый сигнал после ЦАП должен быть опять-таки отфильтрован — частоты выше половины частоты квантования подавляются. К устройству ЦАП предъявляют те же требования по разрядности, линейности и монотонности. Разрядности АЦП и ЦАП могут и не совпадать — эффективная разрядность тракта будет определяться наименьшим значением (включая разрядность находящегося между ними цифрового канала передачи или хранения информации). Заметим, что достаточно быстродействующий (около 20 мкс/преобразование) 16-разрядный АЦП стал широко доступным только в конце 80-х годов. ЦАП реализуется проще, поэтому проигрыватели CD получили массовое распространение довольно давно (поначалу используя более доступные 14-битные устройства ЦАП).

Как указывалась выше, частота дискретизации и разрядность квантования определяются требованием к полосе пропускания и динамическому диапазону тракта при заданном отношении сигнал/шум. Простейший способ цифрового представления сигналов называется *импульсно-кодовой модуляцией* (Pulse-Code Modulation, PCM), или *ИКМ.* Поток данных PCM представляет собой последовательность мгновенных значений, или выборок, в двоичном коде. Если применяемые преобразователи имеют линейную характеристику (мгновенное значение напряжения сигнала пропорционально коду), то данная модуляция называется *линейной* (Linear PCM, LPCM). В случае PCM кодер и декодер не выполняют преобразования информации, а только занимаются упаковкой/распаковкой бит в байты и слова данных. Интенсивность потока (bit rate) определяется как произведение частоты дискретизации (sample rate) на разрядность и на число каналов. Аудио-CD дает поток 44100 x 16 x 2 = 1411200 бит/с (стерео). При этом обеспечиваются диапазон воспроизводимых частот 5-20 000 Гц и динамический диапазон 96 дБ. Ленточные цифровые накопители (DAT) работают с частотами дискретизации 32, 44,1 или 48 кГц и разрядностью 16 бит. Соответственно, потоки данных — 1 024 000, 1 411 200 или 1 536 000 бит/с (стерео).

Если такой поток покажется слишком интенсивным, можно «смирить гордыню» — понизить частоту и разрядность квантования. Очевидно, что с понижением частоты дискретизации пропорционально снизится и доступная полоса

частот. Снижение разрядности приведет к повышению погрешности — уровня шумов квантования. Каждый отброшенный двоичный разряд повысит уровень этого шума на 6 дБ. Если нас интересует только разборчивая передача речи, можно «опуститься» до 8-битного преобразования с частотой 5 кГц — в моно это даст поток около 5 Кбайт/с. В телефонной связи используется 7-битные преобразования с частотой 8 кГц — поток 56 Кбит/с.

Для PCM требуется подавление частот, превышающих половину частоты дискретизации, иначе появятся ложные частоты (aliases). Чтобы сохранить широкую полосу пропускания (до 20 кГц) при частоте дискретизации 44,1 кГц перед АЦП (и после ЦАП) требуются фильтры (аналоговые!) с большой крутизной характеристики (что не так-то просто обеспечить). Повышение частоты дискретизации, используемое в современных цифровых системах (48, 96 и даже 192 кГц), позволяет расширить диапазон воспроизводимых частот и упростить фильтры (но не АЦП!).

Для дисков SACD (Super-Audio CD), предложенных фирмами Sony и Philips на смену традиционным аудио-CD, решили вернуться к однобитному преобразованию. Фирмы предложили метод кодирования, названный *DSD* (Direct Stream Digital encoding), позволяющий обойти ряд проблем кодирования PCM. Здесь используется так называемый *дельта-сигма-АЦП* (рис. 12.3, а), состоящий из квантизатора Q (компаратора) и фильтра-интегратора F, охваченных отрицательной обратной связью. Если уровень входного сигнала, аккумулированный за период дискретизации, превышает значение в цепи обратной связи, накопленное за тот же период, то формируется единица. Если значение входного сигнала падает ниже, формируется ноль. Максимальное положительное значение представляется сплошным потоком единиц, максимальное отрицательное — потоком нулей, нулевой входной уровень — чередованием нулей и единиц. Такое цифровое представление можно назвать *плотностно-импульсной модуляцией* (Pulse Density Modulation, PDM). Декодировать такой сигнал просто: достаточно пропустить единичные импульсы через интегрирующую цепочку (рис. 12.3, б), и получится отображение исходного сигнала. Конечно, для того чтобы восстановленный сигнал повторял динамику исходного, должна быть высокая частота дискретизации. В SACD используется частота 2,8224 МГц, то есть битовый поток имеет скорость чуть больше 2,8 Мбит/с на канал. Это в 4 раза больше, чем в CD/DA (705600 бит/с при 41,1кГцх16 бит). Такой формат обеспечивает широкую полосу пропускания (0-100 кГц) и широкий динамический диапазон (120 дБ).

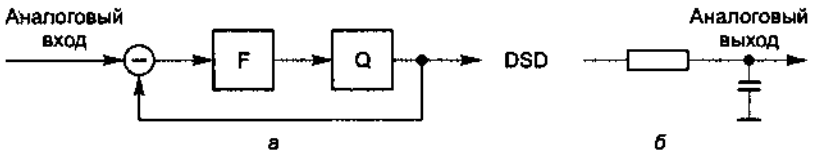


Рис. 12.3. DSD-кодирование: а — кодер, б — декодер

Метод PDM по сравнению с PCM имеет ряд преимуществ. Для PDM не требуется сложный фильтр нижних частот перед АЦП (и после ЦАП) — фильтра-

цию обеспечивает интегратор преобразователя. Поток DSD гораздо лучше передает динамику быстро меняющегося сигнала. Так, меандр (прямоугольник) с частотой 10 кГц, записанный на аудио-CD (полоса пропускания 20 кГц), при воспроизведении выглядит как синусоида. Тот же сигнал, прошедший через тракт DSD, похож на исходный гораздо больше. Поток DSD малочувствителен к битовым ошибкам в тракте передачи или хранения: влияние искаженного бита (и даже их группы) весьма незначительно. В потоке PCM искажение старших битов приводит к значительному искажению отсчета (в DSD все биты «младшие»). Цифровой сигнал (1 бит) потока DSD наглядно отражает передаваемую информацию (рис. 12.4), чего не скажешь о PCM (при осциллографировании поток PCM безлик).

Рис. 12.4. Плотностно-импульсная модуляция

Из потока DSD сравнительно несложно получить традиционные потоки PCM с разными частотами дискретизации. Так, для формата CD/DA (44,1 кГц) требуется из каждых 64 бит (естественно, с учетом предыстории) потока формировать один 16-битный отсчет (пересчет 1/64). Частоты 32, 48 и 96 кГц получаются предварительным умножением частоты потока на 5 («размножением» битов), а затем пересчетами 1/441, 1/294 и 1/147 соответственно. Таким образом обеспечивается сосуществование старых и нового форматов звукозаписи.

Использование ПК для обработки «цифрового» звука

Теперь обсудим, что можно делать с цифровыми потоками полезного и на что способен ПК.

Первое, что приходит в голову, — *запись и воспроизведение.* Аппаратные средства цифровой записи и воспроизведения для PC достаточно просты — АЦП, ЦАП и средства доставки потока данных на диск или с диска. Этими средствами, называемыми *аудиокодеком* (кодер-декодер), обладает любая звуковая карта. В качестве «транспорта» для карт ISA обычно используется канал прямого доступа к памяти контроллера DMA 8237A. На современных картах PCI доставку осуществляет ее собственный контроллер путем прямого управления шиной. Канал DMA справляется с любым аудиопотоком, вопрос лишь в размещении данных — звукозапись с высоким качеством довольно расточительно расходует дисковое пространство. Нетрудно сосчитать, что на дискету 1,44 Мбайт можно уместить около 5 минут речи с низким качеством. А 1 минута стереомузыки с качеством CD занимает около 10 Мбайт (на CD умещается до 74 минут), так что есть повод задуматься о компрессии.

Цифровое хранение обеспечивает богатейшие возможности *нелинейного монтажа аудиозаписей*. Под нелинейностью подразумевается возможность произвольного доступа к любым фрагментам записи. Даже простейшие средства «фонографа» из стандартных утилит Windows позволяют вырезать или вставить любой фрагмент записи, состыковать (один за другим) фрагменты или наложить один на другой. Программная реализация монтажа является «бескровной» и позволяет легко отказаться от проведенных действий. После такого монтажа работа с магнитной лентой, ножницами и скотчем, которой не было альтернативы еще в 80-е годы, кажется кошмаром (а ведь делали!). Поиск начала и конца требуемого фрагмента делается теперь с помощью точного указателя-курсора, а не на слух с протягиванием ленты туда-сюда по магнитной головке вручную.

При цифровом хранении легко реализуются многие э*ффекты*, которые ранее требовали громоздких электромеханических или электроакустических устройств или сложной аналоговой электроники. Прежде всего это *искусственная реверберация* и э*хо*. Известно, что в закрытом помещении (например, зале) от источника до слушателя доходит не только прямой, но и многократно отраженный от различных поверхностей (стен, колонн и т. п.) звук. Отраженные сигналы приходят с различными задержками и затуханием относительно прямого. Это явление называется реверберацией. Акустика зала (реверберационные характеристики) должна соответствовать его назначению — в помещении с малой реверберацией музыка будет звучать «сухо», а в помещении с длительной реверберацией речь окажется неразборчивой. В студиях звукозаписи естественную реверберацию часто подавляют, а в запись добавляют искусственную реверберацию «по вкусу». В доцифровую эпоху использовали реверберераторы на электромеханических линиях задержки, акустические камеры с микрофонами или магнитофонные реверберераторы. Все это было дорого, громоздко, некачественно и трудноуправляемо. При цифровой записи или воспроизведении реверберация достается практически даром. Для этого достаточно на вход ЦАП подавать не просто очередную выборку, а ее сумму с одной или несколькими предыдущими выборками, которые еще присутствуют в памяти. Конечно, эти «отстающие» выборки должны быть снижены по уровню — в цифровой форме эти операции выполнить нетрудно. Задержку (отставание выборки) и уровень каждого эхо-сигнала легко задать программно, количество составляющих ограничивается только производительностью процессора, обрабатывающего эти сигналы. Таким образом, появляется возможность имитации воспроизведения в каком-либо знаменитом зале — для этого достаточно снять его характеристики и заложить их в программу (конечно, на практике это не так просто, но возможно). *Искусственное эхо* технически отличается от реверберации лишь большим временем задержки, так что теперь и с ним нет проблем.

На основе смещения выборок можно делать и более сложные эффекты. В цифровой форме представления легко имитируется эффект Доплера — изменение частоты при быстром приближении источника звука к слушателю или удалении источника от слушателя. С этим эффектом сталкивались все: однотонный свисток приближающегося поезда звучит выше, а удаляющегося — ниже реального тона. В цифровом виде при воспроизведении накопление отставания вы

борок приведет к понижению тона, а сокращение отставания — к повышению. Конечно, как и невесомость в самолете, этот эффект действует недолго — до переполнения или опустошения буфера выборок. Но если отставание «покачивать», возникает эффект частотной модуляции. Управляя «качанием» одной или нескольких выборок, можно получать такие эффекты, как «расстроенный рояль» (плавное покачивание одной-двух выборок), «хорус» (псевдослучайные колебания нескольких выборок), «фэйзинг» и некоторые другие.

Помимо фокусов с задержками, возможна цифровая фильтрация — от реализации простейших темброблоков и эквалайзеров, до «вырезания» голоса из песни (эффект «караоке»). Все упирается в фантазию программиста и вычислительные ресурсы процессора. Раньше обработка сигналов в реальном масштабе времени была уделом лишь специализированных сигнальных процессоров (например, серии TMS). В них предусмотрены специальные средства для таких функций, как, например, быстрое преобразование Фурье. Теперь же эта работа под силу и «обычным» процессорам Pentium, не говоря уже о MMX и Pentium II, III и 4. Специализированные *сигнальные процессоры* (Data Signaling Processor, DSP) входят в состав «продвинутых» звуковых карт. Как и графические акселераторы, они позволяют выполнять действия по обработке сигналов, не загружая центральный процессор. Существует масса программных продуктов, реализующих сложные эффекты и на центральном процессоре PC.

Методы компрессии звуковой информации

В задачу компрессоров/декомпрессоров входит сокращение потока в канале передачи (хранения) относительно потока на выходе АЦП и входе ЦАП. Для реальных звуковых сигналов кодирование с линейной ИКМ (LPCM) является неэкономичным. Поток данных можно сократить, если использовать несложный алгоритм сжатия, применяемый в системе *дельта-ИКМ* (ДИКМ), она же *DPCM* (Differential Pulse-Code Modulation). Упрощенно этот алгоритм выглядит так: в цифровом потоке передаются не сами мгновенные отсчеты, а масштабированная разность реального отсчета и его значения, сконструированного кодеком по ранее сгенерированному им потоку данных. Разность передается с меньшим числом разрядов, чем сами отсчеты. В *АДИКМ* (адаптивная ДИКМ), или *ADPCM* (Adaptive Differential Pulse-Code Modulation), масштаб разности определяется по предыстории — если разность монотонно растет, масштаб увеличивается, и наоборот. Конечно, восстановленный сигнал при таком представлении больше отличается от исходного, чем при обычной ИКМ, но можно добиться существенного сокращения потока цифровых данных. АДИКМ стала широко применяться при цифровом хранении (CD-XA) и передаче аудиоинформации (например, в голосовых модемах). Алгоритм АДИКМ с точки зрения процессора PC может быть реализован как программно, так и аппаратно средствами звуковой карты (модема). Заметим, что на стандартных аудио-CD и DAT компрессия не применяется.

Базовое ПО, обеспечивающее цифровую звукозапись в Windows (приложение «Фонограф»), позволяет выбирать частоту и разрядность преобразования, количество каналов (моно/стерео), а также формат данных (PCM, ADPCM). Для

хранения звука используются файлы с расширением .WAV, заголовки этих файлов содержат информацию о частоте и разрядности квантования, количестве каналов (моно/стерео) и формате записи (методе компрессии).

Более сложные алгоритмы и высокая степень сжатия применяются в аудиокодеках MPEG. В *кодере MPEG-1* входным потоком являются 16-битные выборки с частотой 48 кГц (профессиональная аудиотехника), 44,1 кГц (бытовая техника) или 32 кГц (применяется в телекоммуникациях). Стандарт определяет три «слоя» (layer) сжатия — Layer 1, Layer 2 и Layer 3, работающие один поверх другого. Первоначальная компрессия осуществляется на основе психофизических свойств звуковосприятия. Здесь обыгрывается свойство маскирования звуков: если в сигнале имеются два тона с близкими частотами, существенно различающиеся по уровню, то более мощный сигнал замаскирует слабый (он не будет услышан). Пороги маскирования зависят от удаленности частот. В MPEG весь диапазон звуковых частот разбивается на 32 поддиапазона (sub-band), в каждом поддиапазоне определяются наиболее мощные спектральные составляющие и для них вычисляются пороги частот маскирования. Эффекты маскирования от нескольких мощных составляющих суммируются. Действие маскирования распространяется не только на сигналы, присутствующие одновременно с мощным, но и на предшествующие ему за 2-5 мс (premasking) и последующие в течение до 100 мс (postmasking). Сигналы маскированных областей обрабатываются с меньшим разрешением, поскольку для них снижаются требования к отношению сигнал/шум. За счет этого «загрубления» и происходит сжатие. Компрессию на психофизической основе выполняет слой Layer 1. Следующий этап (Layer 2) повышает точность представления и более эффективно упаковывает информацию. Здесь у кодера в работе находится «окно» длительностью 23 мс (1152 выборки). На последнем этапе (Layer 3) применяются сложные наборы фильтров и нелинейное квантование.

Дополнительные возможности компрессии обусловлены высокой коррелированностью сигналов между парой каналов в стереотракте. С точки зрения поддержки стереофонии, поток может представлять один канал (Mono), два независимых канала, пару обычных стереоканалов (L/R Stereo) или стерео в суммарно-разностном представлении (M/S-Stereo). В последнем варианте, называемом также *joint stereo*, или *intensity stereo*, один канал несет суммарный сигнал обоих стереоканалов, а другой — их разность. Разделение стереоканалов выполняется на выходе декодера. Дополнительная компрессия достигается за счет ограничения (снизу) полосы пропускания разностного канала.

Стандарт определяет ряд фиксированных значений интенсивности потока в канале от 32 до 448 Кбит/с (Layer 1), до 384 Кбит/с (Layer 2) и 320 Кбит/с (Layer 3). Наибольшую степень сжатия обеспечивает слой Layer 3, для которого при высокой достоверности декодирования достигается коэффициент сжатия 11:1. Так, для обеспечения стереофонии с качеством, близким к аудио-CD, достаточно потока 128 Кбит/с (Layer 2 — 160 Кбит/с, Layer 1 — 288 Кбит/с). Layer

1 используется для мини-дисков Sony и цифровых компакт-кассет Philips (384 Кбит/с), Layer 2 — для спутникового вещания и видео-CD (224 Кбит/с), Layer 3 — для потоковой передачи в Интернете и сетях ISDN.

Чем выше уровень компрессии, тем больших вычислительных ресурсов требует кодек. К счастью, ресурсоемкость несимметрична — декодирование значительно проще кодирования. Аудиодекодер может встраиваться в графические карты с MPEG-декодером, при этом графическая карта снабжается выходным разъемом аудиосигнала. В настоящее время стали популярными звукозаписи в формате MPEG-1 Layer 3 (файлы с расширением .MP3), которые могут быть декодированы на современных компьютерах программным способом с выводом сигнала как на ЦАП любой звуковой карты, так и в WAV-файлы. Часто по расширению (или для краткости) их ошибочно называют файлами MPEG-3 (такого стандарта нет!). Файлы с расширением .MP1 и .MP2 представляют данные в формате MPEG-1 Layer 1 и 2 соответственно, но они не так широко распространены. Для декодирования высококачественной стереофонии в реальном времени пригоден любой современный процессор (как минимум 486DX4-100). Процесс кодирования (на Layer 3) в реальном времени обеспечивают специализированные сигнальные процессоры, а также процессоры семейства x86 с тактовой частотой от 500 МГц. На PC компрессии в формат MP3 поддаются WAV- файлы, записанные с разрядностью и частотой, принятыми в MPEG, или же цифровые данные с аудиодисков. Время компрессии на менее мощном процессоре в несколько раз превышает время звучания файла.

В *MPEG*-2 по сравнению с MPEG-1 имеется ряд дополнений. Помимо частот 48, 44,1 и 32 кГц здесь определены частоты дискретизации 16, 22,05 и 24 КГц. Аудиопоток может содержать две пары широкополосных каналов (фронт и тыл), а также один низкочастотный (до 100 Гц). Разрядность входного и выходного потоков может достигать 18 и даже 24 битов. Формат *MPEG2 Layer 3* (обозначается как MP3) стал фактическим стандартом для аудиозаписей, распространяемых в виде файлов (с расширением .MP3) на любых носителях (CD, флэш-карты и т. п.).

Конечно же, за экономию памяти (пропускной способности канала), обеспечиваемую компрессией, приходится расплачиваться потерей достоверности звуковоспроизведения (Hi-Fi). Побочные эффекты «психофизической компрессии» на различных музыкальных фрагментах проявляются по-разному. Искажения при воспроизведении поп-музыки менее заметны, чем для симфонической или фортепианной при тех же параметрах компрессии.

Методы синтеза звуков

Синтезаторы звуков в наше время стали уже привычными инструментами. Их используют как для имитации голосов «естественных» музыкальных инструментов, человеческого голоса, различных шумов, так и для создания оригинальных звуков. Прежде чем рассматривать проблемы и методы синтеза, займемся «препарированием» звука.

Звуки можно разделить на *шумовые* и *тональные* (мелодические). Из теоретических основ электротехники известно, что любой сигнал можно представить в виде ряда гармонических (синусоидальных) составляющих (ряда Фурье), каждая из которых характеризуется своими частотой, амплитудой и фазой. Шумовые звуки имеют спектр, непрерывный в какой-то области. Спектр тонального

звука — дискретный, с *основным тоном* и *гармониками*, частота которых кратна частоте основного тона (первая гармоника является основным тоном). Музыкальный звукоряд представляет собой ряд последовательных *нот*, отличающихся друг от друга частотами основного тона. Ноты, частоты которых отличаются друг от друга в 2 раза, отстоят друг от друга на одну *октаву* («центр» — нота «ля» первой октавы — 440 Гц). В пределах каждой октавы «европейский» звукоряд насчитывает 12 *полутонов* (7 основных нот со знаками альтерации — диезами и бемолями). Частоты соседних полутонов отличаются друг от друга в $\sqrt[12]{2}$ раз. Для более тонкой идентификации тона имеется и единица измерения *цент* — одна сотая (по логарифмической шкале) от полутона.

Сигнал с непрерывным равномерным спектром в широком диапазоне частот называют «белым шумом» (он может охватывать весь слышимый диапазон частот). Поскольку суммарная мощность любого звука конечна, отдельные составляющие белого шума имеют бесконечно малую амплитуду. Если из белого шума выделить узкую спектральную полоску, то звук получит тональную «окраску». Если ширина полосы будет уже, чем расстояние до соседней ноты звукоряда, звук приблизится к мелодическому. Звуки реальных инструментов являются смесью мелодических и шумовых (характерный пример — «придыхающее» звучание саксофона).

Анализ осциллограмм музыкальных звуков позволил построить их обобщенную модель (рис. 12.5). Здесь видны несущая частота, обогащенная гармониками, и ее огибающая. Звук имеет четыре явно выраженные фазы:

- атака (attack) — бурный рост амплитуды несущей, сопровождающийся значительными изменениями (обогащением) ее спектрального состава;
- спад (decay) — процесс, сопровождающийся «смягчением» спектра;
- удержание (sustain) — относительно стационарный, постепенно затухающий процесс (например, удержание нажатой клавиши фортепиано);
- затухание (release) — довольно быстрое уменьшение амплитуды до нуля (демпфирование колебаний при отпускании клавиши).

Рис. 12.5. Типовая осциллограмма звука фортепиано

По первым буквам английских названий фаз такая модель называется ADSR. Для каждого инструмента характерен свой набор параметров, описывающих

эти фазы. Для инструментов с широким диапазоном звучания значения параметров заметно различаются для разных участков частотного диапазона.

Небольшие периодические колебания частоты тона называют *вибрато* (слово «частотное» подразумевается). Модуляция амплитуды называется *тремоло*, или амплитудным вибрато. Если при переходе от ноты к ноте новая частота основного тона не устанавливается скачком, а плавно «подъезжает», такое исполнение называется *портаменто***.** Смещение всего строя в процессе игры (с сохранением интервалов в аккордах) называется *глиссандо*.

Акустическая система любого естественного инструмента имеет свой набор *формант* — областей резонанса, где амплитудно-частотная характеристика имеет подъем. Форманты придают инструментам характерную узнаваемость. В человеческом голосе форманты позволяют, например, различать гласные звуки — каждой гласной соответствует определенная пара формант.

Теперь, после этого беглого «осмотра» звуков, поговорим о синтезе.
Электронным синтезом звуков начали заниматься еще в 1920-е годы Первым синтезатором был *терменвокс,* созданный в России Львом Терменом. В этом инструменте использовались высокочастотные генераторы; оператор управлял частотой одного из генераторов, меняя положение своей руки относительно специального электрода. Выходная (звуковая) частота выделялась как разность частот пары генераторов. Любопытно, что полупроводниковые (а теперь и цифровые) «версии» терменвокса иногда используются и в наши дни. На 1960-80-е годы пришлось бурное развитие аналоговых методов синтеза, в 1990-е годы в основном развивались цифровые и гибридные (аналоговые с цифровым управлением). Введем несколько определений, относящихся к возможностям синтезаторов.

Одноголосный, или *монофонический* (monophonic), синтезатор в каждый момент времени способен воспроизводить только один звук (ноту). При попытке исполнить несколько нот (взять аккорд) будет звучать лишь одна из них. *Многоголосный*, или *полифонический* (polyphonic), синтезатор способен одновременно исполнить несколько нот (не более, чем число его голосов). *Многотембровый* (multitimbral) синтезатор может одновременно издавать звуки с различными тембрами (имитировать несколько разных инструментов).

Сигнал со сложным спектральным составом можно получать самыми разнообразными способами. Если ограничиваться небольшим числом составляющих, то можно воспользоваться *аддитивным методом* синтеза. Его суть очевидна из названия (addition — сложение): сигналы от нескольких управляемых генераторов суммируются. Частоты генераторов могут находиться в гармоническом (быть кратными одной из частот) или ином соотношении. Однако спектрально богатый звук при таком подходе требует применения большого числа согласованно управляемых генераторов, что трудно реализуемо по множеству технических причин. Противоположностью аддитивному является *субтрактивный метод* (subtraction — вычитание). Здесь из шумового (или другого спектрально богатого) сигнала выделяются только нужные области — вспоминается известное объяснение процесса создания скульптуры: взять глыбу и отсечь все лиш

нее. Реализация управляемых фильтров, выполняющих это «художественное отсечение», затруднительна, особенно в аналоговом виде. На практике эти два базисных метода применяются в сочетании с рядом других.

Богатые возможности синтеза предоставляли *модульные синтезаторы*, среди которых наиболее известны Moog-синтезаторы (названы по фамилии создателя). Модули этих синтезаторов представляли собой различные устройства, управляемые напряжением: генераторы (Volt Controlled Oscillator, VCO), фильтры (Volt Controlled Filter, VCF), усилители (Volt Controlled Amplifier, VCA) и генераторы управляющих сигналов различных форм. Для генерации сигналов произвольной формы использовали *секвенсоры* — наборы потенциометров и коммутирующих ключей. Потенциометрами форма «набиралась» по точкам (в Moog использовалось 12 потенциометров), ключи обеспечивали «развертку» этого набора во времени (циклически или однократно по команде). Генераторы и фильтры имели логарифмическую характеристику управления (Roland, Moog) с чувствительностью 1 октава на вольт или линейную (синтезаторы Yamaha, Korg, где частота в герцах пропорциональна напряжению). Клавиатура (и другие управляющие устройства) вырабатывала напряжение с уровнем, определяемым нажатой клавишей. Модули соединялись между собой шнурами (patch), и звук задавался определенной комбинацией этих соединений. С тех пор слово *patch* применительно к синтезаторам означает определенный (загружаемый в память) тип звука (инструмента). Конечно, модульные синтезаторы были дорогими и малопригодными для исполнения произведений в реальном времени. Позже появились синтезаторы с фиксированными соединениями тех же узлов (например, Mini-Moog), с которыми уже могли справляться музыканты-исполнители.

Идеи модульных синтезаторов легли в основу *FM-синтезаторов,* получивших широкое распространение и в простых звуковых картах для PC. Синтез FM построен на модуляции частоты одного звукового генератора (несущей) сигналом от другого звукового генератора. Здесь есть отличия от модуляции, применяемой в радиотехнике, где низкочастотный (звуковой) сигнал управляет частотой высокочастотного, которая выше на несколько порядков, вызывая малое относительное отклонение этой несущей. В FM-синтезаторах частоты соизмеримы, и частота несущей может быть даже ниже модулирующей, глубина модуляции высока. В таком приложении модуляция позволяет из пары гармонических сигналов получить сигнал с богатым набором спектральных составляющих, частоты которых определяются через суммы и разности частот исходных сигналов. Пара управляемых генераторов, имеющих и средства формирования огибающей их колебаний (фазы атаки, спада, удержания и затухания), называется *оператором*. В формировании одного звука (голоса инструмента) может быть задействовано несколько операторов, их можно собирать в цепочки и кольца (в зависимости от сложности звука). Все компоненты синтезатора имеют цифровое управление через набор регистров, доступный управляющей программе. В процессе исполнения программа динамически распределяет имеющиеся ресурсы (операторы). Количество операторов определяет полифонические и многотембровые возможности синтезатора, при формировании сложных тембров

полифонические возможности сужаются. FM-синтезаторы звуковых карт хороши для создания необычных («компьютерных») звуков, но их возможности в воспроизведении естественных звуков весьма скудны. Существуют и очень сложные FM-синтезаторы с богатыми возможностями, но в картах для PC они не применяются.

Для имитации звуков естественных инструментов больше подходит метод синтеза, основанный на воспроизведении предварительно записанных звуковых выборок (образцов звука). Этот метод используется в *WT-синтезаторах,* которые поначалу применялись лишь в относительно дорогих моделях звуковых карт. Такие синтезаторы имеют память, в которой хранятся *волновые таблицы* (WT) — оцифрованные образцы звуков. Для извлечения звука процессор синтезатора извлекает из памяти подходящий образец и воспроизводит его с требуемыми параметрами. Ограниченный объем памяти заставляет использовать различные ухищрения, направленные на ее экономию. Во-первых, можно хранить образцы не всех нот, доступных для синтезируемого инструмента, а только некоторых из них, распределенных по его диапазону, а промежуточные ноты вычислять по образцам ближайших к ним нот. При этом процессору приходится определять точки воспроизводимого сигнала, попадающие между выборками образцов. Для этого используются различные алгоритмы интерполяции, от сложности которых зависит «правильность» вычисленной волны. Во-вторых, можно хранить образец для звука с минимальной длительностью, при которой фазы ADSR различимы. При воспроизведении в фазе удержания зацикливается фрагмент, кратный периоду основного тона (указатели на его границы хранятся вместе с образцом). При этом процессор должен обеспечить «гладкость» стыковки, а фрагмент для повтора должен быть тщательно подготовлен, чтобы не было слышно периодических «всхлипов» на «швах». Как всегда, чем выше требования к качеству звука, тем больше требуется ресурсов — объема памяти выборок и мощности внутреннего процессора синтезатора. Достоверность звуков получается высокой, если один образец «обслуживает» лишь несколько смежных нот. Хорошо бы хранить и несколько образцов разной силы звука, поскольку у многих инструментов она существенно влияет на характер сигнала. Для придания звуку естественности вычисленные волны пропускают через управляемые фильтры. Оживляет звучание и введение случайных составляющих в алгоритмы вычислений и обработки.

Платы аппаратных волновых синтезаторов имеют постоянную память (ROM) для хранения голосов основных инструментов и оперативную для загрузки произвольного набора голосов, включая оригинальные звуки, созданные пользователем. По старинке они называются «патчами» (patches). Волновой синтезатор для PC может быть не только выполнен в виде самостоятельной карты, но и поставляться в качестве средства расширения карты с FM-синтезатором. Мощность современных процессоров позволяет выполнять волновой синтез программно.

Методы синтеза звуков не исчерпываются перечисленными. В настоящее время развивается новый подход к синтезу — математическое моделирование физических процессов, происходящих в реальных инструментах. Конечно, для реше

ния этой задачи в реальном времени требуются мощные вычислительные ресурсы, предоставляемые современными процессорами.

Описанные методы применимы к синтезу как тональных, так и шумовых звуков (например, звуков ударных инструментов). Конечно же, здесь есть масса нюансов, которые выходят за рамки данного обсуждения.

Стереофоническое и объемное воспроизведение

Для обычной стереофонии достаточно двух колонок, расположенных перед слушателем, и подавляющее большинство звуковых карт имеют стереофонический аудиовыход. Некоторыми ухищрениями перекрестного смешивания сигналов удалось расширить зону стереоэффекта, но добиться объемности звучания таким путем не удавалось.

В *системе объемного*, или *обволакивающего* (surround), звучания Dolby Surround Pro Logic, применяемой в «домашнем кинотеатре» с аналоговой записью звука, используются 4 воспроизводящих канала усилителей. Здесь слушателя окружают колонками со всех сторон: перед ним располагают три колонки (слева, справа и по центру), а за ним еще две тыловые (слева и справа). Для каждой из фронтальных колонок задействуется собственный широкополосный канал, а обе тыловые колонки используют сигнал одного канала с ограниченной полосой пропускания (100-7000 Гц). Все 4 канала «упакованы» в стереосигнал с обычными параметрами каналов. Этот стереосигнал может храниться и передаваться по любому стереотракту — компакт-дисков, радио FM, стерео в телевидении и видеокассетах. На обычных моно- и стереосистемах этот сигнал воспроизводится естественным для них способом, но с помощью специального декодера Dolby Surround Pro Logic он раскладывается на вышеуказанные 4 канала.

Для цифровых систем фирма Dolby разработала систему Dolby Digital, она же AC-3 и 5.1, в которой передается (хранится) в сжатом виде информация шести каналов — пяти широкополосных и одного низкочастотного. Здесь слушателя также окружают колонками со всех сторон: три колонки спереди (слева, справа и по центру), две тыловые (слева и справа) и еще одна колонка, низкочастотная, называемая *сабвуфером* (subwoofer), располагается за спиной (хотя ее положение относительно произвольно). Самая мощная колонка в этой системе — сабвуфер; центральная колонка несколько мощнее (и «широкополоснее») боковых. Тыловые колонки (rear speakers) могут иметь сравнительно небольшую мощность и размеры. В отличие от предыдущей системы здесь сигнал для каждой колонки передается собственным каналом, и не в аналоговом, а в цифровом виде. Как и положено фирме Dolby, в системе приняты эффективные способы шумоподавления. Эта полная схема поддерживается цифровым форматом и называется «5.1». Такая система воспроизведения устанавливается в современных кинотеатрах с «цифровым» звуком.

Название AC-3 означает «аудиокодек-3». Кодер Dolby Digital упаковывает 5 каналов с полосой 20-20 000 Гц (±0,5 дБ, завал -3 дБ на частотах 3 и 20 300 Гц)

и 1 канал с полосой 20-120 Гц. Входные сигналы могут иметь разрядность 20 бит и более и частоту дискретизации 32, 44,1 или 48 кГц. В зависимости от требований к качеству и числу каналов поток данных на выходе кодера имеет скорость от 32 Кбит/с (моно) до 640 Кбит/с. Характерны скорости 384 Кбит/с для полной схемы «5.1» бытового формата и 192 Кбит/с для стереофонической передачи звука. Кодирование 5.1 включено в стандарт MPEG-2 и используется для записи звукового сопровождения DVD-дисков (и супервидео-CD). Выход на 6-канальную акустику (аналоговый или даже цифровой) имеют «продвинутые» модели современных звуковых карт для PC. Аппаратный декодер AC-3, встроенный в звуковую карту, позволяет разгрузить центральный процессор при воспроизведении DVD (или иных источников аудиопотока MPEG-2).

Следующий шаг — системы 7.1, в которых к фронтальному ряду колонок добавляются промежуточные колонки слева и справа (по фронту теперь 5 колонок в ряд).

Промежуточный вариант объемности звука — квадрофония (пара колонок спереди, пара — сзади). Однако все эти варианты — «шумные», и при озвучивании игр их оценка окружающими, не увлеченными игрой, вряд ли будет адекватной техническим достижениям. Остается еще вариант — наушники, в которых опять- таки всего пара излучателей.

Трехмерный звук

В компьютерной аудиотехнике, как и в графике, мощным двигателем прогресса являются игры. Первые игры довольствовались заранее запрограммированными звуками, издаваемыми программно-управляемым динамиком. Потом игры стали озвучиваться звуковыми картами, воспроизводящими цифровые записи и синтезирующими звук (в основном FM). Параметрами звуков уже стали управлять динамически, в зависимости от сюжета. Здесь уже появилась стереофония с ее возможностью панорамирования источников звуков. Но с ростом возможностей цифровой обработки растут и потребности, в результате чего появилось несколько разработок трехмерного звука (3D Sound, или 3D Audio). Их целью является создание у слушателя впечатления локализации источников в окружающем его трехмерном пространстве — не просто линейной стерео-панорамы, а по трем координатам: лево-право, перед-зад, верх-низ.

Разработчики 3D-звука исходили из того, что сферическую локализацию человек ухитряется выполнять по сигналам от пары своих ушей — стало быть, можно найти способ формирования этих сигналов даже через пару выходных акустических устройств. Алгоритмы обработки цифрового аудиопотока с целью создания иллюзии трехмерной локализации при выводе через 2-, 4-, 6- и даже 8-канальные акустические системы строятся на основе психоакустических моделей восприятия звука человеком. Конечно же, как и в случае вывода трехмерной графики через плоский экран монитора, это все-таки иллюзия, но ее убедительность совершенствуется по мере усложнения алгоритмов. В значительной степени эта иллюзия подкрепляется озвучиваемым изображением (большую часть информации человек воспринимает через зрение). Реализация этих алго

ритмов возможна как с помощью сигнальных процессоров (DSP) звуковых карт, так и центральным процессором (аналогия с 3D-графикой). Но центральному процессору трудно справиться с несколькими аудиопотоками, особенно если учесть его загруженность остальной (не звуковой) частью игрового приложения. Заметим, что для формирования объемного звука в играх не используются (пока?) возможности кодирования в AC-3, поскольку для современных процессоров это слишком ресурсоемкая задача.

Для реализации трехмерного звука применяются разнообразные фирменные технологии, различающиеся подходами и способами достижения трехмерности. Однако их объединяет общий принцип — создание идеального 3D-звука невозможно, поскольку звуковое восприятие у людей сугубо индивидуально. Тем не менее, трехмерный звук вносит дополнительное оживление в мультимедийную систему компьютера.

В технологии *SRS 3D Sound* (фирмы SRS Labs) информация извлекается из обычного (двухканального) стереопотока и после обработки выводится через пару каналов, но с большим пространственным эффектом. Функция обработки называется HRTF (Head-Related Transfer Function), она описывает передаточные функции прохождения звуковых сигналов к барабанным перепонкам ушей слушателя в зависимости от положения источника звука относительно его головы. Функция HRTF зависит от трехмерных координат (здесь удобна сферическая система координат) и частоты.

Технология *TruSurround*, опирающаяся на ядро SRS 3D Sound, позволяет 6-ка- нальный поток AC-3 конвертировать для воспроизведения через одну пару колонок с сохранением трехмерности.

Технология *3D RSX* (3D Realistic Sound Experience) фирмы Intel — это программная реализация функции HRTF на процессорах с расширением MMX. Она позволяет позиционировать источник и поддерживает дополнительные эффекты, включая реверберацию, смещение строя и эффект Доплера.

Технология *A3D* фирмы Aureal Semiconductor[1] решает примерно те же задачи, что и SRS 3D Sound. Эта фирма разработала и технологию *Wavetracing*, которая выполняет преобразования звука, напоминающие трехмерное графическое моделирование. Здесь также присутствует сцена, только в данном случае акустическая. На сцене располагаются пассивные акустические объекты, например стены. Объекты представляются акустическими многоугольниками, обладающими определенными свойствами поглощения и отражения звука. Правда, в отличие от графики, количество многоугольников, требуемое для достоверной звукопередачи, ограничивается несколькими десятками. Среди этих объектов располагаются виртуальный слушатель, а также источники звуков. Для каждого источника звука просчитываются все возможные пути прохождения звука до ушей (обоих) слушателя: учитываются затухания в преградах, отражения от предметов, атмосферное затухание. При расчетах учитывается динамика движения и моделируется эффект Доплера. Акустические объекты (многоугольни

[1] К сожалению, новых следов этой технологии в 2005 году найти не удалось.

ки) создаются конвертированием графической модели. Источниками звука могут быть любые средства компьютера, генерирующие цифровой аудиопоток.

Трехмерный звук генерируется прикладными программами, и для совместимости с различными звуковыми картами разработаны специальные интерфейсы (API), одним из которых является DS3D (DirectSound 3D). В DS3D входит собственно интерфейс API (набор команд), интерфейс аппаратных средств звуковой карты (драйвер) и программный эмулятор HEL (Hardware Emulation Layer), который при отсутствии аппаратной поддержки 3D-звука картой программно выполняет команды API с минимальными возможностями (насколько позволяет процессор). Поддержку DS3D (драйверы) имеют все карты, претендующие на 3D-звук. В DS3D используется выход на две колонки (стереосистема).

EAX (разработка Creative Labs) — программный интерфейс (API), позволяющий в реальном времени при воспроизведении звука вводить различные эффекты. EAX 1.0 обеспечивает управляемую реверберацию, создавая эффект нахождения в том или ином пространстве: в комнате, коридоре, трубе, концертном зале, на улице и т. п. Каждому типу помещения соответствует свой набор параметров реверберации (пресетов). Первой картой, поддерживающей EAX

1.0, была Sound Blaster Live!. EAX 2.0 позволяет учитывать объекты, расположенные в трехмерном пространстве между источниками звука и слушателем. При этом учитываются как геометрия, так и акустические свойства (материал) объекта (даже воздуха). Поддержка EAX 2.0 была введена в Sound Blaster Live!

5.1. В EAX Advanced HD 3.0 введены более тонкие механизмы управления реверберацией, здесь переход из одного окружения в другое (например, из коридора в комнату) происходит более гладко (в прежних версиях переключение пресетов изменяло параметры реверберации скачком). Для таких эффектов требуется более мощный сигнальный процессор, который появился в картах Sound Blaster Audigy. В EAX Advanced HD 4.0 реализована возможность одновременного использования нескольких типов окружения. Так, стоя в дверях комнаты, можно слышать звуки из комнаты с одними (комнатными) параметрами реверберации, а звуки из коридора — с другими. Кроме того, появилась возможность вводить эффекты электронной музыки: имитировать вау-приставку к гитаре, флэнджер, фэйзинг, искажение голоса и т. п.

Аудиоданные на дисках CD и DVD

На дисках CD и DVD аудиоинформация может храниться в разных видах: в потоках, воспроизводимых проигрывателями, и файлах (которые современные MP3-проигрыватели тоже могут воспроизводить).

На стандартном диске *аудио-CD* (CD/DA) данные записаны в формате LPCM (прямая оцифровка) с частотой дискретизации 44,1 кГц, два канала разрядностью 16 бит, стандартная длительность — 74 минуты. Данные защищены от ошибок (см. 9.8), приводы CD (и DVD-комбайны) позволяют выводить этот поток через собственный ЦАП на аналоговый интерфейс (с уровнем линейного выхода) или интерфейс S/PDIF. Современные модели приводов и их драйверы позволяют извлекать и цифровые данные с аудио-CD. Производительности ны

нешних компьютеров и их аудиокодеков достаточно для того, чтобы не пользоваться специальным аудиоинтерфейсом привода.

В *дисках DVD-Audio* может применяться запись в формате LPCM, до 6 каналов с частотами дискретизации 48/96/192 кГц (а также 44,1/88,2/176,4 кГц) и разрядностью 16/20/24 бит. При этом теоретический предел качества — полоса до 96 кГц при динамическом диапазоне 144 дБ. Мультиканальные записи проигрывателями могут сводиться в двухканальные (стерео). Частоты 192 и 176,4 кГц доступны только в двухканальном варианте. Максимальная скорость потока — **9,6** Мбит/с.

В дисках DVD-Audio может применяться и *сжатие MLP* (Meridian Lossless Packing) — схема Dolby, обеспечивающая коэффициент сжатия около 2:1 (без потерь) с возможностью восстановления PCM-сигнала. MLP обеспечивает воспроизведение 74-135 минут 6-канального звука (96 кГц, 24 бита) на однослойном диске (без компрессии — 45 минут). Стереозвука с тем же качеством умещается 120-140 минут (без компрессии — 67 минут).

Диски SACD (Super-Audio CD), для которых вводился формат DSD, представляют собой двухслойный носитель информации (аналогично двухслойным дискам DVD). Здесь между обычным отражающим слоем с «ямками» (CD Layer) и прозрачной основой помещается полупрозрачный слой с высокой плотностью хранения (HD Layer) объемом 4,7 Гбайт. Слой CD Layer несет ту же информацию, что и обычный аудио-CD (74 минуты стерео, 16 бит, 44,1 кГц, объем 780 Мбайт). Этот слой считывается обычными приводами CD (лазером с длиной волны 780 нм). Слой HD делится на 3 области: с двухканальным стереопотоком DSD, с 6-канальным потоком и дополнительную область для текста, графики и видео. Обе аудиообласти позволяют хранить те же 74 минуты записей, но в разных форматах (для разных воспроизводящих систем). Считывание HD-слоя осуществляется лазером с длиной волны 650 нм, фокусированным на этом слое. Считывать этот слой могут специальные приводы (и DVD).

В *дисках DVD-Video* с каждым видеосюжетом может быть связано до 8 каналов аудиосопровождения. В каждом канале может использоваться один из перечисленных ниже форматов (форматы DTS и SDDS обязательными для поддержки плеерами не являются):

- *LPCM* с частотой 48 или 96 кГц, разрядность 16, 20 или 24 бит, 1-8 каналов, максимальная скорость потока — 6,144 Мбит/с.
- *Dolby Digital* (AC-3) — компрессия с потерями, исходный сигнал PCM 48 кГц — до 24 бит. Скорость потока от 64 до 448 Кбит/с (есть и нестандартная скорость 640 Кбит/с, поддерживаемая многими декодерами). Для схемы 5.1 обычно используется скорость 384 или 448 Кбит/с, для стерео (с объемным звуком или без него) типично 192 Кбит/с. Комбинации каналов (фронт/ тыл) могут быть различными: 1/0, 1+1/0 (два моноканала), 2/0 (стерео), 3/0, 2/1, 3/1, 2/2 и 3/2, — для всех этих комбинаций может применяться дополнительный низкочастотный канал.
- *MPEG audio* — мультиканальный цифровой звук, компрессия (с потерями) исходного сигнала PCM с частотой 48 кГц, 16 или 20 бит. Поддерживаются

форматы MPEG-1 и MPEG-2 Layer 2, но не MP3. Скорость потока (переменная) — от 32 до 912 Кбит/с, типичная скорость — 384 Кбит/с (это предел для MPEG-1). Комбинации каналов (фронт/тыл): 1/0, 2/0, 2/1, 2/2, 3/0, 3/1, 3/2 и 5/2, — в любой из них может дополнительно использоваться низкочастотный канал (тогда последние два варианта дают схемы 5.1 и 7.1). В MPEG-1 возможен только стереозвук; многоканальный звук MPEG-2 передается в дополнительных каналах, так что обеспечивается совместимость с проигрывателями, поддерживающими только MPEG-1.

- *DTS* (Digital Theater Systems — цифровые театральные системы) — мультиканальный цифровой звук, компрессия (с потерями) исходного сигнала PCM с частотой 48 кГц и разрядностью до 24 бит. Скорость потока — от 64 до 1536 Кбит/с (формат DTS Coherent Acoustics допускает до 4096 Кбит/с, но для DVD эта скорость не поддерживается). Типичными скоростями являются 754,5 и 1509,25 Кбит/с для системы 5.1 и 377 или 754 Кбит/с для двухканальных систем. Комбинации каналов: 1/0, 2/0, 3/0, 2/1, 2/2 и 3/2, — в любой из них может дополнительно использоваться низкочастотный канал. Возможны форматы и для схем 6.1 и 7.1
- *SDDS* (Sony Dynamic Digital Sound) — мультиканальный (5.1 или 7.1) цифровой формат, сжатый из PCM 48 кГц, скорость потока — до 1280 Кбит/с.

12.2. Звуковые карты PC

Современные *звуковые карты* (или аудиосредства, интегрированные в системную плату), представляют собой комбинированные устройства, в той или иной мере исполняющие все функции, перечисленные в начале главы. Ранние модели карт были менее универсальными: они могли иметь синтезатор, но не иметь аудиокодека (Ad Lib) и наоборот (WSS), не иметь порта MIDI или, наоборот, выступать лишь в роли MIDI-интерфейса (MPU-401). «Законодателем мод» на звуковые карты массового применения можно назвать фирму Creative Labs. Ее первые звуковые карты, выпускавшиеся для шины ISA, по нынешним меркам были примитивными. В них основные манипуляции с аудиосигналами выполнялись в аналоговом виде, применялись относительно несложные FM-синтеза- торы, а WT-синтезаторы могли устанавливаться лишь как дорогостоящее расширение. «Штатный» WT-синтезатор появился только на карте AWE. Раньше карт AWE фирма Gravis Ultrasound выпустила своего знаменитого «гуся» — карту *GUS с* WT-синтезатором (но без FM). Первые модели GUS имели только 8-битный АЦП (но ЦАП 16-битный), процессор эффектов отсутствовал, голоса загружались в ОЗУ (наращиваемое). Карта была не совместима ни с какими другими семействами, что осложняло ее немузыкальные применения. Позже появились полностью 16-разрядные карты с поддержкой PnP, процессоры эффектов. Однако теперь фирма Gravis звуковыми картами не занимается.

Современные звуковые карты по своим возможностям ушли уже очень далеко от первых моделей, но зачастую поддерживают программную совместимость с картами, от которых произошли фактические стандарты. Старые звуковые карты в основном выпускались для 16-битной шины ISA (применение 8-бит-

ных нежелательно), все последние их модели поддерживают технологию PnP. Нынешние карты выпускаются для шин PCI и PC Card или встраиваются в системную плату.

Фактическим стандартом стал набор аудиосредств платы Sound Blaster фирмы Creative Labs. Для шины ISA разными фирмами выпускалось большое число моделей звуковых карт, похожих по свойствам. Их совместимость с моделью SB 16 является залогом совместимости с множеством игр, работающих в среде DOS непосредственно с «железом» звуковой карты. В случае же отсутствия такой совместимости игра (ее разработчик) должна «знать» аппаратуру данной карты или пытаться работать с программным эмулятором SB. В среде Windows с аппаратурой принято общаться через системные драйверы, так что проблемы совместимости решаются чисто программно. В защищенном режиме эмуляцию выполняют посредством генерации исключений процессоров при обращении к портам ввода-вывода. Однако в реальном режиме процессора полностью прозрачная эмуляция невозможна, и именно DOS-игры оказались под грузом наследия, который в XXI век тащить уже не хотели.

Традиционно на звуковых картах устанавливают порт аналогового джойстика (игровой порт, см. 11.6), при этом сохраняя интерфейс, пришедший со времен первых ПК. Этот интерфейс реализует аналого-цифровое преобразование сигналов от резистивных датчиков чисто программно, расходуя ресурс процессора. В некоторых звуковых картах преобразование выполняется аппаратно (интерфейс с джойстиком сохраняется), но программа должна снимать показания иным способом.

Современные звуковые карты используют шину PCI, гораздо более мощную по пропускной способности. В звуковых картах широко применяют прямое управление шиной — это разгружает процессор, особенно при озвучивании игр с 3D-звуком. Перенос звуковых карт на PCI вызывал некоторые трудности переходного периода, поскольку на какое-то время требовалось обеспечить совместимость с SB 16, где доставка цифрового потока производится по каналу DMA контроллера 8237A. В PCI любая карта может быть контроллером обмена с памятью. Для совместимости звуковых карт PCI с SB 16 может использоваться один из двух механизмов: PC/PCI или DDMA. Механизм PC/PCI был разработан фирмой Intel, чтобы обеспечить возможность использования слотов ISA в блокнотных ПК, подключаемых к док-станции по шине PCI. Альтернативное решение — механизм DDMA (Distributed Direct Memory Access — распределенный прямой доступ к памяти). Как известно, контроллеры DMA для шины ISA располагаются на системной плате, и управление несколькими каналами выполняется через одни и те же регистры. DDMA позволяет «расчленить» стандартный контроллер и отдельные его каналы эмулировать средствами карт PCI. Оба этих механизма реализуемы только как часть моста первичной шины PCI, поэтому их поддержка может обеспечиваться (или не обеспечиваться) только на системной плате и разрешаться в CMOS Setup. Помимо режима совместимости с SB 16 звуковая карта PCI может работать и в естественном для этой шине режиме, реализуя все ее преимущества.

К аудиосредствам еще в спецификации PC'99 компания Microsoft предъявляла следующие требования:

- Разрядность преобразователей ЦАП/АЦП: 16-бит.
- Разрядность данных для импульсно-кодовой модуляции (PCM): 8 и 16 бит.
- Частоты дискретизации:
  - обязательные — 8, 11,025, 22,050 и 44,1 кГц;
  - рекомендуемые — 16, 32 и 48 кГц (Advanced audio).
- Воспроизведение MIDI: 16-голосная полифония, 6 тембров (24-голосный 16-тембровый синтезатор для Advanced Audio).
- Полоса частот: 20 Гц - 20 кГц.
- Нелинейные искажения: <0,02 %.
- Отношение сигнал/шум: 75 дБ (для Advanced Audio выходной канал должен иметь отношение сигнал/шум 85 дБ, а для аналоговой части микшера — 90 дБ).

Современные карты удовлетворяют и более высоким требованиям к цифровой обработке.

Аналоговые звуковые карты

Упрощенная блок-схема традиционной аналоговой звуковой карты приведена на рис. 12.6. Аналоговые сигналы от различных источников — микрофона, CD (здесь обычно используется аналоговый интерфейс CD-ROM), линейного входа, а также ЦАП и синтезатора — смешиваются микшером. Микшер для каждого входа имеет аналоговые регуляторы с цифровым управлением, позволяющие изменять усиление и баланс стереоканалов. Микшер может быть дополнен регулятором тембра — простейшим регулятором усиления высоких и низких частот или многополосным эквалайзером (на рисунке не показан). С выхода микшера аналоговый сигнал поступает на линейный выход и оконечный усилитель. Мощности усилителя, устанавливаемого на звуковых картах, достаточно для «раскачки» небольших пассивных колонок или наушников. Любопытно, что на некоторых высококачественных картах устанавливается маломощный усилитель, работающий только на наушники или активные колонки, — мощному усилителю, в общем-то, не место рядом со слаботочными сигнальными цепями. Кроме того, мощные усилители для низкоомной нагрузки требуют на выходе разделительных конденсаторов большой емкости, которые трудно разместить на звуковой карте. Малая емкость этих конденсаторов «завалит» низкие частоты, чем испортит звучание хорошей карты.

Собственно цифровые каналы звуковой карты проходят через интерфейсные схемы от шины расширения до ЦАП и от АЦП обратно к шине. Для передачи потоков данных в картах ISA используются каналы DMA — один 8-битный и один 16-битный. Преобразования синхронизируются от программируемого генератора (на схеме не показан), который определяет частоту дискретизации. Частоту дискретизации, разрядность (8 или 16 бит) и режим (моно/стерео)

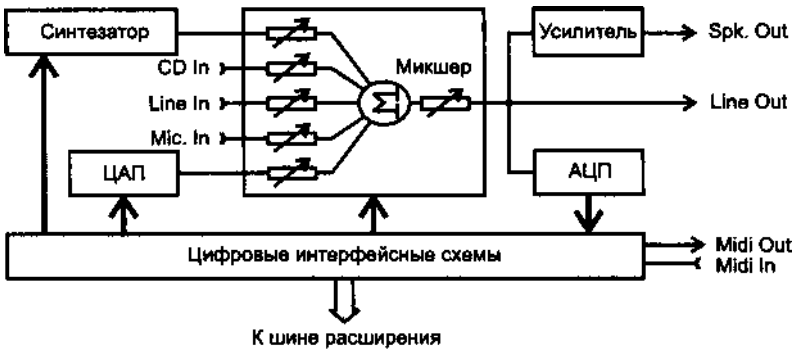


Рис. 12.6. Блок-схема звуковой карты

выбирают при записи. Как уже отмечалось, от этих параметров зависит качество оцифровки и объем информации, занимаемой записью с определенной длительностью. Эти же значения параметров должны устанавливаться и при воспроизведении данной записи (сведения о них хранятся в заголовках звуковых файлов). Изменение параметров воспроизведения относительно параметров записи в общем случае требует программного преобразования файла данных.

Несмотря на наличие двух каналов DMA далеко не все карты позволяли работать в полнодуплексном режиме цифрового канала — одновременно и независимо вводить и выводить цифровой поток. Полный дуплекс нужен, например, для IP-телефонии: аналоговый сигнал от микрофона поступает на АЦП, с которого цифровой поток в сжатом виде укладывается в пакеты IP-транспорта. Одновременно из принятых пакетов данные через ЦАП направляются на аудиовыход. В структуре, изображенной на рис. 12.6, эти потоки пересекаются в микшере. Практически все современные карты поддерживают полный дуплекс. В них имеется два микшера — один для записи, другой для воспроизведения. В сложных картах может быть и пара преобразователей ЦАП (стереофонических), один из которых служит для воспроизведения звукозаписи, а другой обслуживает цифровые синтезаторы. В современных картах, построенных на основе сигнальных процессоров, аналоговые микшеры заменяются цифровыми. В ряде карт число входов у микшеров (4) меньше, чем число возможных источников сигнала. Тогда на карте имеется программно-управляемый коммутатор, с помощью которого можно выбрать источники, посылаемые на микшер записи и микшер воспроизведения.

Для создания специальных эффектов (хор, реверберация и т. п.) на более сложных звуковых картах для обработки сигналов применяются уже упоминавшиеся сигнальные процессоры (DSP).

Цифровые технологии в звуковых картах

По степени вытеснения аналоговой обработки цифровой технологией фирма Intel различает три градации звуковых карт:

- *Аналоговые (analog) карты* имеют аналоговые входные (микрофон, линейный вход, CD) и выходные (линейный вход и вход от усилителя) цепи.

В этих картах чаще всего применяются аналоговые микшеры. На картах располагается и порт традиционного аналогового джойстика и MIDI. В первом поколении карт использовалась шина ISA, аудиокристаллы располагались и на некоторых системных платах. Теперь их сменяют карты для PCI, но при этом обычно сохраняется совместимость с SB 16.

- *Карты Digital Ready* позволяют заменить входные и выходные аналоговые интерфейсы цифровыми, используя шины общего назначения (USB, FireWire) и специальные цифровые аудиоинтерфейсы (S/PDIF, I2S) для подключения цифровой аудиоаппаратуры. В этих картах аудиопоток от любого источника внутри карты представляется в цифровом виде и может перенаправляться как на аналоговые, так и на цифровые внешние интерфейсы или носители информации. В отличие от первых карт, где характеристики АЦП (разрядность, максимальная частота преобразования) часто были хуже, чем у ЦАП, теперь упор сделан на АЦП. Разрядность этого преобразователя повышают до 18 бит и более, сохраняя разрядность ЦАП 16 бит. Качественный АЦП, который остается единственным источником шума («цифра» не шумит по определению), нужен для расширения динамического диапазона. Разрядность обрабатываемых аудиоданных увеличивают до 32 бит, чтобы при вычислениях не терялась точность. Преобразователи ЦАП используются только для прослушивания, так что их погрешности и шумы не накапливаются. Поскольку между такой картой и системной шиной может циркулировать множество цифровых аудиопотоков, в качестве шины расширения PCI альтернатив не имеет. Интерфейс аналогового джойстика с этих карт уходит, подразумевается его замещение цифровым для шины USB.
- В *полностью цифровых (digital only) картах* совершенно отсутствуют аналоговые интерфейсы, в них используются интерфейсы S/PDIF, I^2S, AC-Link, а также ввод-вывод по шинам USB и FireWire. В этих картах от традиционных 16-битных стереостандартов переходят к многоканальным системам большей разрядности и с частотой квантования 48 кГц и выше.

Переход на полностью цифровую обработку аудиосигналов, включая микширование, фильтрацию, позиционирование и применение эффектов, стал возможен даже для программной реализации на современных процессорах (для этого, в частности, предназначалась еще технология MMX). Однако если для одного потока это и приемлемо, то для множества потоков и в многозадачной системе высокого качества обработки не получить (начнутся «заикания»), а загрузка процессора окажется слишком высокой. В графической системе подобная проблема решается видеоакселераторами, которые стали уже традиционной частью видеокарт. То же начинают применять и в аудиокартах. Аппаратные средства, реализующие функции обработки аудиопотоков, называют *аудиоакселератора- ми.* Фактически, это сигнальные процессоры (DSP) со стандартизованным программным интерфейсом (набором драйверов). В среде Windows эффективно взаимодействовать с аппаратными средствами компьютера позволяет технология DirectX. Для звуковых задач в ней имеются интерфейсы DirectSound и DirectSound 3D, названия которых говорят сами за себя. Приложения вызывают аудиосервисы через стандартные программные интерфейсы, а если в сис

теме установлен аудиоакселератор с поддержкой DirectSound и DirectSound 3D, то возможности этих сервисов расширяются, а нагрузка на процессор снижается.

Заметим, что на PC появилась возможность прослушивания музыки без звуковой карты и с высоким качеством (вот она, спираль развития). Достаточно, например, приобрести колонки для USB и подключить их к разъему USB. Под управлением Windows 9x/NT можно наслаждаться воспроизведением аудио- CD, MP3- и WAV-файлов, а при наличии программного WT-синтезатора — еще и проигрывать MIDI. Есть, правда, некоторые неудобства — под DOS колонки будут молчать (шина USB недоступна), трудности могут возникнуть и с другими ОС. Что происходит при отсутствии аудиоакселератора, мы уже обсуждали. Появились и внешние звуковые адаптеры, подключаемые к шинам USB или FireWire, являющиеся полноценными устройствами аудиообработки.

С переходом на цифровые технологии обработки аудиосигналов возникает проблема сведения на микшере сигналов от источников с разными частотами выборки. В аналоговой обработке таких проблем не возникает. Самый простой (но не самый лучший) способ сведения — преобразовывать цифровые сигналы в аналоговые, фильтровать и суммировать уже аналоговые сигналы (а потом их снова преобразовывать в цифровые, например, для записи в файл). Однако возможно и чисто цифровое решение проблемы — преобразование частоты дискретизации, оно же *ресэмплинг* (resampling), или *SRC* (Sample Rate Conversion). Преобразование возможно в сторону как понижения (прореживание выборок), так и повышения частоты. Понижение частоты автоматически приводит к пропорциональному сужению полосы пропускания аудиосигнала (вспомним теорему Котельникова). Повышение частоты выборок путем пересчета аудиосигнал, естественно, не улучшит — новые выборки будут «придуманы» конвертером путем интерполяции реально существующих. Вполне очевидно, что пересчет окажется гораздо проще (и вернее) при удвоении частоты, а не при умножении, скажем, на 8/7. В цифровой аудиосистеме удобно привести все сигналы к единой частоте, а с точки зрения максимального сохранения качества — к самой высокой из используемых. Большое неудобство заключается в том, что диски аудио-CD (считавшиеся еще недавно эталоном качества) задействуют частоту дискретизации 44,1 кГц, а на DVD кодирование MPEG-2 предусматривает частоту 48 кГц, и эту частоту нельзя обходить вниманием. При разработке аудиокодека AC'97 (см. далее) был выбран ряд рекомендованных частот дискретизации 8,000, 11,025, 16,000, 22,050, 32,000, 44,100 и 48,000 кГц. Основной частотой принята 48 кГц, и все остальные сигналы приводятся к ней по схеме, приведенной на рис. 12.7. Самым тяжелым преобразованием в этой цепочке является пересчет 147:160 — из 147 выборок, например, от аудио-CD, получают 160 выборок (разницу «сочиняют»), также равномерно распределенных во времени. Для временного хранения выборок, необходимых при пересчете, требуется буферная память; ее объем зависит от соотношения частот и алгоритма пересчета. Понятно, что даже при самом лучшем алгоритме пересчета сигнал от аудио-CD однозначно потеряет в качестве. Проблема потери качества от ресэмплинга стоит во всех картах, способных обрабатывать несколько цифровых сигналов от разных источников.

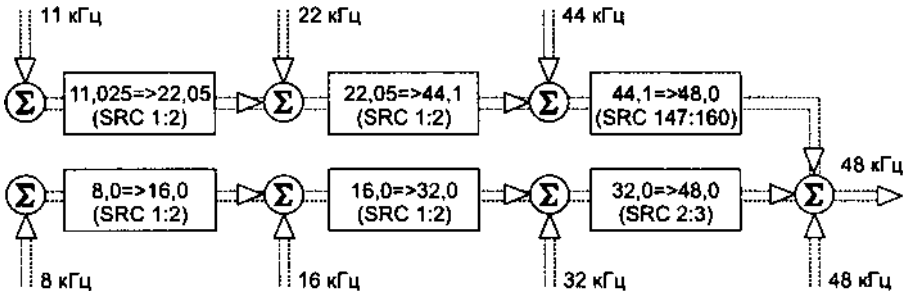


Рис. 12.7. Преобразования частот дискретизации

Аудиокодек AC'97

Аудиокодек AC'97 представляет собой довольно универсальное решение для звуковых карт и модемов, предложенное фирмой Intel в 1997 году; версия 2.2, в которой введена поддержка интерфейса S/PDIF, опубликована в 2000 году. За несколько лет использования аудиокодек AC'97 практически вытеснил дешевые звуковые карты, обеспечивая качество звука, удовлетворяющее большинство пользователей. Аудиосистема на основе AC'97 имеет структуру, приведенную на рис. 12.8. Собственно кодек AC'97 представляет собой микросхему в 48-выводном корпусе с унифицированным назначением выводов, которая выполняет аналого-цифровые и цифроаналоговые преобразования, микширование, аналоговый ввод и вывод сигналов для аудиосистемы и модема. Кодек обеспечивает необходимые преобразования частот дискретизации. Состав функций, реализуемых кодеком, в зависимости от версии микросхемы может варьироваться — от минимально необходимых до расширенных. Кодек подключается к компьютеру по специальному цифровому интерфейсу AC-Link, предоставляемому специализированным *контроллером интерфейса* (AC'97 Digital Controller). Контроллер подключается к шине PCI (как ведущее устройство) или встраивается в чипсет системной платы. В обязательные (минимальные) задачи контроллера входит доставка цифровых аудиоданных в *формате PCM* (см. 12.1) аудиокодеку из памяти и в память от кодека. Для этого, как правило, контроллер должен быть многоканальным контроллером шины PCI (bus master). В память аудиоданные для кодека поступают под управлением центрального процессора от различных источников: из файлов, с внешних цифровых интерфейсов, а также генерируемых программно в реальном времени (например, от игр, от программных декодеров MPEG и т. п.). Из памяти данные от кодека различным потребителям (в файлы, на внешние интерфейсы) доставляет также программа центрального процессора. Более мощный контроллер помимо самой доставки обеспечивает и аппаратное ускорение генерации PCM-данных, например, при декодировании MPEG, формировании 3D-звука, синтезе звуков, а также кодирование PCM-данных в более плотные форматы.

Кодек AC'97 по отношению к контроллеру является подчиненным устройством. Разделение аудиосистемы на две части позволяет собирать ее в соответствии с предъявляемыми требованиями и при необходимости без особых про

блем модернизировать, заменяя лишь микросхему или устанавливая другую (дополнительную) плату кодека с более богатым набором функций или более высокими качественными параметрами. Контроллер интерфейса AC-Link и минимальный кодек AC'97 устанавливаются на многих системных платах, а для добавления новых возможностей интерфейс AC-Link выводится на специальный разъем расширения AMR или CNR (см. 6.2). В интерфейсе есть место и для дополнительных каналов, с помощью которых может быть реализован программный модем. Для этого существует специальный модемный кодек MC'97, который может быть размещен как на системной плате, так и на особой модемной карте. Контроллер способен обслуживать до четырех кодеков; правда, общее количество каналов (тайм-слотов, см. далее) разделяется между всеми кодеками.

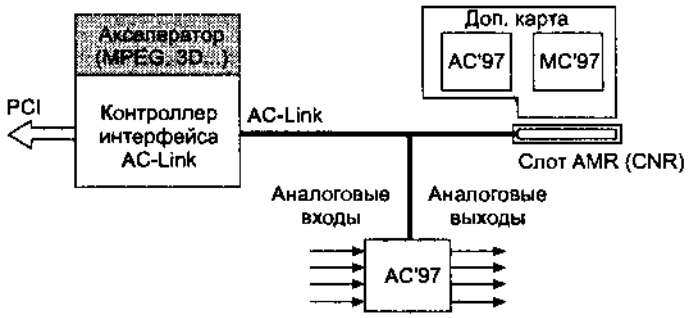


Рис. 12.8. Структура аудиосистемы на базе AC'97

Цифровой интерфейс AC-Link довольно прост; он содержит следующие сигналы:

- BIT_CLK — битовый синхросигнал с частотой 24,576 МГц, источником синхронизации является аудиокодек (он имеет кварцевый генератор);
- SYNC — сигнал синхронизации начала кадра, вырабатывается контроллером, синхронизирующимся от BIT_CLK;
- SDATA_0UT — последовательные данные от контроллера к кодеку;
- SDATA_IN — последовательные данные от кодека к контроллеру;
- RESET# — сигнал аппаратного сброса.

Интерфейс обеспечивает одновременную многоканальную передачу и прием данных по методу временного мультиплексирования (Time Division Multiplexing, TDM). Передача и прием данных происходит в непрерывной последовательности *кадров*. В кадре каждому каналу отводится свой тайм-слот, содержащий 20-битные поля данных. Весь кадр состоит из 13 слотов, нумеруемых от 0 до 12. Частота кадров составляет 48 кГц — это основная частота кодека, так что каждый канал способен нести 20-битные выборки с частотой до 48 кГц. Пропускная способность каждого канала составляет 20 x 48 = 960 Кбит/с. Если частота дискретизации данных, передаваемых каналом, ниже 48 кГц, то отдельные кад

ры (выборки) для данного канала пропускаются и этот факт помечается специальным тегом, передаваемым в начале кадра. Пропуском выборок управляет кодек: при конфигурировании записью в его регистры устанавливается частота выборок для каждого используемого канала. В каждом кадре, посылаемом им контроллеру, кодек отмечает тайм-слоты (каналы) с действительными данными, а также устанавливает биты запросов данных в тех каналах (слотах), которые контроллер должен будет предоставить кодеку в следующем кадре. Для фиксированной частоты 48 кГц все гораздо проще — в используемых каналах данные присутствуют всегда и никаких запросов не требуется. Возможность применения частот, отличных от 48 кГц, появилась только во второй версии AC'97. Объединением пары каналов при необходимости частоту дискретизации можно поднять и до 96 Кбит/с.

Помимо каналов аудиоданных в интерфейсе отводятся слоты для обмена данными контроллера с регистрами кодека. Контроллер может обращаться к любому из 32 регистров (16-битных) кодека (все они имеют четные адреса) как по чтению, так и по записи, причем возможен выбор одного из четырех кодеков, подключенных к контроллеру. Часть регистров кодека имеет стандартизованное назначение: регистры 2-26h относятся к аудиочасти, 28-3Ah — к расширению аудио, 3C-58h — к модему. Регистры 5A-7Ah отдаются в распоряжение разработчика. Если к контроллеру подключено более одного кодека, то входная линия данных для каждого кодека своя, выходная линия данных для всех кодеков общая. Какой из подключенных кодеков будет использовать каждый из каналов, определяется при конфигурировании.

Слоты с данными передаются и принимаются контроллером одновременно, начало кадра отмечается сигналом SYNC, во время которого передается и принимается слот 0. Выходной (для контроллера) слот 0 содержит информацию о кадре: общий бит действительности кадра, теги — биты действительности данных каждого тайм-слота, а также 2-битный идентификатор кодека, к которому обращается команда в данном кадре (если таковая имеется). Выходной слот 1 содержит признак команды (запись или чтение) и 6-битное поле индекса регистра, к которому производится обращение (остальные биты пока в резерве). Выходной слот 2 содержит данные для команды записи — 16-битное слово, прижатое к левому краю (старшие биты). Остальные слоты (3-12) содержат данные выходных каналов контроллера, передаваемые кодеку. Если разрядность PCM- данных меньше 20 бит, то лишние разряды (младшие) обнуляются.

В принимаемых данных слот 0 содержит признак готовности кодека к нормальной работе (неготовность устанавливается по аппаратному сбросу) и теги присутствия данных в последующих слотах. Слот 1 содержит биты запросов данных слотов каналов вывода для последующего кадра, а также эхо индекса регистра (установив его в 0, кодек сообщает об обращении к недопустимому регистру). Слот 2 содержит данные регистра, который считывался командой контроллера. Слоты 3-12 содержат данные каналов ввода.

Назначение каналов (слотов) ввода и вывода приведено в табл. 12.1.

Таблица 12.1. Назначение каналов AC'97

Слот	Назначение
Входные слоты контроллера	
3,4	PCM Playback (L, R) — воспроизведение цифровых данных (из файлов, игр), левый и правый каналы (основные)
5	Modem Line1 DAC - данные для ЦАП модема первой линии
6	PCM Center DAC — данные для ЦАП центрального канала
7, 8	PCM Surround DAC (L, R) — данные для преобразователей ЦАП левого и правого тыловых каналов
9	PCM LFE DAC — данные для ЦАП низкочастотного канала
10[1]	Modem Line2 — данные для ЦАП модема второй линии
11[1]	Modem Handset Output — данные для ЦАП телефонной трубки
12[1]	Данные для управляющих сигналов модема
Выходные слоты контроллера	
3,4	PCM Record (L, R) — запись цифровых данных, левый и правый каналы (основные)
5	Modem Linel ADC — данные от АЦП модема первой линии
6	Dedicated Mic Record — данные от дополнительного АЦП микрофона
7-9	Резерв
10	Modem Line2 ADC — данные от АЦП модема второй линии
11	Modem Handset ADC — данные от АЦП телефонной трубки
12	Данные от сигналов состояния модема

[1] Слоты 10-12 при отсутствии модема могут использоваться для расширения полосы пропускания каналов 3, 4 и 6.

Функциональные возможности аудиокодека хорошо иллюстрирует схема его микшера (рис. 12.9), взятая из спецификации AC'97 2.2. На схеме серым фоном помечены необязательные компоненты. Как видно из схемы, микшер работает с аналоговыми сигналами. Цифровые потоки от контроллера преобразуются в аналоговые сигналы (блоками DAC), предварительно при необходимости блоки SRC приводят выборки к общей частоте 48 кГц. Сигналы от всех источников проходят на главный сумматор микшера (стереофонический) через регуляторы уровня V (Volume control) и выключатели M (Mute). Смешанный сигнал может быть обработан необязательным блоком 3D-эффектов и скорректирован по частотной характеристике необязательным темброблоком (F). Далее сигнал через общий регулятор уровня выводится на аналоговый выход.

Необходимость установки конверторов частот (SRC) в канале воспроизведения (перед ADC) может показаться неочевидной. Однако напомним, что на их вход поступают данные из слотов, которые могут быть заполнены не в каждом кадре. Задача блоков SRC — послать на ЦАП равномерный поток отсчетов (с постоянной частотой).

Тракт цифровой записи не имеет собственного микшера — здесь имеется только коммутатор (MUX), позволяющий оцифровывать (блоком ADC) либо сигналы с одного из входных источников, либо сигнал с главного сумматора (после блока 3D и темброблока). Поток цифровых данных приводится к требуемой частоте выборок (блоком SRC) и выводится из кодека в слотах 3 и 4.

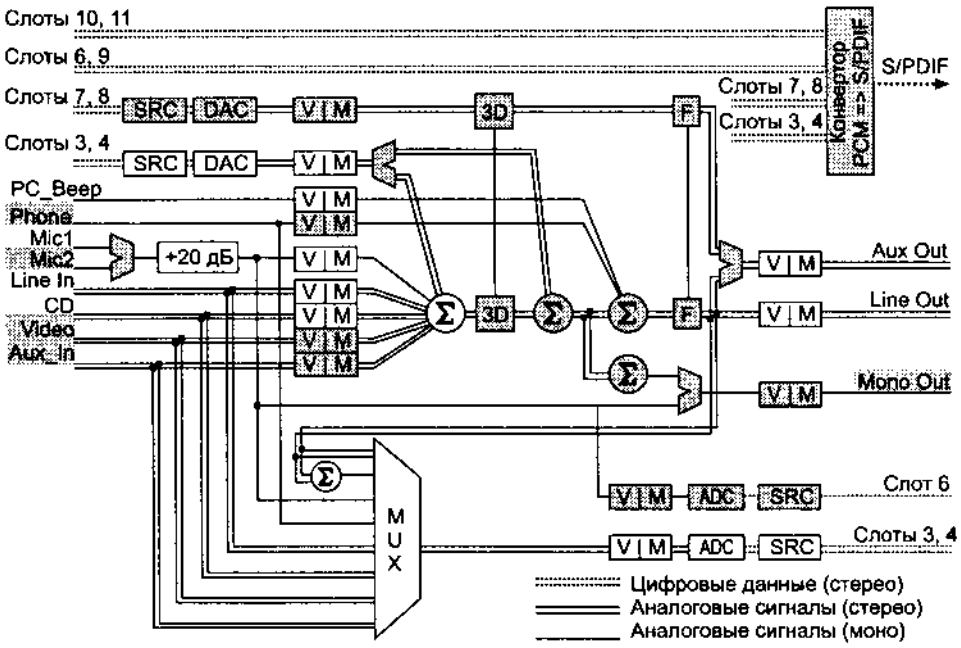


Рис. 12.9. Блок-схема микшера AC'97

Сигнал от микрофона (может предоставляться выбор одного из двух микрофонных входов) усиливается до нормального уровня и, помимо микшера и коммутатора записи, может быть послан на дополнительный специальный микрофонный АЦП. Введение третьего АЦП (вдобавок к паре основных) расширяет возможности эхоподавления: записываемый сигнал от микрофона не будет выходить на акустическую систему, через которую в это же время могут выводиться сигналы от всех остальных источников, также записываемые через основные АЦП. Возможны и другие применения выделенного микрофонного АЦП, например, для интернет-телефонии.

В микшере имеются несколько обходных путей: так, сигнал встроенного динамика системной платы (PC_Beep) и телефона (Phone) незачем пропускать через блок 3D-эффектов. Также основной PCM-сигнал (входные слоты 3, 4) может идти в обход аналогового блока 3D, если эти функции (в цифровом виде) реализуются цифровым контроллером AC-Link.

Кодеку доступны различные конфигурации акустики. В простейшем случае стереосигнал на внешний усилитель снимается с основного выхода (Line Out), а дополнительный выход (Aux Out) может использоваться для подключения стандартных головных телефонов (с сопротивлением 32 Ом). В 4-канальных системах к дополнительному выходу подключаются усилители тыловых колонок; при этом сигнал может быть разложен на 4 канала матричным преобразователем (Dolby Pro Logic). Возможна и иная конфигурация, когда дополнительный выход несет сигналы центрального и тылового каналов. Наконец,

кодек может содержать и передатчик S/PDIF, который б каналов PCM кодирует в единый поток AC-3 для систем 5.1. Передатчик S/PDIF может использоваться и для вывода пары каналов в формате PCM.

Многоканальный звук — High Definition Audio

На смену AC'97 в 2004 году фирма Intel выпустила спецификацию High Definition Audio (сократим ее название до HDA). Ее основная цель — создать инфраструктуру и определить архитектуру аудиоподсистемы, обеспечивающей многоканальные цифровые потоки между кодеками и памятью. В спецификации описан и аудиокодек с более богатыми (по сравнению с AC'97) возможностями. Аудиопотоки позволяют использовать HDA и для подключения кодеков голосовых модемов. Совместимость HDA с AC'97 не предусматривается, хотя общая архитектурная идея та же и даже микросхемы аудиокодеков совпадают по назначению многих выводов.

По сравнению с AC'97 в HDA увеличилась пропускная способность интерфейса и повысилась эффективность его использования за счет отказа от деления кадра на фиксированные 20-битные слоты. Система HDA поддерживает одновременную работу 15 входных и 15 выходных *потоков*, при этом каждый поток может нести цифровую информацию до 16 *каналов*. Поддерживаемые частоты выборок — 6-192 кГц, разрядность отсчетов — 8, 16, 20, 24 и 32 бит.

Как и в AC'97, система HDA начинается с *контроллера* — устройства PCI, обеспечивающего передачу потоков данных между системной памятью и кодеками через интерфейс HDA Link. Контроллер HDA, оставаясь (логически) устройством PCI, может включаться в чипсет или подключаться к какой-либо шине семейства PCI. Кодеков может подключаться несколько — например, аудиокодек, модемный кодек и дополнительный аудиокодек в док-станции мобильного ПК. Интерфейс с контроллером обеспечивается через его регистры, отображенные на пространство памяти, и структуры данных в памяти. *Поток* (stream) представляет собой логическое (виртуальное) соединение между буферами в ОЗУ и кодеком, организуемое отдельным каналом DMA. Поток состоит из одного или нескольких *каналов* данных, каждый канал связан со своим преобразователем (ЦАП, АЦП или конвертором интерфейсов) в кодеке. Например, стереопоток содержит данные левого и правого каналов. Данные каналов потока в памяти расположены рядом и по интерфейсу HDA Link передаются совместно. *Выходной поток* (outbound stream) распространяется по интерфейсу широковещательно, он может приниматься несколькими кодеками. Каждый *входной поток* (inbound stream) может приниматься только от одного кодека.

По каждой входной и выходной сигнальной линии интерфейса передается последовательность *кадров*, частота кадров фиксирована — 48 кГц. Кадр начинается с поля команды/ответа, за которым передаются *пакеты*, содержащие блоки данных потоков. Каждый пакет начинается с *тега потока* (stream tag), идентифицирующего поток и указывающего число блоков в пакете, за тегом следуют блоки отсчетов. *Блок* содержит *отсчеты* (выборки) всех каналов, од

новременно поступающие на ЦАП (или от АЦП). Остаток (если ееть) времени в кадре заполняется нулями. Для потоков с частотой выборки 48 кГц в каждом кадре присутствует по одному блоку данных с комплектом выборок его каналов. Если частота выборок выше или ниже, то число блоков данного потока в кадре оказывается иным (возможно, и переменным от кадра к кадру).

Тактовая частота интерфейса — 24 МГц. В выходном потоке информация передается с двойной синхронизацией. Таким образом, в каждом кадре передается 1000 бит (62,5 16-битных слов), из них 40 бит выделяется для нужд управления, для канальных данных остается 60 слов. Во входном потоке имеет место одиночная синхронизация, в кадре умещается 500 бит (31,25 слов). Из них 36 бит используется для приема ответов, для пакетов данных остается 29 слов.

С каждым потоком связывается канал DMA, обеспечивающий изохронную передачу данных. Передача описывается списком дескрипторов буферов, задающих положение и длину фрагментов потока в физических адресах ОЗУ. Для каждого канала в контроллере имеются регистр-указатель на начало списка и регистр, указывающий на последний действительный дескриптор. Канал DMA должен иметь буферы на 1-2 кадра, предохраняющие контроллер от переполнения (переопустошения) в случае загруженности шины.

После сброса интерфейс HDA Link обеспечивает обнаружение подключенных кодеков. По окончании загрузки соответствующих им драйверов определяются возможности и свойства кодеков. Далее драйвер определяет наличие и типы входных и выходных разъемов кодека. Кодек может определить, подключено ли устройство к данному разъему, а в ряде случаев и тип подключенного устройства. В случае неправильного подключения кодек может исправить ошибку, перекоммутировав сигналы своих входов-выходов.

Взаимодействие драйвера с контроллером осуществляется через регистры и память. В ОЗУ определяется адрес структуры DMA Position in Buffer, в которой *контроллер* регулярно сообщает текущую позицию (адрес исполняемого дескриптора) для каждого потока. С каждым потоком связан набор структур в памяти, включая структуры BDL, CORB, RIRB и структуру формата потока.

Драйвер заполняет в памяти *список дескрипторов буферов* (Buffer Descriptor List, BDL), состоящий из 2-256 элементов. Каждый дескриптор описывает начальный адрес и длину фрагмента в памяти, а также содержит бит IOC, указывающий на необходимость генерации запроса прерывания при выборке данного дескриптора.

Кольцевой буфер CORB (Command Output Ring Buffer) содержит *команды*, посылаемые контроллером в выходной поток в начале каждого кадра. Размер буфера может составлять 2, 16 или 256 элементов. Каждый элемент этого буфера содержит 4-битный адрес кодека, 8-битную *команду* (verb) и 20 бит данных для этой команды. Контроллер аппаратно оперирует парой указателей: CORB WP и CORB RP; по указателю CORB WP (Write Pointer) драйвер добавляет новые команды в буфер (и соответствующим образом модифицирует WP), а по указателю CORB RP (Read Pointer) контроллер задает позицию текущей считанной команды.

Кольцевой буфер RIRB (Response Input Ring Buffer) содержит 32-битные *слова ответов*, приходящих *в* каждом кадре от кодеков, дополняемые 32-битным словом расширения, генерируемым контроллером. Ответы могут быть ожидаемыми (solicit) ответами на команды и неожиданными (unsolicit); последние идентифицируются тегами, размещенными в старших 6 битах ответа и признаком в слове расширения. В слове расширения указывается 4-битный номер кодека, приславшего ответ, и признак неожиданного ответа. Размер буфера может составлять 2, 16 или 256 элементов. Указатели позиций CIRB WP и CIRB RP аналогичны вышеописанным, но по назначению им «зеркальны» (драйвер считывает ответ по RP).

Структура формата потока описывает тип потока (PCM или нет), для PCM- потока задаются частота выборок, разрядность и число каналов. Частота выборок определяется через базовую частоту (48 или 44,1 кГц), ее множитель (1-4) и делитель (1-8).

Данные потока размещаются в памяти определенным образом. Каждый отсчет в памяти упаковывается в контейнер длиной 8, 16 или 32 бита в зависимости от требуемой разрядности. Отсчет «прижимается» к левому краю (старшим битам) контейнера, неиспользуемые биты заполняются нулями. Так, 24-битный отсчет размещается в 32-битном контейнере, и его младшие 8 битов становятся нулевыми. По интерфейсу заполнители не передаются. Отсчеты одного *блока* (одновременные выборки всех каналов) размещаются в смежных контейнерах. Если частота выборок потока превышает 48 кГц, то *пакет*, который передается в кадре, будет содержать более одного блока. Пакеты располагаются в *буферах* длиной, кратной 128 байтам; в каждом буфере должен размещаться хотя бы один полный пакет.

Интерфейс контроллера позволяет *управлять потоками*: запускать, приостанавливать, возобновлять, останавливать и синхронизировать. В системе может быть установлено несколько контроллеров, и их потоки могут быть взаимно синхронизированы (программно, используя значения их счетчиков тактов). Потоки одного контроллера могут быть синхронизированы по запуску и останову. Прерывания от HDA генерируются по событиям контроллера и потоков.

Приведенное описание программного интерфейса HDA очень сильно напоминает изохронную часть контроллера OHCI для шины FireWire, что и неудивительно, поскольку они решают одни и те же задачи. Радует, что идеологически правильный подход к организации потокового взаимодействия становится фактическим стандартом в платформе PC.

Интерфейс HDA Link состоит из следующих сигналов:

- BCLK — сигнал глобальной тактовой синхронизации частотой 24 МГц (вырабатывается контроллером, принимается всеми кодеками);
- SYNC — сигнал синхронизации кадров (48 кГц), с помощью которого передаются и теги выходных потоков (синхронизация и по фронту, и по спаду BCLK);
- SD0 — шина данных, широковещательно передаваемых от контроллера к кодекам (синхронизация и по фронту, и по спаду BCLK);

- SDI — последовательные линии данных, передаваемых кодеками (двухточечные соединения);
- RST# — сигнал сброса.

По составу сигналов интерфейс HDA Link аналогичен AC-Link, но сигналы называются короче и используются иначе. Для масштабирования пропускной способности шина SD0 может иметь разрядность 1, 2 или 4 бита. На широкой (более 1 бита) шине управляющая информация, передаваемая в начале каждого кадра, направляется по всем линиям шины (копируется). Данные распределяются по линиям (striping). Для повышения пропускной способности кодек может использовать более одной линии SDI, при этом с точки зрения интерфейса они независимы.

Организацию обмена на интерфейсе HDA Link иллюстрирует рис. 12.10. Начало кадра отмечается сигналом SYNC длительностью 4 такта. С момента его спада начинается передача пакета потока команд (command) и прием потока ответов (response). Передача пакета каждого *выходного потока* отмечается тегом, который передается по линии SYNC в конце передачи предыдущего пакета. В теге содержится 4-битный идентификатор потока, перед которым передается преамбула (1110). Такая конструкция позволяет отличить передачу тега от передачи метки начала кадра.

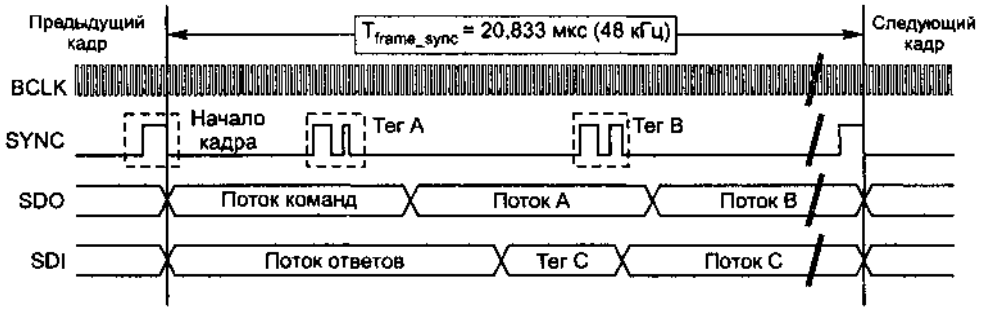


Рис. 12.10. Интерфейс HDA Link

Для *входных потоков* пакеты маркируются иначе. Тег пакета, содержащий 4-бит- ный идентификатор пакета и 6-битное поле длины данных, передается по линии SDI перед самим телом пакета. В конце пакета входного потока могут присутствовать 4 бита-заполнителя (нули).

Спецификация HDA определяет *модульную архитектуру кодеков* и информационные структуры, позволяющие программно выяснять возможности кодеков и управлять их конфигурацией и работой. Кодек состоит из *параметризованных модулей* и их *наборов*, выполняющих определенную функцию. Каждый подключенный кодек получает свой адрес (CAd). Каждый модуль, а также набор представляет собой *узел* (node), уникально адресуемый своим идентификатором N1 (Node Id) в пределах кодека. Узел имеет набор *параметров* (которые можно только читать) и *команд* (с помощью которых модуль подключается, конфигурируется и управляется). *Корневой узел* (root node) обеспечивает ссыл

ки на узлы функциональных групп, содержащихся в кодеке. *Функциональная группа* (function group) — это узел, представляющий собой объединение модулей (каждый из которых адресуем), совместно выполняющих одну прикладную задачу и управляемых одним драйвером. Примеры функциональных групп — *AFG* (аудиогруппа) и *MFG* (модемная группа). Узел функциональной группы имеет общее описание ее возможностей; в его описании содержатся начальный номер и число модулей, входящих в группу.

Модули, выполняющие определенные функции, называются *widget.* Каждый модуль кодека может входить только в одну функциональную группу. В функциональной группе может быть множество однотипных модулей, что позволяет выполнять одновременно операции с несколькими каналами. Каждый модуль имеет параметр конфигурации, который определяет его режим: стерео (два канала) или моно. В описании модуля фигурирует *список соединений* (connection list), в нем перечисляются идентификаторы модулей, выходы которых могут быть поданы на вход данного модуля. Командами управления модули соединяются друг с другом.

В AFG определены следующие типы модулей:

- *Выходной преобразователь* (audio output converter) — ЦАП или устройство, формирующее цифровой поток на интерфейс S/PDIF. На вход модуля поступает поток от интерфейса HDA Link, выход доступен в качестве входного сигнала для других модулей кодека (входит в списки их соединений). Может дополнительно содержать модуль усилителя и процессорный модуль.
- *Входной преобразователь* (audio input converter) — АЦП или устройство, извлекающее цифровой поток из S/PDIF. Его выход передает поток через интерфейс HDA Link, вход подключается к модулю, выбранному в его списке соединений. Дополнительно может содержать модуль усилителя и процессорный модуль.
- *Внешний вывод* (pin widget), обеспечивающий внешнее соединение, аналоговое или цифровое. Вывод может быть входным, выходным или универсальным (переключаемым) и содержать дополнительный входной/выходной усилитель. Вывод может обеспечивать обнаружение факта подключения устройства, определение его импеданса и поддерживать подачу напряжения смещения для микрофонного входа (V_{ref}). Для работы на выход источник сигнала определяется выбором в списке подключения данного модуля. При работе в качестве входа возможность соединения вывода с другими модулями определяется их списками.
- *Микшер* (mixer) — сумматор сигналов от нескольких своих входов, с каждым входом связан список соединений. В каждом входе (и после сумматора) может присутствовать усилитель.
- *Селектор входов* (input selector) — мультиплексор, позволяющий по команде выбирать один из нескольких сигналов (выходов других модулей) в качестве входа. Модуль может дополнительно содержать процессор и усилитель.
- *Ручка уровня* (volume knob) — модуль, позволяющий по внешнему сигналу управлять усилением в других (ведомых им) модулях. Программно по изме

нению положения ручки можно задать либо прямое управление усилением подчиненных модулей, либо генерацию не ожидаемых ответов (unsolicit response), которые должны анализироваться драйвером и отрабатываться соответствующими командами.

- *Генератор тона* (beep generator) обеспечивает подачу приблизительно синусоидального сигнала заданной частоты на все выводы, работающие в качестве выходных. Во время активной работы генератора его сигнал либо смешивается с выводимым потоком, либо временно заменяет его. Генератор тона не указывается в списках соединений. Во время подачи сигнала сброса интерфейса (RST#) аналогичную функцию может выполнять сигнал от входа «PC Speaker».
- *Модуль управления мощностью* (power widget) позволяет управлять энергопотреблением (состоянием) кодека.

В спецификации приводится назначение контактов аудиокодека — микросхемы в 48-выводном корпусе. Помимо сигналов интерфейса HDA Link (по одной линии SDI и SD0) и цепей питания, в «настольном» варианте кодека определены следующие сигналы:

- 8 стереопортов (сигналы PORT-A_L, PORT-A_R, ..., PORT-H_L, PORT-H_R) — могут быть как входами, так и выходами. Выводы, используемые для портов G...H, становятся выводами смещения, если порты A, B, C, D используются как микрофонные входы.
- Стереовход аналогового сигнала воспроизведения CD/DA (CD-L, CD-R).
- Цифровой интерфейс (вход и выход) SPDIF-IN, SPDIF-OUT.
- Вход PCBEEP (от выхода PC Speaker).
- Выходы смещения микрофонных входов (VrefOut-A,..., VrefOut-H) — управляются программно. Могут быть заземлены, находиться в высокоимпедансном состоянии либо получать 50, 80 или 100 % питающего напряжения.
- Входы SENSE_A, SENSE_B — позволяют определять подключение разъемов с помощью кнопок, замыкаемых в гнездах при вставленных «джеках».

Входы SENSE_x дают возможность отслеживать подключение-отключение шнуров (точнее, вилок) за счет функции *jack sense*, которой обладают многие современные звуковые карты (не только с HDA). На гнездах установлены кнопки, которые при вставленной вилке заземляют цепь через резистор; сопротивление резистора кодирует номер порта данного гнезда (рис. 12.11). При изменении состояния подключения кодек посылает сообщение, которое обрабатывается драйвером. Пользователю выводится вопрос о том, что он подключил к данному порту (и на картинке показывают, куда). Если возможности кодека позволяют сконфигурировать данный порт на требуемую функцию, посылаются необходимые конфигурационные команды. В противном случае пользователю показывают, куда это устройство следует подключить.

Конкретные модели кодеков могут использовать не все сигнальные выводы. Параметры кодеков, устанавливаемых в системных платах с интегрированным звуком, как правило, не являются выдающимися среди своих «родственников» — специализированных звуковых карт.

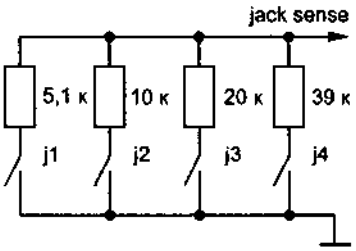


Рис. 12.11. Цепи идентификации подключения

Интерфейсы звуковых карт

Звуковая карта имеет набор разъемов для подключения внешних аналоговых и цифровых аудиосигналов; она может иметь дополнительный интерфейс подключения дочерней карты, расширяющей возможности основной. Дочерние карты могут содержать более качественный синтезатор, дополнительные высококачественные устройства ЦАП, АЦП и иные средства.

Дочерняя карта с MIDI-синтезатором подключается к фактически стандартизованному 26-контактному разъему основной карты, на который выводятся сигнал MIDI-порта (ТТЛ, как и на разъем джойстика) и сигнал аппаратного сброса синтезатора; с дочерней карты принимается стереофонический аналоговый сигнал, который поступает в микшер основной карты. Подключение дочерней карты эквивалентно подключению внешнего синтезатора к MIDI-выходу звуковой карты.

Аналоговые интерфейсы

Аналоговые интерфейсы используются для подключения стандартной бытовой аппаратуры, микрофона, аналогового выхода CD-ROM. На большинстве карт массового потребления для аналоговых сигналов предназначены малогабаритные разъемы — «мини-джеки» (jack) диаметром 3,5 мм, моно и стерео. Эти разъемы универсальны (используются и для бытовой аппаратуры), но имеют весьма низкое качество контактов — являются источником шумов (шорохов и тресков), а также иногда просто теряют контакт. Их полноразмерные 6-мил- лиметровые «родственники», применяемые в профессиональной аппаратуре, имеют весьма высокое качество, но из-за крупных габаритов на звуковые карты не устанавливаются. На некоторых высококачественных картах сигналы линейного входа и линейного выхода выводятся на пары разъемов RCA, что обеспечивает очень хороший контакт (особенно в позолоченном варианте). В просторечии такие разъемы, часто используемые на бытовых видеомагнитофонах, называют «колокольчиками» или «тюльпанами».

Раскладка цепей на мини-джеках унифицирована (рис. 12.12, а): левый канал (Left) — на центральном контакте, экран (GND, «земля») — на внешнем цилиндре, правый канал (Right) — на промежуточном цилиндре. Если стереоджек включить в моно-гнездо и наоборот, сигнал пойдет только по левому каналу (цепь Mono). Все соединения в стереосистемах осуществляются «прямыми» кабелями

(контакты разъемов соединяются «один в один»). Для подключения центрального и низкочастотного каналов в 6-колоночной системе единого подхода нет, и для их правильного соединения может потребоваться перекрестный кабель. Неправильное соединение будет заметно по «писку» сабвуфера и «бормотанию» центральной колонки. Для подключения многоканальных усилителей стали применять и трехканальные 4-контактные мини-джеки, вилки для которых пока что мало распространены, а назначение сигнальных контактов (1, 2, 3) жестко не стандартизовано.

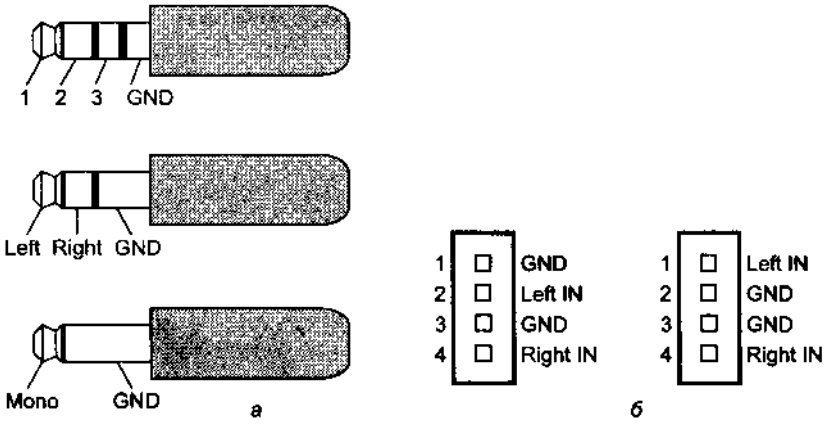


Рис. 12.12. Разъемы подключения аудиосигналов: а — мини-джеки (стерео, моно и трехканальные), *б* — внутренние разъемы подключения CD-ROM

Подключение к звуковой карте устройств через внешние разъемы проблем обычно не вызывает — они унифицированы, и достаточно знать назначение разъемов, маркированных на задней панели:

- *Line In* — линейный вход от магнитофона, тюнера, проигрывателя, синтезатора и т. п. Чувствительность — порядка 0,1-0,3 В.
- *Line Out* — линейный выход сигнала на внешний усилитель или магнитофон, уровень сигнала — порядка 0,1-0,3 В.
- *Speaker Out* — выход на акустические системы или головные телефоны. Подключать к нему внешний усилитель мощности нецелесообразно, поскольку здесь искажения больше, чем на линейном выходе.
- *Mic In* — микрофонный вход, чувствительность — 3-10 мВ. Этот вход обычно монофонический, но иногда используется трехконтактное гнездо (как в стерео), у которого дополнительный контакт (на месте правого канала) предназначен для подачи питания на электретный микрофон.

Подключение внутренних устройств к аналоговым входам может доставить больше хлопот. Для этого используются четырехштырьковые разъемы, различающиеся как шагом межу выводами, так и их назначением. Для подключения CD-ROM часто ставят рядом два, а то и три разъема с параллельно соединенными сигнальными контактами, но и это может не помочь, если кабель имеет

другое расположение сигналов. Спасает только перестановка контактов на разъеме кабеля. Для выполнения такой перестановки иголкой нажимают на фиксирующий выступ контакта, после чего контакт можно вытянуть в сторону кабеля и переставить в другое гнездо. Вид разъемов и варианты расположения сигнальных контактов аудиовходов показаны на рис. 12.12, *б*. Для полноты картины добавим, что разъем может иметь ключ с противоположной стороны (из-за ошибки сборщика кабеля или в соответствии с внутренним стандартом его производителя). Задача подключения все-таки не безнадежна, поскольку требует правильной расстановки только двух сигнальных контактов, а контакты общего провода выделяются тем, что на плате соединяются с шиной, а на кабеле — с экраном. Положение левого и правого каналов аудио-CD в большинстве случаев не принципиально.

Цифровые интерфейсы

Интерфейс S/PDIF (Sony/Philips Digital Interface Format) — это цифровой последовательный интерфейс (и форматы данных) для передачи аудиосигналов между блоками бытовой цифровой аудиоаппаратуры (DAT, CD-ROM и т. п.). Этот интерфейс является упрощенным вариантом студийного интерфейса AES/ EBU (Audio Engineers Society/European Broadcast Union). В интерфейсе AES/ EBU используется симметричный двухпроводной экранированный кабель с импедансом 110 Ом, разъемы XLR, уровень сигнала — 3-10 В, длина кабеля — до 12 м.

В *интерфейсе S/PDIF* используется коаксиальный кабель 75 Ом, разъемы RCA или BNC, уровень сигнала — 0,5-1 В, длина кабеля — до 2 м. В звуковых картах внутренние разъемы S/PDIF проще — это просто пара штырьков (как у джамперов) на плате с соответствующей ответной частью на кабеле. Такие же упрощенные разъемы применяются и на новых приводах CD/DVD, имеющих выход S/PDIF. «Штатная» схема передатчика S/PDIF содержит разделительный импульсный трансформатор (1:1), благодаря которому соединяемые устройства гальванически развязываются. Встречаются и упрощенные варианты без разделительного трансформатора. При стыковке устройств с нестандартными интерфейсами возможны проблемы, связанные с несоответствием уровней сигналов. При этом сигнал может быть неустойчивым (звук будет прерываться) или не приниматься совсем. Эти проблемы могут быть решены подручными средствами — установкой дополнительных формирователей сигнала.

Помимо электрической существует и оптическая версия интерфейса S/PDIF — *Toslink* стандарта EIAJ CP-1201 — с инфракрасными излучателями (660 нм). Применение оптики позволяет обеспечить полную гальваническую развязку устройств, что необходимо для снижения уровня наводок. Для пластикового волокна (POF) длина кабеля не более 1,5 м, для стеклянного волокна — 3 м. В Сети предлагается ряд схем преобразования интерфейсов, одна из которых приведена на рис. 12.13. Здесь первый инвертор посредством обратной связи подведен к линейному участку передаточной характеристики, благодаря чему малый входной сигнал вызывает его переключение. В схеме предлагается использовать микросхему HCT74U04 (6 инверторов, аналог 555ЛН1); вместо све-

тодиода подойдет фирменный трансивер Toslink, его следует подключать без балластного резистора (220 Ом) прямо к выходу инвертора (резистор находится в трансивере).

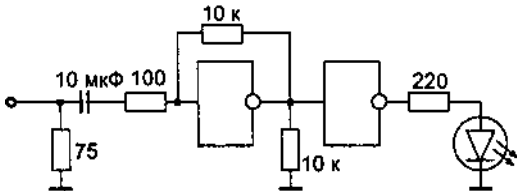


Рис. 12.13. Схема преобразователя электрического интерфейса S/PDIF в оптический (Toslink)

По интерфейсу S/PDIF информация передается в последовательном коде кадрами с обеспечением синхронизации и контролем достоверности передачи (кодами Рида - Соломона). В кадре имеется признак формата данных — PCM или не-PCM, что позволяет по данному интерфейсу передавать и упакованные цифровые данные (например, MPEG для AC-3). Имеется также бит защиты от копирования, признак предискажений и некоторые другие служебные данные. В режиме PCM выборки каждого канала могут иметь разрядность 16, 20 или 24 бита, частота выборок определяет частоту цифрового сигнала. Приемник S/PDIF сам определяет частоту выборок по принимаемому потоку, наиболее популярные частоты — 32, 44,1 и 48 кГц.

Для вывода на акустические системы через интерфейс S/PDIF используется *ресивер* — приемник и декодер сигнала S/PDIF с усилителями мощности, который является довольно дорогостоящим устройством. Есть и сравнительно дешевые активные акустические системы 5.1 (и 7.1) для компьютеров с интерфейсом S/PDIF и встроенным ресивером, но качество их до Hi-Fi не дотягивает.

Шина PS (Inter IC Sound, тоже изобретение Sony и Philips) предназначена для передачи двухканального (стерео) PCM-потока между микросхемами цифровых аудиосистем. Этот интерфейс может использоваться для подключения к звуковым картам дополнительных АЦП и ЦАП. Протокол I^2S крайне прост (рис. 12.14). В интерфейсе I^2S используется всего три сигнальных линии:

- SCK — постоянный Сигнал синхронизации, по фронту которого выбираются остальные сигналы;
- WS (Word Select) — выбор передаваемого слова (0 — левый канал, 1 — правый);
- SD — последовательные данные в дополнительном коде, передаваемые старшим битом вперед. Это позволяет использовать интерфейс при несовпадении разрядности приемника и передатчика. Приемник лишние (для него) биты игнорирует, недостающие считает нулевыми. Начало каждого слова (PCM-отсчета) отмечается сменой состояния сигнала WS (оно происходит на такт раньше передачи старшего бита, что облегчает синхронизацию приемника).

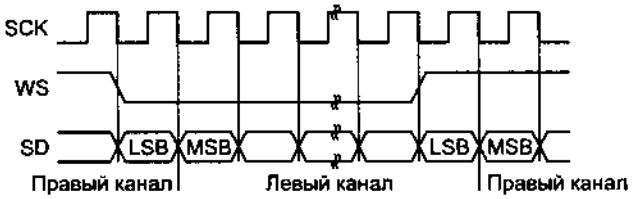


Рис. 12.14. Интерфейс I²S

Сигналами SCK и WS управляет *контроллер шины*, в роли которого может быть либо передатчик, либо приемник, либо внешнее (по отношению к ним) устройство. Передатчик (и контроллер) на шине всегда один, приемников может быть несколько.

Помимо этих интерфейсов в студийной аппаратуре применяют интерфейсы ADAT и TDIF, которые имеются только на дорогих профессиональных звуковых картах.

«Исторические» модели звуковых карт

Рассмотрим кратко основные «столпы совместимости» — массовые звуковые карты. Заметим, что ими не исчерпывается перечень существующих устройств — есть множество других, среди которых имеются и «элитные» карты, удовлетворяющие потребности самых взыскательных музыкантов.

Ad Lib

Ad Lib — первая звуковая карта для PC, имеющая только FM-синтезатор на микросхеме YM3812 (OPL2), занимающая адреса 388—389h. Карта Ad Lib Gold имеет стереофонический синтезатор на микросхеме YM262 (OPL3), занимает адреса 388—38B. Аудиокодеков эти карты не имели; регистровая совместимость с Ad Lib поддерживалась (или эмулировалась) многими последующими картами.

MPU-401 UART и MT-32

Карта *MPU-401* фирмы Roland — первая карта расширения для PC с интерфейсом MIDI, получившая широкое распространение. Контроллер MPU (MIDI Processing Unit — устройство обработки сообщений MIDI) помимо асинхронного последовательного порта UART (Universal Asynchronous Receiver Transmitter — универсальный асинхронный приемопередатчик), реализующего физический интерфейс MIDI, имел развитые аппаратные средства для использования PC в качестве секвенсора. Он имел внутренний таймер-счетчик, который маркировал приходящие сообщения по времени. Синхронно с этим таймером мог работать внутренний метроном, подававший звуковые сигналы. Таймер через аппаратные прерывания мог управлять отправкой сообщений MIDI. Ряд моделей имели средства синхронизации с ленточными записывающими устройствами, синхронизации MTC/SMPTE и фильтрации данных. Такая насыщенность аппаратными сервисами была обусловлена низкой производительностью PC на 8086, недостаточной для программной реализации секвенсора.

Эти довольно навязчивые сервисы, доступные в интеллектуальном режиме (intelligent mode) работы платы, высокопроизводительными компьютерами не использовались (те же функции стало проще реализовать программно и не разбираться с источниками прерываний, приходящих от MPU). Контроллер MPU- 401 имел и простой режим работы — *UART mode*, — в котором применялся только двунаправленный асинхронный порт.

В пространстве ввода-вывода MPU-401 занимает два смежных адреса, MPU (обычно 330h) и MPU+1:

- Порт DATA (адрес MPU+0) обеспечивает запись и считывание байтов, передаваемых и принимаемых по интерфейсу MIDI. В интеллектуальном режиме через этот же порт считываются и вспомогательные данные от MPU (не поток MIDI).
- Порт STATUS/COMMAND (адрес MPU+1) служит для чтения состояния и записи команд (запись — только для интеллектуального режима). В байте состояния определены следующие биты:
  - бит 7 — DSR (Data Set Ready) — готовность (DSR = 0) принятых данных для чтения (бит устанавливается в единицу, когда все принятые байты считаны из регистра данных);
  - бит 6 — DRR (Data Read Ready) — готовность (DRR = 0) UART к записи в регистр данных или команд (условие готовности к записи не возникнет, если приемник имеет непрочитанный байт данных).

По включении питания «настоящий» контроллер MPU-401 устанавливается в интеллектуальный режим, из которого в режим UART его можно перевести командой с кодом 3Fh. Программный сброс MPU-401 (опять-таки в интеллектуальный режим) осуществляется командой RESET (код FFh), на эту команду MPU отвечает подтверждением ACK (FEh). Байт подтверждения извлекается из регистра данных, до его прихода следующую команду MPU не воспримет. На команду с кодом 3Fh контроллер MPU подтверждением не отвечает (некоторые эмуляторы отвечают и на эту команду).

Ввод данных может осуществляться по программному опросу бита DSR или по прерываниям. *Аппаратные прерывания* от MPU UART вырабатываются по приему байта. Обработчик прерывания должен считать все поступившие байты, проверив перед выходом, что DSR = 1 (иначе возможны потери принятых байтов).

Вывод данных разрешается битом DRR, прерывания по готовности вывода не вырабатываются.

Совместимость с MPU-401, имеющаяся у большинства современных звуковых карт с интерфейсом MIDI, означает наличие приемопередатчика, программно совместимого с MPU-401 в режиме UART; функции интеллектуального режима обычно не поддерживаются.

MT-32 — синтезатор с MIDI-интерфейсом, подключаемый к плате MPU-401, обеспечивает одновременное звучание до 32 нот, имеет встроенный ревербератор и поддерживает 33 звуковых эффекта. Набор инструментов — 128 мелодиче

ских и 30 ударных, по раскладке инструментов частично совместим с GM (General MIDI, см. далее). *Совместимость с MT-32* означает поддержку того же набора инструментов с той же раскладкой по номерам.

Windows Sound System

Windows Sound System (WSS) — это карта аудиокодека, обеспечивающая стереофоническую запись и воспроизведение в 16-битном режиме с частотой до 48 кГц. Имеет 4 непосредственно адресуемых регистра, по умолчанию базовый адрес WSS — 534h. Через первые два регистра обеспечивается косвенный доступ к 32 регистрам. Занимает прерывания IRQ 10. Реализует отработку команд стандарта WSS.

Семейство Sound Blaster для шины ISA

Карта *Sound Blaster* имела 8-битный монофонический цифровой тракт с частотой дискретизации 11/22 кГц (запись/воспроизведение), двухоператорный (но многоголосный) FM-синтезатор OPL2, интерфейс MIDI, аналоговые аудиовходы и выходы; полный дуплекс не поддерживался. Позже появилась карта *Sound Blaster Pro* — уже стереофоническая, в которой частота дискретизации была поднята до 44,1 кГц в режиме моно и 22,05 кГц, но только 8 бит, в режиме стерео. Эта модель была принята во многих играх, и требования совместимости с Sound Blaster зачастую заставляют автоматически устанавливать эти весьма посредственные параметры звуковоспроизведения. Синтезаторов стало два — по одному на каждый канал. Эта карта стала основой для программно-аппаратной модели совместимости SB: в пространстве ввода-вывода 220-22F находятся регистры синтезатора, микшера и, дополнительно, DSP. Использует прерывание (IRQ5), стандартным средством доставки цифрового потока являлся 8-бит- ный канал DMA (обычно DMA 1) шины ISA.

Карта *Sound Blaster 16* стала основой фактического стандарта SB16, это устройство является базовым в плане обеспечения совместимости звуковых карт и программного обеспечения. В карте применены 16-разрядные АЦП/ЦАП, частота дискретизации — 4-44,1 кГц в любом режиме. Аудиокодек задействует два канала DMA (один 8-битный и один 16-битный) и один запрос прерывания. Для 16-битного режима используется 16-битный канал DMA (обычно DMA 5). Поддерживает дуплексный режим, но при этом запись идет в 8-битном формате, поскольку требуются два канала DMA, один из которых 8-битный. В аналоговый тракт введены управляемые регуляторы тембра по низким и высоким частотам. Карта имеет аналоговые входы и выходы: линейный вход и выход, микрофонный (моно) и внутренний входы (для CD-ROM) и выход на наушники (колонки), а также FM-синтезатор (OPL-2). В пространстве ввода-вывода карта занимает 16 смежных адресов (синтезатор, микшер и DSP), базовый адрес (SB) стандартно назначается на адрес 220h (карта занимает область 220- 22Fh). Для совместимости с Ad Lib и регистры синтезатора имеют копии по адресам 388-38Bh. Плата эмулирует MPU-401 в режиме UART (адреса 330, 33lh). На плату может быть установлен «продвинутый» сигнальный процессор (ASP), а также дочерняя плата с WT-синтезатором. Разъем дочерней платы

унифицирован, на него выводятся выход MIDI (с уровнями ТТЛ), аналоговые входы и питание. Карты обычно снабжаются портом для подключения джойстика — игровым портом с программной реализацией обработки сигналов от резистивных датчиков положения. На контакты 12 и 15 разъема игрового порта выводятся сигналы MIDI с уровнями ТТЛ. Похожие возможности имел целый ряд моделей карт от Creative Labs и иных производителей. Они различались в некоторых деталях и характеристиках (особенно по уровню шумов) и развивали возможности FM-синтезаторов (увеличивалось число операторов и голосов). Совершенствовались и применяемые сигнальные процессоры. Обычный процессор DSP при записи и воспроизведении выполняет компрессию-декомпрессию данных. «Продвинутый» процессор ASP (Advanced Signal Processor), он же CSP (Creative Signal Processor), позволяет расширить возможности сжатия и помогает в распознавании речи. С его помощью реализовывался объемный звук Qsound, позже эти функции стал выполнять процессор 3DSound.

На многих картах присутствует и интерфейс IDE для подключения CD-ROM (в старых моделях устанавливали специфические интерфейсы CD-ROM Sony, Panasonic, Mitsumi). Более поздние модели исполнялись в варианте PnP, что большинству пользователей облегчало установку карты.

Для музыкантов больший интерес представляли карты с WT-синтезатором. Карта *Sound Blaster AWE32* (Advanced Wave Effects) сочетает в себе SB16 с 32-голосным WT-синтезатором и развитым процессором эффектов на микросхеме EMU8000. Сигнальный процессор для каждого голоса (а также сигнала с FM-синтезатора) реализует управляемые эффекты реверберации, хоруса, резонансного фильтра. Голоса инструментов хранятся в ПЗУ, имеется и ОЗУ (наращиваемое) для загрузки произвольных наборов инструментов. Существует целое семейство карт AWE (включая и AWE64, и SB32) с поддержкой технологии PnP и без таковой, с разным объемом памяти, с нюансами в процессорах эффектов, в поддержке 3D-Sound и т. п.

Карты PCI — SB Live!

Среди массовых карт PCI отметим карты *Sound Blaster Lively* опять-таки от Creative Labs. Эти карты, появившиеся в 1999 году, содержат волновой синтезатор-сэмплер, процессор эффектов, цифровой микшер, многоканальный ау- диорекордер/проигрыватель, процессор трехмерного позиционирования звука, аналоговые и цифровые аудиоинтерфейсы, порты MIDI. Все эти функции реализуются на одном кристалле EMU10K1 — 32-разрядном процессоре DSP, выполняющем все операции при частоте квантования 48 кГц. Архитектура карты ориентирована на средний (по меркам того времени) компьютер: минимум Pentium 133, ОЗУ — 16 Мбайт, шина PCI 2.1.

Синтезатор использует часть ОЗУ (до 32 Мбайт) компьютера (на плате нет ни ПЗУ, ни ОЗУ для сэмплов), общаясь с ней через собственный интеллектуальный контроллер шины PCI. Аппаратно обеспечивается 64-голосная полифония с 8-точечной интерполяцией, позволяющей сократить объем сэмплов не в ущерб качеству. Программно могут быть реализованы еще 512 голосов. Доступ к синтезатору обеспечивают 2 внутренних порта MIDI для аппаратного и 1 порт

для программного синтезатора, что позволяет одновременно использовать до 48 разных инструментов (3 x 16 каналов). Высокое качество WT-синтеза обеспечивается применением многослойной структуры инструментов, управляемых фильтров и прочих технологий «оживления» звука. Карта имеет средства редактирования сэмплов и банков инструментов в формате SF2 (SoundFont 2).

Процессор эффектов в реальном времени обеспечивает реверберацию, хорус, флэнжер, смещение высоты тона и ряд других. Эффекты доступны для любых аудиоисточников (включая звукозаписи) в различных сочетаниях. При обработке, микшировании и позиционировании используются аппаратные каналы числом до 131, вся обработка выполняется в цифровом виде. Эффекты (их параметры) управляемы в реальном времени. Структура микшеров позволяет одновременно использовать как «чистые» сигналы источников, так и сигналы с наложенными эффектами. Система трехмерного позиционирования дает возможность выбрать оптимизацию для 2 или 4 колонок или наушников. Карта имеет акселератор DirectSound и DirectSound3D и поддерживает интерфейс EAX 1.0.

Канал записи-воспроизведения *Sound Blaster Live!* обеспечивает разрядность 8 или 16 битов, частоту квантования 8-48 кГц, полный дуплекс. Более того, поддерживается до 32 одновременных аудиосеансов с аппаратным микшированием. Возможно применение внешних 20-битных АЦП с интерфейсом S/PDIF, при этом все критичные к наводкам цепи выносятся из корпуса компьютера.

Карта имеет богатый набор внешних интерфейсов. Это традиционные миниджеки для микрофонного и линейного входов, линейных выходов фронта и тыла, разъем джойстика/MIDI. На дополнительной плате установлены разъе- мы-«колокольчики» S/PDIF (вход и выход), MIDI (вход и выход), цифровой выход на 8-колоночное расширение. Для внутренних источников имеются входы с приводов CD/DVD (аналоговый, S/PDIF, I^2S) и еще пара аналоговых входов. Отношение сигнал/шум — 90 дБ, динамический диапазон — 94 дБ.

Со временем появились новые модели Sound Blaster Live!, в их числе Live! 1024, Live!5.1 (поддерживает интерфейс EAX 2.0) и Live! 24bit. В них применяются более качественные преобразователи, и они реализуют более сложные алгоритмы обработки сигналов. Карты *Sound Blaster Live124bit,* как и следует из их названия, содержат 24-битные ЦАП и АЦП, работающие на частотах до 96 кГц. Динамический диапазон — более 100 дБ по аналоговым интерфейсам и до 133 дБ — по цифровым. Карта имеет набор аналоговых интерфейсов, позволяющих выводить звук для систем 7.1. С помощью внешнего блока можно подключить цифровые интерфейсы S/PDIF (вход и выход), блок подключается через гнездо микрофонного/линейного входа.

Устройства USB и FireWire

В аудиоустройствах для шин USB и FireWire средствами изохронной доставки потоков данных являются контроллеры этих шин. Заметим, что изохронные передачи более эффективно поддерживаются на FireWire, контроллеры USB (кро-

ме нечасто встречающегося ОНС) требуют для работы значительных ресурсов центрального процессора.
Аудиоустройства (картами их не назовешь) для данных шин выпускаются с широким диапазоном технических характеристик, от простых до профессиональных (это больше относится к FireWire). Примером устройств от Creative Labs является Sound Blaster Live! 24-bit External (24 бит при 96 кГц).

12.3. Интерфейс MIDI

Интерфейс *MIDI* (Musical Instrument Digital Interface — цифровой интерфейс музыкальных инструментов) является последовательным асинхронным интерфейсом с частотой передачи 31,25 Кбит/с. Этот интерфейс, разработанный в 1983 году, стал фактическим стандартом для сопряжения компьютеров, синтезаторов, записывающих и воспроизводящих устройств, микшеров, устройств специальных эффектов и другой электромузыкальной техники. В настоящее время интерфейс MIDI имеют и дорогие синтезаторы, и дешевые музыкальные клавиатуры, применяемые в качестве устройств ввода компьютера. Сообщения MIDI широко используются и для передачи музыкальных записей (на дисках и по сети), и как выходной аудиоинтерфейс игр и прочих «звучащих» приложений. Описание музыкальных фрагментов в формате MIDI очень компактно: минута MIDI (файлы с расширением .MID) может занимать менее 10 Кбайт, в то время как минута оцифрованного звука (файлы с расширением .WAV) с качеством аудио-CD занимает около 10 Мбайт. Однако формат MIDI позволяет воспроизводить лишь звуки, на которые способен синтезатор на исполняющей стороне, в то время как механизм цифрового аудио воспроизводит любые звуки. В отличие от оцифрованного звука при воспроизведении данных MIDI пользователь может независимо изменять тональность (транспонировать музыкальный текст) и темп исполнения, причем без искажения тембра и характера звучания инструментов. Современные процессоры и методы обработки сигналов позволяют транспонировать и оцифрованный звук даже в реальном времени, но все-таки с рядом ограничений.
В *физическом интерфейсе* применяется *токовая петля 5 мА* (возможно до 10 мА) с гальванической (оптронной) развязкой входной цепи. Логическому нулю соответствует наличие тока, логической единице (и покою) — отсутствие тока (в «классической» токовой петле телекоммуникаций — наоборот).
Интерфейс определяет три типа портов: *MIDI-In*, *MIDI-Out* и *MIDI-Thru* (рис. 12.15).

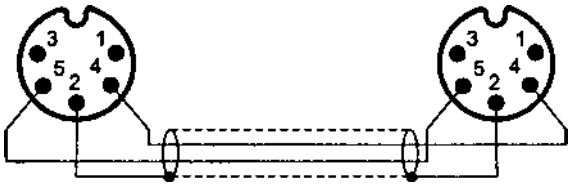


Рис. 12.15. Соединительные кабели MIDI

Входной порт MIDI-In представляет собой вход интерфейса «токовая петля», гальванически развязанного от приемника оптроном с быстродействием не хуже 2 мкс. Устройство отслеживает информационный поток на этом входе и реагирует на адресованные ему команды и данные.

Выходной порт MIDI-Out представляет собой выход источника тока, гальванически связанного со схемой устройства. Ограничительные резисторы предохраняют выходные цепи от повреждения при замыкании на «землю» или источник 5 В. На выход подается информационный поток от данного устройства. При специальной настройке устройства в этом потоке может содержаться и транслированный входной поток, но это нетипично.

Транзитный порт MIDI-Thru служит только для ретрансляции входного потока, по электрическим свойствам он аналогичен выходному. Его наличие обязательным для всех устройств не является.

В качестве разъемов применяются 5-контактные разъемы DIN, распространенные в бытовой звуковой аппаратуре.

На большинстве плат звуковых адаптеров сигналы порта MIDI выведены на неиспользуемые контакты (12 и 15) разъема игрового адаптера (DB-15S). При этом для подключения стандартных устройств MIDI требуется *переходной адаптер*, реализующий интерфейс «токовая петля» (на разъеме карты интерфейс ТТЛ). Переходной адаптер обычно встраивается в специальный кабель, вариант схемы которого приведен на рис. 12.16. Некоторые модели PC имеют встроенные адаптеры и стандартные 5-штырьковые разъемы MIDI. На некоторых системных платах применяются БИС контроллеров интерфейсов, в которых UART-порт, используемый в качестве COM-порта, конфигурированием через CMOS Setup может быть переведен в режим MIDI-порта. Программно MIDI-порт обычно совместим с UART MPU-401 (см. 12.2).

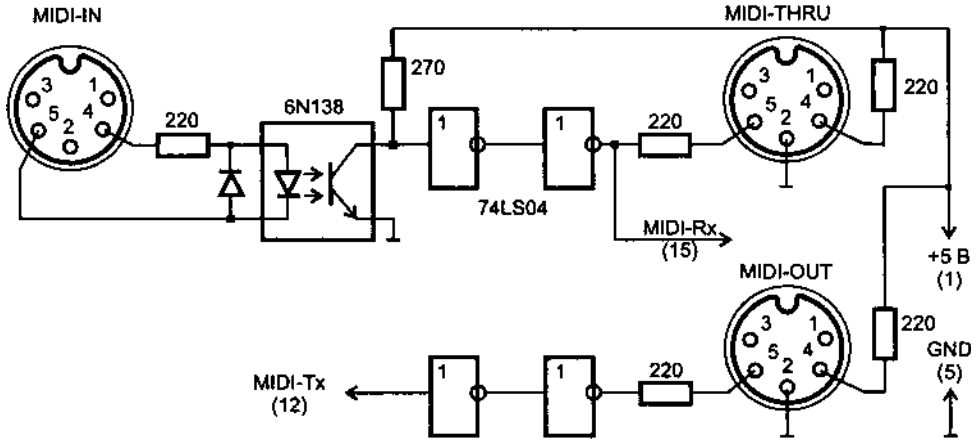


Рис. 12.16. Вариант схемы кабеля-адаптера MIDI

Поток данных MIDI исходит от MIDI-контроллера или от MIDI-секвенсора. *MIDI-контроллер* представляет собой устройство, на котором играют, как на

музыкальном инструменте (чаще всего это клавиатура). MIDI-контроллер формирует поток сообщений, отражающий события исполнения (нажатие и отпускание клавиш) в реальном времени. *MIDI-секвенсор* (sequencer) — устройство, позволяющее перехватывать, хранить и редактировать (включая комбинирование, наложение и генерацию) поток MIDI-сообщений и воспроизводить его в заданном темпе. Исходящий поток MIDI-контроллера или MIDI-секвенсора поступает на его разъем *MIDI-Out.* Для использования в качестве секвенсора компьютер должен иметь порт MIDI и соответствующее прикладное ПО, а также иметь производительность, достаточную для работы в реальном времени.

Конечным приемником потока MIDI-сообщений является *звуковой модуль* (sound module) — синтезатор, формирующий аудиосигнал на основе приходящих к нему команд. Применительно к PC синтезатор может входить в состав звуковой карты, а может быть и внешним.

Каждому звуковому модулю назначается свой номер *логического канала* (MIDI Channel) в диапазоне 1-16.

Порт *MIDI-Thru* позволяет соединять устройства в цепочки и более сложные структуры. Возможные варианты топологии должны подчиняться главному правилу: вход *MIDI-In* одного устройства должен подключаться к выходу *MIDI- Out* или *MIDI-Thru* другого устройства. При планировании MIDI-сети необходимо руководствоваться знаниями информационных потоков и связей устройств. Управляющие устройства — клавиатуры, секвенсоры (в режиме воспроизведения), источники синхронизации — должны находиться, естественно, перед управляемыми. Возможно применение и специальных устройств-мультиплексоров, позволяющих логически коммутировать множество входных потоков в один выходной.

Сообщения MIDI разделяются на канальные (channel messages) и системные (system messages).

Канальные сообщения в свою очередь подразделяются на *голосовые* и *управляющие* (channel mode messages). *Голосовые сообщения* несут основную исполнительную информацию. Исполнение ноты начинается по приему сообщения *Note On,* которое содержит номер ноты и скорость нажатия клавиши. Это сообщение инициирует фазы атаки, спада и удержания (см. рис. 12.5). Скорость нажатия определяет уровень (громкость) исполнения, качественные синтезаторы в зависимости от скорости могут корректировать и характер исполнения (спектр, форму огибающей, выбирать нужный образец). Нота снимается (отрабатывается фаза затухания) по приеме с ее же номером сообщения *Note Off* или *Note On* с нулевой скоростью. Голосовые сообщения управляют и характером исполнения (громкость, вибрато, смещение строя и т. п.). Каждому параметру синтезатора соответствует *номер контроллера* (в данном контексте — номер управляемого параметра), с помощью которого можно им управлять. Эти контроллеры не следует путать с MIDI-контроллерами — устройствами, с помощью которых исполнитель генерирует MIDI-сообщения. *Управляющие сообщения* определяют, как именно звуковой модуль должен отрабатывать голосовые сообщения.

Системные сообщения служат для синхронизации с видеооборудованием и ленточными записывающими устройствами, выбора произведения и позиции воспроизведения, настройки аналоговых синтезаторов. Сообщения реального времени образуют систему синхронизации *MIDI Sync*, используемую секвенсорами, ритм-машинами и другими тактируемыми устройствами MIDI. Есть и специальные сообщения для обмена различными данными, например для загрузки патчей.

Для обеспечения совместимости различных звуковых модулей был принят стандарт *General MIDI* (GM). Он определяет минимальные требования к модулям и оставляет место для расширений, среди которых широко распространены стандарты GS и XG. Стандарт GM поддерживают все звуковые карты, «понимающие» MIDI. Расширение GS поддерживает довольно большое число производителей, а XG — лишь синтезаторы (и звуковые карты) фирм Yamaha и Korg. Для унификации методов хранения и передачи информации MIDI были приняты и стандарты на форматы файлов.

Хранение и передача данных MIDI имеют свою специфику. Когда источником сообщений MIDI является MIDI-контроллер и генерируемый им поток исполняется сразу в реальном времени, никакой маркировки сообщений обычно не требуется. Для хранения данных MIDI в виде файлов нужны специальные отметки времени.

Секвенсоры оперируют *треками* — строками нот, исполняемыми параллельно. Трек примерно соответствует партии (или ее части) одного инструмента. Такое представление естественно для композиторов и дает богатые возможности редактирования. Программный секвенсор может сохранять партитуры как в собственном (ни с кем не совместимым), так и в стандартизованном формате.

При сопряжении устройств MIDI с «инородными» устройствами (видеосистемы, аудиорекордеры) возникает задача их синхронизации. В MIDI используется система MIDI Sync с однобайтными сообщениями SRT. Помимо этой системы в музыкальных инструментах применяется и аппаратная синхронизация Sync24, известная и под названием «DIN Sync», однако несмотря на однотипность разъемов, никакой совместимости интерфейсов MIDI и Sync24 нет. Выделить из потока MIDI сообщения SRT может несложный микроконтроллер (или же аппаратная логика).

В мультимедийных приложениях возникает необходимость синхронизации MIDI- звука с движущимся видеоизображением. Существует несколько систем синхронизации, принятых организацией SMPTE (Society of Motion Picture and Television Engineers — общество инженеров движущихся изображений и телевидения).

Для передачи данных счетчиков SMPTE по интерфейсу MIDI разработана система синхронизации *MTC (MIDI Time Code),* которая представляет собой мост, соединяющий систему синхронизации MIDI, построенную на отметках долей тактов (beat), с данными в форматах SMPTE. Отметим, что синхронизация MIDI Sync является темпозависимой — частота посылки синхросообщений пропорциональна темпу исполнения, который может меняться. Система SMPTE,

а следовательно, и MTC, привязана к абсолютному времени — ее частота сообщений определяется только частотой кадров.

Для *синтезаторов с аналоговым управлением* интерфейс MIDI непосредственно не подходит. Для их подключения существуют специальные микроконтроллеры с ЦАП, формирующие управляющее напряжение (линейное или экспоненциальное) и разрешающий сигнал на основе потока сообщений MIDI, адресованных к выбранному каналу. Встроенное программное обеспечение такого контроллера может реализовать все команды, связанные с изменением частоты тона: вибрато, глиссандо, портаменто и т. п.

Наиболее распространенные MIDI-контроллеры представляют собой 4-6-ок- тавные клавиатуры с полноразмерными (как у рояля) или уменьшенными клавишами. Каждая клавиша имеет датчик нажатия: простейший дискретный (кнопочный), фиксирующий лишь факт нажатия-отпускания, или же динамический, измеряющий скорость (силу удара) и передающий это значение в качестве параметра сообщения. Клавиши простейших клавиатур подпружинены, более качественные «взвешенные» клавиатуры для исполнителя по ощущениям близки к настоящим рояльным. Клавиатура может иметь датчик давления на клавишу после удара, «колеса» (wheels) для управления строем канала, а также подачи сообщений управления контроллерами (например, глубины модуляции). К клавиатуре могут подключаться педаль удержания нот (sustain), органы включения различных эффектов, переключения номера канала. На клавиатуре могут быть установлены небольшой дисплей или отдельные индикаторы режима работы.

Интерфейс MIDI в значительной степени ориентирован на клавишные музыкальные инструменты. Однако даже самая хорошая «взвешенная» клавиатура с датчиками скорости и давления не позволяет передать все нюансы исполнения, например, духовых или струнных (щипковых или смычковых) инструментов. В настоящее время существуют MIDI-контроллеры с интерфейсами, отличными от клавишных, например в виде гитар, где датчики определяют место прижима струн и силу удара (щипка). Есть и MIDI-контроллеры с «духовым интерфейсом» — датчики устанавливаются на клапанах, а специальный мундштук измеряет расход проходящего воздуха. Существуют и преобразователи аналогового сигнала в поток MIDI-инструкций. Они имеют АЦП, оцифровывающий входной сигнал, снятый микрофоном с реального инструмента, и внутренний сигнальный процессор. Для гитар выпускаются специальные звукосниматели, сопряженные с сигнальными процессорами. Задача разложения звука на голосовые (не спектральные!) составляющие довольно сложна и легче решается для одноголосных инструментов. Если преобразователь использовать как устройство ввода для секвенсора, то ошибки преобразования можно исправлять средствами редактирования музыкального текста в секвенсоре.

Самым примитивным MIDI-контроллером может являться обыкновенная компьютерная клавиатура — она способна генерировать сообщения по фактам нажатия и отпускания клавиш. Ценным свойством клавиатуры является способность реагирования на одновременные нажатия-отпускания клавиш, в том числе и во время удержания нескольких клавиш нажатыми. Конечно, она не воспри

нимает динамику удара, и расположение клавиш отнюдь не фортепианное, но при отсутствии настоящей клавиатуры поиграть можно (даже аккордами). Возможность ввода MIDI-команд с клавиатуры имеется во многих программах, работающих с MIDI-синтезатором.

Поток сообщений MIDI проигрывается в компьютере с помощью синтезаторов, аппаратных или программных. Чисто аппаратная реализация MIDI выполняется на звуковых картах с собственным процессором. Этот процессор интерпретирует каждое сообщение MIDI в команды управления синтезатором (FM или WT). Простые карты имеют только сами синтезаторы, а управление ими для интерпретации сообщений MIDI осуществляет программный драйвер карты, исполняемый центральным процессором. Современные процессоры способны поддерживать чисто программную интерпретацию MIDI (Soft MIDI), выполняя и собственно синтез звука в цифровой форме. Правда, при этом занимаются ресурсы (процессорное время и память).

Направление потоков MIDI задается программно (в среде Windows через панель управления, значок Multimedia, вкладка MIDI). В простейшем варианте все сообщения MIDI можно посылать на одно из присутствующих в системе устройств, которые включают аппаратные (FM, WT и иные) синтезаторы установленных звуковых карт, их порты с подключенными дочерними картами, внешние порты MIDI, а также программные синтезаторы. Более сложную конфигурацию можно задать, указав для каждого из 16 каналов MIDI свой синтезатор (порт назначения).

В последнее время в аудиотехнике намечается тенденция к использованию шины USB, которая пригодна как для обмена данными в традиционном виде, так и для изохронной передачи (с равномерной скоростью поступления) аудиоданных в цифровом виде. Интерфейс MIDI имеет ограничение на число каналов (16) и его невысокая скорость передачи ограничивает полифонические возможности (большое число нот в аккорде не может звучать строго одновременно). В то же время производительности современных PC хватает на то, чтобы справляться и с более мощными потоками данных. Для подключения устройств MIDI к компьютеру через USB фирма Roland выпускает 64-канальный процессорный блок S-MPU64, который помимо шины USB имеет 4 входных и 4 выходных порта MIDI. Программное обеспечение допускает объединение до 4 блоков на одной шине USB, что увеличивает число каналов до 256.

ГЛАВА 13

Коммуникационные устройства

Коммуникационные устройства ПК предназначены для обмена данными между компьютерами, компьютером и удаленным устройством ввода-вывода, а также для объединения компьютеров в локальную (Local Area Network, LAN) или глобальную (Wide Area Network, WAN) сеть (включая Интернет). Обмен данными требуется для различных целей: передачи файлов, совместного использования периферийных устройств (например, принтеров), доступа к разнообразным информационным услугам Интернета и частных сетей, приема и передачи факсимильных сообщений, посылки сообщений на пейджеры и мобильные телефоны, установления голосовой связи (IP-телефонии), видеосвязи и даже совместных игр по сети. Современные технологии, используемые для этих целей, рассмотрены в [4], а в этой главе описаны коммуникационные устройства: модемы и адаптеры проводных и беспроводных локальных сетей. Связь между компьютерами — правда, с рядом ограничений может быть установлена и другими средствами: через LPT-порты, последовательные шины FireWire и USB. Конечно, практическую (прикладную) пользу из подключения компьютера к сети можно извлечь только при наличии сетевого программного обеспечения, но его рассмотрение не является темой данной книги.

13.1. Модемы и факс-модемы

Для передачи данных на большие расстояния (в пределах всего мира) издавна используют телефонные сети общего пользования (ТФОП). Однако для непосредственной передачи цифровых данных обычные аналоговые телефонные сети непригодны — требуются модемы на сторонах обоих абонентов. Модем имеет *хост-интерфейс,* которым он подключается к компьютеру (или периферийному устройству), и *интерфейс линии,* согласованный с используемым каналом связи (телефонной линией).

Модем (модулятор-демодулятор) служит для передачи информации на большие расстояния, недоступные локальным сетям, с использованием выделенных и коммутируемых телефонных линий. *Модулятор* поступающую от компьютера двоичную информацию преобразует в аналоговые сигналы с частотной и/ или фазовой модуляцией, спектр которых соответствует полосе пропускания обычных голосовых телефонных линий. *Демодулятор* из этого сигнала извле

кает закодированную двоичную информацию и передает ее в принимающий компьютер.
Факс-модем (fax-modem) позволяет передавать и принимать факсимильные изображения, совместимые с обычными факс-машинами. Передача факсов подразумевает также передачу цифровых данных, хотя «цифра» не видна конечным пользователям: факс-машина сканирует изображение, оцифровывает его (1 бит на точку), сжимает данные и через модем передает в телефонную линию. На приемной стороне выполняются обратные преобразования. Факс-модем работает аналогично, только вместо сканирования его программная поддержка принимает графические или текстовые данные от других программ. Принятые факсы оформляются в виде файлов графических форматов, доступных приложениям для дальнейшей обработки или печати.
Современные модемы имеют ряд дополнительных возможностей, расширяющих сферу их применения. *Голосовой модем* (voice modem) способен преобразовывать звуковой сигнал в цифровой вид, в котором он передается по линии связи. На приемной стороне выполняются обратные преобразования. Аудиосигнал сжимается, например по методу ADPCM (Adaptive Differential Pulse Code Modulation — адаптивная дифференциальная импульсно-кодовая модуляция), или АДИКМ. С помощью голосового модема могут быть реализованы звуковая почта, автоответчик и другие речевые функции. Звуковое сообщение может передаваться по электронной почте или в диалоге реального времени и воспроизводиться голосовым модемом через внутренний динамик, дополнительный телефонный аппарат или мультимедийные средства компьютера (звуковую карту). Средства обработки звуковых сигналов позволяют модему автоматически определять номер вызывающего абонента (АОН), распознавать сигналы тонального набора номера.
Модемы во время сеанса связи могут работать в симплексном, полнодуплексном или полудуплексном режиме. Для повышения эффективной скорости используются различные методы сжатия информации, реализуемые как самими модемами, так и коммуникационным ПО.
В [4] довольно подробно описаны свойства телефонных сетей с точки зрения модемной связи, а также работа аналоговых коммутируемых линий с импульсным и тональным набором, схема телефонного аппарата, сблокированные телефоны и принцип автоматического определения номера. Там же описаны и распространенные стандарты, обеспечивающие совместимость модемов. Здесь ограничимся лишь краткими характеристиками стандартов на модуляцию (табл. 13.1) и отметим, что практически все современные модемы поддерживают стандарт V.90 или V.92, исчерпывающий теоретические возможности обычных телефонных линий. Технология K56flex примерно с теми же параметрами стандартом не стала.

Таблица 13.1. Стандарты на модуляцию

Стандарт	Скорость, бит/с Примечания
Bell 103	300
Bell 212A	1200

Стандарт	Скорость, бит/с	Примечания
V.17	14 400, 1200,9600, 7200,4800	Полудуплекс, Fax Group III (аналоговый), обратно совместим с V.29
V.21	300	Несовместим с Bell 103
V.22	1200	Несовместим с Bell 212A
V.22bis	2400	-
V.23	1200/75	Асимметричный в дуплексном режиме
V.27ter	4800 2400	Полудуплекс, Fax Group III (аналоговый)
V.29	9600, 7200	Полудуплекс, Fax Group III (аналоговый)
V.32	9600, 4800	Дуплекс, дополнительный контроль
V.32bis	14 400, 1200,9600, 7200, 4800	Помехоустойчивый, быстрый
V.32fast	19 200	Расширение V.32Bis
V.34	28 800	-
V.34+	33 600	Расширение V.34
V.90 (x2), V.92	56 000/33 600	Цифровое подключение со стороны АТС
K56flex	56 000/33 600	То же, но не в стандарте
HST	16 800	При дуплексе в обратном направлении скорость 300/450. Удобен для диалога. Используется в U.S. Robotics

Теоретически возможный предел скорости передачи данных через аналоговые телефонные линии достигнут в стандарте V.34+. Здесь скорость изменения сигнала в линии достигает 3429 бод (символов в секунду), и группа символов кодирует группу передаваемых битов. При наилучших условиях 79 битов данных кодируется восемью символами, что при символьной скорости 3429 бод и дает 33 600 бит/с.

В стандарте V.90 (на базе x2) скорость 56 Кбит/с достигается только в направлении к модему при условии, что его партнер по связи (провайдер) подключен к цифровому каналу телефонной сети. В этом направлении из тракта сигнала исключен грубый АЦП телефонной станции (АЦП модемов имеют более высокую разрядность и частоту дискретизации), являющийся источником погрешностей. Телефонная сеть в своей цифровой части обеспечивает передачу 7 бит (отсчетов голоса) с частотой 8 кГц, отсюда и предел в 56 Кбит/с. Стандарт V.92 (развитие V.90) обеспечивает те же предельные скорости, в нем описаны новые средства, позволяющие легче и быстрее установить соединение (Dial-UP), и протокол сжатия данных V.44.

Модемы обеспечивают *коррекцию ошибок* — обнаружение ошибок и организацию повторов на уровне обмена между модемами (протокол V.42, на базе MNP-4). Таким образом, на внешнем интерфейсе модема данные передаются без ошибок.

Для повышения эффективности использования линии модемы выполняют *сжатие данных* (V.42bis или V.44): в линию передаются символы, представляющие сжатый поток данных с интерфейса хоста, а на интерфейс хоста выводятся распакованные данные. Из этого следует, что внешний интерфейс должен

обеспечивать скорость в несколько раз выше битовой скорости в линии (текстовые данные сжимаются до 10 раз).

Модемы для телефонных линий

Функциональная схема аналогового модема с подробностями телефонной части приведена на рис. 13.1. К телефонной линии модем подключается через гнездо RJ-11 «Line» (или «Telco»), телефонные аппараты следует подключать к гнезду «Phone». При работе модема это гнездо отключается от линии, и модем остается единственным устройством, нагружающим линию. Это создает благоприятные условия для настройки модема на линию и позволяет паре модемов выполнить коррекцию характеристик линии.

ВНИМАНИЕ ----------

Если при работе модема к линии остаются подключенными телефонные аппараты, то снятие трубки (и тем более попытки набора номера) могут приводить к снижению скорости и даже разрыву соединений.

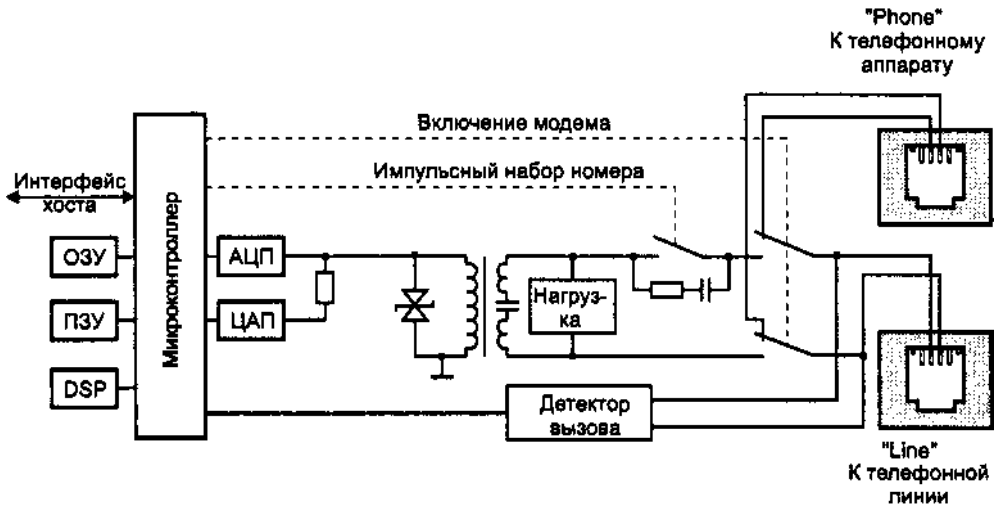


Рис. 13.1. Функциональная схема модема

Модемы, используемые для коммутируемых линий, имеют средства набора номера и определения состояния линии (гудок, занято и т. п.). Набор номера может быть импульсным (pulse dialing) или тональным (tone dialing). В модемах для импульсного набора обычно применяют малогабаритное реле, его характерные щелчки можно услышать при работе модема. Иногда в качестве прерывателя используют электронный ключ (оптрон). При тональном наборе каждая цифра номера кодируется короткими сигналами определенных пар частот, эти «аккорды» можно услышать в телефонной трубке. Цепи сигналов звуковых частот, генерируемых и анализируемых модемом, гальванически развязывают

ся с телефонной линией с помощью трансформатора. Индикатор вызова срабатывает от вызывных импульсов.

Модемы первых поколений имели довольно сложные аналоговые цепи, обеспечивающие требуемые преобразования для модуляции-демодуляции. Управление модемом и некоторые функции протоколов выполнялись микроконтроллером. Современные модемы строятся иначе: аналоговые схемы используются только для телефонной сигнализации, а вся обработка для модуляции-демодуляции выполняется цифровыми методами. Для этого в состав модема входят ЦАП и АЦП. Обработку сигналов в профессиональных модемах выполняет специализированный сигнальный процессор (DSP). Общее управление модемом обеспечивает микропроцессор, в распоряжении которого имеется локальная оперативная память значительного объема. Функции модема определяются возможностями встроенного процессора и его программного обеспечения. Микропрограммное обеспечение модема (firmware) хранится в ПЗУ (EPROM) или флэш-памяти. Такое построение позволяет относительно легко наращивать функциональные возможности модема перезаписью его программного кода. Правда, эти модернизации всегда имеют предел (на каком-то этапе, например, может уже не хватать производительности DSP). Новые версии ПО модема обычно доступны через Интернет (платно или бесплатно).

Более развитые устройства имеют в своем составе оперативную память значительного размера, позволяя в автономном режиме (без компьютера) принимать факсимильные и голосовые сообщения, которые сохраняются для дальнейшей обработки. Такие модемы могут иметь интерфейс для подключения принтера; в результате объединения модема и принтера получается факс-машина.

По конструктивному исполнению модемы для PC делятся на внешние (external) и внутренние (internal).

Внешние модемы, имеющие собственный корпус и блок питания, подключаются кабелем к какому-либо интерфейсу компьютера. Для их установки не требуется вскрытие системного блока. Недостатком является более высокая цена, необходимость отдельного питания и наличие на рабочем месте дополнительного устройства и кабеля. Однако только внешние модемы, как правило, являются «полноценными», то есть содержат микроконтроллер, сигнальный процессор и т. п.

Традиционный интерфейс подключения внешнего модема — COM-порт. В первых модемах выход передатчика (TxD) порта соединялся прямо с аналоговым модулятором, а вход приемника (RxD) — с выходом аналогового демодулятора. COM-порт PC поддерживает только асинхронный режим работы, и в первых протоколах модуляции он и использовался. В современных модемах передача по линии связи происходит в синхронном режиме, хотя и они могут подключаться к COM-порту. В этом нет противоречия, поскольку между внешним интерфейсом современного модема, обращенным к хосту (ПК), и аналоговой телефонной линией находится микроконтроллер модема со своей буферной памятью. Интерфейсы независимы друг от друга, так что в качестве интерфейса подключения к хосту подходит любой интерфейс, способный передавать в обе стороны потоки данных и команд (в том числе и LPT-порт, и шина USB). По

скольку современные модемы обеспечивают сжатие данных, скорость передачи по интерфейсу с хостом может быть в несколько раз выше скорости передачи через линию. При этом интерфейс хоста может оказаться узким местом. Так, при скорости передачи в линии 33 600 бит/с и коэффициенте сжатия 10:1 (конечно, такое высокое сжатие вряд ли окажется постоянным) COM-порт с «потолком» 115 200 бит/с будет сдерживать скорость передачи данных.

Внутренние модемы устанавливаются в слот шины расширения. До недавних пор в основном использовалась шина ISA, теперь эта шина понемногу изживается, и модемы выпускают для шины PCI. Для блокнотных ПК модемы выпускают в виде карт шины PC Card (PCMCIA).

Внутренний модем может быть как аппаратным («честным железным» модемом), так и программным (Win- или Soft-модемом).

Внутренний модем для шины ISA всегда был полноценным, он представлял собой комбинацию COM-порта (UART 16550) и внешнего модема. Его можно «разрезать» по границе интерфейса UART (вставив преобразователи уровня RS-232C-ТТЛ), и получится внешний модем. Устанавливаемый внутренний модем для системы выглядит как еще один COM-порт, у которого базовый адрес регистров (или номер COM-порта) и номер линии запроса прерывания (IRQ) задаются джамперами или переключателями на плате модема. На модеме необходимо установить номер COM-порта, еще не занятого в системе. Как правило, в компьютере имеются два порта — COM1 и COM2 (на системной плате или карте расширения), и модем можно сконфигурировать как COM3 или COM4. Если модем позволяет выбирать лишь между COM1 и COM2, то для него необходимо освободить место — отключить соответствующий порт компьютера или же переместить его на место COM3 или COM4. При отключении портов (особенно джамперами) следует обратить внимание и на освобождение занимаемой ими линии прерывания (особо актуально для карт ISA). Напомним, что портам COM1 и COM3, а также COM2 и COM4 по умолчанию назначаются совпадающие линии запросов прерывания, и этим парам портов приходится использовать разделяемые прерывания (которые на ISA неработоспособны). Модему линия прерывания нужна обязательно. Если к COM-порту подключена мышь, которая без прерывания тоже обходиться не может, то мышь и модем должны быть «разведены» по разным прерываниям. Если мышь стоит на COM1/COM3, то модем нужно ставить на COM2/COM4, и наоборот. Добиться работоспособности в иных комбинациях, в принципе, можно, но слишком хлопотно.

Модем для шины PCI может иметь аналогичную структуру, при этом PCI позволяет устанавливать произвольный базовый адрес UART и линию запроса прерывания (из-за этого возможны проблемы настройки старого коммуникационного ПО).

Мощности современных ПК и пропускной способности PCI достаточно, чтобы решать часть задач управляющего и даже сигнального процессора модема на центральном процессоре. При этом аппаратная часть модема сводится к схеме сопряжения с телефонной линией, ЦАП и АЦП.

Win-модем — карта PCI, выполняющая модуляцию/демодуляцию (с точки зрения ЦП — аппаратно). При этом протоколы модема (команды, коррекция, сжатие) выполняются программно на ЦП. Это вызывает ряд проблем: драйверы неустойчивы, могут оказаться доступными только под Windows (откуда и название). Модем вызывает дополнительную загрузку ЦП, работе модема мешают другие задачи (как и он задачам). *Soft-модем* в этом плане еще хуже, поскольку и модуляция/демодуляция осуществляется программно (на плате модема остается только телефонная часть и АЦП/ЦАП). Win- и Soft-модемы могут быть картами для AMR или CNR (специальный слот на системной плате), связанными со средствами доставки (AC-Link) AC'97.

Преимущества встроенных модемов — низкая цена и отсутствие дополнительных блоков на рабочем месте. Недостатки — необходимость вскрытия системного блока для установки модема и возможные сложности конфигурирования системных ресурсов, а иногда и отсутствие свободного слота. Следует отметить и низкую защищенность компьютера в случае попадания высокого электрического потенциала на телефонный вход модема (например, при ударе молнии в открытую телефонную линию). Правда, если линия не защищена ограничителем перенапряжений, то и внешнее подключение модема не станет надежной защитой компьютера. Недостатки Win- и Soft-модемов отмечены выше.

Модемы для портативных компьютеров имеют интерфейс PC Card (PCMCIA). Ряд моделей позволяют работать с телефонными каналами мобильной связи (старой аналоговой линии NMT-450), имеющими свои специфические особенности. Для мобильных телефонов цифровых сетей (стандартов GSM и CDMA) модем не нужен — требуется лишь интерфейсный кабель подключения к COM- порту или USB. Телефоны с интерфейсом Bluetooth связываются с ПК через соответствующий беспроводной адаптер.

Технологии xDSL и кабельные модемы

Технологии xDSL основаны на превращении абонентской линии обычной телефонной сети из аналоговой в цифровую, что и отражено в их названии (Digital Subscriber Line — цифровая абонентская линия). Общая идея заключается в том, что на обоих концах абонентской линии — на АТС и у абонента — устанавливаются разделительные фильтры, или *сплиттеры* (splitter). Низкочастотная (до 3,5 кГц) составляющая сигнала заводится на обычное телефонное оборудование (порт АТС и телефонный аппарат у абонента), а высокочастотная (выше 4 кГц) используется для передачи данных с помощью xDSL-модемов. Поскольку физическая линия (пара проводов) между абонентом и АТС позволяет пропускать сигнал в полосе даже до 1 МГц, достижимые скорости передачи гораздо выше, чем предел в 56 Кбит/с, установленный и достигнутый для обычных модемов. Высокочастотная часть полосы пропускания сигнала может разделяться между встречными потоками данных различными способами. При частотном разделении каналов (FDM) часть спектра отдается на передачу в одном направлении, часть — в другом. При эхоподавлении (echo-cancellation) вся полоса используется для передачи в обе стороны, а каждое устройство при приеме из общего сигнала вычитает сигнал собственного передатчика. Пропускная

способность может быть как симметричной, так и асимметричной. В случае подключения пользователя к Интернету асимметрия выгодна, поскольку поток к абоненту (страницы текста, аудио- и видеоданные) гораздо больше обратного потока (запросы URL).

- Наибольшее распространение получила асимметричная технология *ADSL* (Asymmetric Digital Subscriber Line), где скорость к абоненту (downstream) достигает 6,1 Мбит/с, от абонента — 16-640 Кбит/с. Достижимая скорость связана с длиной абонентской линии и ее качеством (сечение проводов, материал изоляции, шаг скрутки, однородность и т. п.). Минимальная скорость обеспечивается на линиях длиной до 5,5 км при диаметре провода 0,5 мм (24 AWG) и до 4,6 км при 0,4 мм (26AWG). Скорость 6,1 Мбит/с достигается на линиях длиной до 3,7 км при диаметре провода 0,5 мм и до 2,7 км — при 0,4 мм.
- Технология *UADSL* (Universal ADSL), она же DSL Lite, — улучшенный вариант ADSL с меньшими скоростями (при длине линии до 3,5 км — скорости 1,5 Мбит/с и 384 Кбит/с в разных направлениях; при длине линии до 5,5 км — 640 и 196 Кбит/с). Устройства просты в установке и относительно недороги.
- *RADSL* (Rate Adaptive Digital Subscriber Line) — технология с адаптивным изменением скорости передачи в зависимости от качества линии.
- *HDSL* (High Data-Rate Digital Subscriber Line) — высокоскоростная технология, обеспечивающая скорости 1,536 или 2,048 Мбит/с в обоих направлениях. Протяженность линии — до 3,7 км, требует четырехпроводной линии.
- *SDSL* (Single-Line Digital Subscriber Line) — симметричная высокоскоростная (1,536 или 2,048 Мбит/с) технология, но на двухпроводной линии при длине до 3 км.
- *VDSL* (Very High Data-Rate Digital Subscriber Line) — очень высокоскоростная (до 56 Мбит/с) симметричная технология. Расстояние — до 1,5 км. Технология весьма дорогая, но рассчитана и на коллективное использование линий. После разделяющего фильтра на абонентской стороне может стоять одиночный модем (или концентратор) или группа модемов, подключаемая через специальную кабельную проводку (коаксиальный кабель или витую пару) и разделяющая полосу пропускания предопределенным образом.

Для того чтобы использовать xDSL, провайдер (оператор связи) должен установить свое оборудование на территории АТС обслуживаемого абонента и соединить его с базовой сетью передачи данных каналом достаточной производительности. Конечно, возможны и частные случаи, когда с помощью xDSL объединяются локальные сети в зданиях, охваченных одной АТС. Установка модема xDSL на стороне абонента практически не отличается от установки обычного внешнего модема. Однако технологии xDSL позволяют одновременно и независимо использовать одну и ту же телефонную линию и для передачи данных, и для телефонных переговоров, чего не позволяют обычные модемы для коммутируемых линий.

ВНИМАНИЕ ----------

Сплиттер должен подключаться к витой паре категории 3, приходящей от АТС. Обычная квартирная (и поэтажная) разводка ведется невитым проводом (называемым «лапшой», или «хлоркой»), являющимся для высокочастотного сигнала причиной больших потерь. Для качественной связи следует заменить «лапшу» витой парой до места установки сплиттера (это проще, чем тянуть интерфейсный кабель к ADSL-модему издалека).

Кабельные модемы предназначены для работы через сети кабельного телевидения (Cable Television, CATV), для которых требуется широкополосный коаксиальный кабель с импедансом 75 Ом. Передача данных ведется параллельно с видеовещанием. Эти модемы используют кабельное хозяйство операторов услуг кабельного телевидения. Как и ADSL, кабельные модемы асимметричны: скорость к пользователю может достигать десятков мегабайт в секунду, от пользователя — значительно ниже. Кабельные модемы могут быть и симплексными — модем пользователя только принимает нисходящий (downstream) поток данных от модема оператора кабельного TV. При этом восходящий (upstream) поток данных от пользователя должен передаваться по иным каналам (например, ISDN или аналоговым модемам). Для упрощения структуры коммуникаций (но не оборудования) желательно оба потока передавать по одной и той же кабельной сети. Развитием идеи раздельной передачи потоков является передача нисходящего потока по спутниковым каналам, но пока что это слишком дорогая технология. Кабельные модемы в основном предназначены для предоставления пользователям доступа к Интернету с высокими скоростями получения информации.

Модемы для выделенных линий

Выделенные физические линии имеют гораздо более широкую полосу пропускания, чем коммутируемые. Для них выпускаются специальные модемы, обеспечивающие передачу данных со скоростями до 2048 Кбит/с и на значительные расстояния. Модемы могут работать в синхронном или асинхронном режиме. Асинхронный режим используется на относительно низких скоростях (до 115,2 Кбит/с). В качестве цифровых интерфейсов применяются последовательные — RS-232C, RS-423A, RS-422A, RS-449, RS-485, RS-530, V.35 и др. Есть модемы и с интерфейсом Ethernet — их использование обходится дешевле, чем установка в компьютер специальной карты синхронного последовательного интерфейса (напомним, что стандартный COM-порт синхронный режим не поддерживает). Возможная дальность связи и скорость передачи зависят от типа линии (2-проводная или 4-проводная), диаметра проводников (0,4 мм/ 26AWG или 0,5 мм/24AWG) и способностей модема. Для 4-проводной линии с диаметром проводников 0,4-0,5 мм при скорости 2 Мбит/с достижима дальность 2-2,4 км, при 256 Кбит/с — 9-12 км, при 32 Кбит/с — 15-20 км. Для 2-провод- ной линии при 160 Кбит/с — 4,2-5,6 км, при 144 Кбит/с — 6,5-8,5 км. Данные приведены для модемов Zelax зеленоградского производства, для других модемов цифры могут отличаться. Для этих модемов допустимое напряжение галь

ванической развязки достигает 1500 В. Модемы для выделенных линий заметно дороже массово используемых модемов для коммутируемых линий, зато обеспечивают более высокие скорости передачи и более высокое качество соединений (устойчивость связи).

13.2. Подключение к проводным локальным сетям

Для подключения к проводной локальной сети в компьютере должен присутствовать *сетевой адаптер*, поддерживающий технологию и физический стандарт передачи имеющейся сети. В современных локальных сетях используют различные варианты технологии Ethernet; технологии ARCNet, Token Ring и FDDI встречаются все реже. О сетевых технологиях довольно подробно рассказано в [4]. Из всех разновидностей Ethernet здесь кратко рассмотрим только самые распространенные и перспективные стандарты на витой паре.

Организация сетей Ethernet

Сетевой адаптер подключается к *кабельной сети*, которая по «правилам хорошего тона» должна оканчиваться розетками на рабочих местах пользователей. Кабельная сеть соединяет розетки с активным оборудованием сети. Кабельная сеть может быть и упрощенной: адаптеры (сетевые карты) подключаются кабелями прямо к активному оборудованию. Устройство, к которому подключается множество кабелей от других устройств, удобно называть *концентратором*[1]. Разъем подключения кабеля к этому устройству называют *портом*. Устройство может выполнять функции повторителя, коммутатора или маршрутизатора.

- *Повторитель*, или *хаб* (hub), обеспечивает трансляцию *сигнала* (битового потока), принятого на одном порту, на все остальные порты. При этом в связке повторителей одновременно может происходить только одна передача и полнодуплексный режим работы невозможен. Все узлы сети, подключенные к связке повторителей, находятся в одном *домене коллизий*. Они разделяют общую пропускную способность сети. Формальное ограничение на количество узлов в домене коллизий — до 1024, большое число узлов достигается применением многопортовых повторителей и их каскадным соединением. Однако при числе узлов более 20-30 эффективность сети начинает заметно снижаться.
- *Коммутатор* (switch) транслирует *кадры*, руководствуясь их адресной информацией. В сети на коммутаторах коллизии отсутствуют (каждый порт коммутатора — отдельный домен коллизий). Кадр транслируется только на тот порт, который ведет к его получателю; широковещательные кадры транслируются на все порты, так что коммутаторы объединяют все узлы в домен широковещания. Все узлы, соединенные коммутаторами и повторителями, могут обмениваться кадрами MAC-уровня (по MAC-адресу кадра). Одновре

[1] К сожалению, концентраторы, как правило, отождествляют с хабами.

менно через коммутатор могут проходить несколько передач, если в них задействованы разные порты. Коммутаторы обеспечивают возможность *полнодуплексного режима* (full duplex mode) обмена данными между двумя точками — режима одновременных приема и передачи. В отличие от обычного (полудуплексного) режима, в полнодуплексном режиме коллизий не бывает.

- *Маршрутизатор* (router) транслирует *пакеты* (анализирует заголовки сетевого уровня в кадрах) в соответствии с сетевыми адресами. Каждый порт маршрутизатора может представлять отдельную *локальную сеть* (1Р-под- сеть, для которой задаются адрес и маска).

На интеллектуальных коммутаторах можно строить *виртуальные локальные сети* (Virtual Local Area Network, VLAN), или ВЛС, разбивающие сеть на группы узлов — *домены широковещательных кадров.* Узлы разных групп «не видят» друг друга на MAC-уровне. Связи между VLAN (если необходимо) обеспечиваются дополнительными средствами (вплоть до маршрутизаторов).

ПРИМЕЧАНИЕ ----------
Наиболее эффективны сети, построенные на коммутаторах Ethernet.

Топология сетей Ethernet древовидная, петлевидные связи запрещены — между любой парой узлов должен быть лишь один путь. Исключения из этого правила могут быть только при использовании интеллектуальных коммутаторов. В простейшем случае топология — «звезда», в центре которой находится повторитель (хаб) или коммутатор (рис. 13.2, *а).* Возможно двухточечное соединение пары узлов без применения концентратора (рис. 13.2, *б).*

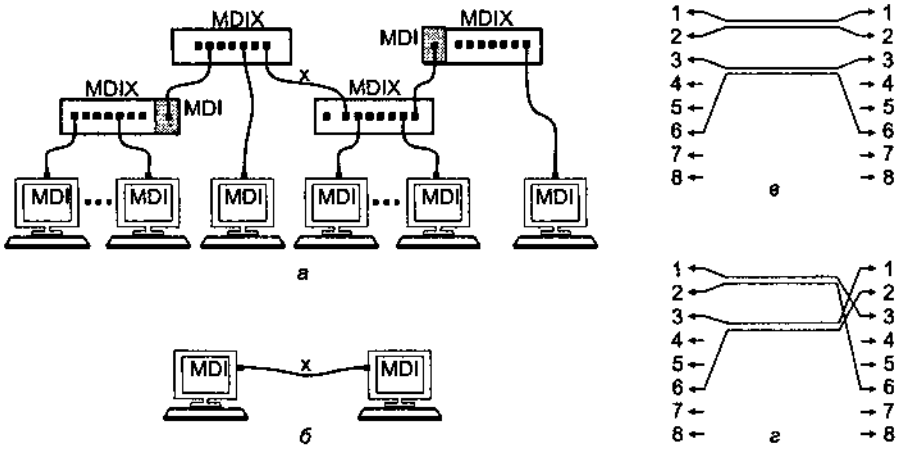


Рис. 13.2. Сеть 10BaseT/100BaseTX: а — звезда, б — двухточечное соединение, *в* — прямой кабель, г — перекрестный кабель

На адаптерах и сетевом оборудовании устанавливаются 8-позиционные модульные гнезда RJ-45 (табл. 13.2). Для соединения адаптера с обычным портом концентратора используется *прямой кабель* (рис. 13.2, в), при непосредствен

ном соединении двух адаптеров (связи пары компьютеров) применяется *перекрестный* (crossover) кабель (рис. 13.2, г). На рисунке перекрестные кабели помечены буквой «х». Минимальная длина кабеля — 2,5 м, максимальная — 100 м.

Таблица 13.2. Интерфейсы 10BaseT, 100BaseTX (разъем RJ-45)

Контакт	Сигнал MDI	Сигнал MDIX
1	Tx+	Rx+
2	Tx–	Rx–
3	Rx+	Tx+
4	Не подключен	Не подключен
5	Не подключен	Не подключен
6	Rx–	Tx–
7	Не подключен	Не подключен
8	Не подключен	Не подключен

В настоящее время используется несколько вариантов Ehernet:

- *10BaseT*— Ethernet на витой паре (Twisted-Pair Ethernet) категории не ниже 3, используется 2 пары проводов. Для топологии соединения действует «правило четырех хабов»: между любыми двумя узлами не должно быть более четырех хабов. Повторители 10BaseT могут подключаться и к коаксиальным сегментам. Для протяженных многосегментных сетей существует ряд топологических ограничений, подробно описанных в [4].
- *100BaseTX* — наиболее популярная версия Fast Ethernet с двумя витыми парами категории 5. Длина витой пары не должна превышать 100 м, хабов может быть не более двух, диаметр домена коллизий — не более 205 м (повторитель класса 1 может быть только один). По использованию разъемов полностью соответствует 10BaseT.
- *100BaseT4* — малораспространенная версия с четырьмя витыми парами категории не ниже 3.
- 1000*BaseT* — Gigabit Ethernet (GE), использует 4 пары категорий 5е и выше (возможно применение и кабеля 5-й категории). Сеть обычно строится на коммутаторах, хотя возможно применение и повторителей (диаметр домена коллизий — не более 200 м).

Для приведенных реализаций Ethernet на витой паре предусмотрен *протокол согласования режимов* (autonegotiation), по которому порт может выбрать самый эффективный из режимов, доступных обоим участникам обмена. В качестве рабочего выбирается самый приоритетный из доступных обоим узлам. Приоритеты режимов в порядке убывания: 1000BaseT, 100BaseTX полнодуплексный, 100BaseT4, 100BaseTX полудуплексный, 10BaseT полнодуплексный, 10BaseT полудуплексный.

Существуют и оптические варианты Ethernet со скоростями 10, 100 и 1000 Мбит/с на многомодовом и одномодовом волокне. Есть устройства, преобразующие тип интерфейса, — медиаконверторы.

Сетевые адаптеры

Сетевые адаптеры, или сетевые интерфейсные карты (Network Interface Card, NIC), предназначены для *передачи и приема кадров* Ethernet. В кадре адаптер имеет дело с MAC-адресами источника (Source Address, SA) и получателя (Destination Address, DA), а также CRC-кодом, с помощью которого контролируется целостность кадра. «Продвинутые» адаптеры интересуются и «начинкой» кадра, что позволяет им осуществлять фильтрацию по типам сетевых протоколов, а также, например, аппаратно проверять (и генерировать) контрольные коды для IP-пакетов. Сетевой адаптер содержит следующие узлы:

- Физический интерфейс подключения (PHY) состоит из разъема и трансформаторов гальванической развязки. Современные адаптеры имеют один разъем RJ-45 для работы на скоростях 10, 100 и даже 1000 Мбит/с; есть карты и с оптическими интерфейсами. Старые адаптеры имели и разъемы BNC (для коаксиального кабеля), а также разъемы AU 1-интерфейса (DB-15) для подключения внешних трансиверов, электрических или оптических.
- MAC-контроллер доступа к среде передачи обеспечивает доступ по методу CSMA/CD или полнодуплексный обмен (с управлением потоком), а также автоматическое согласование режимов.
- Буферная память предназначена для передаваемых и принимаемых кадров.
- Средства доставки обеспечивают передачу данных между буфером кадров и системной памятью компьютера.
- Схема прерываний служит для уведомления ЦП об асинхронных событиях: завершении передачи, приеме кадра и т. п.

При *передаче кадра* (Tx) работа хоста с адаптером выглядит следующим образом. Хост готовит в буфере данных кадр (поля адресов DA, SA и все поля до конца данных) и указывает адаптеру на положение кадра в памяти. MAC-контроллер получает доступ к среде, начинает передачу преамбулы, затем передает тело кадра (из буфера) и контрольный код, вычисленный им по предыдущим полям. В случае обнаружения коллизии он организует повторные попытки. О завершении передачи (успешном или нет) адаптер сигнализирует прерыванием (если оно разрешено), а также установкой соответствующих битов состояния. Отметим, что MAC-адрес источника (SA) в кадре формируется программно, драйвер его может считать из регистра (или энергонезависимой памяти) адаптера или «придумать» сам.

Для *приема кадров* (Rx) хост настраивает фильтр адресов (поле DA) MAC-контроллера. Приемная часть адаптера, просматривая заголовки всех кадров, проходящих в линии, «выуживает» из этого потока кадры, адресованные данному узлу индивидуальным, широковещательным или групповым способом. Можно разрешать прием направленных кадров (по MAC-адресу, записанному в регистр карты), широковещательных кадров и кадров группового вещания (по спискам групп). Возможен и режим приема всех (!) кадров (promiscuous mode). Эти кадры полностью принимаются в буфер и проверяются на отсутствие ошибок (длина кадра, корректность CRC). О приеме корректных кадров уведомляется

центральный процессор. Ошибочные кадры, как правило, игнорируются, хотя адаптер может собирать статистику их появления. На практике попадаются и адаптеры, не обнаруживающие ошибок в поврежденных кадрах. Диагностика сети с таким адаптером непроста.

Сетевой адаптер может вырабатывать *прерывания* по разным событиям: по завершению приема, передачи, обнаружению ошибок и т. п. Число запросов прерываний желательно минимизировать, генерировать прерывания по факту передачи и приема каждого кадра — неэффективно. Время передачи самого короткого кадра Ethernet на скорости 10 Мбит/с составляет около 58 мкс, на скорости 100 Мбит/с — 5,8 мкс, в Gigabit Ethernet — 4 мкс (здесь минимальный размер увеличен, иначе было бы 0,58 мкс)[1]. С таким периодом компьютер (PC-совместимый) прерывания обслуживать не способен.

Эффективная скорость обмена данными по сети сильно зависит от архитектуры сетевых адаптеров. При прочих равных условиях на эту скорость влияют скорость передачи данных между локальной памятью адаптера и системной памятью компьютера, а также возможность параллельного выполнения нескольких операций. В качестве «средств доставки» используются каналы прямого доступа к памяти (DMA), программный ввод-вывод (PIO), прямое управление шиной.

Первые модели сетевых адаптеров, применявшиеся еще с шиной ISA, были довольно примитивными. Буферы принимаемых и передаваемых кадров располагались в локальной памяти адаптера, для ускорения обмена между ними и ОЗУ использовались каналы стандартного контроллера DMA. Стандартные 8-бит- ные каналы прямого доступа шины ISA способны поддерживать скорость до 2 Мбайт/с, 16-битные — до 4 Мбайт/с. Кадр максимальной длины они передают примерно за 1,3 или 2,6 мс соответственно. По сравнению с 12 мс передачи кадра в среде Ethernet это время относительно невелико. Однако для Fast Ethernet, где максимальный кадр в среде передается за 1,2 мс, такая транспортировка оказывается слишком медленной. Более высокую скорость обмена с буфером адаптера обеспечивает режим программного ввода-вывода (PIO), но он полностью загружает центральный процессор на время передачи. Более эффективны интеллектуальные адаптеры, использующие прямое управление шиной (bus mastering) ISA/EISA и относительно высокую скорость (до 8 Мбайт/с ISA 16 бит и до 33 Мбайт/с EISA). Однако для скорости 100 Мбит/с производительности шины ISA уже недостаточно. На сегодняшний день широко применяются адаптеры шины PCI, где для 32-разрядного интерфейса при частоте 33 МГц пропускная способность достигает 132 Мбайт/с. Но для технологии Gigabit Ethernet и этого только-только хватает; правда, у PCI есть резервы — переход на частоту 66 МГц и разрядность 64 бит. Переход на PCI-X и PCI Express решает проблемы пропускной способности шины. Существуют чипсеты системных плат, в которые интегрирован адаптер локальной сети, получающий к тому же, привилегированный канал доступа к памяти.

[1] Время передачи самого длинного кадра (1500 байт) на скоростях 10, 100 и 1000 Мбит/с составляет, соответственно, 1,2 мс, 122 мкс и 12,2 мкс.

Особенно эффективны для шины PCI активные адаптеры, имеющие собственный процессор. Они выполняют передачи на полной скорости PCI, практически не загружая центральный процессор. При этом они работают по *программам*, задаваемым драйвером. Программы передачи выглядят как цепочки дескрипторов, описывающих местоположения буферов (в системной памяти) с заготовленными кадрами. Адаптер сам считывает задание (очередной дескриптор), выполняет передачу и по окончании делает соответствующую пометку в дескрипторе (успех или неудача). Программы приема описывают места, куда будут помещаться принятые пакеты. Адаптер помечает в дескрипторе начало и длину пакета, а также статус приема (успех или ошибка). При приеме и передаче могут возникать специфические ошибки доставки: переполнение (при приеме) или переопустошение (при передаче) FIFO-буферов карты из-за перегрузки шины. Эти ошибки также отмечаются. Драйвер динамически может добавлять задания (дескрипторы) и, естественно, забирать результат. Количество прерываний минимизируется, ими драйверу сигнализируют о необходимости вмешательства (но не по каждому кадру!).

Объем локальной памяти сетевого адаптера менялся от поколения к поколению. Для карт ISA и скорости 10 Мбит/с буферная память имела объем в несколько (даже до 64) килобайт, в ней целиком размещались передаваемые и принимаемые кадры. При этом карты с большим размером буфера обеспечивали выигрыш в производительности.

Для шины PCI при эффективных средствах доставки (интеллектуальное прямое управление шиной) даже при скорости 100 Мбит/с большой буфер не нужен — достаточно небольших FIFO-буферов для приема и передачи. Через них кадры доставляются прямо в/из ОЗУ, что существенно сокращает задержки в их доставке (между линией и драйвером). Типовое решение для каждого буфера FIFO — пинг-понг-буфер, состоящий из двух 64-байтных половинок: пока идет обмен между одной половиной и MAC-узлом, другая половина обменивается с ОЗУ; потом роли меняются. Такое решение позволяет сглаживать задержки предоставления доступа к шине PCI, вызванные активностью других ее абонентов. В числе последних и центральный процессор, имеющий, как правило, более высокий приоритет доступа к PCI. Минимизировать программный доступ процессора к устройствам PCI (тем же сетевым адаптерам) позволяет вышеописанный механизм взаимодействия через системную память. Отработка большого числа прерываний от сетевых адаптеров, особенно с идентификацией причины через чтение регистров устройства PCI, явно не улучшит «условия труда» мастеров шины. Шина PCI (32 бит/33 МГц) «выдерживает» до 4-5 адаптеров Fast Ethernet. Однако неудачная архитектура адаптера (и драйверов) могут негативно сказаться на эффективности сетевого обмена. При больших нагрузках могут теряться принимаемые кадры (в случае передачи только снижается эффективная скорость). Дополнительные задержки обслуживания вносят и *мосты PCI*, которые присутствуют на многопортовых (как правило, четырехпортовых) картах Ethernet. На таких картах установлено четыре MAC-контроллера с интерфейсом PCI на локальной шине, связанной мостом со слотом. Если в компьютер необходимо установить несколько сетевых адаптеров, которые

должны работать с большой нагрузкой, то многопортовый адаптер из-за своего моста может оказаться не лучшим решением, особенно при не очень эффективных MAC-контроллерах. Примером удачной архитектуры можно назвать карты на микросхемах DEC21143, однако их почему-то перестали выпускать.

Карты Gigabit Ethernet для PCI снова стали снабжать буфером значительного размера (например, 256 Кбайт), поскольку рассчитывать на своевременность доставки потока по PCI не приходится. Для этих карт больше подходит интерфейс 66 МГц/64 бит или PCI-X. Включение сетевого адаптера в чипсет системной платы позволяет решить эти проблемы доставки и обойтись без локальной памяти.

Помимо выполнения базовых функций, адаптеры могут обеспечивать дополнительные возможности:

- Поддержка полнодуплексного обмена — одновременного приема и передачи кадров. Для современных карт это естественно (как и поддержка управления потоком в этом режиме по протоколу 802.3x), однако старые карты из-за архитектурных ограничений могли и не поддерживать этот режим.
- «Интеллектуальное» согласование режима (smart autosence) позволяет устанавливать скорость и режим (полный дуплекс) не только по результатам обмена с партнером, но и с учетом качества соединительного кабеля. Сюда же относятся функции автоматического управления назначением контактов разъема (включение внутреннего перекрещивания линий), а также автоматической коррекции полярности сигнала в парах (на случай соединения неправильным кабелем).
- Поддержка маркированных кадров — приоритизация трафика по 802.1p и маркированные ВЛС (Tagged VLAN по IEEE 802.3Q). Поддержка ВЛС позволяет узлу, подключенному одной линией к коммутатору, быть членом нескольких ВЛС, определенных на всей локальной сети.
- Поддержка объединения (Port Trunking) линий (для нескольких карт при соответствующей программной поддержке).
- Поддержка резервирования линий (Resilient Link) — резервный адаптер и линия связи заменяют основной канал в случае его отказа. При этом резервному адаптеру присваивается MAC-адрес основного, чтобы сеть «не заметила» подмены. Резервирование линий должно поддерживаться программными драйверами, чтобы замена происходила прозрачно и для серверных приложений. «Самоизлечивающиеся» драйверы (self-healing drivers) в случае обнаружения проблем функционирования (при «зависании») могут автоматически выполнить сброс и повторную инициализацию адаптера.
- Аппаратный подсчет контрольных сумм IP-пакетов (Fast IP) позволяет ускорять работу протокольного стека TCP/IP.
- Пробуждение по сети (remote wake up, или wake on LAN) с поддержкой интерфейсов DMI и ACPI. Для этого карты имеют специальный дополнительный 3-проводной интерфейс — кабель с коннектором, подключаемый к системной плате. По этому кабелю системная плата с питанием в стандарте

ATX подает дежурное питание (линия +5VSB), даже когда основное питание на системную плату и все устройства не подается. От этой линии питается «дежурная» принимающая схема, которая настроена на прием кадра специфического формата (magic packet) по сетевому интерфейсу. По приеме этого кадра сетевой адаптер через кабель подает пробуждающий сигнал PME на системную плату, которая дает сигнал на включение блока питания, — компьютер включается и загружается ОС с поддержкой DMI. После этого администратор может выполнить все запланированные действия, по окончании которых ОС на компьютере, завершая свою работу, выключает питание.

- Фильтрация трафика группового вещания (multicast filtering).
- Удаленная загрузка (remote boot). На карте размещают микросхему ПЗУ удаленной загрузки (boot ROM), и она становится одним из устройств начальной загрузки. На сервере размещают файлы-образы загрузочных дисков, с которых может удаленно загружать ОС компьютер с ПЗУ удаленной загрузки (он может быть даже бездисковым). В то же ПЗУ иногда включают и антивирусный модуль, контролирующий обращение по записи в системную область жесткого диска (master boot и boot record). Эта антивирусная защита хороша тем, что запускается до загрузки ОС, но только при включенном ПЗУ удаленной загрузки.

Сетевые адаптеры (NIC) для PC выпускаются для шин ISA, EISA, MCA, VLB, PCI, PC Card. Существуют адаптеры, подключаемые к стандартному LPT-пор- ту PC и шине USB, их преимущество — отсутствие потребностей в системных ресурсах (портах, прерываниях и т. п.) и легкость подключения (без вскрытия компьютеров), недостаток — при обмене они значительно загружают процессор и не обеспечивают высокой скорости передачи. Сетевые адаптеры интегрируются и в некоторые модели системных плат.

Адаптеры разных моделей (без учета дорогих карт с оптическим интерфейсом) могут существенно различаться по цене. Их можно разделить на две группы — адаптеры для рабочих станций и адаптеры для серверов, в зависимости от производительности и эффективности. Деление условно — адаптеры для рабочих станций могут иметь черты, относящиеся к серверным. Использовать простые карты в серверах не стоит — они могут стать узким местом сети, «пожирая» ресурсы ЦП.

Интерфейсные карты потребляют *системные ресурсы* компьютера: регистры, приписанные к пространству ввода-вывода или памяти, линии запросов прерываний, адресное пространство для буферной памяти и ПЗУ удаленной загрузки. Карты ISA/EISA пользовались каналами прямого доступа к памяти (DMA 8237A), более совершенные из них задействовали каналы только для получения права управления шиной (для этого пригодны только 16-битные каналы 5-7).

Для карт ISA буферная память (adapter RAM) обычно приписывается к области верхней памяти (UMA), лежащей в диапазоне A0000h-FFFFFh. Память карт PCI может располагаться в любом месте адресного пространства, не занятого оперативной памятью компьютера. Разделяемую память используют не

все модели карт. Теневую память (shadow RAM) и кэширование на область буферной памяти в CMOS Setup задавать нельзя, поскольку ее содержимое при приеме кадра модифицируется неожиданно для контроллера памяти.

Область адресов для Boot ROM (adapter ROM) размером 4/8/16/32 Кбайт всегда находится в диапазоне C0000-DFFFFh (или запрещается). Специально применять для нее теневую память (shadow ROM) в современных компьютерах не требуется: POST при инициализации перенесет ее содержимое в ОЗУ (да и скорость исполнения программы удаленной загрузки не так уж критична).

Под конфигурированием адаптера подразумевается его настройка на использование системных ресурсов PC и выбор среды передачи. Конфигурирование карт выполняется одним из общепринятых способов (от джамперов до автокон- фигурирования шины PCI).

Выбор среды и скорости передачи может быть ручным (программным) или автоматическим. В ряде случаев имеет смысл делать явные назначения, чтобы избегать сюрпризов излишней автоматизации. Автоматическая настройка вносит дополнительные задержки в процесс инициализации (при загрузке) и не со всяким сетевым оборудованием работает корректно. Однако назначение должно быть корректным. В утилитах конфигурирования и на вкладках свойств карты могут предлагаться и дополнительные возможности — оптимизация для клиента или сервера, поддержка модема и некоторые другие. Их настройка должна соответствовать конкретному применению.

ВНИМАНИЕ---------- ----------

Установка полнодуплексного режима допустима только при подключении адаптера к коммутатору или другому адаптеру, также поддерживающему полный дуплекс. Полный дуплекс при подключении к хабу (повторителю) или другому полудуплексному узлу приведет к практической неработоспособности сети из-за постоянных коллизий.

13.3. Подключение к беспроводным сетям (Wi-Fi)

В последнее время возросла популярность *беспроводных локальных сетей* (Wireless LAN, WLAN), в которых физический канал обеспечивается связью в инфракрасном или радиочастотном диапазоне электромагнитных волн. Независимо от физического канала беспроводные сети могут иметь три основных варианта топологии:

- *Неплановая* (*ad-hoc*) *сеть* — группа узлов с беспроводными адаптерами, которая находится в зоне взаимной «видимости» и может без всякого централизованного управления взаимодействовать между собой. В любой момент новый пользователь может как войти (в прямом смысле) в группу, так и покинуть ее. Здесь все узлы равноправны.
- *Сеть с инфраструктурой* — в «поле зрения» группы узлов имеется *точка доступа* (access point), которая централизованно координирует их работу. Здесь точка доступа находится в заведомо привилегированном положении.

- *Сеть с расширенной инфраструктурой* — несколько точек доступа, соединенных между собой (проводной или беспроводной сетью); клиентские узлы находятся в радиусе действия одной или нескольких точек доступа. Клиентские узлы по мере перемещения (или изменения условий приема) могут переходить от одной точки доступа к другой.

Для беспроводных сетей существует ряд стандартных и фирменных решений, их основные характеристики приведены в табл. 13.3. Основным стандартом считается IEEE 802.11, но его исходная версия была несовершенной. Когда альянс WECA (Wireless Ethernet Compatibility Alliance) принял спецификацию IEEE 802.11b, отличающуюся более привлекательными скоростями, был разработан набор тестов на совместимость с ней. Оборудование WLAN, прошедшее эти тесты, получает логотип Wi-Fi (Wireless Fidelity), так что Wi-Fi часто используют как синоним 802.11b. Технологии WLAN используют различные методы кодирования, модуляции и расширения спектра сигналов. Среди перечисленных в таблице в спецификациях IEEE-802.11-FH и HomeRF описан тот же метод перескоков частоты (и те же частоты), что и в Bluetooth.

Таблица 13.3. Параметры технологий WLAN

Название, год	Физическая скорость, Мбит/с	Радиус охвата[1], м	Частота, ГГц
IEEE-802.11 DFIR, 1997	1; 2	10	ИК-лучи
IEEE-802.11-FH, 1997	1; 2	50	2,4
IEEE-802.11-DS, 1997	1; 2	50	2,4
IEEE-802.11b Wi-Fi, 1999, совместим с 802.11-DS	5,5; 11 (1; 2)	100/30	2,4
IEEE-802.11a, Wi-Fi5, 1999	6; 9; 12; 18; 24; 36; 48; 54	50/15	5
IEEE-802.11g, 2003	6; 9; 12; 18; 24; 36; 48; 54	50/15	2,4
HomeRF	1,6	50	2,4
HomeRF 2.0	1,6; 5; 10	50	2,4
MMAC (HiSWANa), HiperLAN/2	6; 9; 12; 18; 24; 36; 48; 54	50/15	5

[1] Максимальное удаление/удаление при максимальной скорости передачи.

Как видно из таблицы, помимо стандарта 802.11 существуют и используются иные фирменные технологии. В таблице указана скорость передачи информации в физическом канале. Полезная скорость передачи данных, достающаяся пользователю, может оказаться значительно ниже: все пользователи одного канала делят его пропускную способность между собой. Кроме того, возникают издержки из-за коллизий и просто сторонних помех приема радиосигнала: поврежденные пакеты (или кадры) приходится повторять.

Адаптер беспроводной сети с точки зрения пользователя отличается от привычного проводного — его также легко установить в компьютер и сконфигурировать, только не надо искать розетку подключения к кабельной сети. Адаптеры беспроводной связи встраиваются практически во все современные блокнотные

ПК; они появляются и в новых системных платах для настольных ПК. Адаптеры беспроводных сетей выпускаются в различных вариантах:

- Адаптеры для настольных ПК — карты PCI, к которым подключается внешняя антенна.
- Адаптеры для блокнотных ПК — карты PC Card, у которых антенна может быть и встроенной. Этот же конструктив используется для сменных приемопередатчиков в ряде моделей точек доступа.

Точки доступа — устройства различного исполнения (настольные, настенные, специальные уличные) — это, как правило, мосты, связывающие WLAN с сетями Ethernet. Специально для точек доступа разработан стандарт IEEE 802.3af — питание к точкам доступа может подаваться по тому же медному кабелю (витая пара), которым они подключаются к коммутаторам или хабам Ethernet. Точки доступа потребляют не так уж много энергии, и проводить к ним питающую сеть не всегда удобно.

Антенны для беспроводных сетей весьма разнообразны по назначению, свойствам и конструкции: направленные и ненаправленные, комнатные и уличные, большие и маленькие. Направленные антенны, как правило, дорогие. Антенны для частот 5 и 2,4 ГГц разные — это следует учитывать при смене технологий. Выбирая подходящий вариант антенны и приемопередатчика, можно построить требуемую беспроводную сеть: обслуживать мобильных пользователей, подключать отдельных пользователей и целые локальные сети к интернет-провайдерам, объединять умеренно удаленные друг от друга локальные сети, не прибегая к дорогостоящей прокладке кабелей или аренде выделенных линий. В большинстве случаев применяются ненаправленные антенны (зона охвата — круг); для организации зон охвата определенной формы используют направленные антенны, у которых зона охвата — сектор; есть и более сложные варианты. Расстановка антенн со сложной диаграммой направленности может оказаться нелегкой задачей, для решения которой потребуется радиоинженер. При двухточечных соединениях радиолинии 802.11 с направленными антеннами позволяют преодолевать расстояние, исчисляемое даже километрами. При множественном доступе сети Wi-Fi работают в радиусе около 100 м, сети Wi-Fi5 высокие скорости (36-54 Мбит/с) обеспечивают в радиусе 15 м, а в радиусе 50 м скорость снижается до 9 Мбит/с.

Стремиться к увеличению чувствительности приемника не всегда полезно: в «густонаселенном» эфире приемник будет принимать сигналы от большего числа узлов (ему не интересных), что приведет к снижению эффективной скорости обмена и увеличению задержек. В нашей стране беспроводные сети пока используются не очень широко. В других странах во многих общественных местах (гостиницы, выставки, вокзалы, аэропорты) работают провайдеры беспроводного доступа к Сети. Правда, цены предлагаемых услуг довольно высоки (например, $10 за час доступа).

Беспроводные сети являются *открытой средой передачи данных*, что порождает определенные проблемы обеспечения *безопасности* (конфиденциальности).

Для решения проблемы применяют различные методы шифрования данных и проверки подлинности, используя ключи доступа, смарт-карты и т. п.

ИМАНИЕ ----------

В поставляемом оборудовании беспроводных сетей параметры, установленные по умолчанию, фактически, отключают средства безопасности. Это позволяет быстро развернуть беспроводную сеть и начать работу. Для обеспечения конфиденциальности пользователь должен сам выполнить необходимую настройку. В ОС Windows параметры безопасности доступны на вкладках окна свойств адаптера беспроводного подключения.

13.4. ПК и Интернет

Персональный компьютер, подключенный к Интернету, становится мощным персональным средством телекоммуникаций. Здесь не будем перечислять всю пользу (и вред), которую можно получить от Сети, а ограничимся технической стороной вопроса.

Варианты подключения

Для подключения отдельного (например, домашнего) компьютера к Интернету необходимо обеспечить его связь с интернет-провайдером (поставщиком услуг). Для такой связи существуют несколько вариантов, различающихся по доступности, пропускной способности, стоимости:

- Подключение через модем по обычной (коммутируемой) телефонной линии — самый массовый и доступный способ. Для этого необходимо установить модем (внутренний или внешний) и настроить интернет-браузер (прикладную программу) на данный тип связи (мастер настройки в Windows задаст все необходимые вопросы). Также нужно заключить договор с провайдером и получить у него номер телефона, по которому модем должен дозваниваться к провайдеру, имя пользователя и пароль. Провайдер, в принципе может находиться в любом месте, но не следует выбирать слишком далекого (в плане телефонной сети) провайдера, тем более иногороднего или зарубежного (будет слишком дорого). Далее при запуске браузера модем автоматически (или с запросом подтверждения) по команде браузера будет дозваниваться до провайдера и устанавливать соединение (на это могут уходить минуты, если, конечно, легко дозвониться до провайдера). После установления соединения можно пользоваться всеми благами Сети. Условия оплаты могут быть различными: фиксированная оплата при ограниченном объеме пересылаемой информации (трафика); повременная оплата подключения; оплата по трафику и их различные комбинации. При повременной оплате не следует забывать разрывать связь по окончании активной работы в Сети (это экономит деньги). У подключения через модем есть ряд недостатков: скорость приема данных из сети не может превышать 56 Кбит/с, а передачи — еще ниже. При плохой АТС пользователя или плохих линиях связи, а также с плохим провайдером связь окажется неустойчивой, соеди

нения будут разрываться, и копирование больших файлов в таких условиях может быть не только длительным, но и невозможным. Во время работы в Сети данной телефонной линией пользоваться для разговоров, естественно, невозможно. Извне дозвониться до увлеченного абонента Сети тоже проблематично (хотя можно по электронной почте послать ему письмо). Абоненты сблокированных телефонов могут испытывать технические трудности подключения, а также сложности в дележе телефонного времени с соседями. Вероятный переход на повременную оплату телефонных разговоров может заметно удорожить этот пока что самый доступный способ подключения.

- Подключение через xDSL-модемы по обычной телефонной линии. Для этого должны быть установлены соответствующие модемы у пользователя и провайдера, но провайдер, кроме того, должен физически находиться на территории АТС, обслуживающей данного пользователя. Такие модемы дороже обычных, но зато обеспечивают более высокую скорость передачи. К тому же модем работает независимо от телефона, подключенного к той же линии. К xDSL-модему можно подключать и небольшую локальную сеть компьютеров (ряд xDSL-модемов подключаются прямо через Ethernet).
- Подключение через выделенную двух- или четырехпроводную телефонную линию с помощью специального модема. Это подключение уже никак не связано с телефоном, здесь используются только телефонные кабели. Подключение может быть организационно сложным, поскольку далеко не всегда в телефонных кабелях имеются свободные пары. Качество и скорость связи, как правило, выше, чем у обычных модемов, но выше и цена оборудования.
- Подключение через сеть кабельного телевидения и кабельный модем. Пока что не очень распространенный способ, поскольку он выгоден провайдеру (владельцу сети кабельного телевидения) лишь при существенном числе абонентов, желающих подключиться к Сети.
- Подключение через цифровую сеть ISDN. Для этого требуются адаптер подключения к ISDN (часто называемый ISDN-модемом) и собственно линия ISDN, проложенная к пользователю. Скорость передачи — 64 или 128 Кбит/с (для абонентов с интерфейсом BRI), но даже для этого начального уровня сеть ISDN — дорогое удовольствие.
- Спутниковое подключение. Провайдер обеспечивает высокоскоростную передачу нисходящего трафика (из Сети к пользователю) через спутник, для приема требуются спутниковая антенна и специальный приемник, подключаемый к компьютеру. Обратный канал организуется одним из традиционных проводных способов (чаще — через коммутируемые телефонные линии).
- Подключение через оптоволоконную линию связи — самое дорогое, но и качественное соединение. Для этого требуется прокладка оптоволоконного кабеля от провайдера до пользователя, причем одному пользователю требуется только пара волокон. Оконечная аппаратура дорогостоящая, но скорость

передачи упирается только в физические возможности провайдера (и финансовые — пользователя).

- Подключение к локальной сети, являющейся IP-подсетью Интернета. Технически это самое простое подключение — необходим сетевой адаптер, подключенный к локальной сети. К адаптеру подключается сетевой протокол IP, назначается IP-адрес, и компьютер становится полноправным членом Сети. Вопросы связи с провайдером ложатся на администратора сети, который должен позаботиться об отделении локальной сети от глобальной маршрутизатором. Связь маршрутизатора с провайдером может выполняться одним из вышеописанных способов.

IP-телефония и передача факсов по IP-сетям

IP-телефония, она же интернет-телефония, означает голосовую телефонную связь между удаленными абонентами с использованием сетей передачи данных, работающих по протоколу IP. Эти два названия можно считать почти синонимами, поскольку в Интернете протокол IP является основным.

В традиционных телефонных сетях цифровой канал для абонента предоставляет полосу 64 (56) Кбит/с в каждом направлении. Такая большая (по меркам передачи данных при удаленном доступе) полоса требуется из-за принятого в телефонии простейшего способа кодирования — ИКМ (PCM). Применение адаптивной дельта-модуляции АДИКМ (ADPCM) позволяет сократить поток до 16 Кбит/с, но сетям передачи данных (учитывая неопределенность задержки и колебания нагрузки) и этот поток трудно выдержать в течение длительного времени. Современные алгоритмы сжатия, реализуемые на достаточно мощных процессорах, позволяют сжимать речевой сигнал до полосы 4-8 Кбит/с с приемлемым качеством. Такой поток уже можно передавать по обычным IP-сетям, что и реализуется в IP-телефонии.

Аудиокодек для IP-телефонии должен решать довольно сложную задачу — обеспечивать значительное сжатие и формировать пакеты данных небольшого размера, а алгоритм упаковки-распаковки должен быть устойчивым к потере отдельных пакетов. Изощренные методы сжатия учитывают специфику речевого сигнала при разговоре: активная речь чередуется с паузами, не несущими информации. Вполне очевидно, что вместо незначащего потока оцифрованной паузы выгоднее передать информацию только об ее длительности. Правда, если в паузе будет полная тишина, то у слушателя может возникнуть ощущение потери соединения. Поэтому пауза на приемной стороне заполняется некоторым «комфортным» шумом, спектральные параметры которого передаются в описателе паузы. Активная речь тоже неоднородна — в ней присутствуют и тональные (вокализированные), и шумовые фрагменты, для которых эффективны различные методы сжатия. Кодер должен отслеживать текущее состояние сигнала и выбирать соответствующий метод представления данного фрагмента. Декодер из простого ЦАП, применяемого при PCM (и с несложными дополнениями для ADPCM), превращается в синтезатор, воссоздающий аудиосигнал из принятого (возможно, и неполного) потока пакетов. В результате всех этих ухищрений удается из исходного равномерного потока 64 Кбит/с получить не

равномерный поток кадров со средней скоростью 4-8 Кбит/с, который нормально проходит через большинство сетей. Для IP-телефонии существуют стандартизованные методы кодирования и сжатия: G.723.1 определяет кодек для скоростей 5,3 и 6,3 Кбит/с, G.729A — для 8 Кбит/с. Наличие этих стандартов обеспечивает совместимость устройств IP-телефонии в международном масштабе, как это давно имеет место в традиционной (телефонной и телеграфной) связи.

Проще всего организовать IP-телефонию между пользователями ПК, имеющими доступ к глобальной IP-сети (Интернету). Для этого достаточно каждый ПК снабдить обычной звуковой картой с наушниками (колонками) и микрофоном. Здесь все задачи могут решаться чисто программно — от звуковых карт требуются только функции обычного кодека (правда, в полнодуплексном режиме, что «умеют» не все карты). Анализ равномерного цифрового потока и упаковка его в кадры, а также обратное декодирование для современных процессоров (особенно с MMX) не является особо обременительной задачей. Установление соединения между двумя IP-узлами — задача тривиальная. Существует ряд программных продуктов (в их числе и NetMeeting от Microsoft), обеспечивающих связь между пользователями ПК. Внешняя оболочка связи может быть различной: например, можно через веб-браузер обращаться за устными консультациями на сайт какой-нибудь фирмы. «Телефонным номером» пользователя ПК будет IP-адрес этого узла.

Однако IP-телефония не ограничивается только диалогами между пользователями ПК — существуют специальные шлюзы для связи с традиционными телефонными сетями и отдельными телефонами. Задачи шлюза несколько сложнее — помимо установления соединения и передачи собственно речи он должен обрабатывать систему сигнализации телефонной системы и преобразовывать ее сигналы в протокольные сообщения IP-телефонии (и обратно). С традиционной телефонией шлюз может контактировать двояко: к нему могут подключаться телефонные аппараты, и он может подключаться к одной или нескольким линиям обычной телефонной сети (местной или городской). Интерфейс сети передачи данных может быть как портом локальной сети (Ethernet), так и портом глобальной сети. В глобальную сеть передачи данных можно выходить и через обычный модем, подключенный к коммутируемой телефонной линии. При этом пользователь может одновременно через одну линию работать в Сети (просматривать веб-сайты) и вести телефонные переговоры. На рис. 13.3 изображена сеть с парой шлюзов, установленных в разных городах (странах) и подключенных к общей сети (Интернету). Здесь абоненты с обычных телефонов, связанных со шлюзами, могут общаться между собой без оплаты междугородных переговоров. Более того, абоненты телефонных сетей, подключенных к шлюзам, могут связываться между собой, набирая вначале номер (городской!) телефона шлюза, а дальше, через номер противоположного шлюза, — номер абонента в его удаленной телефонной сети. Естественно, возможны и связи между «городскими» абонентами и абонентами, подключенными прямо к шлюзам. ПК-пользователи также могут «звонить» через шлюз, зная его IP-адрес. IP-телефония через шлюзы стала конкурировать с традиционной телефонией,

вызывая споры на технические, экономические и правовые темы. На рисунке видно, как трафик проходит «мимо кассы» междугородной телефонной сети. Главный козырь IP-телефонии — низкая себестоимость разговоров, а потому и тарифы значительно более низкие, чем международные и междугородные телефонные. Однако шлюзы являются довольно дорогими и сложными устройствами, которые исполняются как в виде ПК с дополнительными платами адаптеров, так и в виде специализированных устройств. На рисунке показаны и два узла, подключающиеся к сети непосредственно: один на базе ПК, другой — специальный IP-телефон.

Рис. 13.3. Варианты телефонной связи по IP

Для ПК выпускают платы шлюзовых адаптеров — как правило, на 2 или 4 порта с гнездами RJ-11. Порты могут быть либо жестко специализированными (для подключения телефонных аппаратов или для подключения к АТС), либо конфигурируемыми программно. Каждый порт должен иметь по крайней мере кодек G.711 и устройство распознавания и генерации сигналов телефонной сигнализации. В многоканальных системах сжатие обычно выполняется специализированными сигнальными процессорами — мощности одного универсального (x86) центрального процессора может и не хватить. Связь с сетью передачи данных обеспечивается стандартной сетевой картой или адаптером интерфейса глобальных сетей. Программное обеспечение шлюзов должно работать в среде ОС, от которой требуются «умение» действовать в режиме реального времени и, конечно же, устойчивость (шлюз не должен «падать»). Шлюзы выполняют на базе ПК как настольного, так и промышленного исполнения (архитектуры микроРС, с шиной Compact-PCI и т. п.). Специализированные устройства-шлюзы внешне выглядят так же, как обычные сетевые концентраторы, и имеют либо фиксированный набор портов, либо модульную конструкцию, комплектуемую по необходимости. В принципе, шлюзы с достаточным количеством портов могут выступать и в роли У АТС или малых АТС, но для этой роли они все-таки имеют слишком высокую стоимость портов. IP-телефонный аппарат внешне выглядит как многофункциональный кнопочный телефон с дисплеем — микрокомпьютер с сетевым ПО много места не занимает.

Часть IV

Интерфейсы периферийных устройств

ГЛАВА 14

Шины расширения

Шины расширения (expansion bus) ввода-вывода являются средствами подключения системного уровня: они позволяют адаптерам и контроллерам периферийных устройств непосредственно использовать системные ресурсы компьютера — пространство адресов памяти и ввода-вывода, прерывания, прямой доступ к памяти. Устройства, подключенные к шинам расширения, могут и сами управлять этими шинами, получая доступ к остальным ресурсам компьютера. Шины расширения механически реализуются в виде слотов (щелевых разъемов) или штырьковых разъемов; для них характерна малая длина проводников, то есть они сугубо локальны, что позволяет достигать высоких скоростей работы. Эти шины могут и не выводиться на разъемы, а использоваться для подключения устройств в интегрированных системных платах. В истории шин расширения ПК насчитывается уже 3 поколения.

К первому поколению относится ISA — асинхронная параллельная шина с низкой пропускной способностью (единицы мегабайт в секунду), не имеющая средств обеспечения надежности обмена и автоконфигурирования.

Второе поколение началось с шины EISA (а также MCA), за которой последовали шина PCI и ее расширение PCI-X. Это поколение параллельных синхронных надежных шин со средствами автоконфигурирования. Имеются варианты, снабженные возможностью «горячего» подключения-отключения. Скорость передачи достигает единиц гигабайт в секунду. Для подключения большого числа устройств применяется иерархическое объединение шин с помощью мостов в древовидную структуру.

Для третьего поколения (шина PCI Express, она же 3GIO, Hyper Transport, Advanced Switching и InfiniBand) характерен переход от шин к двухточечным соединениям с последовательным интерфейсом; средством объединения множества абонентов являются «коммутационные фабрики». По сути, третье поколение расширения ввода-вывода приближается к сугубо локальным (в пределах системной платы) сетям.

В современных компьютерах основной шиной расширения пока является шина PCI и ее расширение PCI-X; ее дополняет порт AGP. Намечается переход на PCI Express — это средство подключения графического адаптера постепенно вытесняет AGP. Шина ISA из настольных компьютеров уходит, но она сохраняет свои позиции в промышленных и встраиваемых компьютерах как в традиционном слотовом варианте, так и в «бутербродном» варианте PC/104. В блок

нотных компьютерах широко применяются слоты PCMCIA с шинами PC Card и Card Bus, появляется и Express Card. Все эти шины подробно рассматриваются в данной главе. Более полная информация по этим шинам приведена в [7], информацию по шинам ISA можно найти в [6]. Сравнительные характеристики шин расширения PC-совместимых компьютеров представлены в табл. 14.1.

Таблица 14.1. Сравнительные характеристики шин расширения

Шина	Пиковая пропускная способность, Мбайт/с	Каналы DMA	Bus-Master	ACFG[1]	Разрядность данных	Разрядность адреса	Частота, МГц
ISA-8	4	3	–	–	8	20	8
ISA-16	8	7	+	–	16	24	8
LPC	6,7	7	+	–	8/16/32	32	33
EISA	33,3	7	+	+	32	32	8,33
MCA-16	16	–	+	+	16	24	10
MCA-32	20	–	+	+	32	32	10
VLB	132	–	(+)	–	32/64	32	33–50 (66)
PCI	133–533	–	+	+	32/64	32/64	33/66
PCI-X	533–4256	–	+	+	16/32/64	32/64	66–133
PCI Express	496–15872	–	+	+	1/2/4/8/12/16/32	32/64	2,5 ГГц
AGP 1x/2x/4x/8x	266/533/1066/2132	–	+	+	32	32/64	66
PCMCIA	10/20	+	–	+	8/16	26	10
Card Bus	132	–	+	+	32	32	33

[1] Поддержка автоматического конфигурирования. Для ISA PnP является поздней надстройкой, реализуемой адаптерами и ПО.

14.1. Организация шин PCI и PCI-X

PCI и PCI-X — синхронные параллельные шины расширения ввода-вывода, обеспечивающие надежный высокопроизводительный обмен и автоматическое конфигурирование устройств. Шины PCI и PCI-X являются ближайшими «родственниками» с полной взаимной совместимостью устройств. Большинство положений, относящихся к PCI, относится и к PCI-X, так что в дальнейшем описании термин «PC1» в основном относится к обоим вариантам (различия подчеркиваются особо).

Шина PCI позволяет *объединять одноранговые устройства*. Любое устройство шины может выступать как в роли *инициатора транзакций* (задатчика), так и в роли *целевого устройства*. Целевое устройство отвечает на транзакции, адресованные к его ресурсам (областям памяти и портам ввода-вывода). Ядро компьютера (центральный процессор и память) для шины PCI также представляется устройством — *главным мостом* (host bridge). В транзакциях к устройствам PCI, инициированных центральным процессором, главный мост является за-

датчиком. В транзакциях от устройств PCI, обращающихся к ядру (к системной памяти), главный мост является целевым устройством. Право на управление шиной в любой момент времени дается лишь одному устройству данной шины; арбитраж запросов на управление шиной осуществляется централизованным способом. Арбитр, как правило, является частью моста.

Наличие активных устройств (помимо ЦП) позволяет в компьютере выполнять параллельно несколько операций обмена: одновременно с обращениями процессора могут выполняться транзакции от мастеров шины PCI. Эта параллельность — *PCI Concurrency* — возможна лишь для обменов по непересекающимся путям. Одновременный доступ нескольких инициаторов к одному ресурсу (как правило, к системной памяти) требует довольно сложной организации контроллера этого ресурса, но ради повышения суммарной эффективности работы на эти усложнения приходится идти. В системе с несколькими шинами PCI возможно параллельное функционирование устройств-мастеров на разных шинах — *PCI Peer Concurrency.* Однако если они обращаются к одному ресурсу (системной памяти), то какие-то фазы этих обменов все-таки приходится выполнять последовательно.

Каждая *физическая шина PCI* позволяет объединять лишь небольшое число устройств (обычно не более шести). Для увеличения числа подключаемых устройств применяют *мосты PCI* (PCI-to-PCI Bridge) — устройства PCI с парой интерфейсов, которыми шины объединяются в древовидную структуру. В корне этой структуры находится *хост* — «хозяин шины», в обязанности которого входит конфигурирование всех устройств, включая мосты. В роли хоста, как правило, выступает центральный процессор с главным мостом. Мосты позволяют объединять шины PCI и PCI-X с разными характеристиками, а также подключать к PCI/PCI-X иные шины: (E)ISA, MCA, шины блокнотных ПК, PCI Express, Hyper Transport и др.

Шина PCI/PCI-X имеет несколько *вариантов конструктивного оформления*, некоторые из них при наличии специального контроллера допускают «горячую» замену устройств:

- шина объединения компонентов на печатной плате (системной плате или карте расширения);
- слотовые разъемы для установки карт расширения (в конструктивах PC и MCA);
- разъемы для малогабаритных карт расширения (Card Bus, Small PCI, Mini PCI);
- модульные конструктивы для промышленных и инструментальных компьютеров (CompactPCI, PXI).

Важной частью шины PCI является *система автоматического конфигурирования*; конфигурирование выполняется каждый раз при включении питания и инициализации системы. Специальное конфигурационное ПО позволяет обнаружить и идентифицировать все установленные устройства, а также выяснить их потребности в ресурсах (областях памяти, адресах ввода-вывода, прерываниях). Спецификация PCI требует от устройств способности перемещать

все занимаемые ресурсы (области в пространстве памяти и ввода-вывода) в пределах доступного адресного пространства. Это позволяет обеспечить бесконфликтное распределение ресурсов для множества устройств. Одно и то же функциональное устройство может быть сконфигурировано по-разному, отображая свои операционные регистры либо на пространство памяти, либо на пространство адресов ввода-вывода. Драйвер может определить текущую настройку, прочитав содержимое регистра базового адреса устройства. Драйвер также может определить номер запроса на прерывание, который используется устройством. Для конфигурирования устройств существует специальный набор функций PCI BIOS.

Взаимодействие устройств

С программной точки зрения устройство PCI может иметь следующие компоненты:

- конфигурационные регистры, используемые для идентификации и начального конфигурирования устройства при инициализации системы (для всех устройств предусмотрен обязательный набор конфигурационных регистров, остальные регистры могут применяться для текущего управления);
- операционные регистры (необязательные), отображенные на пространство памяти или/и ввода-вывода (эти регистры используются для текущего управления и взаимодействия с устройством);
- локальная память (необязательная), отображенная на выделенные области физических адресов системной памяти;
- источники запросов на прерывания;
- мастер шины, обеспечивающий прямой доступ к системной памяти (DMA) и взаимодействие с другими устройствами.

С устройством PCI, когда оно является целевым, можно взаимодействовать несколькими способами:

- командами *обращения к памяти и портам ввода-вывода*\ эти команды адресуются к областям, выделенным устройству при конфигурировании;
- командами *обращения к конфигурационным регистрам*; эти команды адресуются по *идентификатору* — номеру *шины*, *устройства* и *функции* (компонентам многофункционального устройства PCI);
- специальными *широковещательными сообщениями*, передаваемыми для всех устройств выбранной шины;
- командами *пересылки сообщений*; команды адресуются по идентификатору устройства (эта возможность появилась в PCI-X 2.0).

Для обращений к пространству памяти используется 32- или 64-битная адресация, причем разрядность адресации не зависит от разрядности шины. Таким образом, шина позволяет адресовать до 2^{32} (4 Гбайт) или 2^{64} (более 1,8 x 10^{19}) байт памяти. На шине PCI фигурирует физический адрес памяти. Для адресации портов ввода-вывода используется 32-битная адресация; в компьютерах на базе процессоров x86 из них задействована только 16 младших битов. В системе

адресации ввода-вывода реализована поддержка особенностей, связанных с адресацией портов в PC-совместимых компьютерах с шиной ISA. Для устройств PCI и PCI-X рекомендуется по возможности избегать использования портов ввода-вывода, отображая операционные регистры устройств на пространство памяти (Memory-Mapped I/O).

Конфигурационные регистры устройств PCI расположены в обособленном пространстве адресов (отдельном от пространства адресов памяти и ввода-вывода). Каждому устройству (точнее, каждой функции сложного устройства) выделяется 256-байтный блок конфигурационных регистров; в спецификации PCI-X 2.0 размер блока увеличен до 4096 байт. Частью этого блока является обязательный набор конфигурационных регистров, с помощью которых осуществляются идентификация устройств, их конфигурирование и управление их свойствами. В конфигурационных регистрах, в частности, указываются адреса, отведенные устройству (как целевому), — через них разрешается работа в роли инициатора и целевого устройства; кроме того, через них конфигурируются прерывания. Конфигурационные регистры обеспечивают возможность автоматической настройки всех устройств шины PCI. К этим регистрам система обращается на этапе конфигурирования — переучета обнаруженных устройств, выделения им неперекрывающихся ресурсов (областей памяти и пространства ввода-вывода) и назначения номеров аппаратных прерываний. При дальнейшей регулярной работе взаимодействие прикладного ПО с устройствами осуществляется преимущественно путем обращений по назначенным в процессе конфигурирования адресам памяти и ввода-вывода. Конфигурационные же регистры в регулярной работе используются для системных целей: настройки параметров, описывающих поведение устройства на шине, обработки ошибок, идентификации источника прерываний.

Обращения к регистрам и памяти устройств PCI выполняются командами шины PCI. Команды может подавать любой инициатор — как хост (главный мост) по командам центрального процессора, так и рядовое устройство PCI. Возможность распространения ряда команд зависит от взаимного расположения инициатора и целевого устройства на ветвях дерева шин PCI. Однако хост может безусловно подать любую команду любому устройству PCI. Только хост всегда имеет доступ к конфигурационным регистрам всех устройств (и мостов), поэтому он и должен заниматься конфигурированием. После конфигурирования любое устройство PCI может безусловно обратиться к системной памяти, то есть реализовать *прямой доступ к памяти* (DMA).

Устройства PCI могут вырабатывать *запросы аппаратных прерываний*:

- обычные маскируемые — для сигнализации событий в устройстве; эти прерывания могут сигнализироваться как традиционным способом — по специальным сигнальным линиям, так и передачей сообщений (MSI);
- немаскируемые — для сигнализации о серьезных ошибках;
- прерывания системного управления (System Management Interrupt, SMI) — для сигнализации о событиях в системе управления энергопотреблением

и некоторых системных целей (например, эмуляции работы стандартного контроллера клавиатуры с помощью устройств USB).

Наиболее эффективно возможности шины PCI используются при применении *активных устройств — мастеров шины (PCI Bus Master).* Только эти устройства могут обеспечить скорость передачи данных, приближающуюся к декларированной пиковой пропускной способности. Максимальная производительность обменов по шине PCI достигается только в пакетных транзакциях значительной длины. Транзакции по инициативе программы, исполняемой на ЦП, проводимые главным мостом, как правило, являются одиночными (или очень короткими пакетными). По этой причине программно-управляемый обмен данными с устройствами PCI по производительности значительно уступает обмену, выполняемому устройством-мастером. Таким образом, применение активных устройств дает двойной эффект: разгружает центральный процессор и обеспечивает лучшее использование пропускной способности шины.

Шины, устройства, функции и хост

Каждое устройство PCI при установке в конкретную систему получает *идентификатор*, однозначно определяющий его положение на дереве шин PCI данного компьютера. Идентификатор имеет иерархическую структуру и состоит из номеров *шины* (bus), *устройства* (device) и *функции* (function). Идентификатор задает положение блока конфигурационных регистров заданной функции выбранного устройства в общем конфигурационном пространстве системы. Идентификаторы фигурируют при обращениях к регистрам конфигурационного пространства, а также при обмене сообщениями между устройствами (DIM в PCI-X).

Шина PCI представляет собой набор сигнальных линий, непосредственно соединяющих интерфейсные выводы группы устройств (слотов, микросхем на плате). В системе может присутствовать несколько шин PCI, соединенных *мостами PCI.* Мосты электрически отделяют интерфейсные сигналы одной шины от другой, соединяя шины логически; главный мост соединяет главную шину PCI с хостом (процессором и памятью). Каждая шина имеет свой *номер шины* (PCI bus number). Шины нумеруются последовательно, начиная от хоста; шина PCI, подключенная к главному мосту, имеет нулевой номер.

Устройством PCI называется микросхема, или карта расширения, подключенная к одной из шин PCI и использующая для доступа к конфигурационным регистрам выделенную ей линию IDSEL, принадлежащую этой шине. Устройство может быть многофункциональным, то есть состоять из множества (от 1 до 8) так называемых *функций.* Каждой функции отводится конфигурационное пространство в 256 байт, в PCI-X оно расширено до 4096 байт. Многофункциональные устройства должны отзываться только на конфигурационные циклы с номерами функций, для которых имеется конфигурационное пространство. При этом функция с номером 0 должна присутствовать обязательно (по результатам обращения к ней определяется присутствие устройства), номера остальных функций назначаются разработчиком устройства произвольно (в диапазоне 1-7). Простые (однофункциональные) устройства в зависимости от

реализации могут отзываться либо на любой из номеров функций, либо только на номер функции 0.

Нумерацией и конфигурированием всех устройств PCI занимается *хост* — «хозяин» шины PCI. Роль хоста, как правило, исполняет центральный процессор, связанный с шиной PCI главным мостом, от которого и начинается нумерация шин. Конфигурирование всех устройств шины возможно только со стороны хоста — в этом заключается его особая роль. Ни с одной из шин PCI ни один задатчик не имеет доступа к конфигурационным регистрам всех устройств PCI, без чего полное конфигурирование недоступно. Даже с нулевой шины PCI задатчику недоступны конфигурационные регистры главного моста, а без доступа к ним невозможно запрограммировать распределение адресов между хостом и устройствами PCI. С других шин PCI возможности доступа к конфигурационным регистрам еще скромнее.

Конфигурирование выполняется для каждой *функции*; как уже отмечалось, полный идентификатор функции состоит из трех номеров: шины, устройства и функции. Короткая форма идентификатора вида PC10:1:2 (например, в сообщениях ОС Unix) означает функцию 2 устройства 1, подключенного к главной (0) шине PCI. Диспетчер устройств (конфигурационное ПО) должен оперировать списком всех функций всех устройств, обнаруженных на всех шинах PCI данной системы (компьютера).

В шине PCI принята *географическая нумерация* — номер устройства определяется местом его подключения. *Номер устройства* (device number, dev) определяется той линией шины AD, к которой подключена его линия сигнала IDSEL. В соседних слотах PCI, как правило, задействуются соседние номера устройств; их нумерация определяется разработчиком системной платы (или пассивной кросс-платы в промышленных компьютерах). Часто для слотов используются убывающие номера устройств, начиная с 20 или 15. Группы соседних слотов могут подключаться к разным шинам; на каждой шине PCI нумерация устройств независимая (могут быть и устройства с совпадающими номерами, но разными номерами шин). В устройствах PCI, интегрированных в системную плату, имеет место та же система нумерации. Их номера «запаяны намертво», в то время как номера устройств на картах расширения можно менять, переставляя их в разные слоты.

Одна *карта PCI* может содержать только одно устройство шины, к которой она подключается, поскольку ей в слоте выделяется только одна линия IDSEL. Если на карте размещают несколько устройств (такова, например, 4-портовая карта Ethernet), то на ней приходится устанавливать мост — устройство PCI, к которому и обращаются по линии IDSEL, выделенной данной карте. Этот мост организует на карте дополнительную шину PCI, к которой можно подключить множество устройств. Каждое из этих устройств получит свою линию IDSEL, но относящуюся уже к дополнительной шине PCI данной карты.

С точки зрения обращения к пространствам памяти и ввода-вывода географический адрес (номера шины и устройства) в пределах одной шины безразличен. Однако номер устройства определяет номер линии запроса прерывания, которой может пользоваться устройство. Подробнее об этом см. в 14.5, здесь же

отметим, что на одной шине устройства с номерами, отличающимися друг от друга на 4, используют одну и ту же линию прерывания. В системах с несколькими шинами PCI перестановка устройства в слоты разных шин может влиять на производительность, что связано с характеристиками данной шины и ее удаленностью от главного моста.

Разобраться с нумерацией устройств и полученных ими линий прерываний на конкретной плате можно, если устанавливать карту PCI поочередно в каждый из слотов (отключая питание) и смотреть на сообщения об обнаруженных устройствах PCI, выводимые на дисплей в конце теста *POST*. В этих сообщениях фигурируют и устройства PCI, установленные непосредственно на системной плате (не отключенные параметрами *CMOS Setup).* Однако чтобы не возникло иллюзии простоты, отметим, что «особо умные» операционные системы (например, Windows) не довольствуются полученными назначениями номеров прерываний и изменяют их по своему усмотрению.

Спецификации PCI и PCI-X

Шина PCI (Peripheral Component Interconnect — взаимодействие периферийных компонентов) имеет уже длинную историю.

- PCI 1.0 (1992 г.) — определена общая концепция, описаны сигналы и протокол 32-битной параллельной синхронной шины с тактовой частотой до **33,3** МГц и пиковой пропускной способностью 132 Мбайт/с.
- PCI 2.0 (1993 г.) — введена спецификация коннекторов и карт расширения с возможным расширением разрядности до 64 бит (пропускная способность до 264 Мбайт/с), предусмотрены варианты питания интерфейсных схем напряжением 5 и 3,3 В.
- PCI 2.1 (1995 г.) — введена частота 66 МГц (только для устройств с напряжением питания 3,3 В), что позволило обеспечить пиковую пропускную способность до 264 Мбайт/с в 32-битном варианте и 528 Мбайт/с в 64-битном.
- PCI 2.2 (1998 г.) — уточнения версии 2.1, введен новый механизм сигнализации прерываний — MSI.
- PCI 2.3 (2002 г.) — определены биты для прерываний, облегчающие идентификацию источника; отменены карты расширения с питанием 5 В (остались только универсальные и 3,3 В); введен низкопрофильный (low profile) конструктив карт расширения; добавлены сигналы дополнительной шины SM-Bus. Эта версия, описанная в документе PCI Local Bus Specification, Revision **2.3,** является базой для современных расширений.
- PCI 3.0 (2004 г.) — отменены системные платы на 5 В (остались только универсальные и 3,3 В).

На базе PCI 2.3 в 1999 году появилось *расширение PCI-X*, призванное существенно повысить пиковую пропускную способность шины за счет увеличения частоты передачи, а также повысить эффективность работы за счет оптимизации протокола. В протокол введены расщепленные транзакции и атрибуты, позволяющие участникам транзакции планировать свои действия. Расширение

PCI-X обеспечивает совместимость (механическую, электрическую и программную) устройств и системных плат с обычной шиной PCI, но, естественно, все устройства шины подстраиваются под самого «слабого» участника:

- PCI-X 1.0 — тактовая частота до 133 МГц (для интерфейса на 3,3 В), что дает варианты, называемые PCI-X66, PCI-X100, PCI-X133. Пиковая пропускная способность достигает 528 Мбайт/с в 32-битном варианте и более 1 Гбайт/с — в 64-битном.
- PCI-X 2.0 — введены новые режимы синхронизации с удвоенной (PCI- X266) и учетверенной (PCI-X533) частотами передачи данных относительно тактовой частоты 133 МГц. Столь высокая частота требует низковольтного интерфейса (1,5 В) и режима коррекции ошибок (ECC). Помимо 32- и 64-бит- ных вариантов появился и 16-битный (для встроенных компьютеров). Добавлен новый тип транзакций — сообщения, адресуемые устройству по его идентификатору (DIM). Конфигурационное пространство функции расширено до 4096 байт.

В дополнение к спецификациям шины имеется ряд дополнительных спецификаций на мосты PPB (PCI-to-PCI Bridge), связывающие шины PCI друг с другом, PCI BIOS (конфигурирование устройств PCI и контроллера прерываний), обеспечение «горячего» подключения/отключения устройств (PCI Hot-Plug), управление энергопотреблением.

На базе шины PCI 2.0 фирмой Intel был разработан выделенный интерфейс для подключения графического акселератора AGP (см. 14.9).

Спецификации PCI публикуются и поддерживаются организацией PCI SIG (Special Interest Group, http://www.pcisig.org).

Шина PCI существует в разных конструктивных исполнениях: слоты и карты расширения обычных PC-совместимых компьютеров; Mini-PCI, Small PCI и Card Bus — для малогабаритных компьютеров; Compact PCI (CPCI) и PXI — для промышленных и инструментальных компьютеров. Более подробную информацию обо всех вариантах PCI можно найти в [7].

14.2. Протокол, команды и транзакции шин PCI и PCI-X

Обмен информацией по шинам PCI и PCI-X организован в виде *транзакций* — логически завершенных операций обмена. В каждой транзакции выполняется одна *команда* — как правило, чтение или запись данных по указанному адресу. Транзакция начинается с *фазы адреса*, в которой инициатор задает команду и целевой адрес. Далее могут следовать *фазы данных*, в которых одно устройство (источник данных) помещает данные на шину, а другое (приемник) их считывает. Транзакции, в которых присутствует множество фаз данных, называются *пакетными*. Есть и одиночные транзакции (с одной фазой данных). Транзакция может завершиться и без фаз данных, если целевое устройство (или инициа-

тор) не готово к обмену. В шине PCI-X добавлена *фаза атрибутов*, в которой передается дополнительная информация о транзакции.

Состав и назначение интерфейсных сигналов шины раскрывает табл. 14.2. Состояния всех сигнальных линий воспринимаются по положительному перепаду CLK. В разные моменты времени одними и теми же сигнальными линиями управляют разные устройства шины.

Таблица 14.2. Сигналы шины PCI

Сигнал	Назначение
AD[31:0]	Address/Data – мультиплексированная шина адреса/данных. В начале транзакции передается адрес, в последующих тактах – данные
C/BE[3:0]#	Command/Byte Enable – команда/разрешение обращения к байтам. Команда, определяющая тип очередного цикла шины, задается четырехбитным кодом в фазе адреса
FRAME#	Кадр. Введением сигнала отмечается начало транзакции (фаза адреса), снятие сигнала указывает на то, что последующий цикл передачи данных является последним в транзакции
DEVSEL#	Device Select – устройство выбрано (ответ целевого устройства (ЦУ) на адресованную к нему транзакцию)
IRDY#	Initiator Ready – готовность ведущего устройства к обмену данными
TRDY#	Target Ready – готовность ЦУ к обмену данными
STOP#	Запрос ЦУ к ведущему устройству на остановку текущей транзакции
LOCK#	Сигнал блокировки (захвата) шины для обеспечения целостного выполнения операции. Используется мостом, которому для выполнения одной операции требуется провести несколько транзакций PCI
REQ#	Request – запрос от ведущего устройства на захват шины
GNT#	Grant – предоставление управления шиной ведущему устройству
PAR	Parity – общий бит четности для линий AD[31:0] и C/BE[3:0]#
PERR#	Parity Error – сигнал об ошибке четности (для всех циклов, кроме специальных). Вырабатывается любым устройством, обнаружившим ошибку
PME#	Power Management Event – сигнал о событиях, вызывающих изменение режима потребления (дополнительный сигнал, введенный в PCI 2.2)
CLKRUN#	Clock running – шина работает на номинальной частоте синхронизации. Снятие сигнала означает замедление или остановку синхронизации с целью снижения потребления (для мобильных применений)
PRSNT[1,2]#	Present – индикаторы присутствия платы, кодирующие запрос потребляемой мощности. На карте расширения одна или две линии индикаторов соединяются с шиной GND, что воспринимается системной платой
RST#	Reset – сброс всех регистров в начальное состояние (по нажатию кнопки Reset и при перезагрузке)
IDSEL	Initialization Device Select – выбор устройства в циклах конфигурационного считывания и записи; на эти циклы отвечает устройство, обнаружившее на данной линии высокий уровень сигнала
SERR#	System Error – системная ошибка. Ошибка четности адреса или данных в специальном цикле или иная катастрофическая ошибка, обнаруженная устройством. Активизируется любым устройством PCI и вызывает NMI

Сигнал	Назначение
REQ64#	Request 64 bit — запрос на 64-битный обмен. Сигнал вводится 64-битным инициатором, по времени он совпадает с сигналом FRAME#. Во время окончания сброса (сигналом RST#) сигнализирует 64-битному устройству о том, что оно подключено к 64-битной шине. Если 64-битное устройство не обнаружит этого сигнала, оно должно переконфигурироваться на 32-битный режим, отключив буферные схемы старших байтов
ACK64#	Подтверждение 64-битного обмена. Сигнал вводится 64-битным ЦУ, опознавшим свой адрес, одновременно с DEVSEL#. Отсутствие этого подтверждения заставит инициатор выполнять обмен с 32-битной разрядностью
INTA#, INTB#, INTC#, INTD#	Interrupt A, B, C, D — линии запросов прерывания, чувствительность к уровню, активный уровень — низкий, что допускает совместное использование линий
CLK	Clock — тактовая частота шины. Должна лежать в пределах 20–33 МГц; начиная с PCI 2.1 — до 66 МГц, в PCI-X — до 100 и 133 МГц
M66EN	66MHz Enable — разрешение частоты синхронизации до 66 МГц (на картах контакт на 33 МГц заземлен, на 66 МГц — свободен)
PCIXCAP	Возможности PCI-X: на платах PCI контакт заземлен, на PCI-X133 соединен с землей через конденсатор 0,01 мкФ, на PCI-X66 — параллельной RC-цепочкой 10 кОм, 0,01 мкФ
SDONE	Snoop Done — сигнал завершенности цикла слежения для текущей транзакции. Низкий уровень указывает на незавершенность цикла слежения за когерентностью памяти и кэша. Необязательный сигнал, используется только устройствами шины с кэшируемой памятью. Исключен, начиная с PCI 2.2
SBO#	Snoop Backoff — попадание текущего обращения к памяти абонента шины в модифицированную строку кэша. Необязательный сигнал, используется только абонентами шины с кэшируемой памятью при алгоритме обратной записи. Исключен начиная с PCI 2.2
SMBCLK	SMBus Clock — тактовый сигнал шины SMBus (интерфейс I^2C). Введен, начиная с PCI 2.3
SMBDAT	SMBus Data — последовательные данные шины SMBus (интерфейс I^2C). Введен начиная с PCI 2.3
TCK	Test Clock — синхронизация тестового интерфейса JTAG
TDI	Test Data Input — входные данные тестового интерфейса JTAG
TDO	Test Data Output — выходные данные тестового интерфейса JTAG
TMS	Test Mode Select — выбор режима для тестового интерфейса JTAG
TRST	Test Logic Reset — сброс тестовой логики

В каждый момент времени шиной может управлять только одно ведущее устройство, получившее на это право от арбитра. Каждое ведущее устройство имеет пару сигналов — REQ# для запроса на управление шиной и GNT# для подтверждения предоставления управления шиной. Устройство может начинать транзакцию (устанавливать сигнал FRAME#) только при полученном активном сигнале GNT# и дождавшись отсутствия активности шины. Заметим, что за время ожидания покоя арбитр может «передумать» и отдать управление шиной другому устройству с более высоким приоритетом. Снятие сигнала GNT# не позволяет устройству начать следующую транзакцию и даже может заставить прекратить начатую транзакцию.

Для адреса и данных используются общие мультиплексированные линии AD. Линии C/BE[3:0] обеспечивают кодирование команд в фазе адреса и разрешение байтов в фазе данных. В фазе адреса (начало транзакции) ведущее устройство активирует сигнал FRAME#, передает целевой адрес по шине AD, а по линиям C/BE# — информацию о типе транзакции (команду). Адресованное целевое устройство отзывается сигналом DEVSEL#. Ведущее устройство указывает на свою готовность к обмену данными сигналом IRDY#, эта готовность может быть выставлена и до получения сигнала DEVSEL#. Когда и целевое устройство оказывается готово к обмену данными, оно устанавливает сигнал TRDY#. Данные по шине AD передаются только при одновременном наличии сигналов IRDY# и TRDY#. С помощью этих сигналов ведущее, и целевое устройства согласуют свои скорости, вводя *такты ожидания* (wait states). На рис. 14.1 приведена временная диаграмма обмена, в которой и ведущее и целевое устройства вводят такты ожидания. Если бы они оба ввели сигналы готовности в конце фазы адреса и не снимали бы их до конца обмена, то в каждом такте после фазы адреса передавалось бы по 32 бита данных, что обеспечило бы выход на предельную производительность обмена.

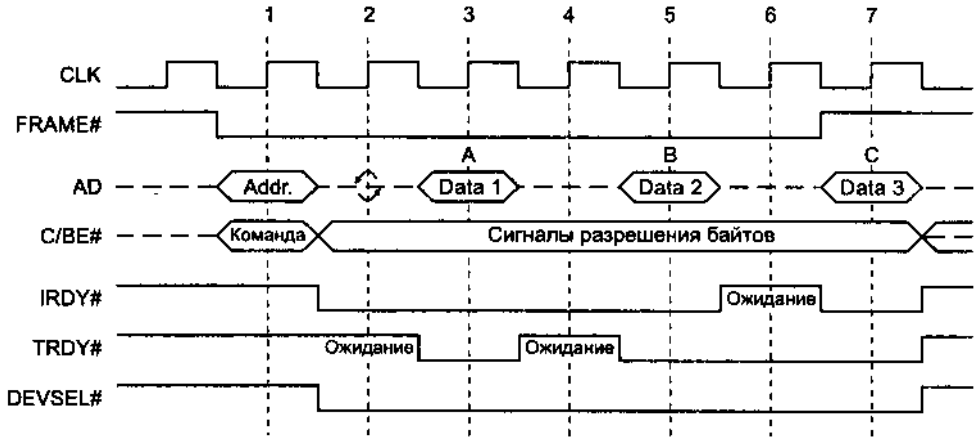


Рис. 14.1. Цикл обмена на шине PCI

На шине PCI все транзакции трактуются как *пакетные*: каждая транзакция начинается фазой адреса, за которой может следовать одна или несколько фаз данных. Количество фаз данных в пакете явно не указывается. Если устройство не поддерживает пакетные транзакции в ведомом режиме, то оно должно потребовать прекращения пакетной транзакции в течение первой фазы данных. В ответ на это ведущее устройство завершает данную транзакцию и продолжает обмен последующей транзакцией со следующим значением адреса. После завершающей фазы данных ведущее устройство снимает сигнал IRDY#, и шина переходит в *состояние покоя* (idle).

Инициатор может начать следующую транзакцию и без такта покоя, такие *быстрые смежные транзакции* (fast back-to-back transactions) могут быть обраще

ны как к одному, так и к разным целевым устройствам. При обмене данными в режиме PCI-X быстрые смежные транзакции недопустимы.

Протокол шины обеспечивает *надежность обмена* — ведущее устройство всегда получает информацию об отработке транзакции целевым устройством. Для *повышения достоверности обмена* применяется механизм контроля четности: линии AD[31:0] и C/BE[3:0]# в фазах адреса и данных защищены битом четности PAR. При обнаружении ошибки устройство вырабатывает сигнал PERR#.

Каждая транзакция на шине должна быть завершена планово или прекращена по инициативе ведущего или целевого устройства.

Ведущее устройство может завершить транзакцию одним из следующих способов.

- *Comletion — нормальное завершение* по окончании обмена данными;
- *Time-out — завершение по тайм-ауту*, если целевое устройство оказалось непредвиденно медленным или запланирована слишком длинная транзакция;
- *Master-Abort — прекращение транзакции* из-за отсутствия ответа от целевого устройства.

Транзакция может быть прекращена *по инициативе целевого устройства* (по сигналу STOP#) по разным причинам:

- *retry — повтор*, целевое устройство из-за внутренней занятости не успевает выдать первые данные в положенный срок (это указание ведущему устройству на необходимость повторного запуска той же транзакции);
- *disconnect — отключение*, целевое устройство не способно своевременно выдать или принять очередную порцию данных пакета (это указание ведущему устройству на необходимость повторного запуска транзакции, но с модифицированным стартовым адресом);
- *target-abort — отказ*, целевое устройство не может обслужить данный запрос (неподдерживаемая команда, фатальная ошибка).

Прекращение типа *retry* служит для организации *отложенных транзакций* (delayed transactions). Отложенные транзакции используются только медленными целевыми устройствами, а также мостами PCI при трансляции транзакций на другую шину. Прекращая (для инициатора) транзакцию условием *retry*, целевое устройство внутренне выполняет данную транзакцию. Когда инициатор повторит эту транзакцию, у целевого устройства (или моста) уже будет готов результат (данные чтения или состояние выполнения записи), который оно быстро вернет инициатору.

Команды шины PCI

Команды PCI определяют направление и тип транзакций, а также адресное пространство, к которому они относятся. Набор команд шины PCI включает следующие:

- *команды чтения и записи портов ввода-вывода* (обращения к пространству портов);

- *команды обращения к памяти* нескольких типов, в том числе к отображенным на память устройствам ввода-вывода и «настоящей» памяти, допускающей предвыборки (чтение строк, множественное чтение, запись с инвалидацией);
- *команды конфигурационных чтения и записи* (обращения к конфигурационному пространству устройств);
- *специальный цикл* (передача широковещательных сообщений);
- *команда подтверждения прерывания* (чтение вектора прерываний).

Для каждого из трех пространств — памяти, портов ввода-вывода и конфигурационных регистров — адресация различна; в специальных циклах адрес игнорируется.

Особенности PCI-X

Протокол шины PCI-X во многом совпадает с PCI, изменения протокола нацелены на повышение эффективности использования тактов шины. В обычной шине PCI все транзакции начинаются одинаково (с фазы адреса) как пакетные с заранее неизвестной длиной. При этом реально транзакции ввода-вывода всегда имеют лишь одну фазу данных; длинные пакеты эффективны только для обращений к памяти (и применяются они именно для этого). В PCI-X транзакции по длине разделены на два типа:

- *пакетные* (burst) транзакции — все команды, обращенные к памяти (кроме команд чтения двойного слова);
- *одиночные* транзакции размером в двойное слово — остальные команды.

В каждой транзакции после фазы адреса присутствует новая *фаза передачи атрибутов* транзакции, в которой инициатор сообщает свой *идентификатор* (номера *шины*, *устройства* и *функции*), *тег, счетчик байтов* и характеристики области памяти, к которой относится транзакция. Идентификатор инициатора вместе с тегом определяют *последовательность* (sequence) — одну или несколько транзакций, обеспечивающих одну *логическую передачу* данных, запланированную инициатором. Каждый инициатор может одновременно выполнять до 32 логических передач. Логическая передача (последовательность) может иметь длину до 4096 байт; в атрибутах каждой транзакции указывается количество байтов, которые должны быть переданы до конца данной последовательности. Общее количество байтов, передаваемых в данной транзакции, заранее не определено.

Характеристики памяти включают флаги R0 и NS:

- R0 (*Relaxed Ordering)* — возможность изменения порядка выполнения отдельных операций записи и чтения;
- NS *(No Snoop)* — область памяти, к которой относится данная транзакция, нигде не кэшируется.

В PCI-X отложенные транзакции заменены *расщепленными транзакциями* (split transactions). Любую транзакцию, кроме всех транзакций записи в память, целевое устройство может завершать либо немедленно (обычным для PCI спосо

бом), либо с использованием протокола расщепленных транзакций. В последнем случае целевое устройство подает сигнал *Split Response* (расщепление), внутренне исполняет команду, а потом инициирует собственную транзакцию (команда *Split Completion)* для пересылки данных или сообщения о завершении инициатору исходной (расщепленной) транзакции. Устройство, вызвавшее расщепленную транзакцию, называется *запросчиком* (Requester), устройство, завершающее расщепленную транзакцию, — *исполнителем* (Completer). Для завершения транзакции исполнитель должен запросить управление шиной у арбитра; запросчик на этапе завершения выступает в роли целевого устройства. *Транзакция завершения* (команда *Split Completion)* во многом напоминает пакетную транзакцию записи, но вместо полного адреса пространства памяти или ввода-вывода по шине AD передаются идентификатор последовательности (с номерами шины, устройства и функции запросчика), к которой относится это завершение, и только младшие 6 бит адреса. По этому идентификатору (номеру шины запросчика) мосты доводят транзакцию завершения до устройства-запросчика. Последовательность может отрабатываться и не одной транзакцией завершения, а серией транзакций.

Запросчик должен быть всегда готов к получению данных начатых им последовательностей, причем данные разных последовательностей могут приходить в произвольном порядке. Исполнитель может выдавать транзакции завершения на несколько последовательностей также в произвольном порядке.

В PCI-X 2.0 вдобавок к вышеописанным изменениям протокола появился новый режим *Mode 2*, отличающийся ускорением блочной записи в память и применением ECC-контроля. Этот режим возможен только при низком (1,5 В) напряжении питания интерфейсных схем. В данном режиме имеет место ECC-контроль адреса и данных (что требует дополнительных тактов). В *транзакциях пакетной записи в память* используется удвоенная или учетверенная скорость передачи данных по отношению к тактовой частоте (режимы PCI-X266 и PCI-X533). В *Mode 2* есть возможность применения 16-битной шины (с 32- или 64-битной адресацией памяти).

В PCI-X 2.0 введена возможность передачи информации (сообщений) устройству, адресуясь с помощью *идентификатора* (номеров шины, устройства и функции). Сообщения передаются последовательностями, в которых используются команды *DIM* (Device ID Message), отличающиеся специфичностью адреса и атрибутов. *Тело сообщения* может иметь длину до 4096 байт. Содержимое тела определяется классом сообщения; класс 0 определяется производителем.

Время выполнения транзакций, таймеры и буферы

Протокол PCI регламентирует время (число тактов), допустимое для различных фаз транзакций. Работа шины контролируется несколькими таймерами, не позволяющими попусту расходовать такты шины и помогающими планировать распределение полосы пропускания.

Каждое целевое устройство должно достаточно быстро отвечать на адресованную ему транзакцию. *Задержка первой фазы данных* (target initial latency) не должна превышать 16 тактов шины. Иногда устройство по своей природе может не успеть уложиться в данный интервал — в этом случае оно должно остановить транзакцию с запросом повтора (retry). Если устройство часто не укладывается в 16 тактов, оно должно выполнять *отложенную транзакцию*. Если целевое устройство не успевает передавать данные каждой фазы за 8 тактов, оно обязано остановить транзакцию. Желательно, чтобы устройство сообщало
о своем «неуспевании» как можно раньше, это экономит полосу пропускания шины.
Инициатор также не должен задерживать поток. Таймер *максимального времени исполнения* задает время (10 мкс), за которое инициатор должен «протолкнуть» хоть одну фазу данных.
В *PCI-X* требования к количеству тактов ужесточились: инициатор не имеет права вводить такты ожидания, целевое устройство имеет право вводить такты ожидания только для начальной фазы данных транзакции.
Право на управление шиной (сигнал GNT#) может быть отобрано у инициатора в любой момент времени. В зависимости от исполняемой команды и состояния сигналов ведущее устройство должно либо прервать транзакцию, либо продолжать ее до запланированного завершения. *Таймер задержки* (master latency timer, или просто latency timer), определяющий поведение мастера при потере права управления, фактически задает ограничение на длину пакетной транзакции и, следовательно, на пропускную способность шины, предоставляемую этому устройству.
При конфигурировании ведущие устройства сообщают свои потребности, указывая максимально допустимую *задержку предоставления доступа к шине* (Max_Lat) и *минимальное время*, на которое им должно предоставляться *управление шиной* (Mi n_GNT). Эти потребности определяются присущим устройству темпом передачи данных и его организацией. Однако будут ли эти потребности реально удовлетворены (по ним должна определяться стратегия арбитража) — не ясно[1].
Для максимального использования возможностей шины устройства должны иметь буферы, чтобы накапливать в них данные для пакетных транзакций. Рекомендуется для устройств со скоростью передачи данных до 5 Мбайт/с иметь буфер по крайней мере на 4 двойных слова. Для более высоких скоростей рекомендуется буфер на 32 двойных слова. Для обмена с системной памятью наиболее эффективны транзакции, работающие с целыми строками кэша, что тоже учитывают при определении размера буфера. Однако увеличение размера буфера может вызвать трудности при обработке ошибок и увеличить задержки в доставке данных (пока устройство не заполнит определенный объем буфера, оно не начнет передачу этих данных по шине, и их потребители будут ожидать).

[1] Автору пока не удалось найти следов управления арбитражем в ОС Windows и UNIX.

В спецификации приводится пример организации карты Fast Ethernet (скорость передачи — 10 Мбайт/с), у которой для каждого направления передачи имеется 64-байтный буфер, разделенный на две половины (ping-pong buffer). Когда адаптер заполняет одну половину буфера приходящим кадром, он выводит в память накопленное содержимое другой половины, после чего половины меняются ролями. Каждая половина выводится в память за 8 фаз данных (около 0,25 мкс на частоте 33 МГц), что соответствует установке MIN_GNT = 1. При скорости прихода данных 10 Мбайт/с каждая половина заполняется за 3,2 мкс, что соответствует установке MAX_LAT = задается в интервалах по 0,25 мкс).

14.1. Прямой доступ к памяти, эмуляция ISA DMA (PC/PCI, DDMA)

Как было сказано выше, шина PCI не предоставляет возможности прямого доступа к памяти с использованием централизованного контроллера в стиле 8237A (как шина ISA). Для разгрузки центрального процессора от рутинных перекачек данных предлагается прямое управление шиной со стороны устройств, называемых *ведущими устройствами*, или *мастерами*, *шипы* (PCI Bus Master). Степень интеллектуальности ведущего устройства может быть разной. В простейшем варианте ведущее устройство обеспечивает пересылку блоков данных между устройством и системной памятью (или памятью других устройств) по указанию от ЦП. Здесь ЦП командами обращения к определенным регистрам ведущего устройства задает начальный адрес, длину блока, направление пересылки и разрешает запуск передачи. После этого пересылка выполняется по готовности (или инициативе) устройства, без отвлечения ЦП. Таким образом выполняется *прямой доступ к памяти* (DMA). Более сложный контроллер DMA может организовывать сцепку буферов при чтении, разбросанную запись и т. п. — возможности, знакомые еще по «продвинутым» контроллерам DMA для ISA/EISA. Более интеллектуальное ведущее устройство, как правило, обладающее собственным микроконтроллером, не ограничивается такой простой работой по указке ЦП — оно выполняет обмены уже по программе своего контроллера. Таким интеллектом обладают, например, хост-контролле- ры последовательных шин USB и IEEE 1394, рассматриваемые в данной книге (см. 17.7 и 18.8).

Для совместимости устройств PCI со старым PC-ориентированным ПО и упрощения устройств PCI фирма Intel разработала специальный *протокол PC/PCI DMA*, позволяющий централизованно эмулировать стандартную (для PC) связку контроллеров DMA 8237. Альтернативное решение — *механизм DDMA* (распределенный доступ DMA) позволяет «расчленить» стандартный контроллер и отдельные его каналы эмулировать средствами карт PCI. Оба этих механизма реализуемы только как часть моста между первичной шиной PCI и шиной ISA, поэтому их поддержка может обеспечиваться (или не обеспечиваться) лишь на системной плате и разрешаться в CMOS Setup.

14.4. Пропускная способность шин PCI и PCI-X

Декларируемая высокая пропускная способность шины достигается только в длинных пакетных циклах, однако пакетные циклы выполняются далеко не всегда. Процессор общается с устройствами PCI инструкциями обращения к памяти или вводу-выводу через главный мост, который шинные транзакции процессора транслирует в транзакции шины PCI. Поскольку у процессоров x86 основные регистры 32-битные, то одна инструкция порождает транзакцию с устройством PCI, в которой передается не более 4 байт данных, что соответствует одиночной передаче. Однако при записи массива данных в устройство PCI (передача с последовательно нарастающим адресом) мост может пытаться организовать пакетные циклы. Пакетные циклы записи можно наблюдать, например, передавая массив данных из ОЗУ в устройство PCI строковой инструкцией MOVSD, используя префикс повтора REP. Тот же эффект даст и цикл последовательных операций LODSW, STOSW (и иных инструкций обращения к памяти). Однако если пересылка данных организуется директивой языка высокого уровня, которая ради универсальности работает гораздо сложнее вышеприведенных ассемблерных примитивов, транзакции, скорее всего, будут уже одиночными. Что касается чтения из устройства PCI, то здесь пакетный режим организовать сложнее. Посмотреть, каким образом происходит обращение к устройству, несложно при наличии осциллографа: в одиночных транзакциях сигнал FRAME# активен в течение всего одного такта, в пакетных он длиннее.

Стремиться к пакетной передаче транзакций записи стоит только в том случае, если устройство PCI поддерживает пакетные передачи в ведомом (target) режиме. Если это не так, попытка пакетной передачи приведет даже к небольшой потере производительности.

При одиночных транзакциях на стандартной шине PCI достижима максимальная скорость чтения 33 Мбайт/с, скорость записи может достигать 66 Мбайт/с. Скорость, соизмеримую с максимальной пиковой, можно получить только при пакетных передачах. При длине пакета 16 байт (4 фазы данных) достижима скорость чтения 76 Мбайт/с и скорость записи 106,6 Мбайт/с. При шестнадцати фазах данных скорость чтения может достигать 112 Мбайт/с, а записи — 125 Мбайт/с. В этих выкладках не учитываются потери времени, связанные со сменой инициатора.

Итак, для выхода на максимальную производительность обмена устройства PCI сами должны быть ведущими устройствами шины, причем способными генерировать пакетные циклы. Радикально повысить пропускную способность позволяет переход на частоту 66 МГц и разрядность 64 бита, что обходится недешево. Для того чтобы на шине могли нормально работать устройства, критичные ко времени доставки данных (сетевые адаптеры, устройства, участвующие в записи и воспроизведении аудио- и видеоданных, и др.), не следует пытаться «выжать» из шины ее декларированную полосу пропускания полностью. Перегрузка шины может привести, например, к потере пакетов из-за несвоевремен

ности доставки данных. Заметим, что адаптер Fast Ethernet (100 Мбит/с) в полудуплексном режиме занимает полосу около 13 Мбайт/с (10 % декларируемой полосы обычной шины), а в полнодуплексном — уже 26 Мбайт/с. Адаптер Gigabit Ethernet даже в полудуплексном режиме вписывается в полосу шины уже с натяжкой (он «выживает» лишь за счет больших внутренних буферов), для него больше подходят 64 бит и 66 МГц. Существенное повышение пиковой скорости и эффективной пропускной способности дает переход на PCI-X с более высокими тактовыми частотами (PCI-X66, PCI-X100, PCI-X133) и быстрой записью в память (PCI-X266 и PCI-X533).

Говоря о пропускной способности шины и эффективной скорости обмена с устройствами PCI, следует помнить об издержках, вносимых дополнительными мостами PCI/PCI. Устройство, находящееся на дальней шине, получит меньшую пропускную способность, чем находящееся сразу за главным мостом устройство, для которого справедливы вышеприведенные рассуждения. Это обусловлено механизмом работы моста — транзакции через мост выполняются поэтапно.

14.5. Прерывания PCI — INTx#, PME#, MSI и SERR#

Устройства PCI имеют возможность сигнализировать об асинхронных событиях с помощью прерываний. На шине PCI возможны четыре типа сигнализации прерываний:

- традиционная проводная сигнализация по линиям INTx;
- проводная сигнализация событий управления энергопотреблением по линии PME#;
- сигнализация с помощью сообщений — *MSI*;
- сигнализация фатальной ошибки по линии SERR#.

В первых версиях (до PCI 2.2 включительно) не было общепринятого способа программной индикации и запрета прерываний. В PCI 2.3 в регистре состояния конфигурационного пространства устройства (функции) определен бит, по которому ОС может определить, что данная функция вызвала прерывание; в регистре команд определен бит запрета прерывания.

Традиционные прерывания PCI — INTx#

Для устройств PCI выделяются четыре проводных линии запросов (IRQX, IRQY, IRQZ, IRQW), соединяемых с контактами INTA#, INTB#, INTC# и INTD# всех слотов PCI с циклическим смещением цепей (см. рис. 4.4 на стр. 114). Мосты PCI просто электрически соединяют одноименные линии INTx своих первичных и вторичных шин. В системах с APIC, в которых число входов запросов увеличено до 24, дополнительные восемь входов могут использоваться периферийными устройствами, установленными на системной плате. На слотах PCI остаются доступными лишь четыре обычных линии запросов.

Устройство PCI вводит сигнал прерывания *низким* уровнем (выходом с открытым коллектором или стоком) на выбранную линию INTx#. Этот сигнал должен удерживаться до тех пор, пока программный драйвер, вызванный по прерыванию, не сбросит запрос прерывания, обратившись по шине к данному устройству.

Линии запросов от слотов и устройств PCI системной платы коммутируются на входы контроллеров прерываний относительно произвольно. Конфигурационное ПО может определить и указать занятые линии запросов и номер входа контроллера прерываний обращением к конфигурационному пространству устройства (см. 14.7).

Драйвер (или иное ПО), работающий с устройством PCI, определяет *номер входа контроллера прерывания*, доставшийся устройству (точнее, функции), чтением конфигурационного регистра Interrupt Line.

Назначение прерываний устройствам (функциям) выполняет процедура POST. Параметрами CMOS Setup (PCI/PNP Configuration) пользователь задает номера запросов прерываний, доступных шине PCI. POST определяет соответствие линий INTA#...INTD# номерам запросов контроллера и соответствующим образом программирует коммутатор запросов. Влияние на аппаратную платформу новых версий ОС настолько велико, что они позволяют себе управлять коммутатором запросов прерываний. Эту возможность можно запретить или разрешить, например, в ОС Windows снятием или установкой флажка Использовать управление IRQ (PCI Interrupt Steering) в окне свойств шины PCI (Панель управления ► Системные устройства ► Шина PCI).

В PCI BIOS имеются функции определения возможностей и конфигурирования прерываний. С их помощью для каждого устройства (на каждой шине) можно определить, с какими входами контроллера прерываний (IRQx) могут быть связаны его линии INTx и с каким именно входом шина связана в данный момент, а также какие входы IRQx отводятся исключительно шине PCI. Есть функция программирования коммутатора запросов, которая может использоваться только конфигурационным ПО (BIOS, ОС), но никак не драйвером устройства.

Сигнализация событий управления энергопотреблением — PME#

Линия PME#, введенная в PCI 2.0, служит для сигнализации в системе управления энергопотреблением (Power Management, PM) — для смены состояния устройств, генерации пробуждения системы по событию. Эта линия электрически доступна всем устройствам PCI и никак не обрабатывается мостами, а лишь доводится до всех абонентов. Обработчик прерывания от PME# может выявить устройство, подавшее сигнал, путем программных обращений к конфигурационным регистрам всех устройств, способных к генерации этого сигнала.

Прерывания сообщениями — MSI

На шине PCI имеется прогрессивный механизм оповещения об асинхронных событиях, основанный на *передаче сообщений MSI* (Message Signaled Inter

rupts). Здесь для сигнализации запроса прерывания устройство запрашивает управление шиной и, получив его, посылает сообщение. Сообщение выглядит как обычная запись двойного слова в ячейку памяти; *адрес* (32-битный или 64-битный) и *шаблон сообщения* на этапе конфигурирования устройств записываются в конфигурационные регистры устройства (точнее, функции). В сообщении старшие 16 бит всегда нулевые, а младшие 16 бит несут информацию об источнике прерывания. Устройство (функция) может нуждаться в сигнализации нескольких типов запросов; в соответствии с его потребностями и своими возможностями система указывает устройству (функции), сколько различных типов запросов оно может вырабатывать.

Прерывания через MSI от одних устройств в одной системе могут генерироваться наряду с обычными прерываниями INTx# от других устройств. Но каждое устройство (функция), генерирующее прерывания через MSI, не должно использовать прерывания через линии INTx#.

Механизм MSI может применяться на системных платах, имеющих «продвинутый» контроллер прерываний APIC. Правда, не все преимущества MSI реально используются. Так, для системных плат на чипсетах с хабом ICH2 и ICH3 фирмы Intel поддержка MSI сводится к организации альтернативных путей подачи запросов IRQ[1:23] на входы APIC (запросы IRQ с номерами 0, 2, 8 и 13 через MSI не передаются). Всем устройствам PCI назначается один и тот же адрес сообщений (Message Address = FEC00020h), по которому в APIC находится регистр IRQ Pin assertion. В сообщении указывается номер взводимого запроса прерывания в диапазоне 1-23 (исключая 2, 8 и 13). Линии запросов для прерываний через MSI не могут использоваться совместно (разделяемо) с прерываниями, полученными другими способами (по линиям запросов от устройств PCI и от других устройств системной платы). Возможно, на других платформах прерывания через MSI используются более эффективно.

14.6. Мосты PCI и PCI-X

Мосты PCI (PCI bridge) — специальные аппаратные средства соединения шин PCI (и PCI-X) между собой и с другими шинами. *Главный мост (host bridge)* используется для подключения PCI к «центру» компьютера (системной памяти и процессору). «Почетной обязанностью» главного моста является генерация обращений к конфигурационному пространству под управлением центрального процессора, что позволяет хосту (центральному процессору) выполнять конфигурирование всей подсистемы шин PCI. В системе может быть несколько главных мостов, что позволяет предоставить высокопроизводительную связь с центром большему числу устройств (число устройств на одной шине ограничено). Из этих шин одна назначается условно главной (bus 0).

Одноранговые мосты PCI *(Peer-to-Peer bridges)* используются для подключения дополнительных шин PCI. Эти мосты всегда вносят дополнительные накладные расходы в передачу данных, так что эффективная производительность при обмене устройства с центром снижается с каждым встающим на пути мостом.

Для подключения шин PCMCIA, CardBus, MCA, ISA/EISA, X-Bus и LPC используются специальные мосты, входящие в чипсеты системных плат или же являющиеся отдельными устройствами PCI (микросхемами). Эти мосты выполняют преобразование интерфейсов соединяемых ими шин, синхронизацию и буферизацию обменов данных.
Применение мостов PCI предоставляет следующие возможности:

- увеличение возможного числа подключенных устройств, преодолевая ограничения электрических спецификаций шины;
- разделение устройств PCI на сегменты — шины PCI — с различными характеристиками разрядности (32/64 бит), тактовой частоты (33/66/100/133 МГц), протокола (PCI, PCI-X Mode 1, PCI-X Mode 2, PCI Express), причем на каждой шине все абоненты равняются на самого слабого участника (правильная расстановка устройств по шинам позволяет с максимальной эффективностью использовать возможности устройств и системной платы);
- организация сегментов с «горячим» подключением/отключением устройств;
- организация одновременного параллельного выполнения транзакций от инициаторов, расположенных на разных шинах.

Каждый мост PCI соединяет только две шины — *первичную* (primary bus), находящуюся ближе к вершине иерархии, со *вторичной* (secondary bus); интерфейсы моста, которыми он связан с этими шинами, называются, соответственно, первичным и вторичным. Допускается только чисто древовидная конфигурация, то есть две шины соединяются друг с другом лишь одним мостом и нет «петель» из мостов. Шины, подсоединяемые ко вторичному интерфейсу данного моста другими мостами, называются *подчиненными* (subordinated bus). Мосты PCI образуют иерархию шин PCI, на вершине которой находится *главная шина* с нулевым номером, подключенная к главному мосту. Если главных мостов несколько, то из их шин (равных друг другу по рангу) условно главной становится шина, которой назначен нулевой номер.
Каждый мост программируется — ему указываются диапазоны адресов в пространствах памяти и ввода-вывода, отведенные устройствам его вторичной и подчиненных шин. В каждом мосте определяется по одной области для адресов пространства ввода-вывода, «настоящей» памяти (допускающей предвыборку) и памяти для отображения регистров ввода-вывода. Если адрес ЦУ текущей транзакции на одной шине (стороне) моста относится к шине противоположной стороны, мост *транслирует транзакцию* на соответствующую шину и обеспечивает согласование протоколов шин. Таким образом, совокупность мостов PCI выполняет *маршрутизацию* (routing) обращений по связанным шинам.
Если в системе имеются несколько главных мостов, то сквозная маршрутизация между устройствами разных шин может оказаться невозможной: главные мосты могут быть связаны друг с другом лишь через магистральные пути контроллера памяти. Поддержка трансляции всех типов транзакций PCI через главные мосты в этом случае становится чересчур сложной, а потому спецификацией PCI строго и не требуется. Таким образом, все активные устройства всех шин PCI могут обращаться к системной памяти, но возможность одноран

гового общения может оказаться в зависимости от принадлежности этих устройств той или иной шине PCI. Передача *сообщений* (команда DIM) с шины на шину через главные мосты реализуется проще, чем передача транзакций всех типов. Для главных мостов PCI поддержка передачи сообщений желательна, но не обязательна.

Мост должен выполнять ряд обязательных функций:

- Обслуживать шину, подключенную к его вторичному интерфейсу:
  - выполнять арбитраж — прием сигналов запроса REQx# от ведущих устройств шины и предоставление им права на управление шиной сигналами GNTx#;
  - парковать шину — подавать сигнал GNTx# какому-то устройству, когда управление шиной не требуется ни одному из задатчиков;
  - генерировать конфигурационные циклы типа 0 с формированием индивидуальных сигналов IDSEL к адресуемому устройству PCI;
  - «подтягивать» управляющие сигналы к высокому уровню;
  - определять возможности подключенных устройств и выбирать удовлетворяющий их режим работы шины (частота, разрядность, протокол);
  - формировать аппаратный сброс (RST#) по сбросу от первичного интерфейса и по команде, сообщая о выбранном режиме специальной сигнализацией.
- Поддерживать карты ресурсов, находящихся по разные стороны моста.
- Отвечать под видом целевого устройства на транзакции, инициированные мастером на одном интерфейсе и адресованные ресурсу, находящемся со стороны другого интерфейса. Транслировать эти транзакции на другой интерфейс, выступая в роли ведущего устройства (мастера), и передавать их результаты истинному инициатору.

Мосты, выполняющие данные функции, называются *прозрачными* (transparent bridge); для работы с устройствами, находящимися за такими мостами, не требуется дополнительных драйверов моста. Именно такие мосты описаны в спецификации PCI Bridge 1.1, и для них как устройств PCI есть специальный класс (06). В данном случае подразумевается «плоская» модель адресации ресурсов (памяти и ввода-вывода): каждое устройство имеет свои адреса, уникальные (не пересекающиеся с другими) в пределах данной системы (компьютера).

Существуют и *непрозрачные мосты* (non-transparent bridges), которые позволяют организовывать обособленные сегменты со своими локальными адресными пространствами. Непрозрачный мост выполняет трансляцию (преобразование) адресов для транзакций, у которых инициатор и целевое устройство находятся по разные стороны моста. Досягаемыми через такой мост могут быть не все ресурсы (диапазоны адресов) противоположной стороны. Непрозрачные мосты используются, например, когда в компьютере выделяется подсистема «интеллигентного ввода-вывода» (1^20) со своими процессором ввода-вывода и локальным адресным пространством.

В адресации портов ввода-вывода есть особенности, связанные с «наследием», доставшимся от шины ISA, — 10-битным декодированием адреса, приводящим к «изрезанности» карты адресов и появлению псевдонимов. В мостах PCI применяются специальные меры, позволяющие сочетать особенности адресации устройств ISA с компактным описанием диапазонов адресов.

В мостах может быть реализована *специальная поддержка графического адаптера VGA*, который может находиться на стороне вторичного интерфейса моста. При включенной поддержке мост осуществляет трансляцию обращений к памяти VGA в диапазоне адресов 0A0000h-0BFFFFh, а также к регистрам ввода-вывода в диапазонах 3B0h-3BBh и 3C0h-3DFh и всем их 64 псевдонимам. Кроме того, для поддержки VGA требуется особый подход к чтению и записи регистров палитр (*VGA Palette Snooping*), которые расположены по адресам 3C6h, 3C8h и 3C9h, и их псевдонимов.

Транслирование транзакций и буферизация

Транслирование транзакций — довольно сложная задача моста, и от способа ее решения зависит производительность системы в целом. Какие именно транзакции следует транслировать с одного интерфейса на другой, решает часть моста, занимающаяся маршрутизацией. При трансляции транзакции мост как целевое устройство PCI сразу отвечает ее инициатору независимо от того, что происходит на другой стороне. Это позволяет мосту, как любому устройству PCI, соблюдать ограничения на время отклика и выполнения транзакций. Далее мост запрашивает управление шиной на противоположной стороне и, получив управление, проводит эту транзакцию от своего имени. Если транслируется транзакция чтения, то мост должен принять ее результаты, чтобы далее переслать их истинному инициатору транзакции. Этот общий сценарий для различных команд реализуется по-разному, но «при всем богатстве выбора» у моста PCI есть всего два варианта ответа инициатору:

- Можно отложить транзакцию, ответив условием *Retry*. Этот вариант называется *отложенной транзакцией*, он заставляет инициатора через некоторое время повторить попытку данной транзакции. За это время мост должен «провернуть» заказанную транзакцию на другой стороне интерфейса.
- Можно сделать вид, что транзакция успешно завершена. Такой вариант, называемый *отправленной записью* (posted write), возможен только для операций записи в память. Реальная запись происходит позже, когда мосту удается получить управление шиной на противоположном интерфейсе.

Если транзакция, транслируется с шины, работающей в режиме PCI-X, мост PCI-X должен ее расщеплять, а не откладывать. Протокол шины PCI-X обеспечивает мостам возможность более эффективной работы.

Интерфейсы моста PCI-X могут работать как в режиме PCI, так и в режиме PCI-X (Mode 1 или Mode 2). Мост должен определить возможность самого слабого устройства на своем вторичном интерфейсе и перевести эту шину (все устройства) в соответствующий режим (по протоколу и частоте синхронизации).

В случае соединения шин PCI и PCI-X мосту приходится преобразовывать некоторые команды и протокол. При трансляции транзакции с PCI на PCI-X мосту нужно формировать атрибуты транзакции.

Порядок выполнения операций и синхронизация

Механизмы отправленных записей и отложенных транзакций нацелены на одновременное (по возможности) выполнение множества операций обмена в системе шин PCI. Каждый мост имеет буферы и очереди отправленных записей и отложенных транзакций для команд, транслируемых в обоих направлениях. При этом мост одновременно может выполнять обмены данными на обоих своих интерфейсах, играя роль как инициатора, так и целевого устройства. Транзакции записи через мост, идущие в противоположном направлении, выполняются независимо друг от друга. Транзакция чтения через мост предварительно «выталкивает» из буферов моста все предыдущие транзакции. Эти (и дополнительные) правила гарантируют правильный порядок выполнения (реального завершения) транзакций.

14.7. Конфигурирование и BIOS устройств PCI и PCI-X

В шину PCI изначально заложены возможности автоматического конфигурирования системных ресурсов (пространств памяти и ввода-вывода, а также линий запроса прерываний). Автоматическое конфигурирование устройств (выбор адресов и прерываний) поддерживается средствами BIOS и ОС; оно ориентировано на технологию PnP. Стандарт PCI определяет для каждой функции конфигурационное пространство размером до 256 8-битных регистров, не приписанных ни к пространству памяти, ни к пространству ввода-вывода. Доступ к ним осуществляется по специальным командам шины *Configuration Read* и *Configuration Write*, вырабатываемым с помощью аппаратно-программного механизма.

Конфигурационное пространство функции начинается со *стандартного заголовка,* в котором содержатся идентификаторы производителя, устройства и его класса, а также описание требуемых и занимаемых системных ресурсов. После заголовка могут располагаться регистры, специфичные для устройства. Для *стандартизованных свойств* (capability) устройств (например, управления энергопотреблением) в конфигурационном пространстве имеются блоки регистров известного назначения. Эти блоки организуются в цепочки; просмотрев цепочку, конфигурационное ПО получает список всех доступных свойств устройства.

В PCI-X для устройств Mode 2 конфигурационное пространство расширено до 4096 байт; в расширенном пространстве могут присутствовать расширенные описания свойств.

После аппаратного сброса (или при включении питания) устройства PCI не отвечают на обращения к пространству памяти и ввода-вывода, они доступны

только для операций конфигурационного считывания и записи. В этих операциях устройства выбираются по индивидуальным сигналам IDSEL, чтением регистров конфигурационное ПО узнает о потребностях в ресурсах и возможных вариантах конфигурирования устройств. После распределения ресурсов, выполняемого программой конфигурирования (во время теста POST или при загрузке ОС), в конфигурационные регистры устройства записываются параметры конфигурирования (базовые адреса). Только после этого устройствам (точнее, функциям) устанавливаются биты, разрешающие им отвечать на команды обращения к памяти и портам ввода-вывода, а также самим управлять шиной. Для того чтобы всегда можно было найти работоспособную конфигурацию, все ресурсы, занимаемые картами, должны быть перемещаемыми в своих пространствах. Для многофункциональных устройств каждая функция должна иметь собственное конфигурационное пространство. Устройство может одни и те же регистры отображать и на память, и на пространство ввода-вывода. При этом в конфигурационных регистрах должны присутствовать оба описателя, но драйвер должен использовать только один способ обращения (предпочтительно через память).

В заголовке конфигурационного пространства описываются потребности в адресах трех типов:

- *Регистры в пространстве ввода-вывода* (*I/O Space).*
- *Регистры ввода-вывода, отображенные на память (Memory Mapped I/O).* Это область памяти, обращения к которой должны производиться в строгом соответствии с тем, что запрашивает инициатор обмена. Обращения к этим регистрам могут изменять внутреннее состояние периферийных устройств.
- *Память, допускающая предвыборку (Prefetchable Memory).* Это область памяти, «лишнее» чтение которой (с неиспользуемыми результатами) не приводит к побочным эффектам (то есть это память в чистом виде).

Потребности в адресах указываются устройствами в *регистрах базовых адресов* (Base Address Register, BAR), в этих же регистрах конфигурирующая программа устанавливает начальный адрес области.

В PCI имеется *поддержка устаревших* (*legacy) устройств* (VGA, IDE), которые сами себя таковыми объявляют по коду класса в заголовке. Их традиционные (фиксированные) адреса портов не заявляются в конфигурационном пространстве, но как только устанавливается бит разрешения обращения к портам, устройствам разрешается ответ и по этим адресам.

PCI BIOS

Для облегчения взаимодействия с устройствами PCI имеются дополнительные функции BIOS, доступные как из реального, так и из защищенного режима работы процессора. Функции PCI BIOS применяются только для поиска и конфигурирования устройств PCI — процедур, требующих доступа к их конфигурационному пространству. Функции приходится поддерживать и использовать потому, что циклы конфигурационных обращений, как и специальный цикл, выполняются специфическим образом. Кроме того, PCI BIOS позволяет управ

лять коммутатором запроса прерываний (PCI Interrupt Steering), скрывая специфический программный интерфейс чипсета системной платы. Остальное взаимодействие с устройствами через их пространства памяти и ввода-вывода, а также обработка прерываний в поддержке со стороны BIOS не нуждаются, поскольку выполняются непосредственно командами процессора и не зависят от платформы (чипсета системной платы). Регулярная работа с этими устройствами выполняется через обращения к регистрам устройств по адресам, полученным при конфигурировании, и в ходе обработки известных номеров прерываний от этих устройств. Функция проверки наличия PCI BIOS позволяет определить доступные конфигурационные механизмы, и, зная их работу, программа в дальнейшем может обходиться без вызовов PCI BIOS.

Программы с помощью функций PCI BIOS могут искать интересующие их устройства по идентификаторам или кодам класса. Если стоит задача полного «переучета» установленных устройств, то она решается чтением конфигурационной информации по всем функциям всех устройств всех шин — это быстрее, чем перебирать все возможные сочетания идентификаторов или классов кодов. Для найденных устройств программы должны определять реальные параметры чтением регистров конфигурационного пространства, учитывая возможность перемещения ресурсов по всему пространству и даже между пространствами памяти и ввода-вывода.

Expansion ROM карт PCI

Для содержимого ПЗУ расширений BIOS, установленных на картах PCI, принят стандарт, несколько отличающийся от традиционных дополнительных модулей ROM BIOS. Заголовок ПЗУ соответствует традиционному, но дополнительно имеет указатель на *структуру данных PCI*. Идентификаторы производителя и устройства, а также код класса совпадают с описанными в конфигурационном пространстве устройства PCI. Поскольку шина PCI используется не только в PC, в ПЗУ карты может храниться несколько программных модулей. Каждый модуль начинается со структуры данных, сам модуль следует сразу за структурой. За ним начинается структура для следующего модуля (если у предыдущего не установлен признак последнего модуля) и т. д. Тип платформы (процессора) указывается в заголовке модуля, и при инициализации BIOS активизируется только нужный тип. Такой механизм позволяет, например, один и тот же графический адаптер устанавливать и в IBM PC, и в Power PC.

14.8. Слоты и карты PCI/PCI-X

Стандартные слоты PCI и PCI-X представляют собой щелевые разъемы, имеющие контакты с шагом 0,05 дюйма. Слоты расположены несколько дальше от задней панели, чем ISA/EISA или MCA. Компоненты карт PCI размещены на левой поверхности плат. По этой причине крайний PCI-слот обычно совместно использует посадочное место адаптера (прорезь на задней стенке корпуса) с со

седним ISA-слотом. Такой слот называют *разделяемым* (shared slot), в него может устанавливаться либо карта ISA, либо карта PCI.

Карты PCI могут предназначаться для интерфейсных сигналов уровня 5 и 3,3 В, а также быть универсальными. Слоты PCI имеют уровни сигналов, соответствующие питанию микросхем PCI-устройств системной платы (включая главный мост): либо 5 В, либо 3,3 В. Во избежание ошибочного подключения слоты имеют ключи, определяющие номинал напряжения. Ключами являются пропущенные ряды контактов 12, 13 или/и 50, 51:

- для *слота на 5 В* ключ (перегородка) расположен на месте контактов 50, 51 (ближе к передней стенке корпуса); такие слоты отменены в PCI 3.0;
- для *слота на 3,3 В* перегородка находится на месте контактов 12, 13 (ближе к задней стенке корпуса);
- на *универсальных слотах* перегородок нет;
- на краевых разъемах *карт на 5 В* имеются ответные прорези только на месте контактов 50, 51; такие карты отменены в PCI 2.3;
- на *картах 3,3 В* прорези имеются только на месте контактов 12, 13;
- на *универсальных картах* имеются оба ключа (две прорези).

Ключи не позволяют установить карту в слот с неподходящим напряжением питания. Карты и слоты различаются лишь питанием буферных схем, которое поступает с линий +V I/O.

На слотах всех типов присутствуют *питающие напряжения* +3,3, +5, +12 и -12 В на одноименных линиях. В PCI 2.2 определена дополнительная линия 3.3Vaux — «дежурное» питание +3,3 В для устройств, формирующих сигнал PME# при отключенном основном питании.

ПРИМЕЧАНИЕ ----------

Выше приведены положения из официальных спецификаций PCI. На современных системных платах пока чаще всего встречаются слоты, по ключу являющиеся 5-вольтовыми, однако при этом напряжение на линиях +V I/O и уровни сигналов интерфейса являются 3,3-вольтовыми. В этих слотах нормально работают все современные карты с 5-вольтовыми ключами — их интерфейсные схемы подходят под оба напряжения питания (как 3,3, так и 5 В). Интерфейс с 5-вольтовым питанием может работать только на частоте до 33 МГц. «Настоящие» 5-вольтовые системные платы были только для процессоров 486 и первых моделей Pentium.

Наибольшее распространение получили 32-битные слоты, заканчивающиеся контактами A62/B62. 64-битные слоты встречаются реже, они длиннее и заканчиваются контактами A94/B94. Конструкция разъемов и протокол позволяют устанавливать 64-битные карты как в 64-битные, так и в 32-битные разъемы, и наоборот, 32-битные карты как в 32-битные, так и в 64-битные разъемы. При этом разрядность обмена будет соответствовать слабейшему компоненту.

Карты и слоты PCI-X по механическим ключам соответствуют 3,3-вольтовым картам и слотам; напряжение питания +V I/O для PCI-X Mode 2 устанавливается 1,5 В.

На рис. 14.2 изображены карты PCI в конструктиве PC/AT-совместимых компьютеров. Полноразмерные карты (Long Card, 107 x 312 мм) используются редко, чаще применяются укороченные платы (Short Card, 107 x 175 мм), но размеры многих карт еще меньше. Карта имеет обрамление (скобку), стандартное для конструктива ISA (раньше встречались карты и с обрамлением в стиле MCA IBM PS/2). У низкопрофильных карт (Low Profile) высота не превышает 64,4 мм; их скобки также имеют меньшую высоту. Такие карты могут устанавливаться вертикально в 19-дюймовые корпуса высотой 2U (около 9 см).

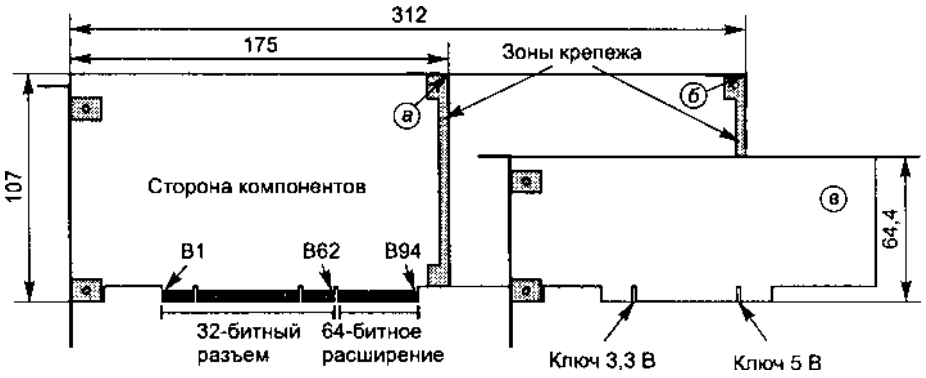


Рис. 14.2. Карты расширения для шины PCI: а — укороченная (обычная), *б* — полноразмерная, *в* — низкопрофильная

Назначение выводов разъема карт PCI/PCI-X приведено в табл. 14.3.

Таблица 14.3. Разъемы шины PCI

Ряд B	№	Ряд A	Ряд B	№	Ряд A
-12 В	1	TRST#	REQ#	18	GND
TCK	2	+12 В	+V I/O	19	PME#3
GND	3	TMS	AD31	20	AD30
TDO	4	TDI	AD29	21	+3,3 В
+5 В	5	+5 В	GND	22	AD28
+5 В	6	INTA#	AD27	23	AD26
INTB#	7	INTC#	AD25	24	GND
INTD#	8	+5 В	+3,3 В	25	AD24
PRSNT1#	9	ECC5[2]	C/BE3#	26	IDSEL
ECC4[2]	10	+V I/O	AD23	27	+3,3 В
PRSNT2#	11	ECC3[2]	GND	28	AD22
GND/Ключ 3,3 В	12	GND/Ключ 3,3 В	AD21	29	AD20
GND/Ключ 3,3 В	13	GND/Ключ 3,3 В	AD19	30	GND
ECC2[2]	14	3.3Vaux[3]	+3,3 В	31	AD18
GND	15	RST#	AD17	32	AD16
CLK	16	+V I/O	C/BE2#	33	+3,3 В
GND	17	GNT#	GND	34	FRAME#

продолжение

Таблица 14.3 (*продолжение)*

Ряд B	№	Ряд A	Ряд B	№	Ряд A
IRDY#	35	GND	C/BE6#	65	C/BE5#
+3,3 B	36	TRDY#	C/BE4#	66	+V I/O
DEVSEL#	37	GND	GND	67	PAR64/ECC72
PCIXCAP4	38	STOP#	AD63	68	AD62
LOCK#	39	+3,3 B	AD61	69	GND
PERR#	40	SMBCLK5	+V I/O	70	AD60
+3,3 B	41	SMBDAT5	AD59	71	AD58
SERR#	42	GND	AD57	72	GND
+3,3 B	43	PAR/ECC0	GND	73	AD56
C/BE1#	44	AD15	AD55	74	AD54
AD14	45	+3,3 B	AD53	75	+V I/O
GND	46	AD13	GND	76	AD52
AD12	47	AD11	AD51	77	AD50
AD10	48	GND	AD49	78	GND
GND/M66EN1	49	AD9	+V I/O	79	AD48
GND/Ключ 5 B/MODE2	50	GND/Ключ 5 B	AD47	80	AD46
GND/Ключ 5 B	51	GND/Ключ 5 B	AD45	81	GND
AD8	52	C/BE0#	GND	82	AD44
AD7	53	+3,3 B	AD43	83	AD42
+3,3 B	54	AD6	AD41	84	+V I/O
AD5	55	AD4	GND	85	AD40
AD3	56	GND	AD39	86	AD38
GND	57	AD2	AD37	87	GND
AD1	58	AD0	+V I/O	88	AD36
+V I/O	59	+V I/O	AD35	89	AD34
ACK64#/ ECC1	60	REQ64#/ ECC6	AD33	90	GND
+5 B	61	+5 B	GND	91	AD32
+5 B	62	+5 B	Резерв	92	Резерв
Конец 32-битного разъема			Резерв	93	GND
Резерв	63	GND	GND	94	Резерв
GND	64	C/BE7#	Конец 64-битного разъема		

[1] Сигнал M66EN определен в PCI 2.1 только для слотов на 3,3 В.

[2] Сигнал введен в PCI-X 2.0 (прежде был резерв).

[3] Сигнал введен в PCI 2.2 (прежде был резерв).

[4] Сигнал введен в PCI-X (в PCI — GND).

[5] Сигналы введены в PCI 2.3. В PCI 2.0 и 2.1 контакты A40 (SDONE#) и A41 (SBOFF#) использовались для слежения за кэшем; в PCI 2.2 они были освобождены (для совместимости на системной плате эти цепи подтягивались к высокому уровню резисторами 5 кОм).

На некоторых старых системных платах позади одного из слотов PCI встречается разъем Media Bus, на который выводятся сигналы ISA. Он предназначен

для размещения на графическом адаптере PCI звукового чипсета для шины ISA.
Большинство сигналов PCI соединяются по чистой шинной топологии, то есть одноименные контакты слотов одной шины PCI электрически соединяются друг с другом. Из этого правила есть несколько исключений: сигналы REQ#, GNT#, IDSEL и CLK заводятся на каждый слот индивидуально, сигналы INTA#, INTB#, INTC#, INTD# циклически сдвигаются по контактам (см. рис. 4.4).
Когда обычная системная плата используется в низкопрофильных корпусах, для подключения карт расширения можно применить *пассивный переходник* (riser card), устанавливаемый в один из слотов PCI. Если в переходник устанавливается более одной карты, то для реализации упомянутых исключений используют выносные разъемы PCI (маленькие печатные платы), с помощью которых вышеперечисленные сигналы берутся от других, свободных слотов PCI на системной плате. Переставляя эти разъемы, можно менять номера устройств на слотах переходника, а главное — их раскладку по линиям запросов прерывания. Беда такого подключения — длинные (10-15 см) шлейфы, соединяющие переходник со слотами. Из-за этого форма сигнала CLK искажается и вносится значительная задержка, что может приводить к внезапным «зависаниям» компьютера без всяких диагностических сообщений.

Инициализация и определение режима работы шины PCI-X

Каждый сегмент PCI-X (физическая шина) должен работать в самом прогрессивном режиме, доступном всем его абонентам, включая и главный для этой шины мост. В стандартной шине PCI «прогрессивность» определяется только допустимой тактовой частотой (33 или 66 МГц), и свои способности карта сообщает по контакту B49 (M66EN, см. выше). В шине PCI-X появляются новые возможности: поддержка собственно протокола PCI-X (*Mode 1* в терминах PCI-X 2.0) и ускоренных передач (*Mode 2).* Эти возможности карта сообщает через контакт B38 (PCIXCAP), который может быть подключен к шине GND через резистор определенного номинала или оставаться неподключенным. Мост, которому подчиняется данная шина, проверяет состояние линий M66EN и PCIXCAP по началу сигнала сброса. В соответствии с выявленными возможностями (они будут соответствовать самому слабому абоненту) мост выбирает режим работы шины. Этот режим доводится до всех абонентов с помощью *шаблона инициализации* (PCI-X Initialization Pattern) — уровней сигналов PERR#, DEVSEL#, STOP# и TRDY# в момент окончания сигнала RST# (по его нарастающему фронту). К этому моменту на слоты уже подается соответствующее напряжение +V I/O.
«Горячее» подключение-отключение устройств PCI *(Hot Plug)* требует наличия в системе специального контроллера *(Hot-Plug Controller*), управляющего слотами «горячего» подключения, и соответствующей программной поддержки — ОС, драйверов устройств и контроллера. Слот с «горячим» подключением должен быть подключен к шине PCI через цепи, обеспечивающие управляемую коммутацию электронными ключами всех сигнальных цепей PCI и управляемую подачу напряжения питания.

Малогабаритные конструктивы с шиной PCI

Стандартный конструктив PCI для настольных PC/AT-совместимых компьютеров для ряда применений является слишком громоздким. Существуют более компактные варианты:

- *Low-Profile PCI* — низкопрофильный вариант карты PCI для системных плат AT /ATX. Эти карты имеют такой же краевой печатный разъем, как и обычные карты, но высота карты уменьшена до 64 мм, уменьшен также размер крепежной скобки. Карты можно устанавливать вертикально (без переходника в низкопрофильные корпуса (например, 19-дюймового формата высотой 2U). Для установки этих карт в полноразмерные (настольные) корпуса на карте следует установить обычную крепежную скобку (в комплект поставки карты может входить дополнительная скобка). Для этих карт в спецификации предусматривается напряжение питания интерфейсных схем только 3,3 В, хотя часто встречаются формально (по ключам) 5-вольтовые низкопрофильные карты.
- Малогабаритные конструктивы для блокнотных компьютеров (см. 14.11):
  - для карт расширения, устанавливаемых пользователем без вскрытия компьютера (с возможностью «горячего» подключения), применяется конструктив PCMCIA, впоследствии переименованный в PC Card (в этом конструктиве, как показано далее, возможно четыре различных варианта интерфейса, одним из которых является CardBus — шина PCI 32 бит/33 МГц);
  - для карт расширения, комплектуемых изготовителем, внутри компьютера (недоступно для пользователя) применяются конструктивы Small PCI и Mini PCI.

14.9. Порт графического акселератора — AGP

Порт AGP (Accelerated Graphic Port — порт ускоренной графики) был введен для подключения *графических адаптеров с 3D-акселераторами*. Такой адаптер содержит: *акселератор* — специализированный графический процессор; *локальную память*, используемую и как видеопамять, и как локальное ОЗУ графического процессора; *управляющие и конфигурационные регистры*, доступные как локальному, так и центральному процессорам. Акселератор может обращаться и к локальной памяти, и к системному ОЗУ, в котором для него могут храниться наборы данных, не умещающиеся в локальной памяти (как правило, текстуры большого объема). Основная идея порта AGP заключается в предоставлении акселератору максимально быстрого доступа к системной памяти (локальная ему и так близка), более приоритетного, чем доступ к ОЗУ со стороны других устройств.

Порт AGP представляет собой 32-разрядный параллельный синхронный интерфейс с тактовой частотой 66 МГц; большая часть сигналов позаимствована

с шины PCI. Однако, в отличие от PCI, интерфейс порта AGP двухточечный, соединяющий графический акселератор с памятью и системной шиной процессора каналами данных чипсета системной платы, не пересекаясь с «узким местом» — шиной PCI. Обмен через порт может происходить как по протоколу PCI, так и по протоколу AGP. Отличительные особенности порта AGP:

- конвейеризация обращений к памяти;
- умноженная относительно тактовой частоты порта частота передачи данных (2x/4x/8x);
- «внеполосная» подача команд (SBA), обеспеченная демультиплексированием шин адреса и данных.

Идею конвейеризации обращений к памяти иллюстрирует рис. 14.3, где сравниваются обращения к памяти по шине PCI и через порт AGP. В PCI во время реакции памяти на запрос шина простаивает (но не свободна). Конвейерный доступ AGP позволяет в это время передавать следующие запросы, а потом получать поток ответов.

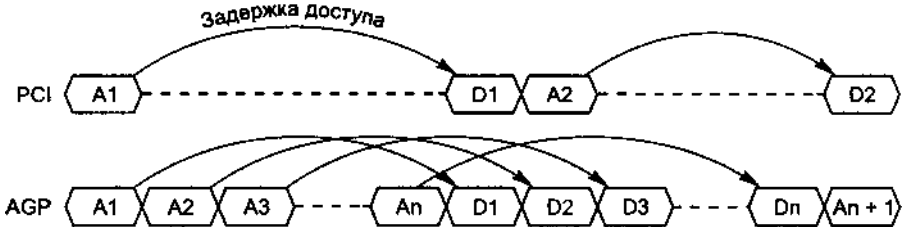


Рис. 14.3. Циклы обращения к памяти PCI и AGP

Умножение частоты передачи данных обеспечивает при частоте 66 МГц пиковую пропускную способность до 533 Мбайт/с в режиме 2x, до 1066 при 4x и до 2132 Мбайт/с при 8x. Выше 66 МГц тактовую частоту официально не поднимают.

Демультиплексирование (разделение) шины адреса и данных реализовано несколько необычно. С целью экономии числа интерфейсных линий шину адреса и команды в демультиплексированном режиме AGP представляют всего 8 линий *SBA* (SideBand Address), по которым команда, адрес и значение длины передачи передаются последовательно за несколько тактов. Поддержка демультиплексированной адресации не являлась обязательной для устройства AGP 1.0, поскольку имеется альтернативный способ передачи адреса по шине AD. В версии AGP 2.0 она стала обязательной, а в 3.0 — это уже единственный способ передачи адреса.

Отметим, что порт AGP обеспечивает только потенциальные преимущества, которые могут быть реализованы лишь при поддержке аппаратными средствами графического адаптера и специального ПО. Графический адаптер с интерфейсом AGP может реально вести себя по-разному:

- не задействовать конвейеризацию, а использовать только *быструю запись PCI* (Fast Write);

- не работать с текстурами, расположенными в системной памяти, но обеспечивать более быстрый обмен данными между памятью и локальным буфером;
- использовать все возможности порта, когда акселератор получает быстрый доступ к системной памяти, а центральный процессор может быстро закачивать данные в локальную память адаптера.

Порт AGP содержит практически полный набор сигналов шины PCI и дополнительные сигналы AGP. Устройство, подключаемое к порту AGP, может предназначаться как исключительно для операций AGP, так и для комбинированных операций AGP и PCI. Акселератор адаптера является *мастером* (ведущим устройством) *порта AGP*, свои запросы он может выполнять как в режиме AGP, так и в режиме PCI. В режиме AGP обмены выполняются с поддержкой (или без поддержки) таких свойств, как внеполосная адресация (SBA) и скорости 2x/4x/8x. Для транзакций в режиме AGP ему доступно только системное ОЗУ (но не локальная память устройств PCI). Кроме того, адаптер является *целевым устройством PCI*, для которого, помимо обычных команд PCI, может поддерживаться (или не поддерживаться) быстрая запись (Fast write) со скоростью 2x/4x/8x стороны процессора. В качестве целевого устройства адаптер выступает при обращениях ЦП к его локальной памяти, регистрам ввода-вывода и конфигурационного пространства.

Порт AGP позволяет акселератору работать в двух режимах — DMA и DIME (Direct Memory Execute). В *режиме DMA* акселератор при вычислениях рассматривает локальную память как первичную, а когда ее недостаточно, подкачивает в нее данные из основной памяти. В *режиме DIME*, он же режим исполнения (executive mode), локальная и основная память для акселератора логически равнозначны и располагаются в едином адресном пространстве. В режиме DMA для трафика порта характерны длительные блочные передачи, в режиме DIME трафик порта насыщен короткими произвольными запросами.

Спецификации AGP разрабатывались фирмой Intel на базе шины PCI 2.1 с частотой 66 МГц; пока имеется три основные версии спецификаций:

- AGP 1.0 (1996 г.) — определен порт с конвейерным обращением к памяти и двумя альтернативными способами подачи команд: внеполосной (по шине SBA) и внутриполосной (по сигналу PIPE#). Режимы передачи — 1x/2x, питание интерфейса — 3,3 В.
- AGP 2.0 (1998 г.) — добавлена возможность быстрой записи в режиме PCI (Fast Writes), а также режим 4x с питанием 1,5 В.
- AGP 3.0 (2002 г., проект назывался AGP8X) — добавлен режим 8x с питанием 0,8 В и динамическим инвертированием байтов, отменены скорости 1x и 2x; оставлен один способ подачи команд — внеполосный (SBA); исключены некоторые команды AGP; введены команды изохронного обмена; введена возможность выбора размера страниц, описанных в таблице GART; введена селективная поддержка когерентности при обращениях к разным страницам в пределах GART.

Порт AGP предназначен только для подключения интеллектуального графического адаптера (причем только одного), имеющего 3D-акселератор. Системная логика порта AGP отличается сложным контроллером памяти, который выполняет глубокую буферизацию и высокопроизводительное обслуживание запросов AGP (от адаптера) и других своих клиентов — центрального процессора (одного или нескольких) и шины PCI. Единственный вариант подключения нескольких адаптеров с AGP — организация на системной плате нескольких портов AGP, что вряд ли когда-нибудь будет применяться.

AGP может реализовать всю пропускную способность 64-битной системы памяти современного компьютера. При этом возможны конкурирующие обращения к памяти со стороны как процессора, так и мостов шин PCI. Фирма Intel впервые ввела поддержку AGP в чипсеты для процессоров P6, конкуренты используют AGP и в системных платах для процессоров с интерфейсом Pentium (сокет *Super 7*). В настоящее время порт AGP имеется во многих системных платах для PC-совместимых компьютеров и других платформ (даже Macintosh). Однако в перспективе его, похоже, заменит PCI Express.

Протоколы транзакций

Транзакции в режиме PCI, инициируемые акселератором, выполняются обычным для PCI способом. Заметим, что при этом на все время транзакции шина AD занята, причем транзакции чтения памяти занимают шину на большее число тактов, чем транзакции записи, — после подачи адреса неизбежны такты ожидания на время доступа к памяти. Запись на шине происходит быстрее — данные записи задатчик посылает сразу за адресом, а на время доступа к памяти они «оседают» в буфере контроллера памяти. Контроллер памяти позволяет завершить транзакцию и освободить шину до физической записи в память.

Конвейерные транзакции AGP (команды AGP) инициируются только акселератором; логикой AGP они ставятся в очередь на обслуживание и исполняются в зависимости от приоритета, порядка поступления запросов и готовности данных. Эти транзакции могут быть адресованы акселератором только к системному ОЗУ.

Обращения со стороны процессора (или задатчиков шины PCI), адресованные к устройству на AGP, отрабатываются им как ведомым устройством PCI, однако имеется возможность *быстрой записи* в локальную память *FW* (Fast Write), в которой данные передаются на скорости AGP (2x/4x/8x), и управление потоком их передач ближе к протоколу AGP, нежели к PCI. Транзакции *FW* инициируются процессором и предназначены для принудительного «заталкивания» данных в локальную память акселератора.

Концепцию *конвейера AGP* иллюстрирует рис. 14.4. Порт AGP может находиться в одном из четырех состояний:

- *IDLE* — покой;
- *DATA* — передача данных конвейеризированных транзакций;
- *AGP* — постановка в очередь команды AGP;
- *PCI* — выполнение транзакции в режиме PCI.

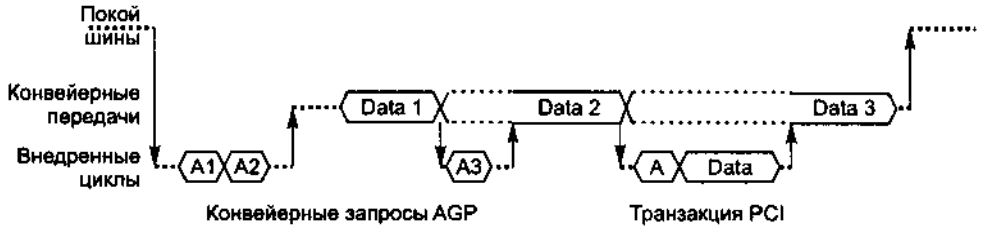


Рис. 14.4. Конвейер AGP

Из состояния покоя *IDLE* порт может вывести запрос транзакции PCI (как от акселератора, так и с системной стороны) или запрос AGP (только от акселератора). В состоянии *PCI* транзакция PCI выполняется целиком, от подачи адреса и команды до завершения передачи данных. В состоянии *AGP* ведущее устройство передает только команду и адрес для транзакции (по сигналу PIPE# или через шину SBA), ставящейся в *очередь*; несколько запросов могут следовать друг за другом. В состояние *DATA* порт переходит, когда у него в очереди имеется необслуженная команда, готовая к исполнению. В этом состоянии происходит передача данных для команд, стоящих в очереди. Данное состояние может прерываться вторжением запросов *PCI* (для выполнения целой транзакции) или *AGP* (для постановки в очередь новой команды), но прерывание[1] возможно только на границах данных транзакций AGP. Когда порт AGP обслужит все команды, он снова переходит в состояние покоя. Все переходы происходят под управлением арбитра порта AGP, реагирующего на поступающие запросы (сигнал REQ# от акселератора и внешние обращения от процессора или других устройств PCI) и ответы контроллера памяти.

Транзакции AGP отличаются от транзакций PCI:

- фаза данных отделена от фазы адреса, чем и обеспечивается конвейеризация;
- используется собственный набор команд;
- транзакции адресуются только к системной памяти;
- длина транзакции явно указывается в запросе;
- конвейерные запросы не гарантируют когерентность памяти и кэша.

Существует *два способа подачи команд AGP* (постановки запросов в очередь), из которых в текущей конфигурации выбирается один, причем изменение способа «на ходу» не допускается:

- запросы вводятся по шине AD[31:0] и C/BE[3:0] с помощью сигнала PIPE#, по каждому фронту CLK ведущее устройство передает очередное двойное слово запроса вместе с кодом команды;
- команды подаются через *внеполосные* (sideband) линии адреса SBA[7:0] («внеполосность» означает, что эти сигналы используются независимо от за

[1] Здесь прерываниями называется вклинивание команд AGP и транзакций PCI в поток передач данных; к прерываниям ЦП они не имеют отношения.

нятости шины AD), причем синхронизация подачи запросов зависит от режима (1x/2x/4x/8x).

При *внеполосной подаче команд* по шине SBA[7:0] передаются 16-битные посылки четырех типов:

- тип 1 — поле длины и младшие биты адреса;
- тип 2 — код команды и средние биты адреса;
- тип 3 — старшие биты адреса;
- тип 4 — дополнительные старшие биты адреса для 64-битной адресации.

Посылки типов 2, 3 и 4 являются «липкими» (sticky) — значения, ими определяемые, сохраняются до введения новой посылки того же типа. Постановку команды в очередь инициирует посылка типа 1, задающая длину транзакции и ее младшие адреса. Такой способ очень экономно использует такты шины для подачи команд при пересылках массивов. Каждая двухбайтная посылка передается по 8-битной шине SBA в два приема (сначала старший, потом младший байт). Синхронизация байтов зависит от режима порта:

- в режиме 1x каждый байт передается по фронту CLK; очередная команда (посылкой типа 1) может вводиться *за каждую пару тактов* CLK, полный цикл ввода команды занимает 10 тактов;
- в режиме 2x для SBA используется двойная синхронизация (по фронту и спаду SB_STB), очередная команда может вводиться *в каждом такте* CLK;
- в режиме 4x частота стробов в два раза выше, чем CLK, в каждом такте CLK может вводиться *пара посылок*, но мастер AGP может ставить в очередь не более одного запроса за такт;
- в режиме 8x частота стробов в 4 раза выше CLK, в каждом такте CLK умещаются уже 4 посылки, но в очередь ставится не более одной команды за такт CLK.

В ответ на полученные команды порт AGP выполняет *передачи данных*, причем фаза данных AGP явно не привязана к фазе команды/адреса. Фаза данных вводится портом AGP по готовности системной памяти к запрашиваемому обмену.

Передачи данных AGP выполняются, когда шина находится в состоянии *DATA*. Фазы данных вводит порт AGP (системная логика), исходя из очередности ранее пришедших к нему команд от акселератора. Акселератор узнает о назначении шины AD в последующей транзакции по сигналам ST[2:0], причем узнает лишь тип и приоритет команды, результаты которой последуют в данной транзакции. Какую именно команду из очереди отрабатывает порт, акселератор определяет сам, так как именно он ставит их в очередь (ему известен порядок). Никаких тегов транзакций (как, например, в системной шине процессоров P6 или в PCI-X) в интерфейсе AGP нет. Имеются только независимые очереди для каждого типа команд (*чтение низкоприоритетное, чтение высокоприоритетное, запись низкоприоритетная, запись высокоприоритетная).* Фазы исполнения команд разных очередей могут чередоваться произвольным образом; порт имеет право исполнять их в порядке, оптимальном с точки зрения производительности. Реальный порядок исполнения команд (чтения и записи памяти) тоже может меняться. Однако для каждой очереди порядок выполнения

всегда совпадает с порядком подачи команд (об этом знают и акселератор, и порт). В AGP 3.0 приоритеты очередей отменили, но ввели возможность изохронных транзакций.
Арбитр системной платы в первую очередь обслуживает высокоприоритетные запросы AGP, затем запросы от центрального процессора и мастеров шины PCI и в последнюю очередь — низкоприоритетные запросы AGP.
При передаче данных AGP управляющие сигналы, заимствованные от PCI, имеют почти такое же назначение, что и в PCI. Передача данных AGP в режиме 1x очень похожа на циклы PCI, но немного упрощена процедура квитирования (поскольку это выделенный порт, и обмен выполняется только с быстрым контроллером системной памяти). В режимах 2x/4x/8x имеется специфика стробирования:

- *в режиме 1x* данные фиксируются получателем по положительному перепаду каждого такта CLK, что обеспечивает пиковую скорость 66,6 x 4 = = 266 Мбайт/с;
- *в режиме 2x* используются стробы, формируемые источником данных, приемник фиксирует данные и по спаду, и по фронту строба, частота стробов совпадает с частотой CLK, что и обеспечивает пиковую скорость 66,6 x 2 x 4 = = 533 Мбайт/с;
- *в режиме 4x* частота стробов в два раза выше, чем CLK, что и обеспечивает пиковую скорость 66,6 x 2 x 2 x 4 = 1066 Мбайт/с;
- *в режиме 8x* пары стробов переключаются с частотой в четыре раза выше CLK, стробы сдвинуты относительно друг друга на половину своего периода, чем и обеспечивается пиковая скорость 66,6 x 4 x 2 x 4 = 2132 Мбайт/с.

Трансляция адресов — апертура AGP и GART

Порт AGP обеспечивает *трансляцию логических адресов*, фигурирующих в запросах акселератора к системной памяти, в *физические адреса*, согласуя видение ОЗУ программой, выполняемой акселератором, и программой, выполняемой центральным процессором. Трансляция осуществляется в постраничном базисе (по умолчанию размер страницы 4 Кбайт), принятом в системе виртуальной памяти с подкачкой страниц по запросу, используемой в процессорах x86 (и других современных процессорах). Трансляции подлежат обращения, попадающие в *апертуру AGP*, — область физических адресов памяти, лежащую выше границы ОЗУ и, как правило, примыкающую к области локальной памяти адаптера (рис. 14.5). Таким образом, при работе в режиме DIME акселератору доступна непрерывная область памяти, часть которой составляет локальная память адаптера. Остальная часть адресуемой им памяти отображается на системное ОЗУ через апертуру с помощью *таблицы GART* (Graphics Address Remapping Table — таблица переопределения графических адресов). Каждый элемент этой таблицы описывает свою страницу в области апертуры. В каждом элементе GART есть признак его действительности; в действительных элементах указывается адрес страницы физической памяти, на которую отображается соответствующая область апертуры. Таблица GART физически находится

в системном ОЗУ, она выровнена по границе 4-килобайтной страницы, на ее начало указывают конфигурационные регистры порта AGP.

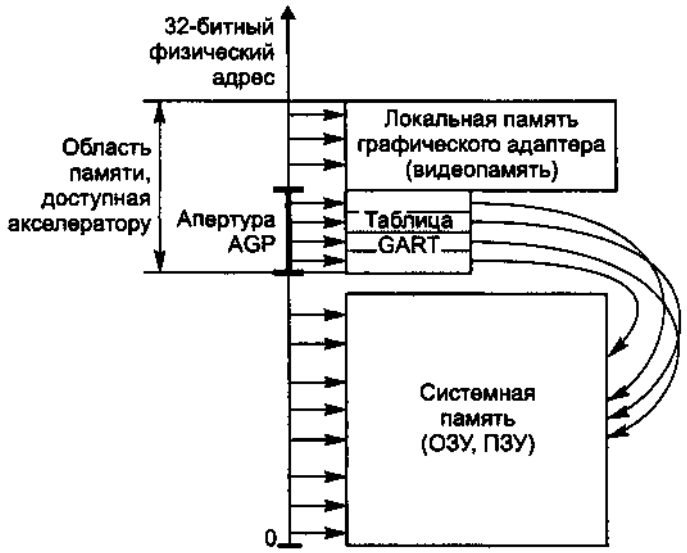


Рис. 14.5. Адресация памяти в системе с AGP

Размер апертуры AGP (определяющий и размер таблицы GART) задается программированием регистров чипсета. Путем настройки параметров CMOS Setup или внешних утилит его можно задать размером 8,16, 32,..., 256 Мбайт и более. Оптимальное значение апертуры зависит от объема памяти и используемых программ, но можно ориентироваться на половину объема системного ОЗУ. Заметим, что назначение размера апертуры в большинстве случаев еще не означает отчуждения указанного объема ОЗУ от системного ОЗУ — это лишь предельный размер памяти, которую ОС будет выделять акселератору по его запросам. Пока акселератору хватает своей локальной памяти, он не запрашивает память в системном ОЗУ. Только при нехватке локальной памяти он начнет динамически запрашивать дополнительную, и эти запросы будут удовлетворяться в пределах установленной апертуры. По мере уменьшения потребности в дополнительной памяти она будет динамически высвобождаться для обычных нужд операционной системы. Правда, если у графического акселератора локальной памяти нет вообще (в дешевых интегрированных адаптерах), то от ОЗУ статически (на все время работы) отчуждается часть памяти (хотя бы под экранный буфер). Это будет видно по уменьшенному размеру памяти, который POST показывает в начале процесса загрузки.

Изохронные транзакции в AGP 3.0

Для поддержки изохронных транзакций в AGP 3.0 введены новые коды команд и состояний, а также конфигурационные регистры, управляющие изохронным

соединением. Изохронные транзакции может выполнять *мастер AGP* только через область апертуры AGP, причем с областью памяти, для которой не обеспечивается когерентность (чтобы избежать непрогнозируемых задержек, связанных с выгрузками «грязных» строк). Изохронные транзакции возможны только на скорости 8x. В зависимости от мощности подсистемы памяти порт AGP может выдерживать изохронный трафик, достаточный для различных применений:

- видеозахват (в настольных ПК) — 128 Мбайт/с;
- видеоредактирование — 320 Мбайт/с;
- поток одного канала HDTV — 384 Мбайт/с;
- поток двух каналов HDTV (в мощных рабочих станциях) — 640 Мбайт/с.

Конфигурационные регистры AGP

Конфигурирование устройств с интерфейсом AGP выполняется так же, как и для обычных устройств PCI, — через обращения к регистрам конфигурационного пространства (см. 14.7). В процессе инициализации процедура POST только распределяет системные ресурсы, но операции AGP оставляет запрещенными. Работу AGP разрешает загруженная ОС, предварительно установив требуемые параметры AGP: режим обмена, поддержку быстрой записи, возможность адресации свыше 4 Гбайт, способ подачи и допустимое число запросов.

В конфигурировании системы с AGP фигурируют два PCI-устройства со своими конфигурационными пространствами:

- собственно *порт AGP* (Core Logic) — целевое устройство в транзакциях AGP;
- *графический адаптер* — инициатор транзакций AGP.

Их специфические конфигурационные регистры частично совпадают по назначению.

Слоты и карты AGP

Графический адаптер с интерфейсом AGP может быть встроен в системную плату, а может располагаться и на карте расширения, установленной в *слот AGP*. Внешне карты с портом AGP похожи на PCI (рис. 14.6), но у них имеется разъем повышенной плотности с «двухэтажным» (как у EISA) расположением ламелей. Сам разъем находится дальше от задней кромки платы, чем разъем PCI.

Порт AGP может использовать три возможных номинала питания интерфейсных схем (Vddq): 3,3 В (для 1x и 2x), 1,5 В (для 2x и 4x) и 0,8 В (для 8x). Сигналы RST# и CLK всегда трехвольтовые. На слотах и картах имеются механические ключи, предотвращающие ошибочные подключения.

Помимо собственно AGP, в порте AGP заложены *сигналы шины USB*, которую предполагается заводить в монитор (линии USB+, USB- и сигнал OVRCNT#, которым сообщается о перегрузке по току линии питания + 5 В, выводимой в монитор).

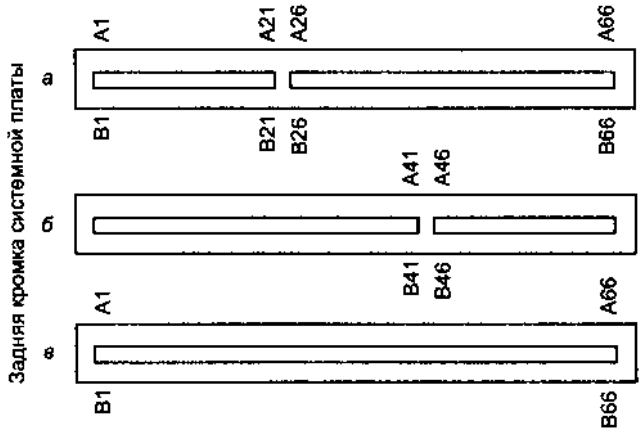


Рис. 14.6. Слоты AGP: а — 3,3 В, *б* — 1,5 В, *в* — универсальные

Спецификация AGP Pro описывает более мощный коннектор (рис. 14.7), позволяющий в 4 раза повысить мощность, подводимую к графической карте. При этом сохраняется односторонняя совместимость: карты AGP могут устанавливаться в слот AGP Pro, но не наоборот. В настоящее время от коннектора AGP Pro отказались, а для подачи питания на графическую карту используется дополнительный кабель с разъемом.

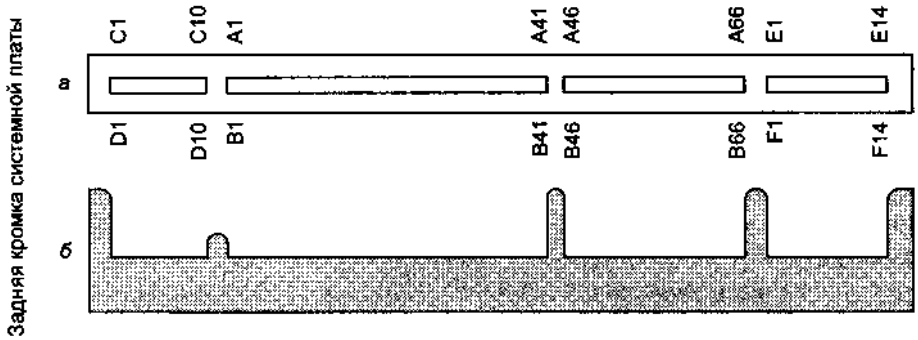


Рис. 14.7. Коннектор карты AGP Pro (показан ключ питания карты 1,5 В): а — вид сверху, *б* — профиль ключей

14.10. PCI Express

PCI Express — новая архитектура соединения компонентов, введенная под эгидой PCI SIG, известная и под названием *3GIO* (3-Generation Input/Output — ввод-вывод 3-го поколения). Здесь шинное соединение устройств с параллельным интерфейсом заменено двухточечными последовательными соединениями через коммутаторы. В этой архитектуре сохраняются многие программные черты шины PCI, что обеспечивает плавный переход от PCI к PCI Express. В архи

тектуре появились новые возможности: управление качеством обслуживания (Quality of Service, QoS), потреблением и бюджетом связей. Протокол PCI Express характерен малыми накладными расходами и малыми задержками выполнения транзакций.

PCI Express позиционируется как универсальная архитектура ввода-вывода для компьютеров разных классов, телекоммуникационных устройств и встроенных систем. Высокая пропускная способность достигается при соизмеримой с PCI цене и даже ниже. Сфера применения — от соединений между микросхемами на плате до межплатных разъемных и кабельных соединений. Высокая пропускная способность на контакт соединения позволяет минимизировать число таких контактов. Малое число сигнальных линий позволяет применять малогабаритные конструктивы. Универсальность дает возможность использования единой программной модели для всех форм-факторов. Спецификация PCI Express Base specification Revision 1.0a опубликована в апреле 2003 года.

Элементы и топология соединений PCI Express

Соединение PCI Express (PCI Express Link) — это пара встречных симплексных каналов, соединяющих два компонента. По этим каналам передаются *пакеты*, несущие команды и данные транзакций, сообщения и управляющие посылки. Канал может быть образован одной или несколькими линиями передачи сигналов (Lane); применение нескольких линий позволяет масштабировать пропускную способность канала. В PCI Express с помощью пакетного протокола реализуются все транзакции чтения и записи, используемые в PCI, причем в расщепленном варианте (как в PCI-X). Таким образом, здесь фигурируют *запросчик* (requester) и *исполнитель* (completer) транзакции. В PCI Express рассматриваются *четыре пространства*: памяти, ввода-вывода, конфигурационное и сообщений. Новое (по сравнению с PCI) *пространство сообщений* (message space) используется для передачи в виде пакетов «внеполосных» сигналов PCI: прерываний по линиям INTx, сигналов управления потреблением и т. п. Таким образом реализуются «виртуальные провода». *Порт PCI Express* содержит передатчик, приемник и узлы, необходимые для сборки-разборки пакетов.

Пример топологии средств ввода-вывода, иллюстрирующий архитектуру PCI Express, приведен на рис. 14.8. Центральным элементом архитектуры является *корневой комплекс* (root complex), соединяющий иерархию ввода-вывода с центром — процессором (одним или несколькими) и памятью. Корневой комплекс может иметь один и более портов PCI Express, каждый из этих портов определяет свой *домен иерархии* (hierarchy domain). Каждый домен состоит из одной *конечной точки* (endpoint) или *субиерархии* — нескольких конечных точек, связанных коммутаторами. Наличие непосредственных одноранговых коммуникаций между элементами разных доменов обязательным не является, но может иметь место в конкретных реализациях. Для обеспечения прозрачных одноранговых коммуникаций в корневом комплексе должны присутствовать коммутаторы. Возможность взаимодействия центрального процессора с любым устройством любого домена безусловна, как и возможность обращения любого

устройства к памяти. Корневой комплекс должен генерировать запросы к конфигурационному пространству — его роль аналогична главному мосту PCI.

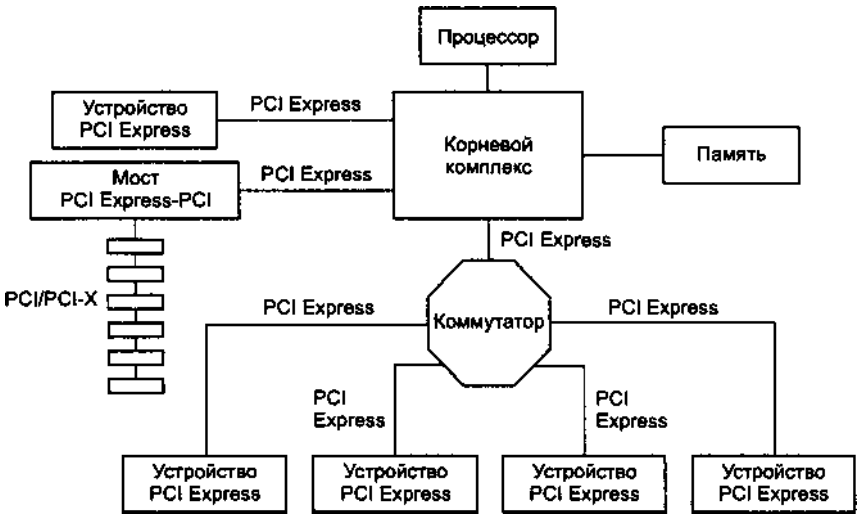


Рис. 14.8. Топология «фабрики» PCI Express

Конечная точка — это устройство, способное инициировать или/и исполнять транзакции PCI Express от своего имени или от имени устройства, не являющегося устройством PCI Express (например, от имени хост-контролера USB). Конечная точка должна быть видима в одном из доменов иерархии — представлять в нем свои конфигурационные регистры и отвечать как исполнитель на конфигурационные запросы. В качестве механизма сигнализации прерываний все конечные точки используют MSI. В PCI Express рассматриваются два типа конечных точек: «наследники» (legacy) и новые точки, построенные по идеологии PCI Express. К «наследным» точкам имеется ряд послаблений в плане адресации памяти, перемещаемости ресурсов (из пространства ввода-вывода в пространство памяти) и некоторых нюансов.

Коммутатор (switch) имеет несколько портов PCI Express. Логически он представляет собой несколько виртуальных мостов PCI-PCI, соединяющих порты коммутатора со своей внутренней локальной шиной. Однако тех издержек, которые вносят «настоящие» мосты PCI, коммутатор не вносит. Коммутатор транслирует между портами пакеты всех типов, основываясь на адресной информации, актуальной для пакета данного типа. Арбитраж между портами коммутатора может учитывать виртуальные каналы и, соответственно, взвешенно распределять пропускную способность. Коммутатор не имеет права разбивать пакеты на более мелкие (у мостов PCI такое право имеется).

Мост PCI-Express-PCI соединяет иерархию шин PCI/PCI-X с «фабрикой» ввода-вывода.

Конфигурирование «фабрики» осуществляется либо со 100-процентной совместимостью с конфигурационным механизмом PCI 2.3, либо с использовани-

ем расширенного конфигурационного пространства PCI-X. Каждое *соединение PCI Express* с помощью виртуальных мостов отображается в виде *логической шины PCI* со своим номером. Логически устройства отображаются в конфигурационном пространстве как устройства PCI, каждое из которых может иметь 1-8 функций со своим набором конфигурационных регистров.

Архитектурная модель PCI Express

Уровень транзакций (transaction layer) — верхний уровень архитектуры, отвечающий за сборку и разборку *пакетов TLP* (Transaction Layer Packet — пакет уровня транзакций). Эти пакеты используются для транзакций чтения и записи, а также для сообщений о событиях некоторых типов. Каждый пакет TLP имеет уникальный идентификатор, который позволяет направить ответный пакет его отправителю. В TLP поддерживаются различные форматы адресации, зависящие от типов транзакций. Пакет может иметь атрибуты отмены слежения за когерентностью NS (No Snoop) и «расслабленной» упорядоченности RO (Relaxed Ordering). Каждая транзакция, требующая ответа, выполняется в виде расщепленной (см. 14.2). Уровень транзакций отвечает и за управление потоком, реализованное на основе механизма кредитов.

Канальный уровень (data link layer), промежуточный в стеке, отвечает за управление связью, обнаружение ошибок и организацию повторных передач вплоть до успеха или признания отказа соединения. К пакетам, полученным от уровня транзакций, канальный уровень добавляет свои заголовки (номера пакетов и контрольные коды). Канальный уровень и сам является генератором и получателем *пакетов DLLP* (Data Link Layer Packet — пакет канального уровня), используемых для управления соединением.

Физический уровень (physical layer) изолирует канальный от всех подробностей передачи сигналов. Он состоит из двух субблоков. *Логический субблок* при передаче выполняет распределение данных по линиям, скремблирование, кодирование по схеме 8B/10B[1], кадрирование и преобразование в последовательный код. При приеме выполняются обратные действия. Символы, добавляемые при кодировании 8B/10B, используются для служебной сигнализации. Логический субблок отвечает и за согласование соединения, инициализацию и т. п. *Электрический субблок* отвечает за электрическое согласование, синхронизацию, обнаружение приемника. Уровневая модель, принятая в PCI Express, позволяет, не затрагивая остальных уровней, сменить физический уровень или его субблоки, когда появятся более эффективные схемы кодирования и сигнализации. Интерфейс между физическим и канальным уровнями зависит от реализации этих компонентов и выбирается их разработчиком. Интерфейс физического уровня четко специфицирован, что обеспечивает возможность соединения устройств разного происхождения. Для тестирования на соответствие электрическим параметрам достаточно подключить устройство PCI Express к специальному тестеру.

[1] Избыточное преобразование 8 бит в 10-битный символ.

Программная совместимость с PCI/PCI-X

Программная модель PCI Express совместима с PCI в следующих аспектах:

- обнаружение, нумерация и конфигурирование устройств PCI Express выполняется тем же конфигурационным ПО, что и в PCI (PCI-X 2.0);
- существующие ОС загружаются без каких-либо модификаций;
- драйверы существующих устройств поддерживаются без каких-либо модификаций;
- конфигурирование и разрешение новых функциональных возможностей PCI Express выполняются в соответствии с общей идеей конфигурирования устройств PCI.

Качество обслуживания и виртуальные каналы

В PCI Express реализована поддержка разных классов *качества обслуживания* (QoS); это дает возможность:

- выделять ресурсы соединения для потока каждого класса (виртуальные каналы);
- конфигурировать политику в соответствии с QoS для каждого компонента;
- указывать QoS для каждого пакета;
- создавать изохронные соединения.

Для поддержки QoS применяется маркировка трафика: каждый пакет TLP имеет 3-битное поле метки *класса трафика* (Traffic Class, TC). Это позволяет различать передаваемые данные по типам, создавать дифференцированные условия передачи трафика для разных классов. Порядок исполнения транзакций соблюдается в пределах одного класса, но не между разными классами. Для дифференцирования условий передачи трафика разных классов в коммутирующих элементах PCI Express могут создаваться виртуальные каналы. *Виртуальный канал* (Virtual Channel, VC) представляет собой физически обособленные наборы буферов и средств маршрутизации пакетов, занятые обработкой только трафика данного виртуального канала. На основе номеров виртуальных каналов и их приоритетов производится арбитраж при маршрутизации входящих пакетов. Каждый порт, поддерживающий виртуальные каналы, отображает пакеты определенных классов на соответствующие виртуальные каналы. При этом на один канал может отображаться произвольное число классов. По умолчанию весь трафик маркируется нулевым классом (TC0) и передается дежурным каналом (VC0). Виртуальные каналы вводятся по мере необходимости.

Сигнализация прерываний и управление энергопотреблением

Основной метод *сигнализации прерываний* в PCI Express — передача сообщений (MSI), причем с 64-битной адресацией (32-битная разрешена только для «наследных» устройств). Однако ради обеспечения программной совместимости устройство может использовать и эмуляцию прерываний через INTx#, передавая

эти запросы с помощью специальных пакетов. Получателем пакетов сигнализации прерываний (как MSI, так и эмуляции INTx#) обычно является контроллер прерываний, расположенный в корневом комплексе. Сигнализация INTx# производится пакетами класса TC0. Прерывания MSI в случае виртуальных каналов должны использовать класс трафика, соответствующий классу трафика данных, к которым относятся данные прерывания. Иначе возможно нарушение синхронизации из-за относительной неупорядоченности трафика разных классов. Синхронизации можно добиваться и теми же средствами, что и в PCI/ PCI-X, — чтением пакетов (пусть даже нулевой длины) через коммутатор (мост). Такой прием неизбежен, если прерывания относятся к данным нескольких разных классов (виртуальных каналов).

Сигнализация событий управления энергопотреблением возможна в двух вариантах: пакетная эмуляция сигнала PME# (аналогично эмуляции INTx#) и естественная сигнализация PCI Express с помощью соответствующих сообщений. При эмуляции PME# идентификация источника сигнала выполняется последовательным чтением конфигурационных регистров устройств, способных генерировать этот сигнал. Естественная сигнализация гораздо удобнее: идентификатор устройства-источника присутствует в сообщении.

Расширенное *управление энергопотреблением* (Power Management, PM) означает возможность:

- идентификации способностей к PM каждой функции;
- перевода функции в указанное состояние потребления;
- получения информации о текущем состоянии потребления функции;
- генерации запроса пробуждения при выключенном основном питании;
- последовательного включения устройств.

«Горячее» подключение

«*Горячие*» *подключение и замена* устройств могут выполняться с использованием как существующих механизмов (PCI Hot-Plug и Hot-Swap), так и естественных механизмов PCI Express, не требующих дополнительных сигналов. Ниже перечислены элементы стандартной модели «горячего» подключения:

- индикатор питания слота — запрещает извлечение/установку карты (мигание указывает на процесс перехода в обесточенное состояние);
- индикатор внимания — указывает на проблемы, связанные с устройством в данном слоте (мигание индикатора служит для поиска нужного слота);
- ручной фиксатор карты;
- датчик состояния ручного фиксатора — позволяет системному ПО обнаружить открытый замок;
- электромеханическая блокировка — не позволяет извлекать карту при включенном питании (специального сигнала для управления блокировкой не предусмотрено — если блокировка имеется, то она должна срабатывать прямо от питания порта);

- кнопка Внимание (Attention) — служит для запроса операции «горячего» подключения;
- программный интерфейс пользователя — позволяет запросить «горячее» подключение;
- система нумерации слотов — дает возможность визуально определить требуемый слот.

Надежность передачи и целостность данных

Для обеспечения *надежности транзакций* и *целостности данных* применяется CRC-контроль всех транзакций и управляющих пакетов. Запросчик считает транзакцию выполненной по получении подтверждающего сообщения от исполнителя (подтверждение отсутствует только для записей, отправленных в основную память). Обработка ошибок в минимальном варианте аналогична PCI, причем обнаруженные ошибки отображаются в конфигурационных регистрах функций (в регистре состояния). Расширенные возможности сообщений об ошибках дают исходную информацию для развитых процедур изоляции отказов и восстановления, а также мониторинга и регистрации (logging) ошибок. Ошибки делятся на три группы, что позволяет использовать адекватные процедуры восстановления:

- *исправимые* (correctable) ошибки автоматически вызывают аппаратную процедуру восстановления (повтора) и не требуют программного вмешательства для нормального исполнения транзакции;
- *неисправимые фатальные* (fatal) ошибки для надежного возобновления работы требуют сброса, в результате которого могут пострадать транзакции, не имеющие прямого отношения к ошибке;
- *неисправимые нефатальные* (non-fatal) ошибки не требуют сброса для возобновления работы — в результате этих ошибок могут быть потеряны лишь несколько транзакций, затронутых ошибкой.

Передача пакетов и пропускная способность соединения

Уровень транзакций формирует *пакеты* TLP, в которых содержатся код команды, адресная информация, данные и некоторые другие поля. Для обеспечения надежной доставки пакетов TLP канальный уровень при передаче снабжает их своим *заголовком*, содержащим 12-битный последовательный номер TLP, и 32-битным полем LCRC (CRC канального уровня). Таким образом, канальный уровень к каждому пакету TLP добавляет 6 байт накладных расходов. На каждый пакет TLP передатчик должен получить *положительное подтверждение Ack* — пакет канального уровня (*DLLP*). Если подтверждение не приходит, то механизм тайм-аута заставляет передатчик повторить посылку пакета. Предусмотрен и пакет *отрицательного подтверждения Nak*, вызывающий повторную передачу без ожидания.

Физический уровень вводит свое обрамление передаваемых пакетов: перед началом пакета передается специальный символ *STP* (для TLP-пакета) или *SDP*

(для DLLP-пакета); после пакета — символ *END*. Эти специальные символы отличаются от символов, представляющих данные после кодирования 8B/10B.

Рассмотрев структуры пакетов, можно оценить «скорострельность» базового соединения PCI Express (разрядность — 1 бит, скорость — 2,5 Гбит/с).

Самая короткая транзакция — запись двойного слова в пространство ввода-вывода — в прямом канале транзакций записи в порт занимает 128 нс (0,128 мкс), в обратном — 80 + 32 = 112 нс. Если подсчитать максимальную скорость передачи данных при непрерывных записях в порт, получаем v = 4/0,128 = 31,25 Мбайт/с. При этом будет занят и встречный канал с коэффициентом загрузки 112/128 = = 0,875. Результат по скорости близок к возможностям стандартной шины PCI (32 бит и 33 МГц), в которой такая транзакция требует четырех тактов шины. Чтение портов ввода-вывода на PCI Express даст те же результаты (на PCI результат будет хуже).

Теперь возьмем самый выгодный (в состязаниях по производительности) вариант транзакции: запись в память пакета 1024 двойных слов (с короткой 32-битной адресацией). Здесь скорость передачи данных составляет 4096/16,5 ≈ 248 Мбайт/с — это уже уровень производительности PCI (32 бит и 66 МГц) при длинных пакетных передачах. Загрузка встречного канала подтверждениями канального уровня в этом случае пренебрежимо мала. Скорость чтения из памяти будет немного ниже, поскольку каждая транзакция чтения состоит из двух пакетов TLP — запроса чтения и пакета завершения с данными.

Если встречный канал удается загрузить полезным трафиком, то можно говорить об удвоении пропускной способности PCI Express за счет возможности работы в полнодуплексном режиме. Однако в примере с записью в порт ввода-вывода о таком удвоении речи быть не может, поскольку встречный канал загружен довольно плотно. Если пересчитать полезную скорость на один сигнальный контакт разъема, то в самом выгодном полнодуплексном варианте получаем 248 x 2/4 = 124 Мбайт/с на контакт. Для сравнения можно взять вариант PCI-X533, обеспечивающий пиковую скорость записи, приближающуюся к 533 x 4 = 2132 Мбайт/с. В операциях чтения памяти PCI-X выглядит гораздо скромнее — пиковая скорость всего 533 Мбайт/с. При этом используется около 50 сигнальных контактов (не считая многочисленных земляных), так что на каждый контакт приходится примерно по 10-40 Мбайт/с. В порте AGP при той же пиковой скорости сигналов еще больше, так что заявления о высокой эффективности использования контактов в PCI Express имеют под собой основу. Полнодуплексный режим работы ни в PCI/PCI-X, ни в AGP невозможен.

Напомним, что данные подсчеты производились для базового соединения (x1,

1 линия); применив интерфейс x32 (32 линии), можно получить максимальную скорость записи в память 248 x 32 = 7936 Мбайт/с. А если брать полную загрузку полнодуплексного соединения, то PCI Express может обеспечить суммарную пропускную способность 15 872 Мбайт/с. Таким образом, в самом мощном варианте PCI Express оставляет далеко позади порт AGP с его пиком 2132 Мбайт/с. Правда, говорить о малом числе контактов уже не приходится — канал PCI Express x32 требует 2 x 2 x 32 = 128 сигнальных контактов (в AGP их меньше).

В настоящее время на системных платах вместо AGP стали устанавливать PCI Express x16.

Физический уровень и конструктивы PCI Express

Физический уровень интерфейса допускает как электрическую, так и оптическую реализацию. *Базовое соединение электрического интерфейса* (x1) состоит из двух дифференциальных низковольтных сигнальных пар — передающей (сигналы PETp0, PETn0) и принимающей (PETp0, PETn0). В интерфейсе применена развязка передатчиков и приемников по постоянному току, что обеспечивает совместимость компонентов независимо от технологии их изготовления и снимает некоторые проблемы передачи сигналов. Для передачи используется самосинхронизирующееся кодирование, что позволяет достигать высоких скоростей передачи. Базовая скорость — 2,5 Гбит/с «сырых» данных (после кодирования 8В/10В) в каждую сторону, в перспективе планируются и более высокие скорости. Для масштабирования пропускной способности возможно *агрегирование сигнальных линий* (сигнальных пар в электрическом интерфейсе) по одинаковому числу в обоих направлениях. Спецификация рассматривает варианты соединений из 1, 2, 4, 8, 12, 16 и 32 линий (обозначаются как xl, x2, x4, x8, x12, x16 и x32); передаваемые данные между ними распределяются побайтно. Таким образом достижима скорость до 32 x 2,5 = 80 Гбит/с, что примерно соответствует пиковой скорости 8 Гбайт/с. Во время аппаратной инициализации в каждом соединении согласуются число линий и скорость передачи; согласование выполняется на низком уровне без какого-либо программного участия. Согласованные параметры соединения действуют на все время последующей работы.

Обеспечение «горячего» подключения на физическом уровне PCI Express не требует каких-либо дополнительных аппаратных затрат, поскольку двухточечное соединение не затрагивает «лишних» участников. Безопасная коммутация сигналов не требуется, возможности подключаемого устройства никак не влияют на режимы работы остальных устройств.

Малое число сигнальных контактов интерфейса дает большую свободу в выборе *конструктивных реализаций* PCI Express:

- соединение компонентов в пределах платы;
- слоты и карты расширения в конструктивах PC/AT и ATX;
- внутренние и внешние карты расширения мобильных ПК;
- малогабаритные модули ввода-вывода для серверов и коммуникационной аппаратуры;
- модули для промышленных компьютеров;
- разъемное подключение «дочерних» карт (mezannine interface);
- кабельные соединения блоков.

Для карт расширения в конструктивах PC/AT и ATX предусматриваются разные модификации разъема-слота PCI Express, различающиеся числом пар сигнальных линий (xl, x4, x8, x16) и, соответственно, размером (рис. 14.9). При этом в слоты большего размера можно устанавливать карты с разъемом того же

размера (или меньшего — это называется *Up-plugging).* Однако противоположный вариант (Down-plugging) — установка большой карты в меньший слот — механически невозможен (в PCI/PCI-X возможен). Как было показано ранее, самый «слабый» вариант PCI Express обеспечивает пропускную способность на уровне стандартной шины PCI. Назначение контактов слотов PCI-express приведено в табл. 14.4.

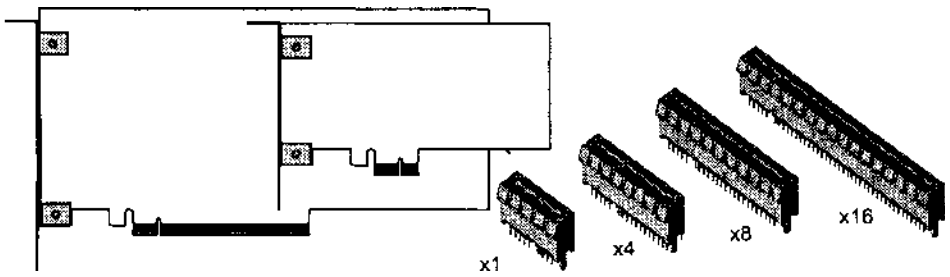


Рис. 14.9. Карты и разъемы PCI Express

Таблица 14.4. Разъемы PCI Express

№	Ряд B	Ряд A	№	Ряд B	Ряд A
1	+12V	PRSNT1#	23	PETp2	GND
2	+12V	+12V	24	PETn2	GND
3	Резерв	+12V	25		
4	GND	GND	26	GND	PERn2
5	SMB_CLK	TCK	27	PETp2	GND
6	SMB_DATA	TDI	28	PETn2	GND
7	GND	TDO	29	GND	PERp3
8	+3,3 V	TMS	30	Резерв	PERn3
9	TRST#	+3,3 V	31	PRSNT2#	GND
10	+3,3 Vaux	+3,3 V	32	GND	Резерв
11	WAKE#	PERST#	*Конец x4-коннектора*		
Ключ			33	PETp4	Резерв
12	Резерв	GND	34	PETn4	GND
13	GND	REFCLK+	35	GND	PERp4
14	PETp0	REFCLK-	36	GND	PERn4
15	PETn0	GND	37	PETp5	GND
16	GND	PERp0	38	PETn5	GND
17	PRSNT2#	PERn0	39	GND	PERp5
18	GND	GND	40	GND	PERn5
Конец x1-коннектора			41	PETp6	GND
19	PETp1	Резерв	42		
20	PETn1	GND	43	GND	PERp6
21	GND	PERp1	44	GND	PERn6
22	GND	PERn1	45	PETp7	GND

№	Ряд B	Ряд A	№	Ряд B	Ряд A
46	PETn7	GND	53	GND	PERn8
47	GND	PERp7	54	PETp9	GND
48	PRSNT2#	PERn7		...	...
49	GND	GND	79	PETn15	GND
Конец x8-коннектора			80	GND	PERp15
50	PETp8	Резерв	81	PRSNT2#	PERn15
51	PETn8	GND	82	GND	GND
52	GND	PERp8	*Конец x16-коннектора*		

Набор сигналов интерфейса PCI Express невелик:

- PETp0, PETn0,..., PETp15, PETn15 — выходы передатчиков сигнальных пар 0...15;
- PERp0, PERn0, ..., PERp15, PERn15 — входы приемников;
- REFCLK+ и REFCLK — сигналы опорной частоты 100 МГц;
- PERST# — сигнал сброса карты;
- WAKE# — сигнал «пробуждения» (от карты);
- PRSNT1#, PRSNT2# — сигналы обнаружения подключения-отключения карты для системы «горячего» подключения. На карте эти цепи соединяются между собой, причем для PRSNT2# выбирается контакт с самым большим номером. Это позволяет точнее отслеживать моменты подключения-отключения (в случае наклона карты). Для определения числа линий подключенной карты данные линии не используются — разрядность линий определяется автоматически при установлении соединения (в процедуре тренировки).

Дополнительно на слоте имеются необязательные сигналы шины SMBus (SMB_CLK и SMB_DATA) и интерфейса JTAG (TCLK, TDI, TDO, TMS, TRST#).

На карты подается основное *питание* +3,3V, +12V и дополнительное +3,3Vaux.

С интерфейсом PCI Express удобно компонуются модули ввода-вывода и сетевых интерфейсов для серверов и коммуникационных устройств стоечного исполнения. Такие модули могут быть довольно компактными (высота 2U не вызывает проблем размещения разъема), при этом производительности интерфейса достаточно даже для таких критичных модулей, как Fibre Channel, Gigabit Ethernet (GbE),10GbE.

Интерфейс PCI Express принимается и для промышленных компьютеров, для чего имеются спецификации PICMG 3.4 (малогабаритные конструктивы для xl, x2 и x4), а также конструктивы в формате Compact PCI.

Интерфейс PCI Express существует и в *кабельном исполнении* для кабельных соединений блоков, находящихся на небольшом удалении друг от друга. Так, по PCI Express можно подключать док-станции к блокнотным ПК. Возможность вывода интерфейса системного уровня за пределы корпуса компьютера из предшественников PCI Express поддерживала только шина ISA, и то лишь при низких скоростях обмена (на частотах до 5 МГц). Из новых последовательных интерфейсов системного уровня эта возможность имеется в InfiniBand. Наличие кабельного варианта высокопроизводительного интерфейса системного

уровня может позволить отойти от традиционной компоновки компьютера, при которой в системном блоке концентрируются все компоненты, требующие интенсивного обмена с ядром компьютера.

14.11. Шины расширения блокнотных ПК

В настоящее время малогабаритные компьютеры (Notebook — блокнотные ПК) получают все более широкое распространение, и для них существует ряд стандартных конструктивов карт расширения и спецификаций интерфейсов. Здесь представлены все три поколения: ISA-подобные шины в PC Card, PCI (CardBus) в различных конструктивах и PCI Express. Основные характеристики этих интерфейсов приведены в табл. 14.5.

Таблица 14.5. Конструктивы и интерфейсы периферии портативных ПК

Параметр	PC Card	Small PC Card	ExpressCard	Small PCI	Mini PCI Type I и II	Mini PCI Type III
Длина, мм	85,6	42,8	75	85,6	70/78	51/44,6
Ширина (по стороне с разъемом), мм	54,0	45,0	34/54	54,0	46	60
Толщина, мм	3,3/5,0/10,5	3,3/5,0/10,5	5	3,3/5,0/10,5	7,5/5,5/17,5	5
Коннектор	Штырьковый, 1,27 × 1,27 мм	Штырьковый, 1,27 × 1,27 мм	Ножевой однорядный, шаг 1 мм	Штырьковый, 0,8 × 1 мм	Штырьковый, 0,8 × 1 мм	Печатный, 0,8 мм
Число контактов	68	68	26	108	100	124
Интерфейсы	Память, ввод-вывод, ATA, CardBus (PCI)	Память, ввод-вывод, ATA, CardBus (PCI)	PCI Express, USB 2.0	PCI	PCI, PCI-X	PCI, PCI-X

Конструктивы Small PCI, Mini PCI и Mini PCI Express

SPCI (Small PCI) — спецификация PCI в миниатюрном исполнении, прежде называвшаяся *SFF PCI* (Small Form-Factor PCI). По размерам карты SPCI *Style A* и *Style B* совпадают с *PC Card* и *Card Bus Type II* и *III* соответственно, но специальные ключи предотвращают ошибки подключения. Карты SPCI могут быть трех видов: с питанием 5 В, 3,3 В и универсальные 5/3,3 В; ошибочная установка предотвращается механическими ключами — прорезями в направляющих на карте и выступами в направляющих сокета (рис. 14.10). Шина SPCI является внутренней (карты расширения находятся под крышкой корпуса и устанавливаются изготовителем при выключенном питании) и поэтому не нацелена на замену Card Bus (шины для внешних подключений с возможностью «горячей» замены).

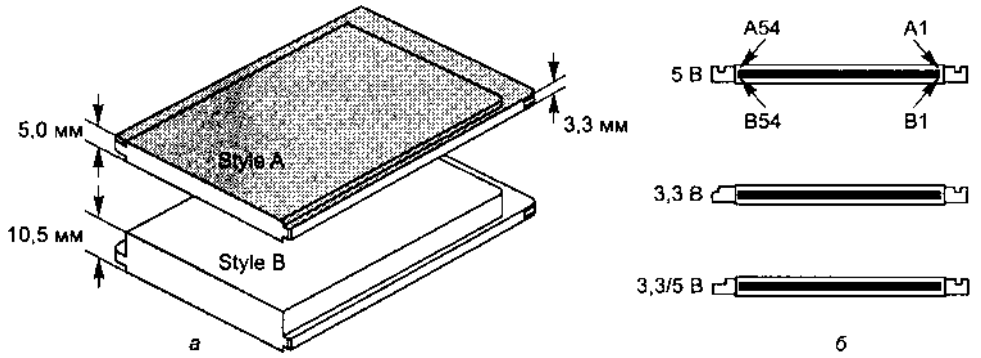


Рис. 14.10. Карты Small PCI: а — карты Style A и Style B, б — разъемы и ключи

Mini PCI Specification — малогабаритный вариант карт расширения с шиной PCI, устанавливаемых внутрь блокнотного ПК изготовителем. На разъеме Mini PCI имеются дополнительные сигналы для подключения аудиокодека AC'97, аналоговых аудиосигналов, телефонной линии, интерфейса локальной сети, а также отдельная линия +5 В для питания аналоговых цепей. В отличие от карт Small PCI и PC Card (PCMCIA), карты Mini PCI не имеют защитного корпуса — их элементы открыты, как и у «больших» карт PCI. В спецификации имеются несколько вариантов конструктивных исполнений:

- *Type IA* и *Type IB* (рис. 14.11, *а)* — для карт, устанавливаемых не у края корпуса компьютера. Системный разъем — штырьковый двухрядный стековый (штырьки контактов перпендикулярны плоскости платы), карты *Type IB* отличается меньшей допустимой высотой компонентов.

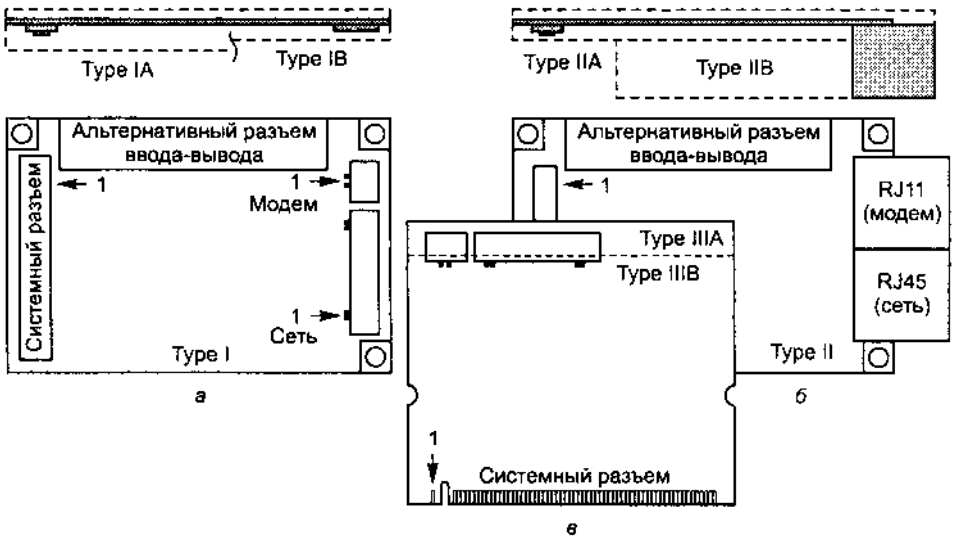


Рис. 14.11. Карты Mini PCI: а —Type I, б —Type II, в —Type I

- *Type IIA* и *Type IIB* (рис. 14.11, *б)* — для карт, устанавливаемых у края корпуса компьютера. Для подключения телефона и локальной сети предусмотрена установка гнезд RJ-11 и RJ-45 в экранированных корпусах. На карте *Type HA* компоненты могут иметь большую высоту (до 13,5 мм) на большей части платы; на картах *Type ИВ* такая высота допустима только для самих гнезд.
- *Type IIIA* и *Type IIIB* (рис. 14.11, *в)* — для карт, устанавливаемых не у края корпуса компьютера. Здесь применен иной системный разъем — двухсторонний печатный с шагом контактов 0,8 мм.

Конструктив *Mini PCI Express* (рис. 14.12) происходит от Mini PCI Type IIIA, но на месте одной карты Mini PCI можно разместить пару карт Mini PCI Express. На разъем карты помимо PCI Express (x1) выведены интерфейсы последовательных шин USB 2.0 и SMBus, питание +3,3 В (750 мА основное и 250 мА дополнительное) и +1,5 В (375 мА). Кроме того, выведены 3 сигнала управления светодиодными индикаторами состояния.

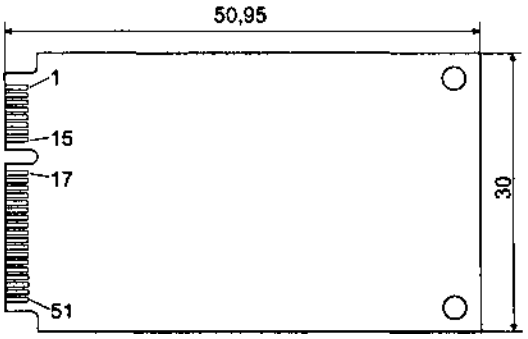


Рис. 14.12. Карты Mini PCI Express

Карты PCMCIA: интерфейсы PC Card, CardBus и Express Card

В начале 90-х годов организация *PCMCIA* (Personal Computer Memory Card International Association — международная ассоциация производителей карт памяти для персональных компьютеров) начала работы по стандартизации шин расширения блокнотных компьютеров, в первую очередь предназначенных для расширения памяти. Первым появился стандарт *PCMCIA* (1990 г.), в котором были описаны 68-контактный интерфейсный разъем и два типоразмера карт памяти: *Type I* и *Type II PC Card.* Следующая версия *PCMCIA 2.0* (1991 г.) для того же разъема определила интерфейс операций ввода-вывода. Позже были добавлены спецификация *PC CardATA* и новый типоразмер *Type III.* Стандарт *PC Card* (1995 г.) явился продолжением предыдущих; в нем введены дополнительные требования для улучшения совместимости и новые возможности: питание 3,3 В, поддержка DMA и 32-битной шины PCI — *CardBus.* В дальнейшем в стандарт были введены и другие дополнительные возможности.

Все карты PCMCIA и PC Card имеют 68-контактный разъем, назначение контактов у которого варьируется в зависимости от типа интерфейса карты. Тип интерфейса «заказывается» картой при установке ее в слот, который, естественно, должен поддерживать требуемый интерфейс. *Интерфейс памяти* обеспечивает 8- и 16-битные обращения с минимальным временем цикла 100 нс, что дает максимальную производительность 10 и 20 Мбайт/с соответственно. *Интерфейс ввода-вывода* имеет минимальную длительность цикла 255 нс, что соответствует 3,92 и 7,84 Мбайт/с для 8- и 16-битных обращений. *Интерфейс CardBus* поддерживает протокол обмена PCI; тактовая частота — 33 МГц, разрядность — 32 бита. Здесь используется та же система автоматического конфигурирования, что и в PCI (через регистры конфигурационного пространства). В интерфейс заложены дополнительные возможности для цифровой передачи аудиосигнала, причем как в традиционной форме ИКМ, так и в новой (забытой старой) форме ШИМ (PWM). Для дисковых *устройств ATA* в формате PC Card имеется специальная спецификация интерфейса.

Существует несколько конструктивных типов PC Card (рис. 14.13); у всех карт этих типов размер в плане составляет 54 x 85,5 мм, но разная толщина (меньшие адаптеры встают в большие гнезда):

- *PC Card Type I* — 3,3 мм — карты памяти;
- *PC Card Type II* — 5 мм — карты устройств ввода-вывода, модемы, адаптеры локальных сетей;
- *PC Card Type III* — 10,5 мм — дисковые устройства хранения;
- *PC Card Type IV* — 16 мм (упоминания об этом типе карт на сайте http:// www.pc-card.com найти не удалось).

Есть еще и маленькие карты *Small PC Card* размером 45 x 42,8 мм с тем же коннектором и теми же типами по толщине.

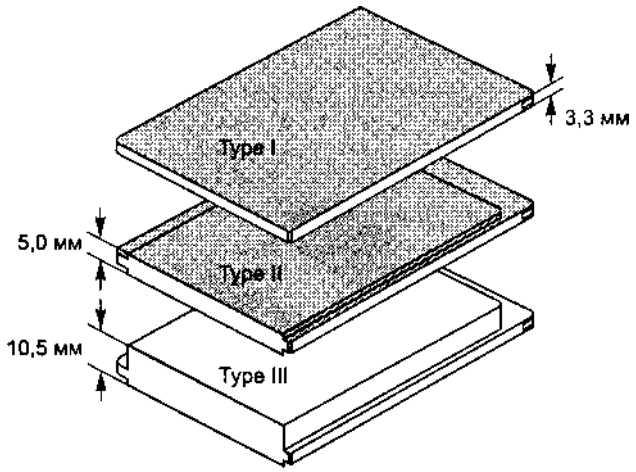


Рис. 14.13. Карты PC Card (PCMCIA)

Большинство выпускаемых карт PC Card поддерживают технологию PnP и предусматривают «горячее» подключение — интерфейсные карты могут вставляться и выниматься без выключения компьютера. Для этого контакты шин питания имеют большую длину, чем сигнальные, обеспечивая их упреждающее подключение и запаздывающее отключение. Два контакта обнаружения карты (card detect), CD1# и CD2#, короче остальных — их замыкание для хоста означает, что карта полностью вставлена в слот. Несмотря на возможность динамического конфигурирования, в некоторых случаях при изменении конфигурации требуется перезагрузка системы.

В стандарте PC Card выпускают самые разнообразные устройства — память, устройства хранения, коммуникационные средства, интерфейсные порты, игровые адаптеры, мультимедийные устройства и т. п.; правда, все они существенно дороже своих крупногабаритных аналогов. Через слот PC Card портативные компьютеры могут подключаться к док-станциям, в которые может быть установлена обычная периферия. Недостаточно строгое следование производителей стандарту иногда приводит к проблемам совместимости.

Слоты PC Card подключаются к системной шине блокнотного ПК через мост; для компьютеров с внутренней шиной PCI это мост PCI/PC Card. В блокнотных ПК могут быть и слоты *SPCI* (см. выше), но они недоступны без вскрытия корпуса и не допускают «горячей» замены устройств.

Настольный ПК можно снабдить слотами PC Card с помощью специальной карты адаптера-моста, устанавливаемой в слот PCI или ISA. Сами слоты (1-2 штуки) оформляются в корпус трехдюймового устройства и выводятся на лицевую панель ПК; этот корпус соединяется с картой-мостом ленточным кабелем- шлейфом.

С появлением PCI Express организация PCMCIA ввела конструктив *ExpressCard* (рис. 14.14), для которого на системный разъем выводится два интерфейса: PCI Express (1x) и USB 2.0. Модули ExpressCard компактнее прежних карт PCMCIA (PC Card и CardBus); предлагаются две модификации, различающиеся по ширине: ExpressCard/34 (34 x 75 x 5 мм) и ExpressCard/54 (54 x 75 x x 5 мм). Толщина модулей всего 5 мм, но если требуется, более длинные модули могут иметь утолщения в части, выходящей за габариты корпуса компьютера (за пределами 75 мм от края разъема). Как и прежние карты PCIMCIA, карты ExpressCard доступны пользователям и поддерживают «горячее» подключение.

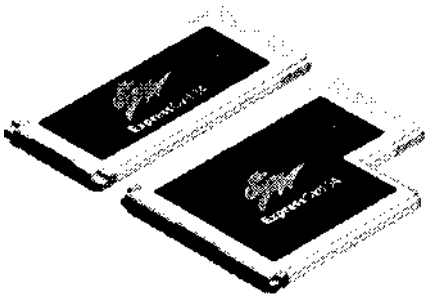


Рис. 14.14. Карты ExpressCard

ГЛАВА 15

Параллельный интерфейс — LPT-порт

Порт параллельного интерфейса был введен в PC для подключения принтера. Отсюда и пошло его название — LPT (Line Printer Terminal — порт построчного принтера). Традиционный, он же стандартный, *LPT-порт* (называемый еще *SPP-портом*) ориентирован на вывод данных, хотя с некоторыми ограничениями позволяет и вводить данные. Существуют различные модификации LPT- порта — двунаправленный, EPP, ECP и др., расширяющие его функциональные возможности, повышающие производительность и снижающие нагрузку на процессор. Поначалу они являлись фирменными решениями отдельных производителей, позднее был принят стандарт IEEE 1284.

С внешней стороны порт имеет 8-битную шину данных, 5-битную шину сигналов состояния и 4-битную шину управляющих сигналов, выведенные на разъем-розетку DB-25S. В LPT-порте используются логические уровни ТТЛ, что ограничивает допустимую длину кабеля из-за невысокой помехозащищенности ТТЛ-интерфейса. Гальваническая развязка отсутствует — схемная «земля» подключаемого устройства соединяется со схемной «землей» компьютера. Из-за этого порт является уязвимым местом компьютера, страдающим при нарушении правил подключения и заземления устройств. Поскольку порт обычно располагается на системной плате, в случае «выжигания» порта зачастую выходит из строя и его ближайшее окружение, вплоть до выгорания всей системной платы.

С программной стороны LPT-порт представляет собой набор регистров, расположенных в адресном пространстве ввода-вывода. Регистры порта адресуются относительно базового адреса порта, стандартными значениями которого являются 3BCh, 378h и 278h. Порт может использовать линию запроса аппаратного прерывания, обычно IRQ7 или IRQ5. В расширенных режимах может использоваться и канал DMA.

Практически все современные системные платы (еще начиная с PCI-плат для процессоров 486) имеют встроенный адаптер LPT-порта. На старых картах ISA LPT-порт чаще всего соседствует с парой COM-портов, а также с контроллерами дисковых интерфейсов (FDC и IDE). Кроме того, LPT-порт обычно присутствует на плате старинного дисплейного адаптера MDA (монохромный текстовый) и HGC (монохромный графический «Геркулес»). Есть и карты PCI с дополнительными LPT-портами.

15.1. Традиционный LPT-порт

Традиционный LPT-порт называется *стандартным параллельным портом* (Standard Parallel Port, SPP), или SPP-портом. Названия сигналов порта и их назначение (графа SPP в табл. 15.1) соответствуют интерфейсу Centronics (см. 11.5), для которого и вводился данный порт. В графах ECP и EPP таблицы приводятся названия сигналов для одноименных режимов, описанных далее.

Таблица 15.1. Разъем и сигналы LPT-порта

Контакт	№ провода	Назначение				
		I/O[1]	Бит[2]	SPP	ECP	EPP
1	1	O/I	CR.0\	Strobe#	HostClk	Write#
2	3	O(I)	DR.0	Data 0	Data 0	Data 0
3	5	O(I)	DR.1	Data 1	Data 1	Data 1
4	7	O(I)	DR.2	Data 2	Data 2	Data 2
5	9	O(I)	DR.3	Data 3	Data 3	Data 3
6	11	O(I)	DR.4	Data 4	Data 4	Data 4
7	13	O(I)	DR.5	Data 5	Data 5	Data 5
8	15	O(I)	DR.6	Data 6	Data 6	Data 6
9	17	O(I)	DR.7	Data 7	Data 7	Data 7
10	19	I[3]	SR.6	Ack#	PeriphClk	INTR#
11	21	I	SR.7\	Busy	PeriphAck	Wait#
12	23	I	SR.5	PaperEnd	AckReverse#	–[4]
13	25	I	SR.4	Select	Xflag	–[4]
14	2	O/I	CR.1\	Auto LF#	HostAck	DataStb#
15	4	I	SR.3	Error#	PeriphRequest#	–[4]
16	6	O/I	CR.2	Init#	ReverseRequest#	Reset#
17	8	O/I	CR.3\	Select In#	1284Active	AddrStb#
18–25	10, 12, 14, 16, 18, 20, 22, 24, 26	–	–	–	–	–

[1] I/O задает направление передачи (вход-выход) сигнала порта. O/I обозначает выходные линии, состояние которых считывается при чтении из портов вывода; O(I) — выходные линии, состояние которых может быть считано только при особых условиях (см. далее).
[2] Символом отмечены инвертированные сигналы (1 в регистре соответствует низкому уровню линии).
[3] Вход Ack# соединен резистором (10 кОм) с питанием +5 В.
[4] Определяется пользователем.

Адаптер SPP-порта содержит три 8-битных регистра, расположенных по соседним адресам в пространстве ввода-вывода, начиная с базового адреса порта BASE (3BCh, 378h или 278h):

- *Регистр данных* (Data Register, DR), адрес = BASE. Данные, записанные в этот регистр, выводятся на выходные линии Data[7:0]. Данные, считанные из этого регистра, в зависимости от схемотехники адаптера соответствуют либо

ранее записанным данным, либо сигналам на тех же линиях, что не всегда одно и то же.

- *Регистр состояния* (Status Register, SR) предназначен только для чтения, адрес = BASE+1. Регистр отображает 5-битный порт ввода сигналов состояния (см. табл. 15.1) и флаг прерывания (SR.2).
- *Регистр управления* (Control Register, CR), адрес = BASE+2, допускает запись и чтение. Регистр связан с 4-битным портом вывода управляющих сигналов (биты 0-3, см. табл. 15.1), для которых возможно и чтение; выходной буфер обычно имеет тип «открытый коллектор». Это позволяет корректно использовать линии данного регистра как входные при программировании их в высокий уровень. Бит CR.5 управляет направлением передачи (1 — ввод, только для портов PS/2, см. далее). Бит CR.4 разрешает прерывание по спаду сигнала на линии Ack# — сигнал запроса следующего байта в протоколе Centronics.

Любые протоколы обмена с устройствами через стандартный LPT-порт реализуются программно, чтением и записью регистров порта. Например, для вывода одного байта по протоколу Centronics требуется как минимум 4-5 операций ввода-вывода с регистрами порта. Отсюда вытекает главный недостаток обмена через стандартный порт — невысокая скорость при значительной загрузке процессора. Порт удается разогнать до скоростей 100-150 Кбайт/с при полной загрузке процессора.

Другой недостаток — неудобство использования LPT-порта в качестве порта ввода. Стандартный порт асимметричен — при наличии 12 линий (и битов), нормально работающих на вывод, на ввод работает только 5 линий состояния. Если необходим ввод данных, на всех стандартных портах работоспособен *режим полубайтного обмена* (Nibble Mode). В этом режиме, называемом также *Hewlett Packard Bi-tronics*, одновременно принимаются 4 бита данных, пятая линия используется для квитирования. Таким образом, каждый байт принимается за два цикла, а каждый цикл требует по крайней мере 5 операций ввода-вывода.

15.2. Расширения параллельного порта

Недостатки стандартного порта частично были преодолены в новых типах портов, появившихся в компьютерах PS/2.

Двунаправленный порт 1 (*Type 1 parallel port)* — интерфейс, введенный в PS/2. Такой порт помимо стандартного режима может работать в режиме ввода или в двунаправленном режиме. Протокол обмена формируется программно, а для указания направления передачи в регистр управления порта введен специальный бит CR. 5: 0 — буфер данных работает на вывод, 1 — на ввод. Данный тип порта «прижился» и в обычных компьютерах, в CMOS Setup он может называться PS/2 или Bi-Di.

Порт с прямым доступом к памяти (*Type 3 DMA parallel port)* применялся в PS/2 моделей 57, 90, 95. Был введен для повышения пропускной способности

и разгрузки процессора при выводе на принтер; при этом аппаратно реализуется протокол Centronics (Fast Centronics).
Позже появились другие адаптеры LPT-портов. Некоторые из них использовали FIFO-буфер данных — *Parallel Port FIFO Mode.* Не будучи стандартизованными, такие порты разных производителей требовали собственных специальных драйверов. Программы, напрямую управляющие регистрами стандартных портов, не могли задействовать их дополнительные возможности. Такие порты часто входили в состав мультикарт VLB. Существуют их варианты с шиной ISA, а также встроенные в системную плату.

15.3. Стандарт IEEE 1284

Стандарт на параллельный интерфейс *IEEE 1284,* принятый в 1994 году, описывает физический и канальный уровни интерфейсов (по модели OSI). Стандарт определяет 5 режимов обмена данными, метод согласования режима, физический и электрический интерфейсы. При описании режимов обмена фигурируют следующие понятия:

- *хост* — компьютер, обладающий параллельным портом;
- *ПУ* — периферийное устройство, подключаемое к этому порту;
- *прямой канал* — канал вывода данных от хоста в ПУ;
- *обратный канал* — канал ввода данных в хост из ПУ;
- *Ptr* — в названиях сигналов обозначает передающее ПУ.

Согласно IEEE 1284, возможны следующие режимы обмена данными через параллельный порт:

- *Режим совместимости* (compatibility mode). В этом режиме используется однонаправленный прямой 8-битный канал, программно управляемый хостом. Это базовый режим порта (исходное состояние порта), а также промежуточное состояние при всех переходах. Режим совместимости полностью соответствует SPP-порту. Остальные режимы рассмотрены в следующих разделах.
- *Полубайтный режим ввода* (nibble mode). В этом режиме используется однонаправленный обратный параллельно-последовательный (4-битный) канал, программно управляемый хостом. Служит дополнением к режиму совместимости, одновременно с ним работать не может (требуется переключение под управлением хоста).
- *Байтный режим ввода* (byte mode). В этом режиме используется однонаправленный обратный 8-битный канал, программно управляемый хостом. Как и предыдущий, байтный режим служит дополнением к режиму совместимости.
- *Режим EPP* (EPP mode). В этом режиме используется двунаправленный 8-битный канал, управляемый процессором. Аппаратно реализованное блокирующее квитирование обеспечивает высокие скорости и ввода, и вывода. Обменом полностью управляет хост, направление передачи определяется инструкцией процессора *(IN* или *OUT),* а тип — регистром, к которому про

исходит обращение. Отдельные линии стробирования позволяют различать передачу данных и адресов.

- *Режим ECP* (ECP Mode). В этом режиме используется двунаправленный симметричный 8-битный канал. Нормальным направлением является прямая передача (от хоста), при которой интерфейсом управляет хост. Обратную передачу запрашивает ПУ; хост подтверждает реверс канала, после чего интерфейсом управляет ПУ, которое передает данные хосту. Блокирующее квитирование организовано аппаратно, но несколько иначе, чем в EPP. Управляющая линия позволяет различать передачу данных и команд; команды могут быть задействованы для адресации каналов и компрессии данных.

Хост является инициатором и исполнителем всех передач в режимах совместимости, байтном, полубайтном и EPP, а также прямых передач в ECP. Хост управляет арбитражем между прямыми и обратными передачами ECP.

ПУ инициирует обратные передачи с рядом ограничений. В байтном и полубайтном режимах оно может инициировать передачу, но только в фазе покоя. В режиме ECP оно может запросить шину и, только получив подтверждение, инициировать обратные передачи. В режиме EPP ПУ «бесправно».

Для определения *готовности данных от ПУ* хост выполняет опрос (чтением регистров порта). Возможно и использование прерываний по сигналу от ПУ, для этого хост должен периодически переводить интерфейс в состояние покоя обратного канала.

Полубайтный режим ввода

Полубайтный режим ввода (nibble mode) обеспечивает программно-управляемый медленный канал ввода по линиям, читаемым через регистр SR. Байт вводится тетрадами (nibble — полубайт, 4 бита) за два приема. По сигналу готовности хоста ПУ выдает сначала младшую тетраду (отвечая сигналом квитирования), затем старшую. После этого ПУ может сигнализировать о наличии следующего байта данных для чтения и о занятости (busy) для прямого канала, а также вызывать прерывание по готовности данных. Взаимно блокирующее квитирование хост отрабатывает программно (через регистры CR и SR). В IEEE 1284.3 для порта, поддерживающего ECP, введен Полубайтный режим ввода с канальной адресацией (channelized nibble mode).

Полубайтный режим сильно нагружает процессор, и поднять скорость обмена выше 50 Кбайт/с не удается. Безусловное его преимущество в том, что он работает *на всех портах*. Его применяют в тех случаях, когда поток данных невелик (например, для связи с принтерами). Однако при связи с адаптерами локальных сетей, внешними дисковыми накопителями и накопителями CD-ROM прием больших объемов данных требует изрядного терпения со стороны пользователя.

Байтный режим ввода

В байтном режиме ввода (byte mode) данные принимаются через двунаправленный порт (в параметрах BIOS Setup — Bi-Di или PS/2), у которого выходной

буфер данных может отключаться установкой бита CR.5 = 1. Как и предыдущие, режим является программно-управляемым — все сигналы квитирования анализируются и устанавливаются драйвером. ПУ может сигнализировать о наличии данных, занятости прямого канала и вызывать прерывание по готовности данных. Достижима скорость до 150 Кбайт/с.

Режим EPP

Протокол *EPP* (Enhanced Parallel Port — улучшенный параллельный порт) был разработан компаниями Intel, Xircom и Zenith Data Systems задолго до принятия стандарта IEEE 1284. Этот протокол предназначен для повышения производительности обмена по параллельному порту, впервые был реализован в чипсете Intel 386SL (микросхема 82360) и впоследствии принят множеством компаний как дополнительный протокол параллельного порта. Версии протокола, реализованные до принятия IEEE 1284, отличаются от нынешнего стандарта (см. далее).

В режиме EPP аппаратная реализация протокола взаимно блокирующего квитирования позволяет поднять скорость до 2 Мбайт/с. В порт введены новые регистры, обращение к которым обеспечивает 4 типа циклов обмена:

- *запись данных* и *чтение данных* при обращениях к регистру EPP_Data инструкциями *OUT* и *IN* (соответственно);
- *запись адреса* и *чтение адреса* при аналогичных обращениях к регистру EPP_ Address.

Цикл записи адреса позволяет организовать внешнюю мультиплексированную шину адреса/данных. Это удобно для подключения ПУ, имеющих в своем составе множество регистров, — записью адреса выбирается один из регистров, после чего циклами чтения и записи данных производится обмен именно с ним. Цикл чтения адреса может быть использован, например, для чтения состояния устройства.

Внешний цикл обмена вложен в шинный цикл чтения/записи, порожденный инструкциями (*IN/OUT)* ЦП. Естественно, ПУ не должно «подвешивать» процессор на шинном цикле обмена. Это гарантирует механизм тайм-аутов PC, который принудительно завершает любой цикл обмена, длящийся более 15 мкс. В ряде реализаций EPP за тайм-аутом интерфейса следит сам адаптер — если ПУ не отвечает в течение определенного времени (5 мкс), цикл прекращается и в регистре состояния адаптера фиксируется ошибка. К сожалению, механизм сообщения об ошибке не стандартизован. В некоторых (но не во всех!) LPT- портах ошибка тайм-аута вызывает установку бита 0 регистра SR.

На рис. 15.1 приведена диаграмма *цикла записи* данных, иллюстрирующая внешний цикл обмена, *вложенный* в цикл записи системной шины процессора (иногда эти циклы называют *связанными*). В адресном цикле записи вместо сигнала DataStb# используется AddrStb#. Пример адресного *цикла чтения* приведен на рис. 15.2. Цикл чтения данных отличается только применением иного стробирующего сигнала. Заметим, что в цикле записи порт выставляет данные одновременно со стробом, так что пользоваться началом строба для защелкива

ния данных в устройстве нельзя. Протокол позволяет автоматически настраиваться на скорость обмена, доступную и хосту, и ПУ. ПУ может регулировать длительность всех фаз обмена с помощью всего лишь одного сигнала Wait#. Протокол автоматически подстраивается и под длину кабеля — вносимые задержки приведут только к удлинению цикла. Поскольку кабели, соответствующие стандарту IEEE 1284 (см. выше), имеют одинаковые волновые свойства для разных линий, нарушения передачи, связанного с «состязаниями» сигналов, происходить не должно. При подключении сетевых адаптеров или внешних дисков к EPP-порту можно наблюдать непривычное явление: снижение производительности по мере удлинения интерфейсного кабеля.

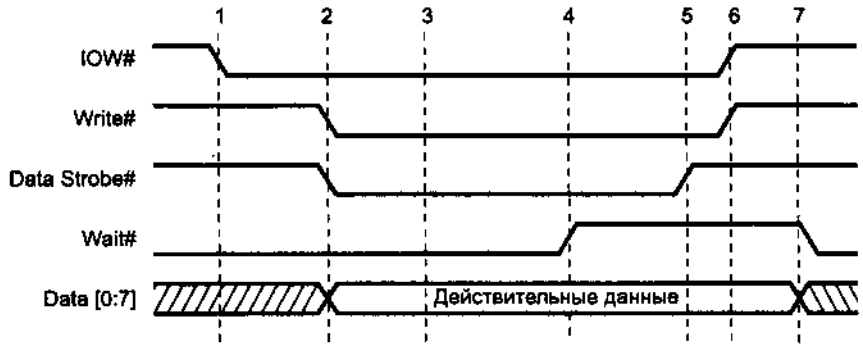


Рис. 15.1. Цикл записи данных EPP

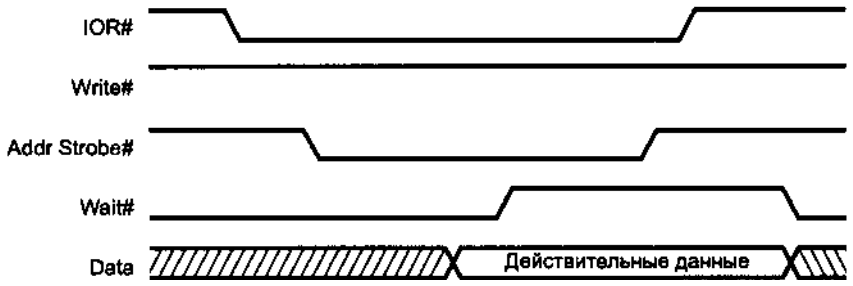


Рис. 15.2. Адресный цикл чтения EPP

EPP-порт имеет *расширенный набор регистров* (табл. 15.2), который занимает в пространстве ввода-вывода 5-8 смежных байтов. Для того чтобы использовать режим EPP, нужно его разрешить (в CMOS Setup) и в регистр CR занести значение, соответствующее покою интерфейса (05h). Далее обращения к регистрам EPP_Data и EPP_Addr будут вызывать соответствующие циклы обмена. К регистрам EPP можно обращаться и инструкциями REP INSB и REP OUTSB. Некоторые адаптеры допускают *16/32-битное обращение* к регистру данных EPP. Таким образом и обеспечивается производительность до 2 Мбайт/с. К регистру адреса обращаются только в 8-битном режиме.

Таблица 15.2. Регистры EPP-порта

Имя регистра	Смещение	Режим	R/W	Описание
SPP_Data (DR)	+0	SPP/EPP	W	Регистр данных SPP
SPP_Status (SR)	+1	SPP/EPP	R	Регистр состояния SPP
SPP_Control (CR)	+2	SPP/EPP	W	Регистр управления SPP
EPP_Address	+3	EPP	R/W	Регистр адреса EPP. Чтение или запись в него генерирует связанный цикл чтения или записи адреса EPP
EPP_Data	+4	EPP	R/W	Регистр данных EPP. Чтение (запись) генерирует связанный цикл чтения (записи) данных EPP
	+5...+7	EPP	N/A	В некоторых контроллерах могут использоваться для 16-32-битных операций ввода-вывода

Устройства с интерфейсом EPP, разработанные до принятия IEEE 1284, отличаются началом цикла: строб DataStb# или AddrStb# устанавливается независимо от состояния WAIT#. Это означает, что ПУ не может задержать начало следующего цикла (хотя может растянуть его на требуемое время). Такая спецификация называется *EPP 1.7* (предложена Xircom). Именно она применялась в контроллере 82360. Периферия, совместимая с IEEE 1284 EPP, будет нормально работать с контроллером EPP 1.7, но ПУ в стандарте EPP 1.7 может отказаться работать с контроллером EPP 1284.

Важной чертой режима EPP является то, что обращение процессора к ПУ осуществляется в реальном времени — нет буферизации. Драйвер способен отслеживать состояние и подавать команды в точно известные моменты времени. Циклы чтения и записи могут чередоваться в произвольном порядке или идти блоками. Такой тип обмена удобен для *регистроориентированных* ПУ, а также ПУ, работающих в *реальном времени*, например устройств сбора информации и управления. Этот режим пригоден и для устройств хранения данных, сетевых адаптеров, принтеров, сканеров и т. п. К сожалению, режим EPP поддерживается не всеми портами: к примеру, ряд блокнотных ПК его не поддерживают. Так что при разработке собственных устройств ради совместимости с компьютерами приходится ориентироваться на режим ECP.

Режим ECP

Протокол *ECP* (Extended Capability Port — порт с расширенными возможностями) был предложен Hewlett-Packard и Microsoft для связи с ПУ типа принтеров или сканеров. Как и EPP, данный протокол обеспечивает высокопроизводительный двунаправленный обмен данными хоста с ПУ, в котором взаимно блокированное квитирование реализовано аппаратно. Протокол ECP в обоих направлениях обеспечивает два типа циклов:

- циклы записи и чтения данных;
- командные циклы записи и чтения, служащие для канальной адресации и компрессии RLC.

Канальная адресация в ECP позволяет на одном интерфейсе организовать до 128 логических каналов в каждом направлении. Канальный адрес передается в командном цикле (бит 7 = 1, остальные биты несут канальный адрес). После согласования режима ECP канальные адреса в обоих направлениях считаются нулевыми и передачи данных в них идут по нулевым каналам. Канальный адрес, переданный хостом в команде записи, становится текущим и относится к последующим обменам данными в обоих направлениях. Если ПУ желает передать данные хосту по другому каналу, оно должно предварительно послать канальный адрес хосту. Этот адрес станет текущим только для передач от ПУ к хосту, для его смены потребуется посылка нового канального адреса хостом (действует в обеих направлениях) или ПУ (действует только на обратные пересылки). Канальная адресация позволяет подключать к порту сложные ПУ, содержащие несколько независимых блоков (например, принтер, сканер и модем — компьютерная факс-машина).

Компрессия RLC (Run-Length Count) позволяет заменить передачу повторяющихся байтов на передачу *команды RLC* (бит 7 = 0, биты [6:0] — счетчик повторов, 0 соответствует 128), за которой следует передача размножаемого байта данных. Таким образом, возможно сжатие до величины 64 : 1. Компрессия при передаче выполняется программно (при обмене по DMA она невозможна), при приеме порт делает декомпрессию аппаратно и помещает декомпрессированный поток в FIFO. ПУ может и не поддерживать компрессию, о чем сообщает при согласовании. Реально компрессия применяется редко.

В отличие от EPP, вместе с протоколом ECP сразу появился стандарт на программную (регистровую) модель его адаптера, изложенный в документе «The IEEE 1284 Extended Capabilities Port Protocol and ISA Interface Standard» компании Microsoft.

Протокол ECP переопределяет сигналы SPP (см. [6], а также табл. 15.1). Адаптер ECP тоже генерирует внешние протокольные сигналы квитирования аппаратно, но его работа существенно отличается от режима EPP.

На рис. 15.3 приведены диаграммы двух циклов прямой и обратной передачи: за циклом данных следует командный цикл. В отличие от диаграмм обмена EPP, на рисунке не приведены сигналы циклов системной шины. В данном режиме обмен программы с ПУ разбивается на два относительно независимых процесса, которые связаны через FIFO-буфер. Обмен драйвера с FIFO-буфером может осуществляться как программным вводом-выводом, так и посредством DMA. Для обмена посредством DMA к порту подключается канал стандартного контроллера DMA 8237A; доступ DMA эффективен при передаче больших блоков данных. Обмен ПУ с буфером аппаратно выполняет адаптер ECP. Драйвер в режиме ECP не имеет информации о точном состоянии процесса обмена, но обычно важно только то, завершен он или нет. За состоянием FIFO-буфера наблюдают либо по регистру ECR, либо по обслуживанию сервисных прерываний от порта.

В регистровой модели адаптера ECP (табл. 15.3) используются свойства архитектуры стандартной шины и адаптеров ISA, где для дешифрирования адресов портов ввода-вывода задействуются только 10 младших линий шины адреса. Поэтому, например, обращения по адресам Port, Port+400h, Port+800h... будут вое-

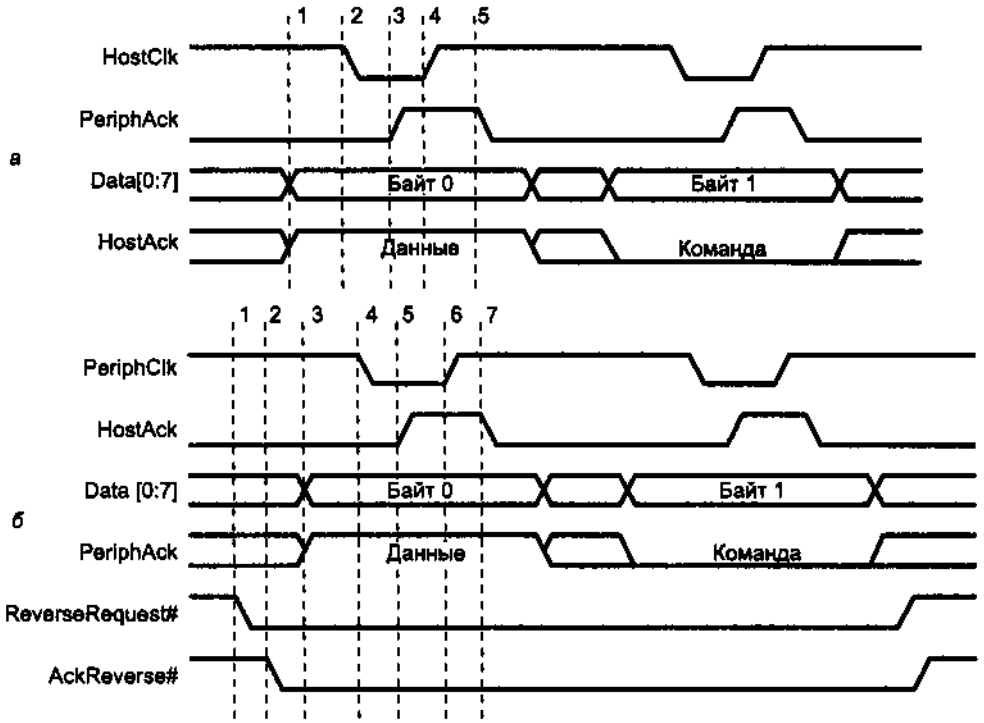


Рис. 15.3. Передача в режиме ECP: а — прямая, *б* — обратная

приниматься как обращения к адресу Port, лежащему в диапазоне 0-3FFh. Современные PC и адаптеры декодируют большее количество адресных битов, поэтому обращения по адресам 0378h и 0778h адресуются двум различным регистрам. Помещение дополнительных регистров ECP «за спину» регистров стандартного порта (смещение 400-402h) преследует две цели. Во-первых, эти адреса никогда не использовались традиционными адаптерами и их драйверами, и их применение в ECP не приведет к сужению доступного адресного пространства ввода-вывода. Во-вторых, этим обеспечиваются совместимость со старыми адаптерами на уровне режимов 000-001 и возможность определения факта присутствия ECP-адаптера посредством обращения к его расширенным регистрам.

Таблица 15.3. Регистры ECP

Смещение	Имя	R/W	Режимы ECP[1]	Название
000	DR	R/W	SPP, Bi-Di	Data Register
000	ECPAFIFO	R/W	ECP	ECP Address FIFO
001	SR	R/W	Все	Status Register
002	CR	R/W	Все	Control Register
400	SDFIFO	R/W	Fast Centronics	Parallel Port Data FIFO
400	ECPDFIFO	R/W	ECP	ECP Data FIFO

Смещение	Имя	R/W	Режимы ECP[1]	Название
400	TFIFO	R/W	Тест	Test FIFO
400	ECPCFGA	R	Конфиг	Configuration Register A
401	ECPCFGB	R/W	Конфиг	Configuration Register B
402	ECR	R/W	Все	Extended Control Register

[1] Регистры доступны только в данных режимах (указаны значения битов 7-5 регистра ECR). Порт ECP может работать в различных режимах: SPP, Bi-Directional, Fast Centronics, ECP, EPP, тестовом и конфигурационном. Каждому режиму ECP соответствуют (и доступны) свои функциональные регистры. Переключение режимов осуществляется записью в регистр ECR. «Дежурными» режимами, включаемыми по умолчанию, являются SPP или Bi-Di. В любом из них работает Полубайтный режим ввода. Из этих режимов всегда можно переключиться в любой другой. Для корректной работы интерфейса перед выходом из старших режимов необходимо дождаться завершения обмена по прямому доступу и очистки FIFO-буфера.

Режим *Fast Centronics* формально в IEEE 1284 не входит. Он обеспечивает вывод через FIFO (можно и с DMA), с аппаратной реализацией протокола Centronics; возможно использование прерываний по состоянию FIFO (по опустошению) или по завершении доступа DMA.

В IEEE 1284.3 введена модификация режима ECP — режим BECP (Bounded ECP), в котором изменены правила смены направления, чтобы избежать дублирования или потерь байта при смене направления в момент, когда ПУ начинает передавать данные.

Согласование режимов IEEE 1284

ПУ в стандарте IEEE 1284 обычно не требуют от контроллера реализации всех предусмотренных этим стандартом режимов. Для определения режимов и методов управления конкретным устройством стандарт предусматривает *последовательность согласования* (negotiation sequence). Последовательность построена так, что старые устройства, не поддерживающие IEEE 1284, на нее не ответят и контроллер останется в стандартном режиме. Периферия IEEE 1284 может сообщить о своих возможностях, и контроллер установит режим, удовлетворяющий и хост, и ПУ. Все переключения режимов выполняются через согласование. В последовательности согласования участвует сигнал Selectln# (контакт 17 порта). В старых (и дешевых) кабелях этот контакт может быть не задействован — с таким кабелем работа по IEEE 1284 невозможна.

Физический и электрический интерфейсы

Стандарт IEEE 1284 определяет физические характеристики приемников и передатчиков сигналов, которые по уровням совместимы с ТТЛ. Расширенные (функционально и по скорости передачи) режимы требуют четких спецификаций. IEEE 1284 описывает два уровня интерфейсной совместимости. *Первый уровень* (Level I) определен для устройств медленных, но использующих смену направления передачи данных. *Второй уровень* (Level И) определен для устройств, работающих в расширенных режимах с высокими скоростями и длинными кабелями.

Стандарт IEEE 1284 определяет три типа используемых *разъемов.* Типы *A (DB-25*) и *B* (*Centronics-36*) характерны для традиционных кабелей подключения принтера, тип *C* — новый малогабаритный 36-контактный разъем.

В стандарте IEEE 1284 определено 2 уровня качества кабелей:

- Level I — обычные кабели (один общий провод GND), как для Centronics, но обязательно со связью 17 (A)-36 (B); на дешевых кабелях этот провод иногда отсутствует, в этом случае работа устройств 1284 невозможна, поскольку программа требует «двунаправленный кабель». Логически кабели достаточны, но при длине более 2 м они не дают работать на высокой скорости.
- Level II — жгут витых пар, каждая сигнальная цепь имеет свой обратный провод (GND). Такой жгут позволяет работать на высоких скоростях (до 2 Мбайт/с) при длине до 10 м.

Подключение цепочек устройств и мультиплексоров

Изначально LPT-порт был рассчитан на монопольное использование: к одному порту подключается одно устройство. Стандарт *IEEE 1284.3* «Standard for Interface and Protocol Extensions to IEEE Std. 1284 Compliant Peripheral and Host Adapter Ports» вводит интерфейсные и протокольные расширения порта и устройств 1284 для прозрачной работы с множеством подключенных устройств. Устройства могут подключаться к порту цепочками или через мультиплексор (рис. 15.4).

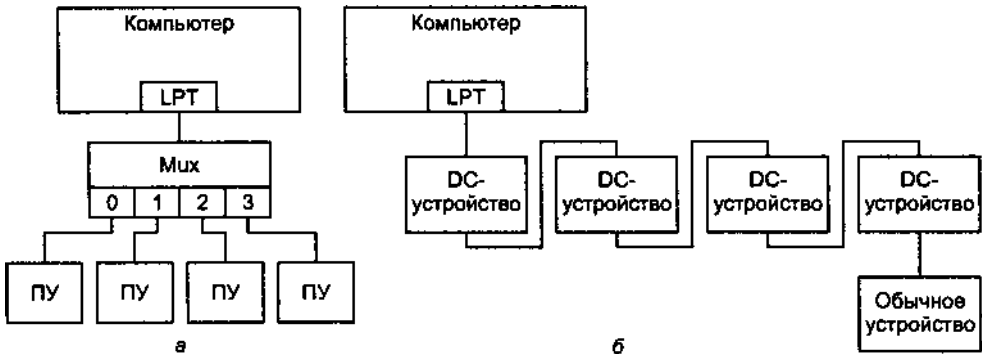


Рис. 15.4. Подключение множества устройств к LPT-порту: а — через мультиплексор, *б* — цепочками

Мультиплексор (Mux) — это специальное устройство, которое может подключаться только к порту хоста и программно коммутировать этот порт на один из своих выходных портов. К портам Mux (их бывает 2-4) могут подключаться любые ПУ (1284 и старые) и цепочки устройств.

Цепочечные устройства (Daisy Chain, DC) имеют 2 порта: один направлен в сторону хоста, к другому — проходному (pass through) — могут подключаться дру

гие DC-устройства. В цепочке может быть до 4 DC-устройств, последнее (пятое) устройство в цепочке может быть обычным (старым). Цепочки могут подключаться к LPT-порту и к портам Mux, но Mux может подключаться только к LPT-порту.

Для адресации, выбора и опроса устройств используется протокол CPP (Command Packet Protocol), прозрачный для устройств 1284 и старых устройств. В мультиплексоре и цепочечном устройстве расположен аппаратный агент, работающий с данным протоколом. В протоколе CPP информация передается по шине данных, но при этом управляющие сигналы остаются в состоянии покоя SPP-порта. Информационные байты передаются с чередованием прямого и инверсного представления на шине, что и позволяет их распознавать.

Организация DC-устройств требует пояснения. Интерфейсные сигналы LPT- порта делятся на 3 группы:

- *Состояние* (Status Line, SL) — входы для хост-порта. DC-устройство может быть для них *прозрачно* (пропускать от своего pass through-порта) или посылать по ним свое состояние.
- *Управление* (Control Line, CL) — выходы хост-порта. *Устройство* может *транслировать* на проходной порт сигналы от хоста или их защелкнутые значения (зафиксированные ранее).
- *Данные* (Data Line, DL) — двунаправленные линии, которые проходят между портами напрямую. Устройство к ним подключается своими шинными формирователями.

DC-устройство по умолчанию (по включению) *не выбрано* (unselected) и *не нумеровано* (unadressed). Не выбранное устройство прозрачно (линии SL и CL пропускаются насквозь, на DL ничего не выводится). Командой CPP всем DC-устройствам назначаются последовательные адреса (по порядку подключения). Далее командой CPP выбирается (по адресу) одно из устройств. Выбранное устройство не транслирует хосту состояние (SL) от «хвоста» цепочки. В «хвост» цепочки оно передает состояние управляющих линий, зафиксированное на момент выбора. Таким образом, диалог хоста происходит только с выбранным устройством. По уговору при *выборе* (select) хост и выбранное ПУ устанавливаются в режим совместимости; далее они могут согласовать любой режим 1284, передавать в нем данные, менять режимы. Перед *отсоединением* (un-select) они должны выполнить завершение (вернуться в режим совместимости). После этого можно выбрать другое устройство и работать с ним. DC- устройство должно выполнять обязательные команды CPP:

- назначить адрес — одним пакетом CPP (но с несколькими байтами команды) хост назначает подключенным устройствам номера (0-3), на каждом шаге он видит, есть ли еще ненумерованные устройства;
- выбрать (select) устройство по адресу;
- отсоединиться (un-select) от устройства;
- проверить прерывание от устройства по адресу.

Дополнительные (необязательные) команды могут использоваться для разрешения/запрета прерываний от выбранного DC-устройства и для генераций прерываний от указанного устройства.

Общение с *мультиплексорами* происходит аналогично. Мультиплексор имеет несколько портов (2-4), только один из них может быть выбран (selected). Выбранный порт подключает ПУ к хосту совершенно прозрачно; не выбранный на линии CL выводит защелкнутое состояние покоя в режиме совместимости, линии данных пассивны.

При подключении к одному порту множества устройств их ПО должно взаимодействовать через программную прослойку — интерфейс SPI (Service Provider Interface), который обеспечивает:

- нумерацию устройств (в полный логический адрес входят номер хост-адаптера, номер порта Mux, адрес DC-устройства в цепочке);
- передачу данных;
- разделение порта между несколькими SPI-клиентами;
- предоставление информации о хост-адаптере (адрес, режим, номер прерывания).

В нашей стране мультиплексоры встречаются редко, а цепочечных устройств довольно много: это и ключи защиты ПО, и внешние устройства хранения, подключаемые к LPT-порту, и другие устройства (не принтеры).

15.4. Системная поддержка LPT-порта

Системная поддержка LPT-порта включает поиск установленных портов и сервисы печати (Int 17h, см. 11.5). В процессе начального тестирования POST BIOS проверяет наличие параллельных портов по адресам 3BCh, 378h и 278h и помещает базовые адреса обнаруженных портов в ячейки BIOS Data Area 0:0408h, 040Ah, 040Ch, 040Eh. Эти ячейки хранят адреса портов LPT1-LPT4, нулевое значение адреса является признаком отсутствия порта с данным номером. В ячейки 0:0478, 0479, 047A, 047B заносятся константы, задающие тайм-аут для этих портов.

Поиск портов обычно ведется довольно примитивно — по базовому адресу (в регистр данных предполагаемого порта) выводится тестовый байт (AAh или 55h), затем производится ввод по тому же адресу. Если считанный байт совпал с записанным, предполагается, что найден LPT-порт; его адрес помещается в ячейку BIOS Data Area. Базовые адреса портов могут быть впоследствии изменены программно. Адрес порта LPT4 система BIOS самостоятельно установить не может, поскольку в списке стандартных адресов поиска имеются только три вышеуказанных.

Обнаруженные порты *инициализируются* — записью в регистр управления активируется и деактивируется сигнал Init#, после чего записывается значение 0Ch, соответствующее исходному состоянию сигналов интерфейса. В некоторых случаях сигнал Init# активен с момента аппаратного сброса до инициали

зации порта при загрузке ОС. Это можно заметить по поведению включенного принтера во время перезагрузки компьютера — у принтера надолго гаснет индикатор On-Line. Следствие этого явления — невозможность распечатки экранов (например, параметров BIOS Setup) по нажатию клавиши Print Screen до загрузки ОС.

Для поддержки режимов порта, определенных в IEEE 1284, введены функции EPP BIOS. До их использования необходимо проверить наличие функций EPP BIOS (эти функции присутствуют не во всех версиях BIOS). Проверка выполняется через функцию 2 Int 17h (опрос состояния LPT) с помещением ключевого слова «EPP» в регистры CH, BL и BH. Если функции присутствуют, то этот вызов даст и точку входа в них. EPP BIOS обеспечивает следующие функции:

- опрос конфигурации (номер прерывания, адрес) и возможностей порта (доступные режимы, версия);
- установку и чтение текущего режима (SPP, BiDi, ECP, EPP, EPP 1.7);
- разрешение/запрет прерываний;
- сброс порта;
- выполнение обменов в режиме EPP — одиночные циклы чтения/записи адреса и данных, ввод-вывод блока данных; комбинированные операции (посылку адреса, ввод-вывод байта или блока данных);
- выбор и захват (lock — unlock) устройства, подключенного в цепочке или через мультиплексор;
- установку обработчиков прерываний для указанных устройств, подключенных в цепочке или через мультиплексор;
- определение устройства-источника прерываний;
- дополнительные функции для работы с мультиплексорами и цепочками: опрос версии, текущего порта (определение выбранного порта и его состояния), блокировки, прерывания, цепочки (определение выбранного устройства и его состояния), установки идентификаторов для устройств, не способных их сообщать.

15.5. Параллельный порт и функции PnP

Большинство современных периферийных устройств, подключаемых к LPT- порту, поддерживает стандарт 1284 и функции PnP. Для поддержки этих функций компьютером в аппаратной части достаточно иметь контроллер интерфейса, соответствующий стандарту 1284. Если подключаемое устройство поддерживает PnP, оно по протоколу согласования режимов 1284 способно «договориться» с портом, представляющим «интересы» компьютера, о возможных режимах обмена. Далее, для работы PnP подключенное устройство должно сообщить операционной системе все необходимые сведения о себе. Как минимум, это идентификаторы производителя, модели и набор поддерживаемых команд. Развернутая информация об устройстве может содержать идентификатор

класса, подробное описание и идентификатор хорошо известного устройства, с которым данное устройство совместимо. В соответствии с принятой информацией для поддержки данного устройства операционная система может предпринять действия по установке требуемого программного обеспечения.

Устройства с поддержкой PnP распознаются ОС на этапе ее загрузки, если, конечно же, они подключены к порту интерфейсным кабелем и у них включено питание. Если ОС Windows обнаруживает подключенное устройство PnP, отличающееся от того, что прописано в ее реестре для данного порта (или просто новое устройство), она пытается установить требуемые для устройства драйверы из дистрибутива ОС или из комплекта поставки нового устройства. Если Windows не желает замечать вновь подключенного устройства PnP, это может свидетельствовать о неисправности порта или кабеля. Система PnP не работает, если устройство подключается дешевым «не двунаправленным» кабелем, у которого отсутствует связь по линии Selectln# (контакт 17 порта LPT и контакт 36 разъема Centronics).

15.6. Применение LPT-порта

Обычно LPT-порт используют для подключения принтера (см. 11.5), однако этим его применение не исчерпывается.

Для связи двух компьютеров по параллельному интерфейсу применяются различные кабели в зависимости от режимов используемых портов. Самый простой и медленный — Полубайтный режим, работающий на *всех* портах. Для этого режима в кабеле достаточно иметь 10 сигнальных и один общий провод. Распайка разъемов кабеля приведена в [6]. Связь двух PC данным кабелем поддерживается стандартным ПО типа Interlnk из MS-DOS или Norton Commander. Заметим, что здесь применяется свой протокол, отличный от протокола полубайтного режима. Высокоскоростная связь двух компьютеров может выполняться и в режиме ECP (режим EPP неудобен, поскольку требует синхронизации шинных циклов ввода-вывода двух компьютеров).

Подключение *сканера* к LPT-порту эффективно только если порт обеспечивает хотя бы двунаправленный режим (*Bi-Di),* поскольку основной поток — ввод. Лучше использовать порт ECP, если этот режим поддерживается сканером (или EPP, что маловероятно).

Подключение *внешних накопителей* (Iomega Zip Drive, CD-ROM и др.), *адаптеров ЛВС* и других симметричных устройств ввода-вывода имеет свою специфику. В режиме SPP наряду с замедлением работы устройства заметна принципиальная асимметрия этого режима: *чтение данных* происходит *в два раза медленнее, чем запись* (тоже, кстати, небыстрая). Применение *двунаправленного* режима *(Bi-Di* или *PS/2 Type 1*) устраняет эту асимметрию — *скорости выравниваются.* Только перейдя на EPP или ECP, можно получить *нормальную* скорость работы. В режиме EPP или ECP подключение к LPT-порту почти не уступает по скорости подключению через ISA-контроллер. Это справедливо и при подключении устройств со стандартным интерфейсом шин к LPT-прртам через преобразователи интерфейсов (например, LPT — IDE, LPT — SCSI,

LPT — PCMCIA). Заметим, что винчестер IDE, подключенный через адаптер к LPT-порту, для системы может быть представлен как устройство SCSI (это логичнее с программной точки зрения).

15.7. Конфигурирование LPT-порта

Управление параллельным портом разделяют на два этапа — *предварительное конфигурирование* (Setup) аппаратных средств порта и *текущее* (оперативное) *переключение* режимов работы прикладным или системным ПО. Оперативное переключение возможно только в пределах режимов, разрешенных при конфигурировании. LPT-порт, расположенный на системной плате, конфигурируется через BIOS Setup.

Ниже перечислены параметры, подлежащие конфигурированию:

- *Базовый адрес* — 3BCh, 378h или 278h. Большинство портов по умолчанию конфигурируется на адрес 378h.
- Используемая *линия запросов прерываний:* для LPT — IRQ7, для LPT2 — IRQ5. Традиционно прерывания от принтера не задействуются, и этот дефицитный ресурс можно сэкономить. Однако в скоростных режимах ECP (или Fast Centronics) работа через прерывания может заметно повысить производительность и снизить загрузку процессора.
- Использование *канала DMA* для режимов ECP и Fast Centronics — разрешение и номер канала DMA.
- Режимы работы порта: *SPP*, *PS/2* (он же *Bi-Di), Fast Centronics*, *EPP*, *ECP*, *ECP+EPP.*

Выбор режима EPP, ECP или Fast Centronics сам по себе не приводит к повышению быстродействия обмена с подключенными ПУ, а только дает возможность драйверу и ПУ установить оптимальный режим в пределах их «разумения».

Принтеры и *сканеры* могут пожелать режима ECP. Windows (3.x, 9x, NT и XP) имеет системные драйверы для этого режима. В среде DOS печать через ECP поддерживается только специальным загружаемым драйвером.

Сетевые адаптеры, *внешние диски* и *CD-ROM*, подключаемые к параллельному порту, могут использовать режим EPP. Для этого режима стандартный драйвер пока еще не применяется; поддержка EPP включается в драйвер самого подключаемого устройства.

15.8. Неисправности и тестирование параллельного порта

Тестирование параллельных портов разумно начинать с *проверки их наличия* в системе. Список адресов установленных портов появляется в таблице, выводимой BIOS на экран перед загрузкой ОС. Список можно также посмотреть с

помощью программ тестирования или прямо в BIOS Data Area с помощью отладчика.
Если BIOS обнаруживает меньше портов, чем установлено физически, скорее всего, двум портам присвоен один и тот же адрес. Программное тестирование порта без диагностической заглушки (loop back) не покажет ошибок, поскольку при этом читаются данные выходных регистров, а они у всех конфликтующих (по отдельности исправных) портов совпадут. Именно такое тестирование производит BIOS при проверке на наличие портов. Разбираться с ситуацией полагается, последовательно устанавливая порты и наблюдая за адресами, появляющимися в списке.
Если физически установлен только один порт, а BIOS его не обнаруживает, то либо порт отключен при конфигурировании, либо он вышел из строя (скорее всего, из-за нарушений правил подключения). Если вам повезет, неисправность можно устранить «передергиванием» платы в слоте — там иногда возникают проблемы с контактами.
Наблюдаются и такие «чудеса» — при «теплой» перезагрузке DOS после Windows 9x порт не виден (и приложения не могут печатать из MS-DOS). Однако после повторной перезагрузки DOS порт оказывается на месте. С этим явлением легче смириться, чем бороться.
Тестирование портов с помощью диагностических программ позволяет проверить выходные регистры, а при использовании специальных заглушек — и входные линии. Поскольку количество выходных (12) и входных (5) линий порта различно, то полная проверка порта с помощью пассивной заглушки принципиально невозможна. Разные программы тестирования требуют применения разных заглушек (рис. 15.5).

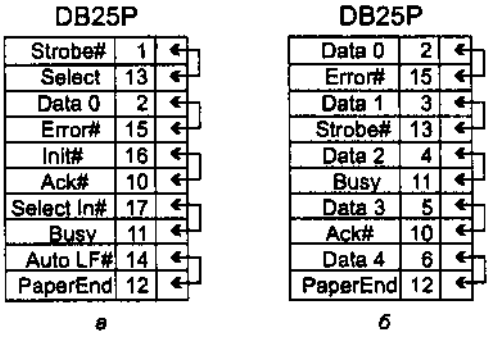


Рис. 15.5. Схема заглушки для тестирования LPT-порта: а — для Checklt, *б* - для Norton Diagnostics

Большинство неприятностей при работе с LPT-портами доставляют *разъемы* и *кабели.* Для проверки порта, кабеля и принтера можно воспользоваться специальными тестами из популярных диагностических программ (Checklt, PC- Check и т. п.). Можно попытаться просто вывести на принтер какой-либо файл, содержащий печатаемые символы.

- Если вывод файла с точки зрения ОС прошел (копирование файла на устройство с именем LPTn или PRN совершается быстро и успешно), а принтер (исправный) не напечатал ни одного символа — скорее всего, это обрыв (неконтакт в разъеме) цепи Strobe#.
- Если принтер включен (находится в состоянии *On Line),* а появляется сообщение о его неготовности, причину следует искать в линии Busy.
- Если принтер, подключенный к порту, в стандартном режиме (SPP) печатает нормально, а при переходе в режим ECP начинаются сбои, следует проверить кабель — соответствует ли он требованиям IEEE 1284 (см. выше). Дешевые кабели с неперевитыми проводами нормально работают на скоростях 50-100 Кбайт/с, но при скорости 1-2 Мбайт/с, обеспечиваемой ECP, имеют полное право не работать, особенно при длине более 2 м.
- Если при установке драйвера PnP-принтера появилось сообщение о необходимости применения «двунаправленного кабеля», проверьте наличие связи контакта 17 разъема DB-25 с контактом 36 разъема Centronics. Хотя эта связь изначально предусматривалась, в ряде кабелей она отсутствует.
- Если принтер искажает информацию при печати, возможен обрыв (или замыкание) линий данных. В этом случае удобно воспользоваться файлом, содержащим последовательность кодов всех печатных символов. Если файл печатается с повтором некоторых символов или их групп, по периодичности повтора можно легко вычислить оборванный провод данных интерфейса. Этот же файл удобно использовать для проверки аппаратной русификации принтера.

Аппаратные прерывания от LPT-порта используются не всегда. Даже DOS- программа фоновой печати PRINT работает с портом по опросу состояния, а ее обслуживающий процесс запускается по прерыванию от таймера. Поэтому неисправности, связанные с цепью прерывания от порта, проявляются не часто. Однако по-настоящему многозадачные ОС стараются работать с портом по прерываниям. Протестировать линию прерывания можно, только подключив к порту ПУ или заглушку. Если к. порту с неисправным каналом прерывания подключить адаптер локальной сети, то он, возможно, будет работать, но с очень низкой скоростью: на любой запрос ответ будет приходить с задержкой в десятки секунд — принятый из адаптера пакет будет обрабатываться не по прерыванию (сразу по приходе), а по внешнему тайм-ауту.

ГЛАВА 16

Проводные и беспроводные последовательные интерфейсы

Эта глава посвящена последовательным интерфейсам: проводным (RS-232C на COM-порте) и беспроводным (IrDA и Bluetooth). Последовательные интерфейсы привлекательны благодаря малому числу сигнальных линий, что упрощает кабельные соединения. Беспроводные (wireless) интерфейсы позволяют освободить устройства от связывающих их интерфейсных кабелей, что особенно привлекательно для малогабаритной периферии, по размеру и весу соизмеримой с кабелями. В беспроводных интерфейсах используются электромагнитные волны инфракрасного (IrDA) и радиочастотного (Bluetooth) диапазонов.

В данной главе рассмотрены только последовательные интерфейсы, используемые для подключения внешних устройств. Есть ряд последовательных интерфейсов, предназначенных для соединения компонентов (микросхем); из них в книге упоминаются I^2C, SPI. Интерфейс I^2C является основой шин ACCESS. Bus и SMBus, он используется и в интерфейсе мониторов VGA для передачи конфигурационной и управляющей информации. Подробно эти интерфейсы рассмотрены в [6].

16.1. Интерфейс RS-232C — COM-порт

В последовательном интерфейсе для передачи данных в одном направлении используется одна сигнальная линия, по которой информационные биты передаются друг за другом — последовательно. Английские названия последовательных интерфейса и порта — *Serial Interface* и *Serial Port*, иногда их неправильно переводят как «серийные». Последовательная передача позволяет сократить количество сигнальных линий и добиться улучшения связи на больших расстояниях.

Начиная с первых моделей в PC имелся последовательный интерфейс — *COM- порт* (Communications port — коммуникационный порт). Этот порт обеспечивает *асинхронный* обмен по стандарту *RS-232C*. *Синхронный* обмен в PC поддерживают лишь специальные адаптеры, например SDLC или V.35. COM-порты

реализуются на микросхемах универсальных асинхронных приемопередатчиков (*UART*), совместимых с семейством *i8250/16450/16550.* Они занимают в пространстве ввода-вывода по 8 смежных 8-битных регистров и могут располагаться по стандартным *базовым адресам* 3F8h (COM1), 2F8h (COM2), 3E8h (COM3), 2E8h (COM4). Порты могут вырабатывать *аппаратные прерывания* IRQ4 (обычно используются для COM1 и COM3) и IRQ3 (для COM2 и COM4). С внешней стороны порты имеют линии последовательных данных передачи и приема, а также набор сигналов управления и состояния, соответствующий стандарту RS-232C. COM-порты имеют внешние *разъемы-вилки* (male — «папа») *DB25P* или *DB9P,* выведенные на заднюю панель компьютера. Характерной особенностью интерфейса является применение не ТТЛ-сигналов — все внешние сигналы порта двуполярные. Гальваническая развязка отсутствует — схемная «земля» подключаемого устройства соединяется со схемной «землей» компьютера. Скорость передачи данных может достигать 115 200 бит/с.

Компьютер может иметь до четырех последовательных портов COM1—COM4 (для машин класса AT типично наличие двух портов) с поддержкой на уровне BIOS. Сервис Int 14h BIOS обеспечивает инициализацию порта, ввод и вывод символа (без прерываний) и опрос состояния. Через Int 14h скорость передачи программируется в диапазоне 110-9600 бит/с (меньше, чем реальные возможности порта). Для повышения производительности широко используется взаимодействие программ с портом на уровне регистров, для чего требуется совместимость аппаратных средств COM-порта с программной моделью i8250/16450/ 16550.

Название порта указывает на его основное назначение — подключение коммуникационного оборудования (например, модема) для связи с другими компьютерами, сетями и периферийными устройствами. К порту могут непосредственно подключаться и периферийные устройства с последовательным интерфейсом: принтеры, плоттеры, терминалы и другие. COM-порт широко используется для подключения мыши, а также организации непосредственной связи двух компьютеров. К COM-порту подключают и электронные ключи.

Практически все современные системные платы (еще начиная с PCI-плат для процессоров класса 486) имеют встроенные адаптеры двух COM-портов. Один из портов может использоваться и для беспроводной инфракрасной связи с периферийными устройствами (IrDA). Существуют карты ISA с парой COM- портов, где они чаще всего соседствуют с LPT-портом, а также с контроллерами дисковых интерфейсов (FDC и IDE). «Классический» COM-порт позволял осуществлять обмен данными только программным способом, при этом для пересылки каждого байта процессору приходилось выполнять несколько инструкций. Современные порты имеют FIFO-буферы данных и позволяют выполнять обмен по каналу DMA, существенно разгружая центральный процессор, что особенно важно на больших скоростях обмена.

В спецификации PC’99 применять традиционные COM-порты не рекомендуется, но пока не запрещается. Если они есть, то должны быть совместимыми с UART 16550A и обеспечивать скорость до 115,2 Кбит/с. Устройства, которые традиционно задействуют COM-порт, рекомендуется переводить на последова

тельные шины USB и FireWire. Однако и поныне COM-порты продолжают широко использоваться. На современных системных платах присутствуют два COM-порта: COM1 выводится на внешний разъем, а COM2 используется для инфракрасной связи.

Протокол RS-232C

Стандарт *RS-232C* описывает несимметричные передатчики и приемники: сигнал передается относительно общего провода — схемной «земли» (симметричные дифференциальные сигналы используются в других интерфейсах — например, *RS-422*). Интерфейс *не обеспечивает гальванической развязки* устройств. Логической единице (состояние *MARK)* на *входе данных* (сигнал RxD) соответствует диапазон напряжения от -12 до -3 В; логическому нулю — от +3 до + 12 В (состояние *SPACE).* Для *входов управляющих сигналов* состоянию *ON* («включено») соответствует диапазон от +3 до +12 В, состоянию *OFF* («выключено») — от -12 до -3 В. Диапазон от -3 до +3 В — зона нечувствительности, обусловливающая гистерезис приемника: состояние линии считается измененным только после пересечения порога. Уровни сигналов на выходах передатчиков должны быть в диапазонах от -12 до -5 В и от +5 до +12 В. Разность потенциалов между схемными «землями» (SG) соединяемых устройств должна быть менее 2 В, при более высокой разности потенциалов возможно неверное восприятие сигналов. Заметим, что сигналы ТТЛ-уровней (на входах и выходах микросхем UART) передаются в прямом коде для линий TxD и RxD и в инверсном — для всех остальных.

Интерфейс предполагает наличие *защитного заземления* соединяемых устройств, если они оба питаются от сети переменного тока и имеют сетевые фильтры.

ВНИМАНИЕ----------

Подключение и отключение интерфейсных кабелей устройств с автономным питанием должно производиться при отключенном питании. Иначе разность невыровненных потенциалов устройств в момент коммутации может оказаться приложенной к выходным или входным (что опаснее) цепям интерфейса и вывести из строя микросхемы.

В табл. 16.1 приведено назначение контактов разъемов COM-портов (и любой другой аппаратуры передачи данных, *АПД).* У модемов название цепей и контактов такое же, но роли сигналов (вход-выход) меняются на противоположные.

Таблица 16.1. Разъемы и сигналы интерфейса RS-232C

Обозначение цепи			Контакт разъема		№ провода кабеля выносного разъема PC				Направление
COM-порт	RS-232	V.24 Стык 2	DB-25P	DB-9P	1[1]	2[2]	3[3]	4[4]	I/O
PG	AA	101	1	5	(10)	(10)	(10)	1	–
SG	AB	102	7	5	5	9	1	13	–

Обозначение цепи			Контакт разъема		№ провода кабеля выносного разъема PC				Направление
COM-порт	RS-232	V.24 Стык 2	DB-25P	DB-9P	1[1]	2[2]	3[3]	4[4]	I/O
TD	BA	103	2	3	3	5	3	3	O
RD	BB	104	3	2	2	3	4	5	I
RTS	CA	105	4	7	7	4	8	7	O
CTS	CB	106	5	8	8	6	7	9	I
DSR	CC	107	6	6	6	2	9	11	I
DTR	CD	108/2	20	4	4	7	2	14	O
DCD	CF	109	8	1	1	1	5	15	I
RI	CE	125	22	9	9	8	6	18	I

[1] Ленточный кабель 8-битных мультикарт.

[2] Ленточный кабель 16-битных мультикарт и портов на системных платах.

[3] Вариант ленточного кабеля портов на системных платах.

[4] Широкий ленточный кабель к 25-контактному разъему.

Подмножество сигналов *RS-232C*, относящихся к асинхронному режиму, рассмотрим с точки зрения COM-порта PC. Для удобства будем пользоваться мнемоникой названий, принятой в описаниях COM-портов и большинства устройств (она отличается от безликих обозначений RS-232 и V.24). Назначение сигналов интерфейса приведено в табл. 16.2.

Таблица 16.2. Назначение сигналов интерфейса RS-232C

Сигнал	Назначение
PG	Protected Ground — защитная «земля», соединяется с корпусом устройства и экраном кабеля
SG	SG Signal Ground — сигнальная (схемная) «земля», относительно которой действуют уровни сигналов
TD	Transmit Data — последовательные данные — выход передатчика
RD	Receive Data — последовательные данные — вход приемника
RTS	Request To Send — выход запроса передачи данных: состояние «включено» уведомляет модем о наличии у терминала данных для передачи. В полудуплексном режиме используется для управления направлением — состояние «включено» служит сигналом модему на переключение в режим передачи
CTS	Clear To Send — вход разрешения терминалу передавать данные. Состояние «выключено» запрещает передачу данных. Сигнал используется для аппаратного управления потоками данных
DSR	Data Set Ready — вход сигнала готовности от аппаратуры передачи данных (модем в рабочем режиме подключен к каналу и закончил действия по согласованию с аппаратурой на противоположном конце канала)
DTR	Data Terminal Ready — выход сигнала готовности терминала к обмену данными. Состояние «включено» поддерживает коммутируемый канал в состоянии соединения
DCD	Data Carrier Detected — вход сигнала обнаружения несущей удаленного модема
RI	Ring Indicator — вход индикатора вызова (звонка). В коммутируемом канале этим сигналом модем сигнализирует о принятии вызова

Нормальная последовательность управляющих сигналов для случая подключения модема к COM-порту приведена на рис. 16.1. Напомним, что положительному уровню соответствует логическое состояние «включено», а отрицательному — «выключено».

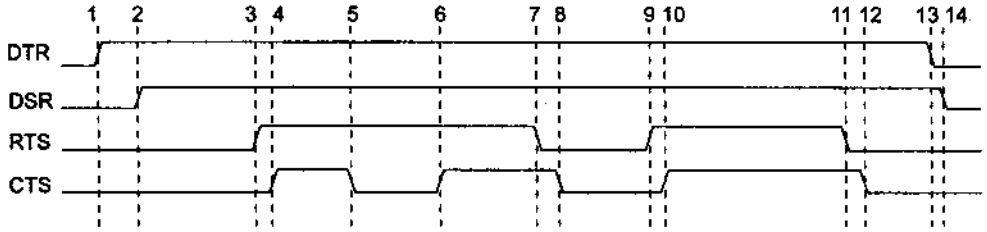


Рис. 16.1. Последовательность управляющих сигналов интерфейса RS-232C

Рассмотрим эту последовательность управляющих сигналов:

1. Установкой сигнала DTR компьютер указывает на желание использовать модем.
2. Установкой сигнала DSR модем сигнализирует о своей готовности к установлению соединения.
3. Сигналом RTS компьютер запрашивает разрешение на передачу и заявляет о своей готовности принимать данные от модема.
4. Сигналом CTS модем уведомляет о своей готовности к приему данных от компьютера и передаче их в линию.
5. Снятием сигнала CTS модем сигнализирует о невозможности дальнейшего приема (например, буфер заполнен) — компьютер должен приостановить передачу данных.
6. Восстановлением сигнала CTS модем разрешает компьютеру продолжить передачу (в буфере появилось место).
7. Снятие сигнала RTS может означать как заполнение буфера компьютера (модем должен приостановить передачу данных в компьютер), так и отсутствие данных для передачи в модем. Обычно в этом случае модем прекращает пересылку данных в компьютер.
8. Модем подтверждает снятие сигнала RTS сбросом сигнала CTS.
9. Компьютер повторно устанавливает сигнала RTS для возобновления передачи.
10. Модем подтверждает готовность к этим действиям.
11. Компьютер указывает на завершение обмена.
12. Модем отвечает подтверждением.
13. Компьютер снимает сигнал DTR, что обычно означает необходимость разрыва соединения («повесить трубку»).
14. Модем сбросом сигнала DSR сообщает о разрыве соединения.

Из этой последовательности становятся понятными соединения DTR-DSR и RTS-CTS в нуль-модемных кабелях.

При асинхронной передаче каждому байту предшествует *старт-бит*, сигнализирующий приемнику о начале посылки, за которым следуют *биты данных* и, возможно, *бит четности* (parity). Завершает посылку *стоп-бит*, гарантирующий паузу между посылками (рис. 16.2). Старт-бит следующего байта посылается в любой момент после стоп-бита, то есть между передачами возможны паузы произвольной длительности. Старт-бит, имеющий всегда строго определенное значение (логический 0), обеспечивает простой механизм синхронизации приемника по сигналу от передатчика. Подразумевается, что приемник и передатчик работают на одной скорости обмена.

Рис. 16.2. Формат асинхронной передачи

Формат асинхронной посылки позволяет выявлять возможные *ошибки передачи*: ложный старт-бит, потерянный стоп-бит, ошибку паритета. Контроль формата позволяет обнаруживать обрыв линии: при этом принимаются логический нуль, который сначала трактуется как старт-бит, и нулевые биты данных потом срабатывает контроль стоп-бита.

Для асинхронного режима принят ряд *стандартных скоростей обмена*: 50, 75, 110, 150, 300, 600, 1200, 2400, 4800, 9600, 19 200, 38 400, 57 600 и 115 200 бит/с. Иногда вместо единицы измерения «бит/с» используют «бод» (baud), но при рассмотрении двоичных передаваемых сигналов это некорректно. В бодах принято измерять частоту изменения состояния линии, а при недвоичном способе кодирования (широко применяемом в современных модемах) в канале связи скорости передачи бит (измеряемая в битах в секунду) и изменения сигнала (бод) могут отличаться в несколько раз (подробнее см. 13.1).

Количество *битов данных* может составлять 5, 6, 7 или 8 (5- и 6-битные форматы распространены незначительно). Количество *стоп-битов* может быть 1, 1,5 или 2 («полтора бита» означает только длительность стопового интервала).

Асинхронный режим является *байт-ориентированным* (символьно-ориентированным): минимальная пересылаемая единица информации — байт (символ). В отличие от него синхронный режим (не поддерживаемый COM-портами) является бит-ориентированным — кадр, пересылаемый по нему, может иметь произвольное количество битов.

Управление потоком данных

Для управления потоком данных (flow control) могут использоваться два варианта протокола — аппаратный *RTS/CTS* и программный *XON/XOFF*. Иногда управление потоком путают с квитированием. *Квитирование* (handshaking) подразумевает посылку уведомления о получении элемента, в то время как *управление потоком* предполагает посылку уведомления о возможности или невозможности последующего приема данных. Квитирование характерно для параллельных интерфейсов (см. главу 15), его применение может избавить от необходимости управления потоком.

Так называемый *аппаратный протокол управления потоком RTS/CTS* (hardware flow control) использует сигнал CTS, который позволяет остановить передачу данных, если приемник не готов к их приему (рис. 16.3). Передатчик «выпускает» очередной байт только при включенной линии CTS. Байт, который уже начал передаваться, задержать сигналом CTS невозможно (это гарантирует целостность посылки). Аппаратный протокол обеспечивает самую быструю реакцию передатчика на состояние приемника. Микросхемы асинхронных приемопередатчиков имеют не менее двух регистров в приемной части — сдвигающий, для приема очередной посылки, и хранящий, из которого считывается принятый байт. Это позволяет реализовать обмен по аппаратному протоколу без потери данных.

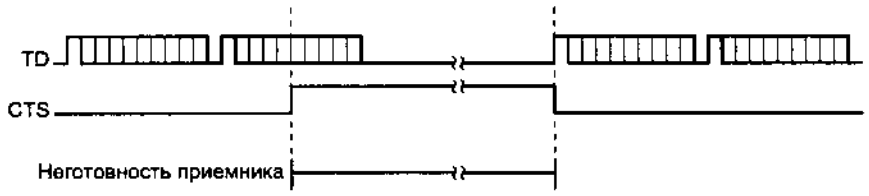


Рис. 16.3. Аппаратное управление потоком

Аппаратный протокол удобен при подключении принтеров и плоттеров, если они его поддерживают. При непосредственном (без модемов) соединении двух компьютеров аппаратный протокол требует перекрестного соединения линий RTS-CTS, что обеспечит состояние «включено» на линии CTS. В противном случае передатчик будет «молчать».

Применяемые в IBM PC приемопередатчики 8250/16450/16550 сигнал CTS аппаратно не отрабатывают, а только показывают его состояние в регистре MSR. Реализация протокола RTS/CTS возлагается на драйвер (например, Int 14h BIOS), и называть его «аппаратным» не совсем корректно. Если же программа, пользующаяся COM-портом, взаимодействует с UART на уровне регистров (а не через BIOS), то обработкой сигнала CTS для поддержки данного протокола она занимается сама. Ряд коммуникационных программ позволяет игнорировать сигнал CTS (если не используется модем), и для них не требуется соединение входа CTS с выходом своего сигнала RTS. Однако существуют и иные приемопередатчики (например, 8251), в которых сигнал CTS отрабатывается аппа

ратно. Для них, а также для «честных» программ использование сигнала CTS на разъемах (а то и на кабелях) обязательно. Преимущество протокола RTS/CTS во времени реакции по сравнению с программным протоколом XON/XOFF остается лишь для буферизованной (в режиме FIFO) передачи.

Программный протокол управления потоком XON/XOFF предполагает наличие двунаправленного канала передачи данных. Работает протокол следующим образом: если устройство, принимающее данные, обнаруживает причины, по которым оно не может их дальше принимать, оно по обратному последовательному каналу посылает байт-символ XOFF (13h). Противоположное устройство, приняв этот символ, приостанавливает передачу. Когда принимающее устройство снова становится готовым к приему данных, оно посылает символ XON (11h), приняв который, противоположное устройство возобновляет передачу. Время реакции передатчика на изменение состояния приемника по сравнению с аппаратным протоколом увеличивается по крайней мере на время передачи символа (XON или XOFF) плюс время реакции программы передатчика на прием символа (рис. 16.4). Из этого следует, что данные без потерь могут приниматься только приемником, имеющим дополнительный буфер принимаемых данных и заблаговременно сигнализирующим о неготовности (имея в буфере свободное место).

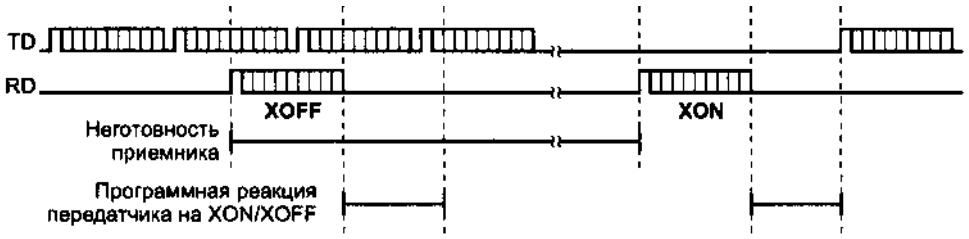


Рис. 16.4. Программное управление потоком XON/XOFF

Преимущество программного протокола заключается в отсутствии необходимости передачи управляющих сигналов интерфейса — кабелю для двустороннего обмена достаточно иметь всего 3 провода. Недостатком, помимо необходимости буфера и большего времени реакции (снижающего общую производительность канала из-за ожидания сигнала XON), является сложность реализации полнодуплексного режима обмена. В этом случае из потока принимаемых данных должны выделяться (и обрабатываться) символы управления потоком, что ограничивает набор передаваемых символов или требует дополнительных ухищрений в плане кодировки.

Помимо этих двух распространенных стандартных протоколов, поддерживаемых и ПУ и ОС, существуют и другие.

Микросхемы асинхронных приемопередатчиков

Процессор взаимодействует со всеми подсистемами компьютера параллельными кодами, минимальная длина адресуемой посылки составляет один байт.

В COM-портах преобразование параллельного кода в последовательный при передаче и обратное преобразование при приеме данных выполняют специализированные микросхемы UART (Universal Asynchronous Receiver-Transmitter — универсальный асинхронный приемопередатчик). Эти же микросхемы формируют и обрабатывают управляющие сигналы интерфейса. COM-порты IBM PC XT/AT базируются на микросхемах, совместимых на уровне регистров с UART i8250 — 8250/16450/16550A. Это семейство представляет собой усовершенствование начальной модели, направленное на повышение быстродействия, снижение потребляемой мощности и загрузки процессора при интенсивном обмене. Отметим некоторые моменты:

- Микросхема 8250 имеет ошибки (появление ложных прерываний), учтенные в XT BIOS.
- В микросхеме 8250A ошибки исправлены, но в результате потеряна совместимость с BIOS. Эта микросхема работает в некоторых моделях AT, но непригодна для скорости 9600 бит/с.
- В микросхеме 8250B исправлены ошибки 8250 и 8250A, но восстановлена ошибка в прерываниях, благодаря чему возвращена совместимость с XT BIOS. Работает в AT под DOS (кроме скорости 9600 бит/с).

Микросхемы 8250x имеют невысокое быстродействие по обращениям со стороны системной шины. Они не допускают обращения к своим регистрам в смежных шинных циклах процессора — для корректной работы с ними требуется введение программных задержек между обращениями CPU.

В компьютерах класса AT применяют микросхемы UART перечисленных ниже модификаций:

- 16450 — быстродействующая версия 8250 для AT. Ошибок 8250 и полной совместимости с XT BIOS не имеет.
- 16550 — развитие 16450. Может использовать канал DMA для обмена данными. Имеет FIFO-буфер, но некорректность его работы не позволяет им воспользоваться.
- 16550A — имеет работающие 16-байтные FIFO-буферы приема и передачи и возможность использования DMA. Именно этот тип UART должен применяться в AT при интенсивных обменах на скоростях 9600 бит/с и выше. С этой микросхемой совместимо большинство микросхем контроллеров портов ввода-вывода, входящих в современные чипсеты.

Микросхемы UART 16550A с программной точки зрения представляют собой набор регистров, доступ к которым определяется адресом (смещением адреса регистра относительно базового адреса порта) и значением бита DLAB (бита 7 регистра LCR). В адресном пространстве микросхема занимает 8 смежных адресов. Описание регистров UART 16550A к ним приведены в [4], [6].

Системная поддержка COM-портов

COM-порты поддерживаются *сервисом* Int 14h BIOS, который обеспечивает следующие функции:

- *Инициализацию* (установку скорости обмена и формата посылок, заданных регистром AL; запрет источников прерываний). На сигналы DTR и RTS влияния не оказывает (после аппаратного сброса они пассивны).
- *Ввод и вывод символа* без аппаратных прерываний с контролем тайм-аута. Активируются сигналы DTR и RTS.
- *Опрос состояния* модема и линии.

В процессе начального тестирования POST BIOS проверяет наличие последовательных портов (регистров UART 8250 или совместимых) по стандартным адресам и помещает базовые адреса обнаруженных портов в ячейки BIOS Data Area 0:0400, 0402, 0404, 0406. Эти ячейки хранят адреса портов с логическими именами C0M1—COM4. Нулевое значение адреса является признаком отсутствия порта с данным номером. В ячейки 0:047C, 047D, 047E, 047F заносятся константы, задающие тайм-аут для портов.
Обнаруженные порты *инициализируются* на скорость обмена 2400 бит/с, 7 бит данных с контролем на четность (even), 1 стоп-бит. Управляющие сигналы интерфейса DTR и RTS переводятся в исходное состояние («выключено» — отрицательное напряжение).

Конфигурирование COM-портов

Компьютер может иметь до четырех последовательных портов COM 1—COM4, для машин класса AT типично наличие двух портов. Управление последовательным портом разделяется на два этапа — предварительное конфигурирование (Setup) аппаратных средств порта и текущее (оперативное) переключение режимов работы прикладным или системным ПО. Конфигурирование COM- порта зависит от его исполнения. Порт на плате расширения конфигурируется джамперами на самой плате. Порт на системной плате конфигурируется через CMOS Setup.
Параметры конфигурирования перечислены ниже.

- *Базовый адрес* для портов COM1—COM4 обычно имеет значение 3F8h, 2F8h, 3E8h и 2E8h. При инициализации BIOS проверяет наличие портов по адресам именно в этом порядке и присваивает обнаруженным портам логические имена C0M1, COM2, COM3 и COM4. Для портов COM3 и COM4 возможны альтернативные адреса 3E0h, 338h и 2E0h, 238h соответственно.
- *Линия запросов прерываний.* Для C0M1 и COM3 обычно используется линия IRQ4 или IRQ11, для COM2 и COM4 — IRQ3 или IRQ10. В принципе, номер прерывания можно назначать в произвольных сочетаниях с базовым адресом (номером порта), но некоторые программы и драйверы (например, драйверы последовательной мыши) настроены на стандартные сочетания. Каждому порту, нуждающемуся в аппаратном прерывании, назначают отдельную линию, не совпадающую с линиями запроса прерываний других устройств. Прерывания необходимы для портов, к которым подключаются устройства ввода, UPS или модемы. При подключении принтера или плоттера прерываниями пользуются только многозадачные ОС (не всегда), и этот дефицитный ресурс PC можно сэкономить. Также прерывания обычно не нужны при связи

двух компьютеров нуль-модемным кабелем. Возможность разделения одной линии запроса несколькими портами (или портом и другими устройствами) зависит от реализации аппаратного подключения и ПО. При использовании портов, установленных на шину ISA, разделяемые прерывания обычно не работают.

- *Канал DMA* (для микросхем UART 16450/16550, расположенных на системной плате) — разрешение использования DMA и номер канала. Режим DMA при работе с COM-портами задействуют редко.

Режим работы порта по умолчанию (2400 бит/с, 7 бит данных, 1 стоп-бит и контроль четности), заданный при инициализации порта во время теста POST системы BIOS, может изменяться в любой момент при настройке коммуникационных программ или консольной командой MODE COMx: с указанием параметров (синтаксис можно узнать, запустив команду MODE с ключом /?).

Использование COM-портов

COM-порты широко применяются для подключения различных периферийных и коммуникационных устройств, связи с технологическим оборудованием, объектами управления и наблюдения, программаторами, внутрисхемными эмуляторами и прочими устройствами по протоколу RS-232C. COM-порт может функционировать и как двунаправленный интерфейс, у которого имеются 3 программно-управляемых выходных линии и 4 программно-читаемых входных линии с двуполярными сигналами. Порядок их использования определяется разработчиком. Существует, например, схема однобитного широтно-импульсного преобразователя, позволяющего записывать звуковой сигнал на диск PC, используя входную линию COM-порта. Воспроизведение этой записи через обычный динамик PC позволяет передать речь. В настоящее время, когда звуковая карта стала почти обязательным устройством PC, это не впечатляет, но когда-то такое решение казалось интересным.

Непосредственное подключение устройств

COM-порты чаще всего применяют для подключения *манипуляторов* (мышь, трекбол). В этом случае порт используется в режиме последовательного ввода. Мышь с последовательным интерфейсом — *Serial Mouse* — может подключаться к любому исправному порту. Для согласования разъемов порта и мыши допускается применение переходника DB-9S-DB-25P или DB-25S-DB-9P. Для мыши требуется прерывание, для порта COM1 — IRQ4, для COM2 — IRQ3. То, что для работы мыши порту COM1 требуется прерывание IRQ4, является особенностью ее драйвера, но для пользователя важен сам факт ограничения. Каждое событие — перемещение мыши или нажатие/отпускание кнопки — кодируется двоичной посылкой по интерфейсу *RS-232C*. Применяется асинхронная передача; двуполярное питание обеспечивается от управляющих линий интерфейса.

Для подключения *внешних модемов* требуется полный (9-проводный) кабель *АПД-АКД,* схема которого приведена на рис. 16.5. Этот же кабель используется для согласования разъемов (по количеству контактов); возможно применение

переходников 9-25, предназначенных для мышей. Для работы коммуникационного ПО обычно нужны прерывания, но здесь есть свобода выбора номера (адреса) порта и линии прерывания. Если предполагается работа на скоростях 9600 бит/с и выше, то COM-порт должен быть реализован на микросхеме UART 16550A или совместимой. Возможности работы посредством FIFO-буферов и обмена по каналам DMA зависят от коммуникационного ПО.

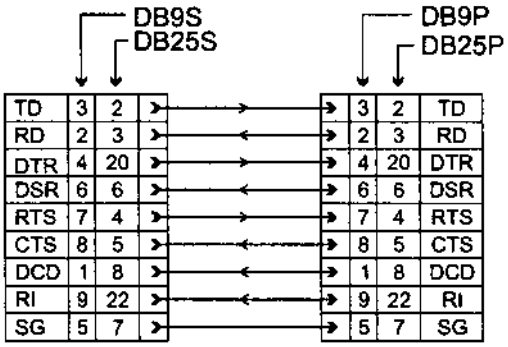


Рис. 16.5. Кабели подключения модемов

Для связи *двух компьютеров*, удаленных друг от друга на небольшое расстояние, используют и непосредственное соединение их COM-портов нуль-модемным кабелем (рис. 16.6). Программы MS-DOS типа Norton Commander и Interlnk позволяют обмениваться файлами со скоростью до 115,2 Кбит/с без применения аппаратных прерываний. Это же соединение может использоваться и сетевым пакетом Lantastic, предоставляющим более развитый сервис.

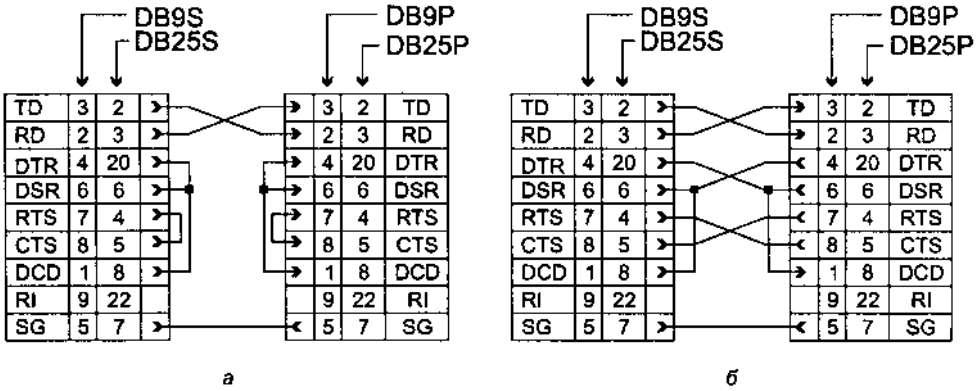


Рис. 16.6. Нуль-модемный кабель: а — минимальный, *б* — полный

COM-порт пригоден и для подключения *электронных ключей* (security devices), предназначенных для защиты от нелицензированного использования ПО. Эти устройства могут быть как «прозрачными», то есть обеспечивающими возмож-

ность подключения периферии к этому порту, так и полностью занимающими порт.

COM-порт при наличии соответствующей программной поддержки позволяет превратить PC в *терминалу* эмулируя систему команд распространенных специализированных терминалов (VT-52, VT-100 и т. д.). Простейший терминал получается, если замкнуть друг на друга функции BIOS обслуживания COM- порта (Int 14h), телетайпного вывода (Int 10h) и клавиатурного ввода (Int 16h). Однако такой терминал будет работать лишь на малых скоростях обмена, поскольку функции BIOS хотя и универсальны, но не слишком быстры.

Преобразования последовательных интерфейсов

На *физическом уровне* последовательный интерфейс имеет различные реализации, различающиеся способом передачи электрических сигналов. Существует ряд международных стандартов, родственных *RS-232Cy* о них подробнее можно узнать в [6]. На рис. 16.7 приведены схемы соединения их приемников и передатчиков, а также показаны ограничения на длину линии (I) и максимальную скорость передачи данных *(v)*. Несимметричные линии интерфейсов *RS-232C* и *RS-423A* имеют самую низкую защищенность от синфазной помехи, хотя дифференциальный вход приемника *RS-423A* несколько смягчает ситуацию. Лучшие параметры имеют двухточечный интерфейс *RS-422A* и его магистральный (шинный) аналог *RS-485y* работающие на симметричных линиях связи. В них для передачи каждого сигнала используются дифференциальные сигналы с отдельной (витой) парой проводов для каждой сигнальной цепи. Поскольку логически эти интерфейсы родственны, допустимо применение несложных преобразователей сигналов, обеспечивающих переход от одного интерфейса к другому.

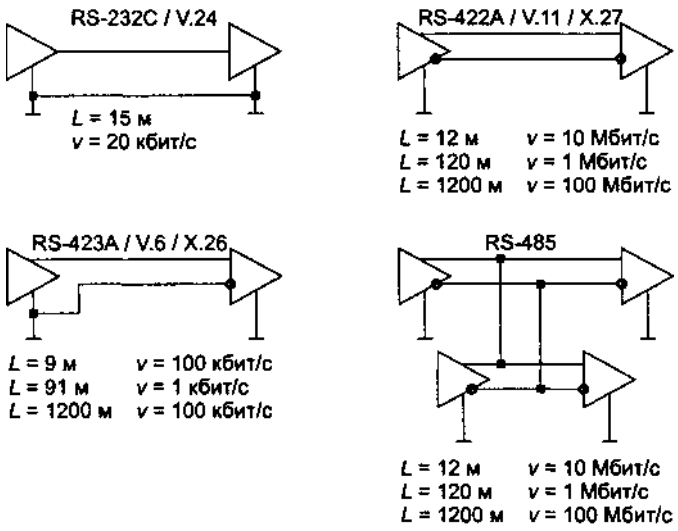


Рис. 16.7. Стандарты последовательных интерфейсов

В перечисленных выше стандартах сигнал представляется *потенциалом*. Существуют последовательные интерфейсы («токовая петля» и MIDI), где информативен ток, протекающий по общей цепи «передатчик-приемник» (см. 12.3).

«Токовая петля» — распространенный вариант последовательного интерфейса. В ней электрическим сигналом выступает не уровень напряжения относительно общего провода, а *ток* в двухпроводной линии, соединяющей приемник и передатчик. Логической единице (состоянию «включено») соответствует ток 20 мА, а логическому нулю — отсутствие тока. Такое представление сигналов для вышеописанного формата асинхронной посылки позволяет обнаружить обрыв линии — приемник заметит отсутствие стоп-бита (обрыв линии действует как постоянный логический нуль).

Токовая петля обычно предполагает *гальваническую развязку* входных цепей приемника от схемы устройства. При этом источником тока в петле является передатчик (этот вариант называют активным передатчиком). Возможно и питание от приемника (активный приемник), при этом выходной ключ передатчика может быть также гальванически развязан с остальной схемой передатчика. Существуют упрощенные варианты без гальванической развязки, но это уже вырожденный случай интерфейса.

Токовая петля с гальванической развязкой позволяет передавать сигналы на расстояния до нескольких километров. Расстояние определяется сопротивлением пары проводов и уровнем помех. Поскольку интерфейс требует пары проводов для каждого сигнала, обычно используют только два сигнала интерфейса. В случае двунаправленного обмена применяются только сигналы передаваемых и принимаемых данных, а управление потоком реализуется программным методом XON/XOFF. Если двунаправленный обмен не требуется, занимают одну линию данных, а обратная линия задействуется для сигнала управления потоком (CTS для аппаратного протокола или RXD для программного протокола). При надлежащем ПО одной токовой петлей можно обеспечить двунаправленную полудуплексную связь двух устройств. При этом каждый приемник «слышит» как сигналы передатчика на противоположной стороне канала, так и сигналы своего передатчика. Они расцениваются коммуникационными пакетами просто как эхо-сигнал. Поэтому для безошибочного приема передатчики должны работать поочередно.

COM-порт и PnP

Современные ПУ, подключаемые к COM-порту, могут поддерживать спецификацию PnP. Основная задача ОС заключается в идентификации подключенного устройства, для чего разработан несложный протокол, реализуемый на любых COM-портах чисто программным способом [6].

Протокол позволяет определить факт подключения устройства, считать его строку идентификатора PnP и определить факт отключения. Строка идентификатора PnP должна иметь маркеры начала (28h или 08h) и конца (29h или 09h), между которыми располагается тело идентификатора в стандартизованном формате. Перед маркером начала могут находиться до 16 символов, не относя

щиеся к идентификатору PnP. Протокол разрабатывался фирмой с учетом совместимости не только с PnP-устройствами — он обеспечивает устойчивость системы к сообщениям, не являющимся PnP-идентификаторами. Например, обычная мышь Microsoft Mouse по включении питания от интерфейса отвечает ASCII-кодом символа «M» (трехкнопочная — строкой «M3»).

Неисправности и тестирование COM-портов

Неполадки с COM-портами случаются (выявляются) при установке новых портов или неудачном подключении внешних устройств.

Проверка конфигурирования

Тестирование последовательных портов (как и параллельных) начинают с их опознавания системой. Список *адресов* установленных портов обычно появляется в таблице, выводимой BIOS перед загрузкой ОС. Список можно посмотреть с помощью тестовых программ или прямо в BIOS Data AREA с помощью отладчика.

Если BIOS обнаруживает меньше портов, чем установлено физически, значит, двум портам присвоен один и тот же адрес или установлен нестандартный адрес какого-либо порта. Проблемы могут возникать с адресами портов COM3 и COM4: не все версии BIOS будут искать порты по альтернативным адресам 3E0h, 338h, 2E0h и 238h; иногда не производится поиск по адресам 3E8h и 2E8h. Нумерация найденных портов, отображаемая в заставке, может вводить в заблуждение: если установлены два порта с адресами 3F8h и 3E8h, в заставке они могут называться COM1 и COM2, и по этим именам на них можно ссылаться. Те же порты в заставке могут называться COM1 и COM3 (поскольку 3E8h является штатным адресом для COM3), однако попытка сослаться на порт COM3 окажется неудачной, поскольку в данном случае адрес 3E8h будет находиться в ячейке 0:402h BIOS Data Area, соответствующей порту COM2, а в ячейке порта COM3 (0:404h) будет нуль — признак отсутствия такового порта. «Объяснить» системе, где какой порт, можно вручную в любом отладчике, занеся правильные значения базовых адресов в ячейки BIOS Data Area (это придется делать каждый раз после перезагрузки ОС перед использованием «потерянного» порта). Существуют тестовые утилиты (например, Port Finder), позволяющие находить порты.

Если двум портам назначен один и тот же адрес, тестовая программа обнаружит ошибки порта только с помощью внешней заглушки *(External LoopBack).* Программное тестирование порта без заглушки не покажет ошибок, поскольку при этом включается диагностический режим (см. описание UART в [6]) и конфликтующие (по отдельности исправные) порты будут работать параллельно, обеспечивая совпадение считываемой информации. В «реальной жизни» нормальная работа конфликтующих портов невозможна. Разбираться с конфликтом адресов удобно, последовательно устанавливая порты и наблюдая за адресами, появляющимися в списке.

Если физически установлен только один порт и его не обнаруживает BIOS, причины те же, что в случае LPT-порта: либо он отключен при конфигурирова

нии, либо вышел из строя. Неисправность может самоустраниться при «передергивании» платы адаптера в слоте системной шины.

При работе с COM-портом задействуются соответствующие *аппаратные прерывания* — их используют при подключении модема, мыши и других устройств ввода. Неработоспособность этих устройств может быть вызвана некорректной настройкой запроса прерывания. Здесь возможны как конфликты с другими устройствами, так и несоответствие номера прерывания адресу порта.

Функциональное тестирование

В первом приближении COM-порт можно проверить диагностической программой (Checklt) без использования заглушек. Этот режим тестирования проверяет микросхему UART (внутренний диагностический режим) и прерывания, но не входные и выходные буферные микросхемы, которые являются более частыми источниками неприятностей. Если тест не проходит, причину следует искать в конфликте адресов/прерываний или в самой микросхеме U ART.

Для более достоверного тестирования рекомендуется использовать *внешнюю заглушку*, подключаемую к разъему COM-порта (рис. 16.8). В отличие от LPT- порта у COM-порта количество входных сигналов превышает количество выходных, что позволяет выполнить полную проверку всех цепей. Заглушка соединяет выход приемника со входом передатчика. Обязательная для всех схем заглушек перемычка RTS-CTS позволяет работать передатчику — без нее символы не смогут передаваться, если применяется «аппаратное» управление потоком[1]. Выходной сигнал DTR обычно используют для проверки входных линий DSR, DCD и RI.

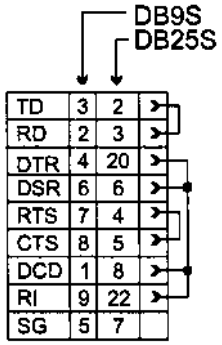


Рис. 16.8. Заглушка для проверки COM-портов (LoopBack для Checklt и Norton Diagnostics)

Если тест со внешней заглушкой не проходит, причину следует искать во внешних буферах, их питании или в ленточных кабелях, служащих для подключе

[1] Это не всегда так, поскольку в COM-портах сигнал CTS обрабатывается программно и протокол RTS/CTS можно отменить.

ния внешних разъемов. Здесь может помочь осциллограф или вольтметр. Последовательность проверки может быть следующей:

1. Проверить наличие двуполярного питания выходных схем передатчиков (этот шаг логически первый, но поскольку он технически самый сложный, его можно отложить на крайний случай, когда появится желание заменить буферные микросхемы).
2. Проверить напряжение на выходах TD, RTS и DTR: после аппаратного сброса на выходе TD должен быть отрицательный потенциал около -12 В (по крайней мере, ниже -5 В), а на выходах RTS и DTR — такой же положительный. Если этих потенциалов нет, возможна ошибка подключения разъема к плате через ленточный кабель. Распространенные варианты:
   - ленточный кабель не подключен;
   - ленточный кабель подключен неправильно (разъем перевернут или вставлен со смещением);
   - раскладка ленточного кабеля не соответствует разъему платы.

   Первые два варианта проверяются внимательным осмотром, третий же может потребовать некоторых усилий. В табл. 16.1 приведены три варианта раскладки 10-проводного ленточного кабеля для разъема COM-порта, известных автору; для COM-портов на системных платах возможно существование и других. Теоретически, ленточный кабель должен поставляться в соответствии с разъемом. Если дело в ошибочной раскладке, то эти три выходных сигнала удастся обнаружить на других контактах разъемов (на входных контактах потенциал совсем небольшой). Если эти сигналы обнаружить не удалось, очевидно, вышли из строя буферные формирователи.
3. Соединив контакты линий RTS и CTS (или установив заглушку), следует попытаться вывести небольшой файл на COM-порт — например, такой командой:

   COPY C:\AUTOEXEC.BAT COM1:

   С исправным портом эта команда выполнится за несколько секунд с сообщением об успешном копировании. При этом потенциалы на выходах RTS и DTR должны измениться на положительные, а на выходе TD должна появиться пачка двуполярных импульсов с амплитудой более 5 В. Если потенциалы RTS и DTR не изменились, ошибка кроется в буферных формирователях. Если на выходе RTS (и входе CTS) появился отрицательный потенциал, а команда COPY завершается с ошибкой, скорее всего, вышел из строя приемник линии CTS (или опять-таки ошибка в ленточном кабеле). Если команда COPY успешно проходит, а изменения на выходе TD не обнаруживаются (их можно увидеть с помощью стрелочного вольтметра, но оценить амплитуду импульсов не удастся), виноват буферный передатчик сигнала TD.

Замена микросхем приемников и передатчиков существенно облегчается, если они установлены в «кроватки». Перед заменой следует с помощью осциллографа или вольтметра удостовериться в неисправности конкретной микросхемы.

Если буферные элементы включены в состав интерфейсной БИС (что теперь весьма широко распространено), то такой порт ремонту не подлежит (по край

ней мере, в обычных условиях). Неисправный COM-порт, установленный на системной плате, можно попытаться отключить в CMOS Setup, но если порт сгорает вместе со схемой своего отключения, он остается «живым мертвецом» в карте портов ввода-вывода и прерываний. Иногда он полностью выводит из строя системную плату.

Источниками ошибок могут являться *разъемы* и *кабели.* В разъемах встречаются плохие контакты, а кабели помимо возможных обрывов могут иметь плохие частотные характеристики. Частотные свойства кабелей обычно сказываются при большой длине (десятки метров) на высоких скоростях обмена (56 или 115 Кбит/с). При необходимости использования длинных кабелей на высоких скоростях сигнальные провода данных должны быть перевиты с отдельными проводами схемной «земли».

Питание от интерфейса

При подключении к COM-порту устройств с небольшим энергопотреблением возникает соблазн использовать питание от выходных линий интерфейса. Если линии управления DTR и RTS не заняты по прямому назначению, их можно задействовать как питающие с напряжением около 12 В. Ток короткого замыкания на схемную «землю» ограничен буферной микросхемой передатчика на уровне 20 мА. При инициализации порта эти линии переходят в состояние «включено», то есть вырабатывают *положительное* напряжение. Линия TD в покое находится в состоянии логической единицы, так что на выходе вырабатывается *отрицательное* напряжение. Потенциалами линий можно управлять через регистры COM-порта (выход TD вырабатывает положительное напряжение, если установить бит BRCON).

Двуполярным питанием от линий интерфейса (+V от DTR и RTS, -V от TD) пользуются все манипуляторы, подключаемые к COM-портам. Зная это, в случае неработоспособности мыши с данным портом следует проверить напряжения на соответствующих контактах разъема. Бывает, что с конкретным портом не работает только конкретная мышь (модель или экземпляр), хотя другие мыши с этим портом и эти же мыши с другими портами работают нормально. Здесь дело может быть в уровнях напряжений. Стандарт требует от порта выходного напряжения не менее 5 В (абсолютного значения), и если данный порт обеспечивает только этот минимум, некоторым мышам может не хватить мощности для питания светодиодов (главных потребителей энергии).

Порт получает двуполярное питание через системную плату от блока питания компьютера. Отсутствие на выходе блока питания напряжения +12 В обычно обнаруживается по неработоспособности дисков. Отсутствие напряжения -12 В «заметят» только устройства, подключенные к COM-портам. Блок питания теоретически контролирует наличие этих напряжений на своем выходе (сообщая о неполадках сигналом Power Good, вызывающим аппаратный сброс). Встречаются упрощенные схемы блоков питания, у которых контролируются не все напряжения. Кроме того, возможны плохие контакты в разъеме подключения питания к системной плате.

16.2. Инфракрасный интерфейс IrDA

Применение излучателей и приемников инфракрасного (ИК) диапазона позволяет осуществлять беспроводную связь между парой устройств, удаленных на расстояние нескольких метров. Инфракрасная связь — *IR (InfraRed) Connection* — безопасна для здоровья, не создает помех в радиочастотном диапазоне и обеспечивают конфиденциальность передачи. ИК-лучи не проходят через стены, поэтому зона приема ограничивается небольшим, легко контролируемым пространством. Инфракрасная технология привлекательна для связи портативных компьютеров с периферийными устройствами. Инфракрасный интерфейс имеют некоторые модели принтеров, им оснащают многие современные малогабаритные устройства: карманные компьютеры (PDA), мобильные телефоны, цифровые фотокамеры и т. п.

Различают инфракрасные системы низкой (до 115,2 Кбит/с), средней (1,152 Мбит/с) и высокой (4 Мбит/с) скорости. Низкоскоростные системы служат для обмена короткими сообщениями, высокоскоростные — для обмена файлами между компьютерами, подключения к компьютерной сети, вывода на принтер, проекционный аппарат и т. п. Ожидаются более высокие скорости обмена, которые позволят передавать «живое видео».

В 1993 году была создана ассоциация *IrDA* (Infrared Data Association — ассоциация разработчиков систем инфракрасной передачи данных), призванная обеспечить совместимость оборудования от различных производителей. В настоящее время действует стандарт *IrDA 1.1*, наряду с которым существуют и собственные системы фирм Hewlett-Packard — *HP-SIR* (Hewlett-Packard Slow Infra Red) и Sharp — *ASK IR* (Amplitude Shifted Keyed IR).

Излучателем для ИК-связи является светодиод с длиной волны 880 нм; светодиод дает конус излучения с углом около 30°. В качестве приемника используют PIN-диоды, эффективно принимающие ИК-лучи в конусе 15°. Помимо полезного сигнала на приемник воздействуют помехи, в том числе засветка от солнечного освещения или ламп накаливания, дающая постоянную составляющую оптической мощности, и засветка от люминесцентных ламп, дающая переменную (но низкочастотную) составляющую. Эти помехи приходится фильтровать. Поскольку передатчик почти неизбежно вызывает засветку своего же приемника, вводя его в насыщение, приходится прибегать к полудуплексной связи с определенными временными зазорами при смене направления обмена. Для передачи сигналов используют двоичную модуляцию (есть свет — нет света) и различные схемы кодирования.

Спецификация IrDA определяет многоуровневую систему протоколов, которую рассмотрим снизу вверх. Ниже перечислены возможные варианты IrDA на физическом уровне:

- *IrDA SIR* — для скоростей 2,4-115,2 Кбит/с используется стандартный асинхронный режим передачи с 8-битной посылкой. Нулевое значение бита кодируется импульсом длительностью 3/16 битового интервала (1,63 мкс на скорости 115,2 Кбит/с), единичное — отсутствием импульсов (режим IrDA SIR-A). Таким образом, в паузе между посылками передатчик не светит,

а каждая посылка начинается с импульса старт-бита. В спецификации 1.1 предусмотрен и иной режим — IrDA SIR-B с фиксированной длительностью импульса 1,63 мкс для всех этих скоростей.

- *ASK IR* — для скоростей 9,6-57,6 Кбит/с также используется асинхронный режим, но кодирование иное: нулевой бит кодируется посылкой импульсов с частотой 500 КГц, единичный — отсутствием импульсов.
- *IrDA HDLC* — для скоростей 0,576 и 1,152 Мбит/с используется синхронный режим передачи и кодирование, аналогичное SIR, но с длительностью импульса 1/4 битового интервала. Формат кадра соответствует протоколу HDLC, начало и конец кадра отмечаются флагами 01111110, внутри кадра эта битовая последовательность исключается методом вставки битов (bit stuffing). Для контроля достоверности кадр содержит 16-битный CRC-код.
- *IrDA FIR* (IrDA4PPM) — для скорости 4 Мбит/с также применяется синхронный режим, но кодирование несколько сложнее. Здесь каждая пара смежных битов кодируется позиционно-импульсным кодом: 00 — 1000, 01 — 0100, 10 — 0010, 11 — 0001 (в четверках цифр единица означает посылку импульса в соответствующей четверти символьного интервала). Такой способ кодирования позволяет вдвое снизить частоту включения светодиода по сравнению с предыдущим. Постоянство средней частоты принимаемых импульсов облегчает адаптацию к уровню внешней засветки. Для повышения достоверности применяется 32-битный CRC-код.

Над физическим уровнем расположен *протокол доступа IrLAP* (IrDA Infrared Link Access Protocol) — модификация протокола HDLC, отражающая нужды ИК-связи. Он инкапсулирует данные в кадры и предотвращает конфликты устройств: при наличии более двух устройств, «видящих» друг друга, одно из них назначается первичным, а остальные — вторичными. Связь всегда полудуплексная. IrLAP описывает процедуру установления, нумерации и закрытия соединений.

Над IrLAP располагается *протокол управления соединением IrLMP* (IrDA Infrared Link Management Protocol). С его помощью устройство сообщает остальным о своем присутствии в зоне охвата (конфигурация устройств IrDA может изменяться динамически: для ее изменения достаточно поднести новое устройство или отнести его подальше). Протокол IrLMP позволяет обнаруживать сервисы, предоставляемые устройством, проверять потоки данных и выступать в роли мультиплексора для конфигураций с множеством доступных устройств. Приложения с помощью IrLMP могут узнать, присутствует ли требуемое им устройство в зоне охвата. Однако гарантированной доставки данных этот протокол не обеспечивает.

Транспортный уровень поддерживается протоколом *Tiny TP* (IrDA Transport Protocol) — здесь обслуживаются виртуальные каналы между устройствами, обрабатываются ошибки (потерянные пакеты, ошибки данных и т. п.), производится упаковка данных в пакеты и сборка исходных данных из пакетов (протокол напоминает TCP). На транспортном уровне может работать и протокол Ir TP.

Протокол IrCOMM позволяет через ИК-связь эмулировать обычное проводное подключение:

- 3-проводное по RS-232C (TXD, RXD и GND);
- 9-проводное по RS-232C (весь набор сигналов COM-порта);
- Centronics (эмуляция параллельного интерфейса).

Протокол IrLAN обеспечивает доступ к локальным сетям; он позволяет передавать кадры сетей Ethernet и Token Ring. Для ИК-подключения к локальной сети требуются устройство-провайдер с интерфейсом IrDA, соединенное обычным (проводным) способом с локальной сетью, и соответствующая программная поддержка в клиентском устройстве (которое должно войти в сеть).

Протокол объектного обмена IrOBEX (IrDA Object Exchange Protocol) — простой протокол, определяющий команды PUT и GET для обмена «полезными» двоичными данными между устройствами. Этот протокол располагается над протоколом Tiny TP. У протокола IrOBEX есть расширения для мобильных коммуникаций, которые определяют передачу информации, относящуюся к сетям GSM (записная книжка, календарь, управление вызовом, цифровая передача голоса и т. п.), между телефоном и компьютерами разных размеров (от настольного до PDA).

Этими протоколами не исчерпывается весь список протоколов, имеющих отношение к ИК-связи. Заметим, что для дистанционного управления бытовой техникой (телевизоры, видеомагнитофоны и т. п.) используется тот же 880-нано- метровый диапазон, но иные частоты и методы физического кодирования.

Приемопередатчик IrDA может быть подключен к компьютеру различными способами; по отношению к системному блоку он может быть как внутренним (размещаемым на лицевой панели), так и внешним, размещаемым в произвольном месте. Размещать приемопередатчик следует с учетом угла «зрения» (30° у передатчика и 15° у приемника) и расстояния до требуемого устройства (до 1 м).

Внутренние приемопередатчики на скоростях до 115,2 Кбит/с (IrDA SIR, HP- SIR, ASK IR) подключаются через обычные микросхемы UART, совместимые с 16450/16550 через сравнительно несложные схемы модуляторов-демодуляторов. В ряде современных системных плат на использование инфракрасной связи (до 115,2 Кбит/с) может конфигурироваться порт COM2. Для этого в дополнение к UART чипсет содержит схемы модулятора и демодулятора, поддерживающие один или несколько протоколов инфракрасной связи. Чтобы порт COM2 задействовать для инфракрасной связи, в CMOS Setup требуется выбрать соответствующий режим (запрет инфракрасной связи означает обычное использование COM2). Существуют внутренние адаптеры и в виде карт расширения (для шин ISA, PCI, PC Card), для системы они выглядят как дополнительные COM-порты.

На средних и высоких скоростях обмена применяются специализированные микросхемы контроллеров IrDA, ориентированные на интенсивный программный обмен (PIO) или DMA, с возможностью прямого управления шиной. Здесь обычная микросхема UART непригодна, поскольку она не поддерживает синхронный режим и высокую скорость. Контроллер IrDA FIR выполняется

в виде карты расширения либо интегрируется в системную плату; как правило, такой контроллер поддерживает и режимы SIR.

Приемопередатчик подключается к разъему *IR-Connector* системной платы напрямую (если он устанавливается на лицевую панель компьютера) или через промежуточный разъем (mini-DIN), расположенный на скобе-заглушке на задней стенке корпуса. К сожалению, единой раскладки цепей на внутреннем коннекторе нет, и для большей гибкости приемопередатчик (или промежуточный разъем) снабжают кабелем с отдельными контактами разъема. Собрать их в должном порядке предоставляют пользователю; варианты назначения контактов коннектора инфракрасного приемопередатчика приведены в табл. 16.3. Некоторые приемопередатчики, поддерживающие режимы FIR и SIR, имеют раздельные выходы приемников — IRRX для SIR и FIRRX для FIR. Если контроллер поддерживает только один из режимов, один из контактов остается неподключенным.

Таблица 16.3. Коннектор инфракрасного приемопередатчика

Цепь	Назначение	Контакт/вариант			
		1	2	3	4
IRRX	Вход с приемника	1	3	3	3
FIRRX	Вход с приемника FIR	–	–	–	4
IRTX	Выход на передатчик	3	5	1	1
GND	Общий	2	4	2	2
Vcc (+5B)	Питание	4	1	5	5
NC	Свободный	–	2	4	–

Внешние ИК-адаптеры выпускают с интерфейсом RS-232C для подключения к COM-порту или же с шиной USB. Пропускной способности USB достаточно даже для FIR, COM-порт пригоден только для SIR. Внешний ИК-адаптер IrDA SIR для COM-порта не так прост, как казалось бы: для работы модулятора-демодулятора требуется сигнал синхронизации с частотой, равной 16-кратной частоте передачи данных (этот сигнал поступает на синхровход микросхемы UART COM-порта). Такого сигнала на выходе COM-порта, нет и его приходится восстанавливать из асинхронного битового потока. Адаптер ASK IR в этом плане проще — передатчик должен передавать высокочастотные импульсы все время, пока на выходе TD находится сигнал высокого уровня; приемник должен формировать огибающую принятых импульсов.

Для прикладного использования IrDA помимо физического подключения адаптера и трансивера требуется установка и настройка соответствующих драйверов. В ОС Windows 9x/ME/2000 контроллер IrDA попадает в «сетевое окружение». Сконфигурированное ПО позволяет устанавливать соединение с локальной сетью (для выхода в Интернет, использования сетевых ресурсов); передавать файлы между двумя компьютерами; выводить данные на печать; синхронизировать данные PDA, мобильного телефона и настольного компьютера; выгружать отснятые изображения из фотокамеры в компьютер и выполнять ряд других полезных действий, не заботясь ни о каком кабельном хозяйстве.

16.3. Радиоинтерфейс Bluetooth

Bluetooth (синий зуб) — это фактический стандарт на миниатюрные недорогие средства радиопередачи информации на небольшие расстояния между мобильными (и настольными) компьютерами, их периферийными устройствами, мобильными телефонами и любыми другими портативными устройствами. Разработкой спецификаций занимается группа фирм, лидирующих в областях телекоммуникаций, компьютеров и сетей, — 3Com, Agere Systems, Ericsson, IBM, Intel, Microsoft, Motorola, Nokia, Toshiba. Эта группа (названная Bluetooth Special Interest Group) и вывела данную технологию на рынок. Спецификация Bluetooth свободно доступна в Сети (www.bluetooth.org), правда, весьма объемиста (более 1000 страниц базовой спецификации); открытость спецификации способствует ее быстрому распространению. Здесь позволим себе сократить название технологии до «BT» (это не официальное сокращение). Само название берет начало от прозвища датского короля, объединившего Данию и Норвегию, — намек на всеобщую объединяющую роль технологии.

Спецификации версий 1.0 появились в 1999 году, в 2000 году вышла отредактированная версия 1.1. В версии 1.2 (2003 г.) введен ряд усовершенствований: ускорено установление соединения, введена адаптация используемых частот, определены расширенные синхронные соединения; улучшены механизмы обнаружения ошибок, управления потоком и синхронизации. В 2004 году вышла версия, обозначаемая как 2.0 + EDR (Enhanced Data Rate), в которой появились скорости передачи данных 2 и 3 Мбайт/с и связанные с ними новые типы пакетов.

Физические каналы и пикосети

Каждое BT-устройство имеет радиопередатчик и приемник, работающие в диапазоне частот 2,4 ГГц. Этот диапазон в большинстве стран не требует лицензирования, что обеспечивает повсеместную применимость устройств. Передача ведется с перескоком несущей частоты, что помогает в борьбе с интерференцией и замираниями сигнала. В первоначальной версии BT используется 79 несущих частот: $F = 2402 + k$ (МГц), где $k = 0, ..., 78$. Для нескольких стран (например, Франции, где в этом диапазоне работают военные) возможен сокращенный вариант с 23 частотами, здесь $F = 2454 + k$ ($k = 0,..., 22$). В версии 1.2 введена возможность адаптации к текущему радиоокружению: из 79 часть (произвольный набор частот, «населенных» другими радиоустройствами) может быть исключена. При этом необходимость определения специального 23-частотного варианта отпадает.

По мощности передатчики могут быть трех классов: до 1, 2,5 и 100 МВт, причем должна возможность понижения мощности с целью экономии энергии. В зависимости от мощности передатчика обеспечивается дальность связи от единиц до сотни метров.

Данные по радиоканалу передаются с символьной скоростью 1 Мбод. В первоначальной версии используются частотная манипуляция и простое кодирова

ние: логической единице соответствует положительная девиация частоты, нулю — отрицательная. Такое кодирование обеспечивает передачу одного бита в каждом символе — *базовую скорость* (basic rate) 1 Мбит/с. Для *расширенных скоростей* (EDR) используются более сложные схемы с фазовой манипуляцией: с четырьмя возможными символами, обеспечивая скорость 2 Мбит/с, и с восемью символами — скорость 3 Мбит/с. Информация передается пакетами. Для обеспечения совместимости заголовки пакетов (управляющая информация) передаются всегда на базовой скорости, а на высокой скорости передается только поле данных.

Физический канал связи характеризуется определенной псевдослучайной последовательностью используемых частот и кодом доступа, передаваемым в начале каждого пакета. Канал делится на *тайм-слоты* длительностью 625 мкс, слоты последовательно нумеруются с цикличностью 2^{27}. Каждый тайм-слот соответствует одной частоте несущей в последовательности перескоков базового канала (1600 перескоков в секунду). Положение тайм-слотов задает устройство-мастер; под него подстраиваются ведомые устройства. Мастер и ведомые устройства ведут передачу поочередно: в четных слотах передачу начинает мастер, а в нечетных — адресованное им ведомое устройство (если от него требуется ответ). Передача пакета начинается на границе тайм-слота, пакет может занимать 1-5 тайм-слотов. Если пакет занимает более одного тайм-слота, то он весь передается на одной несущей частоте, но отсчет слотов по 625 мкс продолжается, и после длинного пакета следующая частота будет соответствовать очередному номеру слота (то есть несколько частот пропускаются).

Каждое устройство имеет свой *уникальный 48-битный адрес* BD_ADDR (Bluetooth device address), формируемый по стандарту IEEE 802. Из уникального адреса 28 бит используются для генерации последовательности перескоков и формирования кода доступа. Из всего набора возможных значений адреса 64 зарезервированы как стандартные адреса для операции опроса (inquiry). Из них формируются код доступа глобального опроса (GIAC, General Inquiry Access Code) и 63 выделенных кодов доступа (DIAC, Dedicated Inquiry Access Code) для устройств определенных классов.

Код доступа — это 68- или 72-битная последовательность, известная и передатчику, и приемнику физического канала. Приемник, прослушивая эфир и обнаруживая передачу информации, сравнивает ее с кодом доступа. При обнаружении совпадения последующая часть передачи принимается как продолжение ожидаемого пакета, при несовпадении — игнорируется. Таким образом удается отделять пакеты своего канала от пакетов других каналов, которые могут передаваться на той же частоте. Группа устройств, разделяющих один физический канал (то есть «знающих» одну и ту же последовательность перескоков и код доступа), образует так называемую *пикосеть* (piconet), в которую может входить от 2 до 8 устройств. В каждой пикосети имеются одно ведущее устройство (мастер) и до 7 активных ведомых. Кроме того, в зоне охвата ведущего устройства в его же пикосети могут находиться «припаркованные» ведомые устройства: они тоже «знают» последовательность перескоков и синхронизируются (по

перескокам) с мастером, но не могут обмениваться данными до тех пор, пока мастер не разрешит им активность.

Пикосети могут перекрываться зонами охвата, образуя «разбросанную» сеть (scatternet). При этом в каждой пикосети мастер только один, но ведомые устройства могут входить в несколько пикосетей за счет разделения времени (часть времени устройство работает в одной пикосети, часть — в другой). Более того, мастер одной пикосети может быть ведомым устройством другой пикосети. Эти пикосети никак не синхронизированы, каждая из них использует свой канал (последовательность перескоков). Устройство с помощью собственных «часов» поддерживает синхронизацию с каждой из этих пикосетей.

В BT определено несколько типов физических каналов, различающихся правилами перескоков частот и определения кода доступа. В каждый момент времени устройство может находиться только в одном физическом канале. При нормальной работе оно основное время находится в базовом (или адаптивном) канале пикосети, но периодически на некоторое время переключается в канал опроса и канал сканирования страниц, что обеспечивает возможность обнаружения новых устройств и соединения их в пикосети.

В *базовом физическом канале пикосети* псевдослучайная последовательность перескоков по 79 частотам и код доступа определяются по значению адреса мастера. Текущая частота определяется значением собственных часов мастера (CLKN, Clock Native). Ведомые устройства синхронизируются с мастером и отвечают уже на новой (в последовательности перескоков) частоте. Номинальная частота перескоков 1600 1/с, период повтора последовательности очень большой. Для работы базового канала ведомые устройства должны узнать физический адрес мастера, чтобы определить последовательность перескоков, и синхронизировать свои часы с мастером, чтобы их приемопередатчики переключали рабочие частоты синхронно с мастером. В базовом канале могут выделяться тайм-слоты для посылки пакетов-маяков (beacon); эти маяки требуются для ресинхронизации припаркованных устройств (если таковые имеются). В этих пакетах передаются данные широковещательного транспорта припаркованных устройств.

Адаптивный физический канал пикосети по назначению и характеристикам практически аналогичен базовому. Отличие заключается в адаптации перескоков (AFH, Adaptive Frequency Hopping) — мастер вычеркивает нежелательные частоты (оставляя не менее 20), при этом ведомое устройство отвечает на частоте предыдущего принятого им пакета. Адаптивный канал может быть установлен после установления базового канала (разрешением AFH).

В *физическом канале опроса* (inquiry scan) используется сокращенный набор частот (32 равномерно расположенные частоты). Опрашивающее устройство в каждом четном тайм-слоте выполняет передачу пакета опроса (одного и того же) на двух разных частотах (в начале и середине слота, так что частота перескоков составляет 3200 1/с). В последующем (нечетном) слоте это устройство ожидает приема ответа на тех же частотах. Отвечающее устройство через 625 мкс после приема пакета опроса ответит пакетом *FHS* на той же частоте. В пакете опроса используется один из нескольких предопределенных кодов доступа

(GIAC или DIAC). Опрашивающее устройство формирует тайм-слоты по своим собственным часам (CLKN), последовательность перескоков (укороченная псевдослучайная с периодом 32) соответствует используемому коду доступа.

В *физическом канале сканирования страниц* (page scan) также используется сокращенный набор частот (32). Псевдослучайная последовательность перескоков (с периодом 32) и код доступа определяются физическим адресом сканируемого устройства. Сканирующее устройство определяет текущую позицию в последовательности перескоков, используя ожидаемое им значение часов отвечающего (сканируемого) устройства (CLKE, Clock Estimated). Сканирующее устройство (pager) передает пакет в каждом четном тайм-слоте дважды на двух разных частотах, в последующем (нечетном) слоте оно ожидает приема ответа на тех же частотах. Необходимость передачи на двух частотах следует из несинхронизированности моментов смены слотов в двух устройствах; успех процедуры сканирования (установления) приводит к подстройке момента смены слотов у сканируемого устройства.

Синхронизация и установление соединений

Как уже упоминалось, канал в пикосети определяется последовательностью перескоков частот. Текущее положение (фаза) в этой последовательности определяется по «часам» устройства его внутреннему 28-разрядному счетчику импульсов частоты 3,2 кГц (номер слота определяется старшими 27 битами, младший бит определяет полуслот для каналов опроса и сканирования). Такие часы имеются в каждом устройстве, они работают постоянно и друг с другом никак не синхронизированы. Для того чтобы все время иметь представление о часах друг друга, устройства вычисляют текущее смещение относительно своих часов, используя информацию пакета *FHS*, передаваемого в ответ на пакет опроса (inquiry). Суммированием этого смещения со своими собственными часами устройство определяет ожидаемое значение часов партнера, что позволяет ему настраивать текущую частоту своего приемника на ожидаемую частоту сигнала от партнера и отсчитывать тайм-слоты, даже находясь в ином физическом канале. Поскольку генераторы всех устройств работают автономно, их частоты дрейфуют тоже автономно и смещение приходится периодически уточнять и корректировать. Автономный отсчет времени позволяет устройству «посещать» несколько каналов (в том числе и пикосетей) поочередно, имея представление об их времени и положении их тайм-слотов.

Мастером становится устройство, инициирующее соединение с одним или несколькими другими устройствами. Физически мастер и ведомые устройства идентичны, их «титулы» определяют лишь роли в протоколе; мастером может стать любое устройство. Более того, уже после развертывания пикосети может произойти смена ролей устройств.

Для установления соединения в протоколе предусмотрена специальная процедура *paging*. Каждое устройство, готовое войти в какую-нибудь пикосеть в качестве ведомого, периодически входит в *состояние сканирования* (page scan), используя соответствующий физический канал. В этом состоянии оно прослушивает эфир с последовательностью перескоков частот и кодом доступа, опре

деляемым его собственным адресом; перескоки выполняются по его часам. *Сканируемое устройство* ожидает приема пакета *ID*, содержащего только код доступа, соответствующий его собственному адресу. Пакет *ID* посылает *устройство-инициатор* установления соединения, которое предварительно должно было узнать адрес и часы сканируемого партнера с помощью процедуры опроса. На этот пакет сканируемое устройство отвечает таким же пакетом *ID* (на той же частоте). Получив ответ, инициатор посылает специальный пакет *FHS*, в котором сообщает свой адрес и класс устройства, показания своих часов, назначает сканируемому устройству его временный номер (LT_ADDR) и передает некоторые дополнительные параметры. На это сканируемое устройство отвечает все тем же пакетом *ID* и начинает использовать для перескоков адрес и часы инициатора, ставшего для него мастером (оно точно синхронизирует свои часы с моментом получения пакета *FHS*). Заключительным этапом установления соединения является посылка инициатором пакета *POLL*, на который сканируемое устройство обязано ответить (даже если ему нечего сообщить мастеру). С этого момента сканируемое устройство становится ведомым устройством в пикосети, а инициатор сканирования становится для него мастером.

Чтобы определить адреса окружающих устройств, а также «глянуть» на их часы, используется процедура опроса *inquiry*. Устройство-исследователь окружения посылает пакеты *ID* с кодом опроса (GIAC или DIAC) на одной из специальных (тоже коротких) последовательностях перескоков. Опрос может быть глобальным, с кодом GIAC — на него должны отзываться устройства всех классов, или же выборочным, с кодами DIAC. Процедура опроса напоминает процедуру *paging*, но здесь последовательность перескоков определяется уже не адресом устройства, а кодом доступа (GIAC или DIAC). Устройство отвечает на опрос с подходящим (или глобальным) кодом опроса пакетом *FHS*, в котором сообщает свой BD-адрес, 16-битный класс и показания часов на момент отправки. При этом возможно, что один и тот же пакет опроса одновременно примут несколько устройств и одновременно пошлют ответ (*FHS*). Для борьбы с этими коллизиями применяется механизм задержки ответов на случайный интервал времени, так что вероятность одновременного ответа двух устройств (это будет воспринято как помеха) снижается.

Логический транспорт, пакеты и каналы

Каждому активному ведомому устройству мастер назначает трехбитный *первичный логический транспортный адрес* — LT_ADDR, фигурирующий в заголовке пакетов, передаваемых мастером. Ведомое устройство обрабатывает принятые пакеты только со своим адресом (и широковещательные). У припаркованного устройства логического транспортного адреса нет. Между мастером и ведомыми устройствами могут устанавливаться *логические транспортные связи* нескольких типов.

Синхронный транспорт с установлением соединения (Synchronous Connection- Oriented link, SCO), он же изохронный, используется для передачи изохронного трафика (например, оцифрованного звука). Синхронные транспортные связи являются двухточечными; они предварительно устанавливаются мастером

с выбранными ведомыми устройствами, и для каждой связи определяется период (в слотах), через который для нее резервируются слоты. Обычные синхронные связи (SCO) получаются симметричные двусторонние, их скорость фиксирована — 64 Кбит/с в каждом направлении. Повторных передач пакетов в случае ошибок приема нет. Мастер может установить до трех связей SCO с одним или разными ведомыми устройствами. Ведомое устройство может иметь до трех связей с одним мастером или иметь по одной связи SCO с двумя разными мастерами.

Расширенный синхронный транспорт eSCO (Extended SCO) позволяет резервировать слоты для повторных передач пакетов, принятых с ошибкой. Связи могут быть как симметричными, так и асимметричными. На базовой скорости eSCO обеспечивает симметричную скорость до 288 Кбит/с, на скорости 2 Мбит/с — до 576 Кбит/с, а на скорости 3 Мбит/с — до 864 Кбит/с. По сетевой классификации связи SCO и eSCO относятся к *коммутации цепей.*

Асинхронный транспорт без установления соединения (Asynchronous Connection-Less link, ACL) реализует *коммутацию пакетов* по схеме «точка-точка» между мастером и ведомыми устройствами пикосети. Мастер может связываться с любым из ведомых устройств пикосети в слотах, не занятых под SCO, послав ему пакет и потребовав ответа. Ведомое устройство имеет право на передачу только получив обращенный к нему запрос мастера (безошибочно декодировав свой адрес). Для большинства типов пакетов предусматривается повторная передача в случае обнаружения ошибки приема. С каждым из своих ведомых устройств мастер может установить лишь одну связь ACL. На базовой скорости максимальная скорость передачи пользовательских данных (без заголовков) в симметричном варианте достигает 433,9 Кбит/с, в асимметричном —723,2/ 57,6 Кбит/с (от мастера скорость выше). На скорости 2 Мбит/с достижима симметричная скорость 869,7 Кбит/с или асимметричная 1448,5/115,2 Кбит/с. На скорости 3 Мбит/с достижима симметричная скорость 1306,9 Кбит/с или асимметричная 2178,1/177,1 Кбит/с.

Широковещательный транспорт позволяет мастеру передавать информацию для всех ведомых устройств своей пикосети, активных и припаркованных. Здесь обеспечивается связь «точка-множество точек».

Защита данных от искажения и контроль достоверности производятся несколькими способами. Данные ряда типов пакетов защищаются двухбайтным CRC-кодом, и приемник информации должен подтверждать прием правильного пакета или сообщать об ошибке приема. Для сокращения числа повторов в некоторых типах пакетов применяется упреждающая коррекция ошибок FEC (Forward Error Correction code) — избыточное кодирование. При использовании FEC 1/3 каждый полезный бит (включая и CRC) передается трижды, что позволяет выбрать наиболее похожий вариант мажорированием (совпадением двух из трех копий). Эта схема применяется и в заголовках пакетов, и в полях данных. Схема FEC 2/3 несколько сложнее, здесь используется код Хэмминга (15, 10) — из каждых полезных 10 бит генерируется 15-битный символ, что позволяет исправлять все однократные и обнаруживать все двукратные ошибки

в каждом 10-битном блоке. Если длина защищаемого битового поля не кратна **10,** то к нему добавляются дополнительные биты-заполнители.

В Bluetooth определены 5 логических каналов:

- Управляющий канал *LC* (Link Control), «спрятанный» в заголовках пакета, служит для низкоуровневого контроля (управление потоком, подтверждение приема, определение характеристик контейнеров).
- Управляющий канал *ACL-C* (в ранних версиях называвшийся LM, Link Manager) служит для обмена информацией между менеджерами соединений ведущего и ведомых устройств.
- Асинхронные и изохронные пользовательские каналы *ACL-U* (прежде называвшиеся UA/UI, User Asynchronous/Isochronous Data) несут пользовательские данные (пакеты 2-го уровня протокольного стека), которые могут передаваться и во фрагментированном виде (занимая более одного пакета нижнего уровня). Изохронность обеспечивается верхними протокольными уровнями.
- Синхронные каналы *SCO* (прежнее название US, User Synchronous Data) и *eSCO* прозрачно передают синхронные потоки данных, используя связи SCO или eSCO с соответствующими типами пакетов.

Каждый *голосовой капал* SCO обеспечивает скорость по 64 Кбит/с в обоих направлениях. В канале может использоваться кодирование в формате PCM (импульсно-кодовая модуляция) или CVSD (Continuous Variable Slope Delta Modulation — вариант адаптивной дельта-импульсно-кодовой модуляции). Кодирование PCM допускает компрессию по алгоритму G.711, оно обеспечивает лишь сугубо «телефонное» качество сигнала — имеется в виду цифровая телефония, 8-битные выборки с частотой 8 к/с (киловыборок в секунду). Кодер CVSD предлагает более высокое качество — он упаковывает входной PCM-сигнал с частотой выборок 64 к/с, однако и при этом спектральная плотность сигнала в полосе частот 4-32 кГц должна быть незначительной. Для передачи высококачественного аудиосигнала голосовые (речевые) каналы BT непригодны, для них предназначены eSCO или изохронные каналы ACL.

Для безопасности в BT применяются *аутентификация* и *шифрование данных* на уровне связи (link layer), которые, конечно же, могут дополняться средствами верхних протокольных уровней. На уровне связи средствами безопасности используется 48-битный уникальный адрес устройства (BD_ADDR), 128-битный личный ключ аутентификации пользователя, 8-128-битный личный ключ для шифрования данных и 128-битное часто сменяемое случайное (псевдослучайное) число.

Протоколы Bluetooth

Мы кратко рассмотрели, что происходит на нижних «этажах» технологии BT — каким образом организуются соединения по радиоканалу. Над этими этажами находятся другие протоколы, реализуемые в хост-контроллере, к которому подключено BT-устройство. Согласующей прослойкой является *протокол адаптации и управления логической связью* (Logical Link Control and Adaptation Pro

tocol, L2CAP), над которым располагаются все остальные «полезные» протоколы. Протокол L2CAP обеспечивает создание (и уничтожение) асинхронных и изохронных каналов между приложениями устройств BT и передачу по ним блоков пользовательских данных. На уровне L2CAP присутствуют менеджеры ресурсов и каналов, заведующие обеспечением требуемого качества обслуживания (QoS). Синхронные (аудиоканалы) BT стоят особняком — их данные не передаются через протокол L2CAP, аудиокодек «спрятан» в нижних этажах BT.

Важной частью BT является *протокол обнаружения сервисов* (Service Discovery Protocol, SDP), позволяющий устройству найти «интересного собеседника». В дальнейшем, установив с ним соединение, устройство сможет воспользоваться требуемыми сервисами (например, выводить документы на печать, подключаться к Сети и т. п.).

Интерфейс хост-контроллера (Host Controller Interface, HCI) — это единообразный метод доступа к аппаратно-программным средствам нижних уровней BT. Он предоставляет набор команд для управления радиосвязью, получения информации о состоянии и собственно передачи данных. Через этот интерфейс происходит взаимодействие протокола L2CAP с аппаратурой BT. Физически аппаратура BT может подключаться к различным интерфейсам: шине расширения (например, PC Card), шине USB, COM-порту. Для каждого из этих подключений имеется соответствующий протокол транспортного уровня HCI — прослойка, обеспечивающая независимость HCI от способа подключения.

Поскольку физический уровень (сам радиоинтерфейс) BT совпадает с одним из вариантов интерфейса беспроводных сетей (IEEE 802.11-FH), интерфейс Bluetooth можно получить на компьютере, оснащенном адаптером беспроводной сети, путем загрузки соответствующего программного драйвера.

Протокол RFCOMM обеспечивает эмуляцию последовательного порта (9-про- водного RS-232) через L2CAP. С его помощью традиционные кабельные соединения устройств (в том числе и нуль-модемные) могут быть легко заменены радиосвязью без каких либо модификаций ПО верхних уровней. Протокол позволяет устанавливать и множественные связи (одного устройства с несколькими), в этом случае радиосвязь заменит громоздкие и дорогие мультиплексоры и кабели. Через протокол RFCOMM может работать протокол OBEX, используемый в инфракрасных беспроводных соединениях (в иерархии протоколов IrDA). Через RFCOMM может работать и протокол PPP, над которым стоят протоколы стека TCP/IP, — это открывает дорогу во все приложения Интернета. Через RFCOMM работают и AT-команды, управляющие телефонными соединениями и сервисами передачи факсов (эти же команды используются в модемах для коммутируемых линий).

Специальный *бит-ориентированный телефонный протокол* (Telephony Control protocol — Binary, TCS BIN), определяющий сигнализацию вызова для связи BT-устройств (речевой связи и обмена данными), тоже работает через L2CAP. В протоколе имеются и средства управления группами устройств TCS.

ГЛАВА 17

Шина USB

USB (Universal Serial Bus — универсальная последовательная шина) является промышленным стандартом расширения архитектуры PC, ориентированным на интеграцию с телефонией и устройствами бытовой электроники. Версия стандарта 1.0 была опубликована в 1996 году, большинство устройств поддерживают стандарт 1.1 (1998 г.), в нем были устранены обнаруженные проблемы первой редакции. В спецификации USB 2.0 радикально повышена пропускная способность шины. Первоначально (в версиях 1.0 и 1.1) шина обеспечивала две скорости передачи информации: *полную скорость* (Full Speed, FS) 12 Мбит/с и *низкую скорость* (Low Speed, LS) 1,5 Мбит/с. В версии 2.0 определена еще и *высокая скорость* (High Speed, HS) 480 Мбит/с, что позволяет существенно расширить круг устройств, подключаемых к шине. В одной и той же системе могут присутствовать и одновременно работать устройства со всеми тремя скоростями. Шина позволяет с использованием промежуточных хабов соединять устройства, удаленные от компьютера на расстояние до 30 м. Подробную и оперативную информацию по USB (на английском языке) можно найти на сайте http://www.usb.org. Разработку устройств, их классификацию и стандартизацию координирует организация *USB-IF* (USB Implemented Forum, Inc.).

17.1. Архитектура USB

Шина USB представляет собой хост-центрическую аппаратно-программную систему подключения множества периферийных устройств. Хост-центричность понимается в нескольких аспектах:

- хост отвечает за конфигурирование всех устройств;
- хост управляет всеми обменами (транзакциями) на шине;
- обмен информацией возможен только между хостом (его памятью) и устройствами — однорангового взаимодействия устройств шина USB не позволяет.

Ниже перечислены компоненты *аппаратной части USB*:

- *Периферийные устройства USB* выполняют полезные *функции* (USB-functions).
- *Хост-контроллер* (host controller) обеспечивает связь шины с ядром компьютера. Хост-контроллер объединяется с *корневым хабом* (root hub), органи

зующим точки подключения устройств USB. Существует 2 варианта хост-контроллеров USB 1.x — универсальный (Universal Host Controller, UHC) и открытый (Open Host Controller, OHC). Оба варианта поддерживают скорости FS/LS; высокую скорость шины USB 2.0 (HS и только) поддерживает расширенный хост-контроллер (Enhanced Host Controller, EHC).

- *Хабы USB* (USB hubs) обеспечивают дополнительные точки подключения устройств.
- *Кабели USB* соединяют устройства с хабами.

Ниже перечислены компоненты *программной части USB*:

- *Клиентское ПО* (Client Software, CSw) — это драйверы устройств USB, обеспечивающие доступ к устройствам со стороны прикладного ПО. Драйверы взаимодействуют с устройствами только через программный интерфейс с общим драйвером USB (USBD). Непосредственного обращения к каким-либо регистрам аппаратных средств драйверы устройств USB не выполняют.
- *Драйвер USB* (USB Driver, USBD) «заведует» всеми устройствами USB системы, их нумерацией, конфигурированием, предоставлением служб, распределением пропускной способности шины, мощности питания и т. п.
- *Драйвер хост-контроллера* (Host Controller Driver, HCD) преобразует запросы ввода-вывода в структуры данных, размещенные в коммуникационной области оперативной памяти, и обращается к регистрам хост-контроллера. Хост-контроллер выполняет физические транзакции, используя эти структуры данных.

Работой всех устройств шины USB управляет *хост-контроллер* — программно-аппаратная подсистема *хост-компьютера*. Хост-контроллер является интеллектуальным устройством шины PCI или составной частью «южного» хаба (моста) системной платы, интенсивно взаимодействующей с оперативной памятью.

Программная часть хоста в полном объеме реализуется операционной системой. До загрузки ОС может функционировать лишь усеченный фрагмент программной части USB, поддерживающий только те устройства, которые требуются для загрузки. В спецификации PC'2001 к BIOS выдвигается требование поддержки USB в такой мере, чтобы ОС могла загружаться с устройств USB. Современные версии BIOS обеспечивают возможность загрузки с устройств хранения, подключенных к USB: винчестеров, CD/DVD и флэш-карт. После загрузки системы эта «дозагрузочная» поддержка игнорируется — система начинает работу с контроллером «с чистого листа», то есть со сброса и определения всех подключенных устройств.

В BIOS современных системных плат имеется *поддержка традиционного интерфейса клавиатуры и мыши*, подключаемых через контроллер 8042. В хост- контроллерах UHC и OHC для этого имеются аппаратные средства, перехватывающие обращения к портам 60h и 64h пространства ввода/вывода (это порты контроллера 8042). При разрешенной эмуляции старых устройств (legacy input devices) по обращениям ПО к этим портам контроллер вызывает прерывание *SMI* (System Management Interrupt — прерывание системного управления),

обрабатывающееся в ПК на процессорах x86 в режиме *SMM* (System Management Mode — режим системного управления) невидимо для обычных программ. Обработчик SMI, перехватывающий эти обращения, формирует последовательности действий, необходимые для их исполнения с помощью клавиатуры и/или мыши USB. В ОНС имеются специальные регистры, упрощающие задачу эмуляции.

17.2. Топология шины

Физическое устройство USB должно иметь интерфейс USB, обеспечивающий полную поддержку протокола USB, выполнение стандартных операций (конфигурирование и сброс) и предоставление информации, описывающей устройство. Физические устройства USB могут быть *комбинированными* (compound devices): включать в себя несколько устройств-функций, подключенных к внутреннему хабу, а также предоставлять своим внутренним хабом дополнительные внешние точки подключения.

Физическая топология USB — многоярусная «звезда» (рис. 17.1, а). Ее вершиной является хост-контроллер, объединенный с корневым хабом. Хаб является устройством-разветвителем; к тому же он может служить источником питания для подключенных к нему устройств. К каждому порту хаба может непосредственно подключаться периферийное устройство или промежуточный хаб; шина допускает до 5 уровней каскадирования хабов (не считая корневого). Поскольку комбинированные устройства содержат внутри себя хаб, их подключение к хабу 5-го уровня уже недопустимо. Каждый промежуточный хаб имеет несколько *нисходящих* (downstream) портов для подключения периферийных устройств (или нижележащих хабов) и один *восходящий* (upstream) порт для подключения к корневому хабу или нисходящему порту вышестоящего хаба.

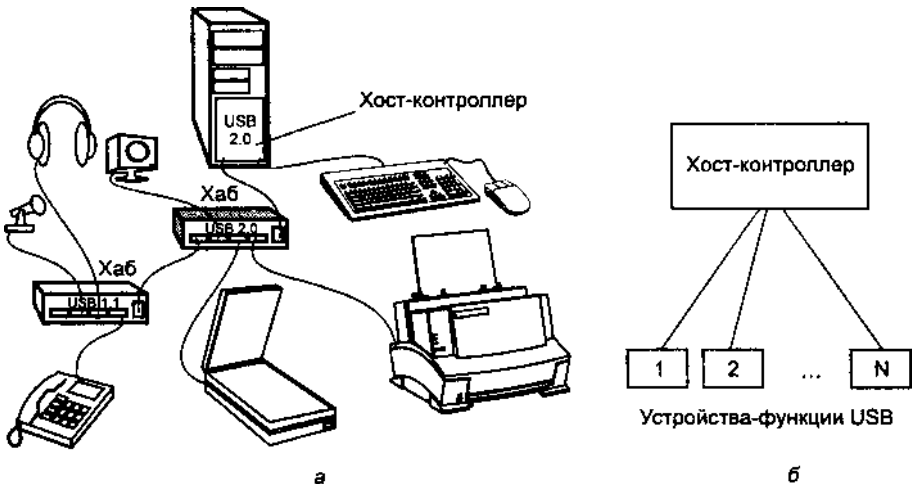


Рис. 17.1. Различные взгляды на отношения в USB: а — физическая топология, *б* — логическая топология

Логическая топология USB — «звезда». Хабы (включая корневой) создают иллюзию непосредственного подключения каждого логического устройства к хост- контроллеру (рис. 17.1, *б)*. В этой «звезде» устанавливаются сугубо подчиненные отношения по системе опроса-ответа: хост-контроллер по своей инициативе передает данные выбранному устройству или принимает их. Устройство по своей инициативе передавать данные не может; непосредственные передачи данных между устройствами невозможны. Устройство по своей инициативе может лишь сигнализировать о *пробуждении* (wakeup), для чего используется специальная сигнализация, но не передача данных.

Физический интерфейс USB прост и изящен. Конструкция кабелей и коннекторов USB (рис. 17.2, а и б) не дает возможности ошибиться при подключении устройств. Для распознавания разъема USB на корпусе устройства ставится стандартное символическое обозначение (рис. 17.2, *в)*. Гнезда типа А устанавливаются только на нисходящих портах хабов, вилки типа А — на шнурах периферийных устройств или восходящих портов хабов. Гнезда и вилки типа В используются только для шнуров, отсоединяемых от периферийных устройств и восходящих портов хабов (от «мелких» устройств — мышей, клавиатур и т. п. кабели, как правило, не отсоединяются). Для малогабаритных устройств имеются разъемы mini-B, а для поддержки расширения OTG (см. 17.8) имеются вилки mini-A и розетки mini-AB. Хабы и устройства обеспечивают возможность «горячего» подключения и отключения с сигнализацией об этих событиях хосту.

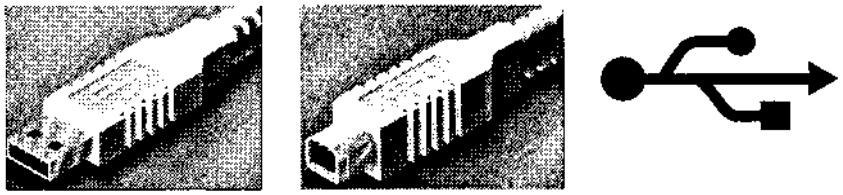


Рис. 17.2. Коннекторы USB: а — вилка типа А; *б* —вилка типа В; *в* — символическое обозначение

При планировании соединений следует учитывать способ питания устройств: устройства, питающиеся от шины, как правило, подключают к хабам, питающимся от сети. К хабам, питающимся от шины, подключают лишь маломощные устройства — так, к клавиатуре USB, содержащей внутри себя хаб, подключают мышь USB и другие устройства-указатели (трекбол, планшет).

Логическое устройство USB представляет собой набор независимых *конечных точек* (Endpoint, EP), с которыми хост-контроллер (и клиентское ПО) обменивается информацией. Каждому логическому устройству USB (как функции, так и хабу) конфигурационная часть ПО хоста назначает свой адрес (1-127), уникальный на данной шине USB. Каждая конечная точка логического устройства идентифицируется своим номером (0-15) и направлением передачи *(IN* — передача к хосту, *OUT* — от хоста). Точки *IN4* и *OUT4*, к примеру, представляют

собой разные конечные точки, с которыми могут общаться даже разные модули клиентского ПО. Набор конечных точек зависит от устройства, но всякое устройство USB обязательно имеет *двунаправленную конечную точку 0* (*EP0),* через которую осуществляется его общее управление. Для прикладных целей используются конечные точки с номерами 1...15 (1, 2 для низкоскоростных устройств). Адрес устройства, номер и направление конечной точки однозначно идентифицируют приемник или источник информации на данной шине при обмене хост-контроллера с устройствами USB. Каждая конечная точка имеет набор характеристик, описывающих поддерживаемый тип передачи данных (изохронные данные, массивы, прерывания, управляющие передачи, см. далее), размер пакета, требования к частоте обслуживания.

Устройство может решать несколько функциональных задач: например, привод CD-ROM может обеспечивать проигрывание аудиодисков и работать как устройство хранения данных. При этом в устройстве определяются *интерфейс* — набор конечных точек, предназначенных для решения данной задачи, и правила их использования. Таким образом, каждое устройство должно предоставлять один или несколько интерфейсов. Наличие нескольких интерфейсов позволяет нескольким драйверам, каждый из которых обращается только к своему интерфейсу (представляющему часть устройства USB), работать с одним и тем же устройством USB. Каждый интерфейс может иметь один или несколько *альтернативных вариантов* (альтернативных установок — alternate settings), из которых в данный момент активным может быть только один. Варианты различаются наборами (возможно, и характеристиками) используемых конечных точек.

Набор одновременно поддерживаемых интерфейсов составляет *конфигурацию устройства.* Устройство может иметь одну или несколько возможных конфигураций, из которых на этапе конфигурирования хост выбирает одну, делая ее *активной.* От выбранной конфигурации зависят доступная функциональность и зачастую — потребляемая мощность. Пока устройству не назначен номер выбранной конфигурации, оно не может функционировать в прикладном смысле, и ток потребления от шины не должен превышать 100 мА. Хост выбирает конфигурацию, исходя из доступности всех ресурсов, затребованных данной конфигурацией, включая и ток потребления от шины.

17.3. Модель передачи данных

Архитектура USB допускает четыре базовых *типа передач* данных между хостом и периферийными устройствами:

- *Изохронные передачи* (isochronous transfers) — потоковые передачи в реальном времени, занимающие предварительно согласованную часть пропускной способности шины с гарантированным временем задержки доставки. На полной скорости (FS) можно организовать один канал с полосой до 1,023 Мбайт/с (или два по 0,5 Мбайт/с), заняв 70 % доступной полосы (остаток можно занять и менее емкими каналами). На высокой скорости (HS) можно получить канал до 24 Мбайт/с (192 Мбит/с). Надежность доставки не гаранта-

руется — в случае обнаружения ошибки изохронные данные не повторяются, недействительные пакеты игнорируются. Шина USB позволяет с помощью изохронных передач организовывать *синхронные соединения* между устройствами и прикладными программами. Изохронные передачи нужны для потоковых устройств: видеокамер, цифровых аудиоустройств (колонки USB, микрофон), устройств воспроизведения и записи аудио- и видеоданных (CD и DVD). Видеопоток (без компрессии) шина USB способна передавать только на высокой скорости.

- *Прерывания* (interrupts) — передачи спонтанных сообщений, которые должны выполняться с задержкой не большей, чем требует устройство. Предел времени обслуживания устанавливается в диапазоне 10-255 мс для низкой и 1-255 мс для полной скорости. На высокой скорости можно заказать и 125 мкс. Доставка гарантирована, при случайных ошибках обмена выполняется повтор (правда, при этом время обслуживания увеличивается). Прерывания используются, например, при вводе символов с клавиатуры или передаче сообщений о перемещениях мыши. Прерываниями можно передавать данные и к устройству (как только устройство сигнализирует о потребности в данных, хост своевременно их передает). Размер сообщения может составлять 0-8 байт для низкой скорости, 0-64 байт — для полной и 0-1024 байт — для высокой скорости передачи.
- *Передачи массивов данных* (bulk data transfers) — это передачи без каких-либо обязательств по своевременности доставки и по скорости. Передачи массивов могут занимать всю полосу пропускания шины, свободную от передач других типов. Приоритет этих передач самый низкий, они могут приостанавливаться при большой загрузке шины. Доставка гарантированная — при случайной ошибке выполняется повтор. Передачи массивов уместны для обмена данными с принтерами, сканерами, устройствами хранения и т. п.
- *Управляющие передачи* (control transfers) используются для конфигурирования устройств во время их подключения и для управления устройствами в процессе работы. Протокол обеспечивает гарантированную доставку данных и подтверждение устройством успешности выполнения управляющей команды. Управляющая передача позволяет подать устройству команду (запрос, возможно, с дополнительными данными) и получить на него ответ (подтверждение или отказ от выполнения запроса и, возможно, данные). Только управляющие передачи на USB обеспечивают *синхронизацию запросов и ответов*; в остальных типах передач явной синхронизации потока ввода с потоком вывода нет.

Каждая единица клиентского ПО (обычно представляемая драйвером) связывается с одним интерфейсом своего устройства (функции) монопольно и независимо (рис. 17.3). Связи на этом рисунке обозначают *коммуникационные каналы* (communication pipes), которые устанавливаются между драйверами устройств и их конечными точками. Каналы могут устанавливаться только с конечными точками устройств, относящимися к выбранным (из альтернативных) вариантам интерфейсов активной конфигурации. Другие конечные точки недоступны.

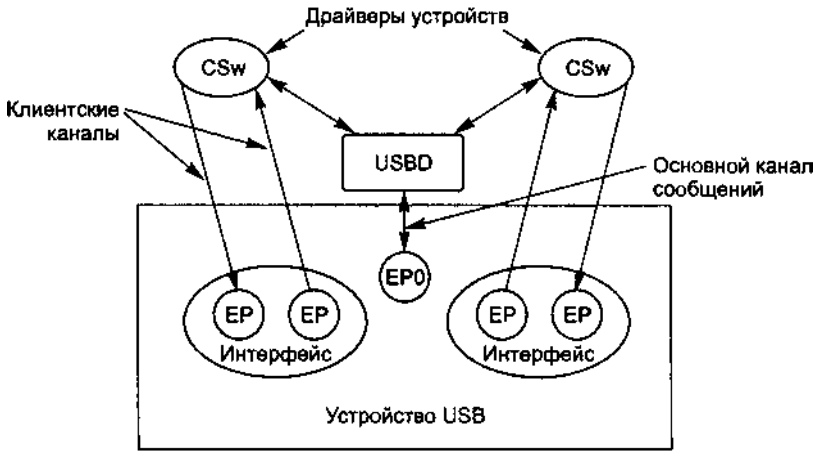


Рис. 17.3. Отношения клиентского ПО (CSw) с интерфейсами устройств USB

Запросы, пакеты и транзакции

Для передачи или приема данных клиентское ПО посылает каналу пакет *запроса ввода-вывода* (Input/Output Request, IRQ) и ждет уведомления о завершении его отработки. Формат IRP определяется реализацией драйвера USBD в конкретной ОС. В IRP имеются только сведения о запросе (местоположение буфера передаваемых данных в оперативной памяти и длина передачи); от свойств конкретного текущего подключения (скорость, допустимый размер пакета) драйвер устройства абстрагируется. Отработкой запроса в виде транзакций на шине USB занимается драйвер USBD; он при необходимости длинные запросы разбивает на части (пакеты), пригодные для передачи за одну транзакцию. *Транзакция* на шине USB — это последовательность обмена пакетами между хостом и ПУ, в ходе которой может быть передан или принят один *пакет данных* (возможны транзакции, в которых данные не передаются). Отработка запроса считается завершенной в случае успешного выполнения всех связанных с ним транзакций. «Временные трудности», встречающиеся при их выполнении (неготовность к обмену данными), до сведения клиентского драйвера не доводятся — ему остается только ждать завершения обменов (или выхода по тайм-ауту). Однако устройство может сигнализировать о серьезных ошибках (ответом *STALL),* что приводит к аварийному завершению запроса; о последнем уведомляется клиентский драйвер. В этом случае отбрасываются и все последующие запросы к данному каналу. Возобновление работы с данным каналом возможно лишь после явного уведомления об обработке ошибочной ситуации, которое драйвер устройства делает с помощью специального запроса (тоже вызова USBD).

Длинные запросы разбиваются на транзакции так, чтобы размер пакета был максимальным. Последний пакет с остатком может оказаться короче максимального размера. Хост может считать короткий пакет либо разделителем, указывающим на конец блока данных, либо признаком ошибки, по которому канал останавливается. При передаче массивов использование укороченных пакетов

в качестве разделителей наиболее естественно. Например, в одном из вариантов протоколов для устройств хранения данных укороченные пакеты известной длины применяются в качестве управляющих.

Каналы

Коммуникационные каналы USB разделяются на 2 типа:

- *Потоковый капал* (streaming pipe) доставляет данные от одного конца канала к другому, он всегда *однонаправленный*. Передачи данных в *разных* потоковых каналах друг с другом *не синхронизированы*. Это означает, что запросы клиентских драйверов для *разных каналов*, поставленные в определенном порядке относительно друг друга, могут выполняться другом порядке. Если во время исполнения какого-либо запроса происходит серьезная ошибка (об этом устройство сообщает ответом *STALL),* поток останавливается. Поток может реализовывать передачи массивов, изохронные передачи и прерывания.
- *Канал сообщений* (message pipe) является *двунаправленным*. Передачи сообщений во встречных направлениях *синхронизированы* друг с другом и строго *упорядочены*. На каждое сообщение противоположная сторона обязана ответить подтверждением его приема и отработки. Форматы сообщений определяются спецификацией USB: имеется набор стандартных сообщений (запросов и ответов) и зарезервированных идентификаторов сообщений, формат которых определяется разработчиком устройства или интерфейса.

С каналами связаны характеристики, соответствующие конечной точке (полоса пропускания, тип сервиса, размер пакета и т. п.). Каналы организуются при конфигурировании устройств USB. Полоса пропускания шины делится между всеми установленными каналами. Каналы различаются и по назначению:

- *Основной канал сообщений* (default pipe, он же control pipe 0), владельцем которого является USBD, используется для доступа к конфигурационной информации всех устройств. Этот канал устанавливается с *нулевой конечной точкой* (Endpoint Zero, EP0), которая у всех устройств всегда поддерживает только управляющие передачи.
- *Клиентские каналы* (client pipes) — каналы, владельцами которых являются драйверы устройств. По этим каналам могут передаваться как потоки, так и сообщения; они поддерживают любые типы передач USB (изохронные передачи, прерывания, массивы, управляющие передачи).

Интерфейс устройства, с которым работает клиентский драйвер, представляет собой *связку клиентских каналов* (pipe’s bundle). Для этих каналов драйверы устройств являются единственными источниками и потребителями передаваемых данных.

Владельцем основных каналов сообщений всех устройств является драйвер USB (USBD); по этим каналам передается информация конфигурирования, управления и состояния. Основным каналом сообщений может пользоваться и клиентский драйвер для текущего управления и чтения состояния устройства, но опосредованно — через USBD. Например, сообщения, передаваемые по основному каналу, используются драйвером принтера USB для опроса текуще

го состояния (передаются три признака в формате регистра состояния LPT- порта: *ошибка ввода-вывода, принтер выбран, отсутствие бумаги).*

17.4. Организация обменов по шине Кадры и микрокадры

Хост организует обмены с устройствами согласно своему плану распределения ресурсов. Для этого хост-контроллер циклически с периодом 1 мс формирует *кадры* (frames), в которые укладываются все запланированные транзакции[1] (рис. 17.4). Каждый кадр начинается с посылки пакета-маркера *SOF* (Start Of Frame), который является синхронизирующим сигналом для изохронных устройств, а также для хабов. Кадры нумеруются последовательно, в маркере *SOF* передаются 11 младших битов номера кадра. В режиме HS каждый кадр делится на 8 *микрокадров*, и пакеты *SOF* передаются в начале каждого микрокадра (с периодом 125 мкс). При этом во всех восьми микрокадрах *SOF* несет один и тот же номер кадра; новое значение номера кадра передается в нулевом микрокадре. В каждом кадре (микрокадре) может быть выполнено несколько транзакций, их допустимое число зависит от скорости, длины поля данных каждой из них, а также от задержек, вносимых кабелями, хабами и устройствами. Все транзакции кадров должны быть завершены до начала интервала времени *EOF* (End of Frame). Период (частота) генерации кадров (микрокадров) может немного варьироваться с помощью специального регистра хост-контроллера, что позволяет подстраивать частоту для изохронных передач.

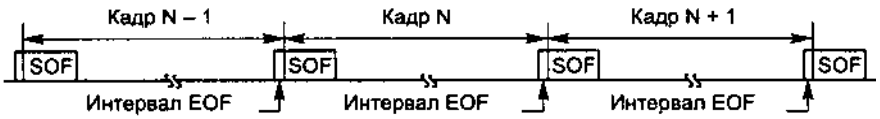


Рис. 17.4. Кадры шины USB

Кадрирование используется и для обеспечения живучести шины. В конце каждого кадра (микрокадра) выделяется интервал времени *EOF,* на время которого хабы запрещают передачу по направлению к контроллеру. Если хаб обнаружит, что с какого-то порта в это время ведется передача данных (к хосту), этот порт отключается, изолируя «болтливое» устройство, о чем информируется USBD.

Хост планирует загрузку кадров так, чтобы помимо запланированных изохронных транзакций и прерываний в них всегда находилось место для транзакций управления. Свободное время кадров может заполняться передачами массивов.

Протокол шины USB

Протокол шины USB обеспечивает обмен данными между хостом и устройством. На протокольном уровне решаются такие задачи, как обеспечение досто-

[1] В отличие от сетевой терминологии, в USB кадр является более крупной организационной единицей, чем пакет.

верности и надежности передачи, управление потоком. Для достоверности применяется избыточное кодирование (дублирование заголовков и CRC-контроль тела пакета). На физическом уровне используется техника вставки битов (bit stuffing), предотвращающая потерю синхронизации. Пакеты обрамляются специальными разделителями (Sync в начале и EOP в конце); при приеме контролируются размер пакета (он должен составлять целое число байтов), допустимость символов, корректность CRC и дублирующих полей заголовка. Любое нарушение заставляет считать пакет недействительным.

Весь трафик на шине USB передается посредством *транзакций,* в каждой транзакции возможен обмен только между хостом и адресуемым устройством (его конечной точкой).

Все транзакции (обмены) с устройствами USB состоят из двух-трех пакетов. Каждая транзакция планируется и начинается по инициативе хост-контроллера, который посылает *пакет-маркер* (token packet). *Маркер транзакции* описывает тип и направление передачи, адрес выбранного устройства USB и номер конечной точки. Адресуемое маркером устройство распознает свой адрес и готовится к обмену. Источник данных, определенный маркером, передает *пакет данных.* На этом этапе транзакции, относящиеся к изохронным передачам, завершаются — здесь нет подтверждения приема пакетов. Для остальных типов передач работает механизм подтверждения, обеспечивающий гарантированную доставку данных. Получатель данных отвечает пакетом *ACK*, означающим успешный прием и готовность, либо пакетом *NAK* или *NYET,* означающим ошибку или неготовность; на ошибочный пакет он не отвечает, и срабатывает механизм тайм-аута. Возможен и ответ *STALL* — серьезная ошибка, при которой «оживление» канала (конечной точки) требует программного вмешательства. В случае неудачи приема (тайм-аут, неготовность) попытка транзакции повторяется в следующем кадре (микрокадре).

Протокол управляющих передач из-за двунаправленности существенно отличается от передач других типов. Управляющие передачи состоят из двух или трех стадий и выполняются с помощью нескольких транзакций:

- *Стадия установки* (setup stage) предназначена для передачи управляющего сообщения от хоста к устройству. Это сообщение (всегда 8 байт) описывает команду (запрос), которую должно выполнить устройство. Команда может быть связана с передачей или приемом данных.
- *Стадия передачи данных* (data stage) предназначена для посылки дополнительной управляющей информации (в передаче *Write Control)* или приема информации от устройства (в передаче *Read Control).* Эта стадия может отсутствовать, если не требуется ввод информации, а выводимая информация умещается в сообщении стадии установки. Длина поля данных для этой стадии не ограничена.
- *Стадия передачи состояния* (status stage) предназначена для уведомлениях оста о факте завершения исполнения команды (устройство посылает пакет *ACK)*. Ответ устройства *NAK* означает, что запрос еще не выполнен, *STALL* —

отказ от выполнения данного запроса (здесь «оживление» канала не требуется).

Пропускная способность и совместная работа устройств с разными скоростями

К одной шине USB можно подключать устройства, работающие на существенно различающихся скоростях передачи. Чтобы обеспечить рациональное распределение времени кадров (микрокадров), для каждой из скоростей приняты соответствующие ограничения на максимальную длину поля данных пакета. Из этих ограничений вытекают ограничения достижимой пропускной способности устройств и шины в целом:

- На н*изкой скорости* (LS), составляющей 1,5 Мбит/с, в пакете может быть не более 8 байт данных, при этом двухстадийная транзакция управления (без фазы данных) занимает 30 % кадра, а транзакция прерывания — 14 %. Максимальная скорость обмена с одной точкой не превышает 8 Кбайт/с.
- На *полной скорости* (FS), составляющей 12 Мбит/с, поле данных для изохронных обменов содержит до 1023 байт (транзакция занимает 69 % кадра), максимальная скорость конечной точки составляет 1023 Кбайт/с. Для остальных типов допустима длина до 64 байт (5 % кадра). При этом, если с конечной точкой в каждом кадре выполняется одна транзакция, достигается скорость 64 Кбайт/с. В один кадр может уместиться до 19 таких транзакций, что соответствует суммарной пропускной способности шины 1,216 Мбайт/с. В принципе такая пропускная способность может достаться и одному устройству (его конечной точке), если другие устройства неактивны.
- На *высокой скорости* (HS), составляющей 480 Мбит/с, поле данных содержит до 3 х 1024 байт для прерываний и изохронных обменов (14 % микрокадра), максимальная скорость конечной точки составляет 24,576 Мбайт/с. Для передач массивов и управляющих передач допустим размер до 512 байт (7-8 % микрокадра), скорость при одной транзакции в микрокадре составляет 4,096 Мбайт/с. На практике одной точке для передач массивов может доставаться пропускная способность около 24 Мбайт/с (при расторопном хост-контроллере и неактивности других устройств). Суммарная пропускная способность шины может достигать 53,2 Мбайт/с.

Приведенные значения максимальной суммарной пропускной способности шины достигаются лишь в идеале. Реально они ниже, поскольку транзакции могут выполняться дольше из-за необходимости передачи дополнительных вставленных битов (в худшем случае они удлиняют пакеты в 7/6 раз), а также задержек в кабелях и хабах.

Низкоскоростные транзакции расходуют время кадра весьма неэффективно, но в USB 1.х с этим мирятся ради возможности подключения дешевых устройств и упрощения хабов, которые являются просто повторителями сигналов. Эффективное сосуществование трех скоростей в USB 2.0 реализуется сложнее и обходится дороже. Во-первых, хост-контроллер USB 2.0 содержит, фактически, два контроллера — контроллер EHC, работающий только на высокой ско

рости, и контроллер-компаньон (возможно, и не один) USB 1.x (UHC или OHC) — для полной и низкой скоростей. Корневой хаб имеет равноправные порты, но в процессе автоконфигурирования в зависимости от свойств подключенного к нему устройства (или хаба) каждый порт соединяется с соответствующим контроллером.

Во-вторых хаб USB 2.0 имеет более сложную структуру: помимо повторителя, он имеет еще и *транслятор транзакций.* Когда восходящий и нисходящие порты хаба работают на одинаковой скорости (FS или HS), хаб функционирует в режиме повторителя. При этом транзакция с устройством, подключенным к хабу, занимает весь канал от хост-контроллера до устройства на все время своего выполнения. Если же к порту хаба USB 2.0, работающего на скорости HS, подключается устройство или хаб 1.1, то применяются *расщепленные транзакции.* Здесь по части канала от хоста до хаба (его транслятора транзакций) обмен проходит на скорости HS, а между транслятором транзакций и устройством (или хабом) USB 1.x обмен идет уже на его «родной» скорости FS или LS. Эти обмены разнесены во времени, между ними могут вклиниваться любые транзакции на высокой скорости (в том числе и расщепленные). Таким образом, расщепленные транзакции позволяют не расходовать попусту пропускную способность высокоскоростной шины: транзакции с хабом на высокой скорости занимают в 40 (для FS) и даже в 320 (для LS) раз меньше времени шины, чем транзакции с самим целевым устройством[1]. От старых (USB 1.x) устройств и хабов все тонкости расщепленных транзакций скрываются, чем и обеспечивается обратная совместимость.

Порт хаба имеет возможность аппаратно определить, какую скорость поддерживает подключенное устройство. Все устройства со скоростью HS по включении работают в режиме FS и только после взаимного согласования с портом хаба переходят на скорость HS. Если устройство со скоростью HS подключается к хабу USB 1.x, который этого согласования не поддерживает, устройство остается в режиме FS (возможно, с усеченной функциональностью). В системе с USB 2.0 у устройства можно спросить (запросом дескрипторов), что изменится в его функциональности, если его подключить на другой скорости (изменив топологию соединений).

Вполне понятно, что устройство USB 2.0 сможет реализовать высокую скорость если по пути от него к хост-контроллеру (тоже 2.0) будут встречаться хабы 2.0. Если это правило нарушить и между ним и контроллером 2.0 окажется старый хаб, то связь может быть установлена только в режиме FS. Если такая скорость устроит и клиентское ПО (к примеру, для принтера и сканера это выльется только в большее время ожидания пользователя), то подключенное устройство работать будет, но появится сообщение о неоптимальной конфигурации соединений. По возможности конфигурацию следует исправить, благо переключения кабелей USB можно выполнять на ходу. Устройства и ПО, критичные к полосе пропускания шины, в неправильной конфигурации работать

[1] На самом деле отношения немного меньше из-за несколько больших накладных расходов как в целом на поддержание скорости HS, так и на расщепление транзакций.

откажутся и категорично потребуют переключений. Если же хост-контроллер старый, то все преимущества USB 2.0 окажутся недоступными пользователю. В этом случае придется менять хост-контроллер (менять системную плату или приобретать PCI-карту контроллера USB 2.0).

Контроллер и хабы USB 2.0 позволяют *повысить суммарную пропускную способность шины* и для старых устройств. Если устройства FS подключать к разным портам хабов USB 2.0 (включая и корневой), то для них возможно повышение суммарной пропускной способности шины USB по сравнению с 12 Мбит/с во столько раз, сколько используется портов высокоскоростных хабов. Конечно, при этом суммарная пропускная способность для всех устройств, включая устройства со скоростью HS, не может превышать общей пропускной способности HS-шины (нужно учитывать и накладные расходы). Кроме того, нужно учитывать архитектурные особенности хост-контроллера и хабов. Хост-контроллер может умножать пропускную способность FS/LS на число своих встроенных контроллеров USB 1.x. «Множительные способности» хаба зависят от реализации его транслятора транзакций.

Синхронизация при изохронной передаче

Изохронная передача данных связана с синхронизацией устройств, объединяемых в одну систему. Возьмем пример использования шины USB, в котором к компьютеру подключены микрофон USB (источник данных) и колонки USB (приемник данных) и эти аудиоустройства связаны между собой через программный микшер (клиентское ПО). Каждый из этих компонентов может иметь собственные «понятия» о времени и синхронизации: микрофон, к примеру, может иметь частоту выборки 8 кГц и разрядность данных 1 байт (поток 64 Кбит/с), стереоколонки — 44,1 кГц и разрядность 2x2 байта (176,4 Кбит/с), а микшер может работать на частоте выборок 32 кГц. Микшер в этой системе является связующим элементом, и его источник синхронизации будем считать главным (master clock). Программный микшер обрабатывает данные пакетами, сеансы обработки выполняются регулярно с определенным периодом обслуживания (скажем, в 20 мс — частота 50 Гц). В микшере должны быть *конверторы частот выборок* (Sample Rate Converter, SRC), которые из *n* входных выборок делают *m* выходных, используя интерполяцию («сочиняя» промежуточные выборки). Эти конверторы позволят микшеру принимать данные от микрофона с его частотой (в нашем случае 8000 выборок/с) и отсылать на колонки с другой (44 100 выборок/с). Естественным решением задачи обеспечения взаимодействия этих компонентов было бы установление между ними *синхронного соединения,* реализующего передачу и потока данных, и сигнала синхронизации. Универсальная шина USB, обеспечивающая одновременное подключение множества устройств, синхронного интерфейса устройствам не предоставляет. *Синхронное соединение* на USB основано на *изохронных передачах.* При этом приходится иметь дело со следующими частотами:

- *Fs* (sample rate) — частота выборки для источников (source clock) и приемников (sink clock) данных;

- *Fb* (bus clock) — частота шины USB: частота кадров (1 кГц) для полной скорости и микрокадров (8 кГц) для высокой — с этой частотой все устройства USB «видят» маркеры начала (микро)кадров *SOF*,
- *частота обслуживания* — частота, с которой клиентское ПО обращается к драйверам USB для передачи и приема изохронных данных.

В системе без общего источника синхронизации между парами синхросигналов возможны отклонения. В цифровой системе передачи данных эти отклонения выливаются в то, что у источника или приемника может образовываться излишек или недостаток данных, колеблющийся или прогрессирующий во времени. В USB по *типу синхронизации* источников или получателей данных с системой различают *асинхронный*, *синхронный* и *адаптивный* классы конечных точек, каждому из которых соответствует свой тип канала USB:

- *Асинхронная точка* не имеет возможности согласования своей частоты выборок с метками *SOF* или иными частотами системы USB. Частота передачи данных фиксированная или программируемая. Примерами асинхронного устройства-источника могут быть CD-плеер с синхронизацией от кварцевого генератора, приемник спутникового телевещания; пример приемника — дешевые колонки, работающие от внутреннего источника синхронизации.
- *Синхронная точка* связана с внутренним генератором, синхронизируемым с маркерами (микро)кадров *SOF* (1 или 8 кГц). Пример синхронного источника — цифровой микрофон с частотой выборки, синтезируемой по маркерам *SOF*. С программной точки зрения организация каналов с такими устройствами проще всего.
- *Адаптивная точка* имеет возможность подстройки своей внутренней частоты под требуемый поток данных (в разумных границах). Пример адаптивного источника — CD-плеер со встроенным конвертором частоты; пример приемника — высококачественные колонки или наушники USB.

Согласование скоростей выполняется с использованием механизма *прямого объявления скорости* (feed forward) или механизма *обратной связи* (feedback). Какой из механизмов используется, зависит от типа синхронизации, поддерживаемого изохронной конечной точкой данного устройства.

Шина USB позволяет устройству и хосту расставлять *метки времени* в непрерывном потоке изохронных передач для любой конечной точки. Для этого хост посылает устройству специальный управляющий запрос *Synch Frame*; в этом запросе хост указывает номер кадра (ожидаемого в обозримом будущем) и номер конечной точки, к которой относится данная метка времени. Устройства и хост имеют общее представление о времени по номеру кадра, передаваемому в маркере *SOF*. Метка времени может использоваться, например, для указания момента начала изохронной передачи. Таким образом, устройство может заранее подготовиться к началу изохронного обмена.

Хост-контроллер USB имеет возможность *подстройки частоты кадров* в пределах ±0,5 %. Естественно, хост-контроллер может подстроиться под частоту внутренней синхронизации только одного устройства.

17.5. Электрический интерфейс

Кабели и разъемы

Кабель USB содержит две пары проводов: одну для сигнальных цепей (D+ и D-) и одну для схемной «земли» (GND) и питания +5 В (Vbus). Допустимая длина сегмента (кабеля от устройства до хаба) — до 5 м. Ограничения на длину сегмента диктуются затуханием сигнала и вносимыми задержками. Максимальное удаление устройства от хост-контроллера составляет 30 м (5 хабов, 6 кабельных сегментов). Оно определяется задержкой, вносимой кабелями, промежуточными хабами и самими устройствами.

В *кабеле USB 1.x* используются витая пара проводов для сигнальных цепей и неперевитая пара для питания; требований к экранированию кабелей не выдвигалось. Для низкой скорости может применяться кабель с неперевитой парой сигнальных проводов (он тоньше и дешевле), но его длина не должна превышать 3 м.

В *кабелях USB 2.0* обязателен экран и связанный с ним дополнительный проводник. Такой кабель пригоден для работы на любых скоростях, включая и HS (480 Мбит/с).

Разъемы USB сконструированы с учетом простоты подключения и отключения устройств. Для реализации «горячего» подключения разъемы обеспечивают более раннее соединение и более позднее отсоединение питающих цепей по отношению к сигнальным. В USB определено несколько типов разъемов:

- *Л.* Гнезда (рис. 17.5, *а)* устанавливаются на нисходящих портах хабов, это стандартные порты подключения устройств. Вилки типа А устанавливаются на шнурах периферийных устройств или восходящих портов хабов.
- *В.* Разъемы этого типа используются для шнуров, отсоединяемых от периферийных устройств, и восходящих портов хабов (от «мелких» устройств — мышей, клавиатур и т. п. кабели, как правило, не отсоединяются). На устройстве устанавливается гнездо (рис. 17.5, *б),* на кабеле — вилка.

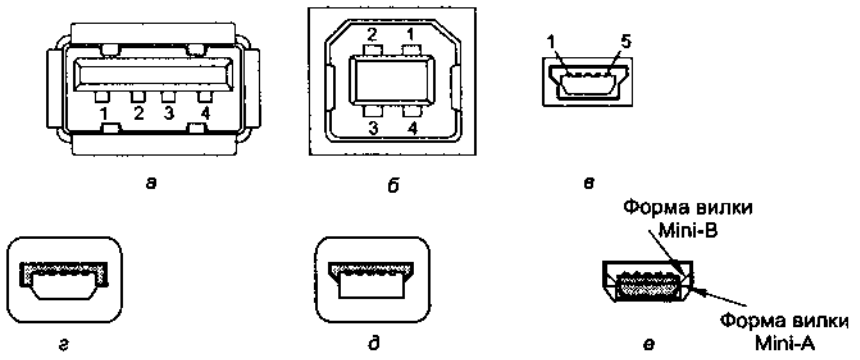


Рис. 17.5. Разъемы USB: а — гнездо А, *б* — гнездо В, *в* — гнездо mini-B, г — вилка mini-B, *д* — вилка mini-A, е — гнездо mini-AB

- *Mini-B.* Разъемы этого типа (рис. 17.5, в, *г)* используются для отсоединяемых шнуров малогабаритных устройств.
- *Mini-A.* Этот тип (рис. 17.5, *д*) введен в спецификации OTG (см. 17.8), вилки используются для подключения к портам малогабаритных устройств с гнездом mini-AB.
- *Mini-AB.* Гнезда (рис. 17.5, е) введены в спецификации OTG для портов двухролевых устройств, которые могут вести себя как хост (если в гнездо вставлена вилка mini-A) или как периферийное устройство (если в гнездо вставлена вилка mini-B).

Назначение выводов разъемов USB приведено в табл. 17.1, нумерация контактов показана на рис. 17.5. Штырьковые разъемы, устанавливаемые на системной плате (рис. 17.6), предназначены для кабелей-«выкидышей», которыми подключаются дополнительные разъемы USB, устанавливаемые на передней или задней стенках корпуса компьютера (иногда на боковых). На эти разъемы порты выводятся парами, причем у разных производителей подход к универсальности и защите от ошибочных подключений различен. Подключение «выкидыша», не подходящего к разъему, приводит к неработоспособности порта (к счастью, как правило, временной). Ошибка в подключении цепей GND и +5V может приводить к нагреванию кабелей и разъемов из-за короткого замыкания питающей цепи.

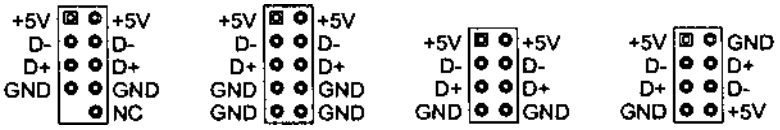


Рис. 17.6. Варианты разъема USB на системной плате

ВНИМАНИЕ ----------

Ошибка в полярности подводимого питания может необратимо повредить подключаемое устройство. По этой причине наиболее безопасными для подключаемого устройства являются внешние разъемы USB, запаянные на системной плате или карте контроллера USB. Их неправильное подключение маловероятно.

Таблица 17.1. Назначение выводов разъема USB

Цепь	Контакт стандартного разъема	Контакт мини-разъема
Vbus (+5 В)	1	1
D–	2	2
D+	3	3
GND	4	5
ID	–	4

Все кабели USB «прямые» — в них соединяются одноименные цепи разъемов, кроме цепи ID, используемой для идентификации роли устройства в OTG. На вилке *mini-A* контакт 4 (ID) соединен с контактом 5 (GND), что заставляет порт,

к которому подсоединена такая вилка, взять на себя роль нисходящего порта хаба. На вилке *mini-B* такого соединения нет.

Сигнальный интерфейс

Приемники и передатчики для организации аппаратного интерфейса имеют множество состояний линии и команд. При этом учитываются не только уровни электрических сигналов, но и время нахождения их в том или ином состоянии. По сочетанию уровней различают дифференциальные состояния *J* и *K* и состояние *SE0* (линейный ноль), когда обе линии на низком уровне. Для передачи данных применяются дифференциальные сигналы, кодирование NRZI и вставка битов. Это позволяет из одного дифференциального сигнала извлекать и данные, и сигналы битовой синхронизации (путем ФАПЧ).

Высокая скорость (480 Мбит/с — всего в 2 раза медленнее, чем Gigabit Ethernet) требует тщательного согласования приемопередатчиков и линии связи. На этой скорости может работать только кабель с экранированной витой парой для сигнальных линий. Для высокой скорости аппаратура USB должна иметь дополнительные специальные приемопередатчики. При подключении сначала HS-устройство ведет себя как FS-устройство (с соответствующими приемопередатчиками), после успешного согласование скорости HS приемопередатчики устройства (и хаба) переключаются на эту скорость.

Хаб обнаруживает *подключение устройства* по уровням напряжений D+ и D-:

- при отключенном устройстве на линиях D+ и D- уровни сигнала низкие, что обусловлено резисторами, нагружающими сигнальные линии в порте хаба;
- при подключении LS-устройства повышается уровень сигнала D- за счет резистора, расположенного в устройстве и подтягивающего его к напряжению питания;
- при подключении FS/HS-устройства повышается уровень сигнала D+ за счет аналогичного резистора в устройстве.

Последовательность обнаружения подключения и сброса устройств FS и LS иллюстрируют рис. 17.7, *а* и *б* соответственно. Хаб следит за сигналами нисходящего порта и сигнализирует об их смене. После обнаружения факта смены состояния системное ПО выжидает около 100 мс (время на успокоение сигналов) и проверяет состояния порта. Обнаружив факт подключения и определив тип устройства (LS или FS/HS), ПО дает для этого порта команду сброса шины.

Для *сброса шины (bus reset)* хаб опускает уровень поднятого устройством сигнала (D+ или D-) на 10-20 мс. Считается, что через 10 мс после этого сброса устройство должно быть готово к конфигурированию (отзываться только на обращения к *EP0* по нулевому адресу устройства).

Сброс шины для HS-устройства (рис. 17.7, *в)* запускает протокол *согласования скорости.* При подключении, как и по сигналу сброса, HS-устройство устанавливает свои схемы в состояние FS. Таким образом, поначалу HS-устройство выглядит для хаба как FS-устройство. Для согласования скорости используется так называемое «чириканье» (chirp-sequence): в ответ на состояние *SE0*, введен

ное хабом для сброса (заземлением линии D+), HS-устройство посылает импульс тока в линию D-. На этот импульс HS-хаб отвечает импульсом на линии D+. Такой обмен импульсами повторяется еще дважды; после успеха согласования и устройство, и хаб переходят на скорость HS, при этом линия переходит в состояние *SE0*. Теперь хосту надо снова опросить состояние порта хаба, чтобы уточнить режим подключенного устройства (FS или HS). Если HS-устройство подключено к FS-порту, хаб на «чириканье» устройства не отвечает.

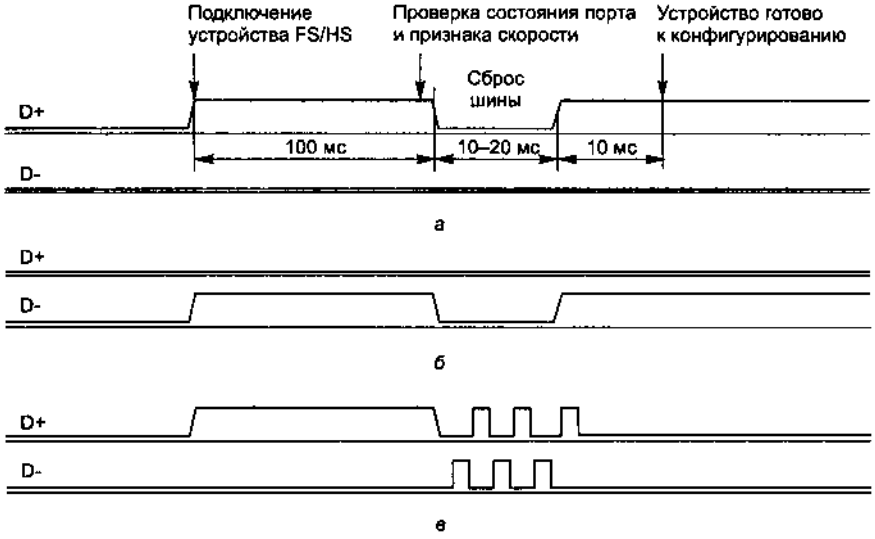


Рис. 17.7. Обнаружение подключения и сброса устройства: *а* — FS, *б* — LS, *в* — HS

Отключение FS/LS-устройств обнаруживается хабом просто по уровням напряжения. *Отключение HS-устройств* обнаружить сложнее, поскольку состояние шины (*SE0*) при отключении устройства не изменяется. Для обнаружения факта отключения HS-устройства используют эффект отражения сигнала при потере согласованности линии.

О *команде приостановки устройства* (suspend) хаб сигнализирует длительным состоянием покоя.

Сигналом к возобновлению работы (resume) является перевод шины в состояние *K* на длительное время (20 мс), достаточное для «оживления» устройств. Сигнал возобновления может подать как хаб, так и приостановленное устройство; последний случай называется *удаленным пробуждением.*

Удаленное пробуждение (remote wakeup) — это единственный случай на USB, когда сигнальную инициативу проявляет устройство (а не хост). Для сигнализации пробуждения устройство на некоторое время (1-15 мс) формирует состояние *K*, которое воспримется хабом как сигнал *Resume* и транслируется им на восходящий порт и на все разрешенные нисходящие порты, включая порт, с которого пришел данный сигнал. Таким образом, сигнал пробуждения достигает хост-контроллера.

Питание от шины

Шина USB обеспечивает устройства питанием по линии Vbus с номинальным напряжением 5 В относительно линии GND. Питание подается на нисходящие порты хабов; устройства-функции могут только потреблять питание (как и хаб по своему восходящему порту). Естественно, устройства могут пользоваться и собственным питанием. Ток питания от шины выделяют единицами по 100 мА, устройству шина может предоставить максимум 0,5 А. При начальном подключении (до конфигурирования) устройство может потреблять до 100 мА. Порт, обеспечивающий 0,5 А, называют *мощным* (high-power port); *маломощный порт* (low-power port) обеспечивает лишь 0,1 А. По отношению к питанию от шины различают следующие типы устройств:

- Корневой хаб получает питание вместе с хост-контроллером. При питании от внешнего источника хаб должен иметь мощные порты; при автономном питании (от батарей) порты могут быть как мощными, так и маломощными.
- Хаб, питающийся от шины (bus-powered hub), может иметь только маломощные порты (и не более четырех). Питание на нисходящие порты этот хаб подает только после конфигурирования.
- Хаб с автономным питанием (self-powered hub) может потреблять от шины лишь 0,1 А. На свои нисходящие порты он подает питание от другого источника; эти порты могут быть как мощными, так и маломощными (у хаба с батарейным питанием).
- Маломощные устройства-функции с питанием от шины (low-power bus-powered functions) могут потреблять не более 0,1 А.
- Мощные устройства-функции с питанием от шины (high-power bus-powered functions) могут потреблять до 0,5 А;
- Устройства-функции с автономным питанием (self-powered functions) могут потреблять от шины не более 0,1 А.

Способ питания устройства (и хаба) и максимальный ток, потребляемый от шины (с точностью до 2 мА), описываются в дескрипторе конфигурации устройства.

Питание на порты хабов подается с защитой от перегрузок из расчета 5 А на порт (это не отменяет нормы потребления). Срабатывание токовой защиты может индицироваться, например, гудком динамика системной платы ПК. Управление подачей питания у хаба может быть как общим (на все порты сразу), так и выборочным.

При питании от шины до устройств доходит напряжение, меньшее, чем дает хаб, из-за сопротивления питающих проводов и контактов разъемов. На каждом кабеле (между вилками A и B) в каждой из линий GND и Vbus может быть падение напряжения до 0,125 В. Худший случай по питанию — когда между источником питания (хабом с собственным питанием) и устройством находится один хаб с питанием от шины. Устройство, питающееся от шины, должно быть способно сообщать конфигурационную информацию при напряжении питания на вилке A своего кабеля от 4,4 В; маломощное устройство при таких ус

ловиях должно и нормально работать. Для нормальной работы мощного устройства требуется как минимум 4,75 В на его вилке.

ВНИМАНИЕ ----------

При недостаточном напряжении питания от шины устройство может быть неработоспособным (хорошо, если полностью, а не с загадочными симптомами). Проблемы питания могут возникать и от плохих кабелей (длинных и с тонкими проводами).

USB имеет развитую *систему управления энергопотреблением*. Хост-компьютер может иметь собственную систему управления энергопотреблением (power management system), к которой логически подключается и одноименная система USB. Программное обеспечение USB взаимодействует с этой системой компьютера, поддерживая такие системные события, как *приостановка* (suspend) и *возобновление* (resume). Кроме того, устройства USB могут сами являться источниками событий, отрабатываемых системой управления энергопотреблением хоста.

17.6. Хабы USB

Хаб является ключевым элементом технологии PnP в архитектуре USB. Хаб выполняет множество функций:

- обеспечивает физическое подключение устройств, формируя и воспринимая сигналы в соответствии со спецификацией шины на каждом из своих портов и транслируя трафик с восходящего порта на нисходящие и наоборот;
- обеспечивает управляемую информационную связь сегментов шины, включая и связь сегментов, работающих на разных скоростях, причем каждому нисходящему порту может быть селективно разрешена или запрещена трансляция трафика;
- отслеживает состояние подключенных к нему устройств, уведомляя хост об изменениях — подключении и отключении устройств;
- обнаруживает ошибки на шине, выполняет процедуры восстановления и изолирует неисправные сегменты шины (благодаря «бдительности» хабов неисправное устройство не может заблокировать всю шину);
- управляет энергопотреблением: подает питающее напряжение на нисходящие порты, селективно генерирует сигнал приостановки портов, транслирует эти сигналы в разных направлениях.

17.7. Хост-контроллер

Хост-контроллер является аппаратным посредником между устройствами USB и хостом. Хост-контроллер выполняет физические транзакции с устройствами по шине USB в соответствии с описаниями (дескрипторами) этих транзакций, помещенными в системное ОЗУ драйвером хост-контроллера. При этом тран

закции разных типов обрабатываются по-разному. В плане обработки ошибок проще всего изохронные транзакции, в которых ошибки не требуют повторов. Транзакции передач с гарантированной доставкой в случае ошибок требуют повторов до «победного конца» или до признания неудачи (исчерпания допустимого числа повторов). С точки зрения планирования выделяются периодические транзакции, которые должны выполняться строго по графику; остальные выполняются как получится, и их ставят в очереди. Из-за особенностей планирования и возможных повторов порядок завершения обработки дескрипторов транзакций (успешного или нет) для разных конечных точек будет отличаться от порядка их помещения в память, что прибавляет забот хост-контроллеру и его драйверу.

Основное взаимодействие драйвера с хост-контроллером происходит с помощью дескрипторов, расположенных в памяти. Прерывания от контроллера могут инициироваться различными событиями, такими как выполнение транзакций (избранных), обнаружение факта приема короткого пакета, прием сигнала возобновления или появление ошибки.

В настоящее время имеется три спецификации хост-контроллеров (UHC, OHC и EHC), каждой из них соответствует свой комплект драйверов хост-части. Приведенные выше задачи они решают по-разному и используют разные стратегии планирования транзакций.

Универсальный хост-контроллер

Разработанный Intel *универсальный хост-контроллер* (Universal Host Controller, UHC) для шины USB 1.x впервые появился в микросхеме PIIX3 (мост PCI-ISA) чипсетов системных плат для процессоров Pentium и продолжает использоваться во многих последующих изделиях Intel. Это хост-контроллер для скоростей FS/LS, который большую часть забот по планированию транзакций перекладывает на ПО — драйвер контроллера UHC (UHCD).

Драйвер UHC формирует для хост-контроллера дескрипторы, называемые в UHCI дескрипторами передач (Transfer Descriptor, TD), реально описывающие каждую *шинную транзакцию*. Одному *запросу ввода-вывода* (IRP) может соответствовать несколько таких «передач». Драйвер UHC разбивает запрос на транзакции и помещает дескрипторы этих транзакций в соответствующую очередь, а очередь включает в ближайшие планы. Драйвер отвечает за балансировку загрузки шины в каждом кадре. Планированием кадров обеспечивается требуемая частота обращений к точкам периодических передач.

Большое неудобство работы с UHC возникает из-за необходимости программного просмотра всех дескрипторов передач на предмет выявления завершенных. Дескрипторы завершенных передач приходится программно извлекать из цепочек, сохраняя связанность элементов. Планирование транзакций (составление списков дескрипторов и заголовков) — тоже достаточно трудоемкая задача для драйвера. Очевидно, преследовалась цель упрощения аппаратных средств хост-контроллера. Однако это оборачивается зависимостью эффективной производительности шины USB от мощности и загрузки центрального про

цессора. Такой подход к организации ввода-вывода трудно назвать эффективным.

Открытый хост-контроллер

Открытый хост-контроллер (Open Host Controller, OHC) для шины USB, как и универсальный хост-контроллер (UHC), предназначен для поддержки скоростей FS/LS. Спецификация его интерфейса (OHCI) разработана компаниями Compaq, Microsoft и National Semiconductor (1999 г.). В отличие от UHC, большую часть забот по планированию берут на себя аппаратные средства OHC, разгружая ЦП от рутины постоянной обработки дескрипторов. Контроллер OHC оперирует дескрипторами конечных точек и дескрипторами передач.

Дескрипторы конечных точек (Endpoint Descriptor, ED) создаются для всех сконфигурированных конечных точек всех подключенных устройств. Эти дескрипторы размещаются в памяти и связываются между собой; конфигурация связей задает порядок их обслуживания хост-контроллером. Дескриптор конечной точки описывает ее полный адрес, направление, тип, допустимый размер пакета, скорость, состояние точки и дескриптора.

Дескрипторы передач OHC, в отличие от UHC, действительно описывают *передачи по USB*. Каждая передача может разбиваться на несколько транзакций, и это разбиение выполняет хост-контроллер, исходя из размера пакета, установленного в дескрипторе конечной точки. Дескрипторы передач собираются в *очереди*, которые присоединяются к дескрипторам конечных точек. Существует и специальная очередь, в которую контроллер автоматически помещает отработанные дескрипторы из очередей запросов.

Контроллер OHC по сравнению с UHC наделен большим интеллектом. К сожалению, контроллер OHC на системных платах встречается нечасто, он более распространен в виде карт расширения (PCI).

Расширенный хост-контроллер

Расширенный хост-контроллер (Enhanced Host Controller, EHC) был введен фирмой Intel для поддержки высокой скорости в USB 2.0. Спецификация его интерфейса (EHCI) была опубликована в 2002 году. Контроллер EHC предназначен для работы с устройствами только на высокой скорости подключения к корневому хабу, при этом для FS/LS-устройств, которые подключены через промежуточный хаб USB 2.0, контроллер EHC выполняет расщепленные транзакции. С теми портами корневого хаба, к которым непосредственно подключены хабы и устройства USB 1.x, работает контроллер-компаньон (UHC или OHC). Коммутацию портов и контроллеров осуществляет маршрутизирующая логика, входящая в состав корневого хаба USB 2.0 (рис. 17.8). Обнаружением фактов подключения устройств к корневому хабу занимается драйвер EHC через регистры EHC. Обнаружив факт подключения FS/LS-устройства, драйвер перекоммутирует данный порт на контроллер-компаньон, и с этого момента порт отдается в ведение компаньону и его драйверу. Компаньон и его драйвер могут и не «знать» о том, что они работают в составе контроллера USB 2.0. Для

портов, остающихся в ведении UNC/OHC, эмулируется внешний хаб — ПО манипулирует портами, используя стандартные запросы к хабам USB.

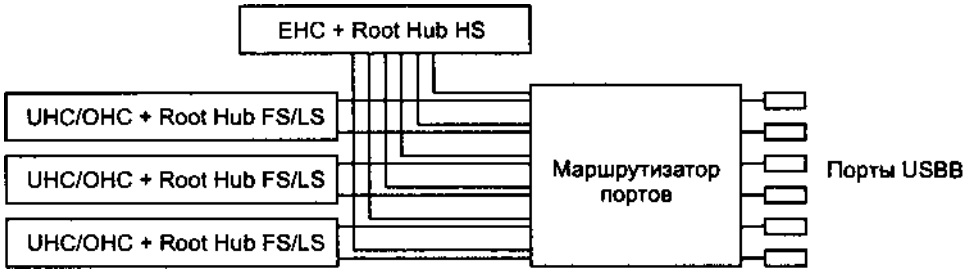


Рис. 17.8. Структура хост-контроллеров для USB 2.0

С точки зрения взаимодействия с драйвером EHC отчасти напоминает UHC, но высокая скорость передачи (480 Мбит/с) потребовала усиления интеллекта контроллера с целью уменьшения числа операций обмена между драйвером, памятью и контроллером. В EHC просматриваются многие идеи, заложенные в OHC. Структуры данных разработаны с учетом минимизации обращений к памяти.

В EHCI с точки зрения планирования транзакций передачи делятся на *периодические* (изохронные передачи и прерывания) и *асинхронные* (управляющие передачи и передачи массивов). Каждый из этих двух планов реализуется по-своему и может быть включен в работу и выключен. Контроллер начинает каждый микрокадр с выполнения периодических передач (если они разрешены), оставшееся от них время выделяется для выполнения асинхронных передач. За то, чтобы в микрокадре оставалось время для асинхронных передач, отвечает драйвер. Хост-контроллер аппаратно следит лишь за тем, чтобы транзакции не пересекали границу микрокадра.

Для всех *передач с гарантированной доставкой* (прерывания, управляющие передачи и передачи массивов) используются *очереди буферов*, в которых автоматически обеспечивается упорядоченное исполнение потоков передач. В EHC под передачей понимается последовательность однотипных транзакций[1]; ограничен лишь суммарный размер передаваемых блоков (20 Кбайт). Драйвер может динамически (во время исполнения плана) добавлять новые передачи в очереди. Для *изохронных передач* используются специальные структуры данных, различающиеся для HS-устройств и FS-устройств (для последних требуется расщепление транзакций).

Основой *планирования периодических транзакций* является *список кадров.* Основой планирования непериодических транзакций является *асинхронный список*, представляющий собой кольцо из заголовков очередей. Обслуживание всех асинхронных очередей выполняется по кругу, возможен и специальный *режим*

[1] По сравнению с UHC это уже некоторый прогресс.

парковки, в котором контроллеру разрешается выполнить подряд несколько транзакций из одной очереди.

Постановка запросов передач в очереди, как и включение изохронных передач в план, а также добавление/удаление очередей, могут выполняться драйвером динамически во время работы хост-контроллера. Для «сбора урожая» — поиска отработанных передач — драйверу приходится просматривать во всех дескрипторах передач признаки активности. Такого сервиса, как очередь исполненных передач (как в OHC), контроллер EHC не предоставляет. Но по сравнению с UHC, конечно же, объем работ драйвера EHC сокращается, поскольку этот контроллер оперирует передачами, а не транзакциями. Однако у драйвера EHC появляется дополнительная довольно сложная задача — планирование расщепленных транзакций.

17.8. USB без ПК — расширение OTG

Протокол шины USB ориентирован на сугубо подчиненные отношения: всеми транзакциями со всеми подключенными устройствами управляет хост — как правило, это ПК с контроллером USB. Никакого равноправия в отношениях на шине USB быть не может, однако в ряде случаев хотелось бы обойтись и без компьютера. Так, напрашивается непосредственное соединение цифровой фотокамеры и фотопринтера, чтобы обеспечивать печать снимков без участия ПК. Практически все периферийные устройства USB имеют встроенные микроконтроллеры, и функциональные возможности этих микроконтроллеров неуклонно растут. Периферийное устройство, имеющее даже простейшие средства диалога с пользователем (дисплей, отображающий хотя бы строку текста и несколько кнопок управления), вполне может взять на себя управляющие функции в плане организации транзакций USB. Функции такого мини-хоста можно упростить, если ориентироваться на двухточечное соединение пары устройств без промежуточных хабов. В этом случае мини-хосту остается лишь идентифицировать одно подключенное устройство, и если ему известно, как это устройство можно использовать, сконфигурировать его. Задача планирования транзакций лишь с одним устройством гораздо проще общей задачи «большого» хоста и хост-контроллера. Именно на создание таких упрощенных связей пары устройств нацелено расширение OTG (On-The-Go — связь «на ходу»).

Спецификация On-The-Go Supplement to USB 2.0 Specification (2003 г.) определяет дополнения к USB 2.0, необходимые для организации упрощенного соединения пары устройств. Большая часть спецификации посвящена описанию разъемов (см. рис. 17.5), и терминология OTG тоже привязана к типам разъемов (собственно, пользователь видит разъемы на устройствах и просто пытается соединить их доступными кабелями):

- *Устройство A (A-Device)* — устройство, в гнездо которого вставлена вилка типа *A* (или *mini-A*). Это устройство подает питание (Vbus) на шину и играет роль хоста, по крайней мере, в первое время после подключения к другому устройству. По ходу сеанса связи *устройство A* может передать функции хоста своему партнеру, а само стать периферийным (в терминах USB).

- *Устройство B (B-Device)* — устройство, в гнездо которого вставлена вилка типа *B* (или *mini-B*). Это устройство при подключении к другому устройству играет роль периферийного (ведомого) устройства USB. Если это устройство является двухролевым, то по ходу сеанса связи ему могут быть переданы функции хоста.
- *Двухролевое устройство* (*dual-role device)* — устройство с единственным гнездом типа *mini-AB*, обеспечивающее питание шины (не менее 8 мА) и обязательно поддерживающее полную скорость. Это устройство имеет усеченные возможности хоста, список поддерживаемых периферийных устройств, средства диалога с пользователем. Для управления связью устройство должно уметь работать по протоколам SRP и HNP (см. далее). Двухролевое устройство может поддерживать и хабы (это усложняет его задачи), однако стандартные хабы USB не позволяют работать протоколам SRP и HNP.

Протокол запроса сеанса (Session Request Protocol, SRP) предназначен для дополнительного энергосбережения: когда *устройство A* не нуждается в обмене по шине, оно может снять питание Vbus. При этом *устройство B* все-таки может «попросить внимания» — запросить сеанс связи. Здесь *сеансом* называется интервал времени, в течение которого двухролевое устройство подает достаточное (для работы) напряжение питания.

Протокол согласования роли хоста (Host Negotiation Protocol, HNP) позволяет *устройству A* и *устройству B* поменяться ролями во время сеанса связи (если они оба двухролевые).

17.8. Автоматическое конфигурирование устройств

USB поддерживает динамическое конфигурирование, отслеживая подключение и отключение устройств. USB позволяет идентифицировать подключаемые устройства, определять их потребности в ресурсах (полоса пропускания, питание от шины), выбирать нужную конфигурацию и управлять устройствами, то есть обеспечивает полную поддержку технологии PnP. Для этих целей определены «правила поведения» подключаемых устройств, система дескрипторов и стандартные управляющие запросы к устройствам. Ключевую роль в технологии PnP играют хабы, позволяющие селективно управлять работой подсоединенных к ним сегментов шины, что требуется на этапе конфигурирования. При работе шины постоянно идет процесс *нумерации* (enumeration) устройств, в ходе которого отслеживаются изменения физической топологии.

Все устройства подключаются через порты хабов. Хабы определяют факты подключения и отключения устройств к своим портам и по запросу сообщают о состоянии портов. Хост выполняет сброс и разрешает работу порта (одного!), на котором обнаружено новое подключение. При начальном подключении или после сброса устройство находится в «дежурном» состоянии — отзывается только на обращения по основному каналу сообщений (*EP0*) и имеет *нулевой адрес*. Таким образом, обращаясь к устройству по нулевому адресу, хост взаимодейству

ет только с одним новоподключенным устройством. Хост стандартными запросами считывает дескрипторы этого устройства и назначает ему *уникальный адрес на шине* (1-127). Таким образом хост заполняет свой перечень подключенных устройств. По назначении уникального адреса устройство переходит в состояние «адресовано», но его полноценное функционирование пока еще не разрешено. Полноценная работа устройства возможна только после управляющего запроса от хоста, выбирающего конфигурацию из числа доступных, — устройство переходит в состояние «сконфигурировано».

Если новое устройство является хабом, хост, сконфигурировав его, таким же способом определяет подключенные к нему устройства, идентифицирует их, назначает адреса и конфигурирует. Если новое устройство является функцией, уведомление о подключении передается заинтересованному ПО и при необходимости для него загружаются клиентские драйверы.

Когда устройство отключается, хаб автоматически отключает соответствующий порт и сообщает об отключении хосту, который удаляет сведения о данном устройстве из всех рабочих структур данных (но не из реестра Windows!).

17.10. Проблемы при подключении устройств USB

Первая проблема, возникающая при подключении устройства USB, — выбор порта подключения, подходящего с разных точек зрения:

- *Удобство подключения.* Для часто подключаемых/отключаемых устройств (например, флэш-памяти) удобны легкодоступные разъемы. Порты USB на задней панели корпуса ATX в этом плане непривлекательны, зато безопасны (см. далее).
- *Безопасность* (и *работоспособность*) *подключения.* На порт должны быть правильно выведены линии питания и сигналы USB (см. 17.5). В этом плане все порты, подключенные к системной плате через кабель, *потенциально опасны* — перепутанное напряжение питания может сжечь недешевое, возможно, устройство USB (флэш-память с ценными данными). При ошибочном подключении неработоспособным может оказаться подключаемое устройство и даже компьютер в целом.
- *Поддерживаемая скорость.* Режим HS может быть доступен не на всех портах компьютеров, поддерживающих USB 2.0 (см. далее). Бывает, что порт, ранее работавший в режиме HS, вдруг отказывается на него переходить. В этом случае обычно помогает перезагрузка ОС (может потребоваться и отключение питания). Для работы в режиме HS требуются поддержка USB 2.0 операционной системой (например, Service Pack 2 для Windows XP) и специальные драйверы хост-контроллера.
- *Питание от шины.* Мощные устройства, питающиеся от шины, нельзя подключать к хабам, не имеющим собственного питания (см. 17.5). Компьютер

с батарейным питанием может не обеспечивать тока питания, требуемого для подключаемых устройств.

ВНИМАНИЕ ----------

Подключение устройства с собственным питанием от сети при нарушении правил заземления или неисправности его блока питания может вывести из строя и устройство, и системную плату.

СОВЕТ ----------

Для первого подключения к каждому порту, расположенному на передней (или боковой) стенке системного блока, лучше использовать дешевую мышь (или специальный разъем-тестер со светодиодной индикацией). Если первое подключение проходит успешно, данному порту можно доверять.

В идеальном варианте с пользователя снимаются все заботы по конфигурированию подключаемых устройств и установке для них программного обеспечения. Однако на практике процесс подключения нового устройства не всегда происходит в таком «безоблачном» варианте. Проблемы, с которыми сталкиваются пользователи готовых устройств, как правило, сводятся к поиску подходящих драйверов и прикладного ПО. Конечно же, не надо забывать и о таких простых неполадках, как случайное отключение (через CMOS Setup) контроллера USB, находящегося на системной плате.

В ОС Windows каждое подключенное устройство USB отображается в окне диспетчера устройств (device manager). Для наглядности удобно в меню Вид выбрать команду Устройства по подключению, тогда устройства будут отображаться в виде деревьев, «растущих» из шины PCI. Каждое дерево начинается со своего хост-контроллера, к которому подключен корневой хаб. Заметим, что для каждого дерева (то есть для каждого хост-контроллера) адреса устройств USB назначаются независимо. К сожалению — ради удобства пользователя! — диспетчер устройств Windows отображает только подключенные устройства и не отображает свободных портов хабов. Это усложняет понимание организации портов и контроллеров. Более информативна утилита USB View (от Microsoft), которая подробнее показывает дерево устройств (хост-контроллеры, корневые хабы, промежуточные хабы и конечные устройства). Это позволяет определить внутреннюю структуру хост-части системы.

В компьютере с шестью портами USB и провозглашенной поддержкой USB 2.0 можно, например, увидеть «расширенный хост-контроллер USB 2.0» (EHC) с 6-портовым корневым хабом и три «универсальных хост-контроллера USB», у каждого из которых имеется по двухпортовому корневому хабу. В этом случае HS-устройство можно подключать к любому порту, и оно маршрутизирующей логикой будет связано с контроллером EHC. Для FS/LS-устройств каждая пара портов подключается к своему контроллеру-компаньону (UHC). Если бы мы в 6-портовой системе увидели расширенный контроллер с четырехпортовым корневым хабом, это означало бы, что поддержка USB 2.0 реализована только на четырех портах из шести.

Устройство, подключаемое к шине USB, должно последовательно пройти от состояния «запитано» (означающего, что на порт хаба, к которому подключено

устройство, подано питание) до «сконфигурировано» (см. выше). По тому, в каком состоянии «застревает» подключаемое устройство, можно судить о событиях, произошедших в системе, и найти неполадки.

Если устройство при подключении «зависает» в состоянии «запитано», это может быть вызвано рядом причин:

- Хост не получает сигнала о подключении устройства (высокий логический уровень на линии D+ или D-), который оно же и должно обеспечить. Этот сигнал может не дойти до хоста при неплотном соединении разъема USB (в нем сигнальные цепи замыкаются позже питающих).
- Хост не реагирует на сигнал подключения. Обычно хост реагирует посылкой сигнала шинного сброса — занулением линии D+ или D- на 10-20 мс, этот импульс легко можно увидеть с помощью осциллографа. «Заснувший» хост может и не реагировать на подключение (не посылать сигнал сброса).
- Устройство не реагирует на сигнал шинного сброса. Обычно устройство должно по сигналу шинного сброса перейти в «дежурное» состояние.

Если устройство «зависает» в «дежурном» состоянии, это означает его неспособность сообщить дескрипторы и исполнить запрос назначения уникального адреса.

Зависание в состоянии «адресовано» может означать, что никто в системе не заинтересовался подключенным устройством и не пытается его сконфигурировать. Это происходит, например, когда ОС не может связать с обнаруженным устройством подходящий драйвер.

В поддержке устройств USB, по крайней мере, в ОС Windows, наблюдается странность (с точки зрения автора) в учете устройств. Каждое вновь подключенное устройство после установки его ПО поддержки оставляет в реестре Windows запись, в которой некоторый идентификатор устройства связывается с именами модулей ПО, его поддерживающих. Если то же устройство переключить в другой порт (даже в другой порт промежуточного хаба), то ОС это воспримет как подключение нового(!) устройства и снова начнет установку ПО поддержки, создавая новую запись в реестре. Отключение устройства не вызывает удаления записи из реестра, так что реестр будет постоянно «обрастать» лишними записями. Само по себе это не так уж важно, но ПО некоторых устройств отказывается загружаться (или работать), когда в реестре прописан его двойник с другим адресом. В этой ситуации приходится либо возвращать устройство в порт первоначального подключения (что не всегда удобно), либо вручную чистить реестр (что рядовому пользователю «не к лицу»).

Для соединения пары компьютеров по USB нужно промежуточное устройство, которое увидят оба хоста и через которые смогут обмениваться пакетами. Кабель связи двух ПК снабжен парой вилок типа A, что для USB нехарактерно. В этом кабеле, как правило, и установлено промежуточное устройство («таблетка»). В системных платах встречается встроенное *коммуникационное устройство* USB Link, которое своим внешним портом USB соединяется с обычным портом хаба USB на другом ПК; для этого соединения используется чисто пассивный кабель. ПО позволяет организовать сеть из нескольких ПК с такими устройствами, соединенных цепочкой.

ГЛАВА 18

Шина IEEE 1394 - FireWire

Высокопроизводительная последовательная шина (high performance serial bus) IEEE 1394 — FireWire создавалась как более дешевая и удобная альтернатива параллельным шинам (SCSI) для соединения равноранговых устройств. Шина без дополнительной аппаратуры (хабов) обеспечивает связь до 63 устройств. Устройства бытовой электроники — цифровые камкордеры (записывающие видеокамеры), камеры для видеоконференций, фотокамеры, приемники кабельного и спутникового телевидения, цифровые видеоплееры (CD и DVD), акустические системы, цифровые музыкальные инструменты, а также периферийные устройства компьютеров (принтеры, сканеры, устройства дисковой памяти) — и сами компьютеры могут объединяться в единую сеть. Шина не требует управления со стороны компьютера. Шина поддерживает *динамическое реконфигурирование* — возможность «горячего» подключения и отключения устройств. События подключения-отключения вызывают сброс и повторную инициализацию: определение структуры шины (дерева), назначение физических адресов всем узлам и, если требуется, избрание ведущего устройства (мастера) циклов, диспетчера изохронных ресурсов и контроллера шины. Менее чем через секунду после сброса все ресурсы становятся доступными для последующего использования, и каждое устройство имеет полное представление обо всех подключенных устройствах и их возможностях. Благодаря наличию линий питания интерфейсная часть устройства может оставаться подключенной к шине даже при отключении питания функциональной части устройства.
По инициативе VESA шина позиционируется как основа «домашней сети», объединяющей всю бытовую и компьютерную технику в комплекс. Эта сеть является одноранговой (peer-to-peer), чем существенно отличается от USB.
Стандарт IEEE 1394 имеет ряд совместимых реализаций под разными названиями: FireWire, iLink, Digital Link, MultiMedia Connection. Наиболее часто используется разработанная фирмой Apple шина FireWire, на основе которой и появился стандарт.

18.1. Спецификации

Спецификация IEEE 1394 официально доступна на сайте http://www.ieee.org (платно). С вопросами лицензирования и интеллектуальной собственности можно ознакомиться на сайте http://www.1394la.com.
Стандарт IEEE 1394-1995 определяет архитектуру шины, основанную на трехуровневой модели (см. 18.2), и протоколы, обеспечивающие автоматическое

конфигурирование, арбитраж и передачу изохронного и асинхронного трафиков. В стандарте определены три возможные скорости передачи сигналов по кабелям: 98,304, 196,608 и 393,216 Мбит/с, которые округляют до 100, 200 и 400 Мбит/с и обозначают как S100, S200 и S400 соответственно. Стандартизованы кабель и 6-контактный разъем, позволяющий передавать сигналы и питание.

В *дополнение IEEE 1394a* (2000 г.) введен ряд усовершенствований:

- Повышена эффективность использования шины (ускоренный сброс, ускоренный арбитраж, конкатенация пакетов, передаваемых на разных скоростях).
- Введен миниатюрный 4-контактный разъем (без питающих линий).
- Расширены средства управления энергопотреблением и введена возможность приостановки и запрета портов. Введена возможность общения с регистрами физического уровня удаленного узла.

Новых скоростей в этом стандарте не появилось; изменения вводились с учетом обеспечения обратной совместимости с устройствами, отвечающими исходному стандарту.

Дополнения IEEE 1394b (2002 г.) в основном касаются повышения скорости и дальности передачи:

- Введен новый метод сигнализации (бета-сигнализация). В этом методе используются пара встречных однонаправленных линий и соответствующий бета-режим работы портов со старыми и новыми (S800, S1600) скоростями (планируется и S3200).
- Введен миниатюрный 9-контактный разъем (для скоростей до 3,2 Гбит/с с подачей питания).
- Введены новые типы среды передачи (для бета-режима):
  - пара пластиковых или стеклянных оптических волокон для расстояний до 50 или до 100 м;
  - кабель UTP-5 (используются 2 пары) с разъемами RJ-45 для расстояний до 100 м с трансформаторной гальванической развязкой.
- Введен новый метод арбитража (BOSS), повышающий эффективность использования пропускной способности шины за счет исключения простоев шины (зазоров арбитража).

Совместимость с 1394 и 1394a обеспечивается «двуязычными» физическими уровнями, способными работать с разными методами сигнализации на разных портах одной шины. При этом возможно построение гибридной шины, состоящей из одного или нескольких «облаков» узлов с бета-сигнализацией, связанных друг с другом фрагментами с традиционной сигнализацией.

18.2. Организация, топология и архитектура

Стандарт IEEE 1394 описывает шину с последовательным интерфейсом, по которой информация передается *пакетами*. Источник пакетов должен получить право передачи пакета, используя механизм арбитража, в котором задействуют

ся все устройства, подключенные к шине. *Арбитраж* предоставляет узлам право доступа в соответствии с запрошенным типом передачи. Для асинхронных транзакций арбитраж обеспечивает справедливое распределение полосы пропускания; для изохронных передач — гарантированную (предварительно согласованную) полосу пропускания для каждого канала. Коллизии (столкновения пакетов от нескольких устройств) в исправной шине отсутствуют. Арбитраж в IEEE 1394 основан на прослушивании шины и определении зазора — покоя шины. В последующих дополнениях механизм арбитража усовершенствован для повышения эффективности использования шины (зазоры арбитража — непродуктивный расход времени).

ТОПОЛОГИЯ

Устройства, подключаемые к шине, имеют один или несколько портов IEEE 1394, объединенных внутренним повторителем. Устройства соединяются друг с другом кабелями (сегментами), при этом допускается большая свобода выбора *топологии физических соединений*. Пример топологии соединения приведен на рис. 18.1. Стандарт накладывает на топологию следующие ограничения:

- на шине может быть не более 63 узлов;
- между любой парой узлов может быть не более 16 кабельных сегментов;
- длина сегмента стандартного кабеля не должна превышать 4,5 м (в IEEE 1394b для ряда типов кабелей допустима длина до 100 м);
- суммарная длина кабеля не должна превышать 72 м;
- топология не должна иметь петель[1].

При этом *логическая топология для передачи данных* остается *шинной* — пакеты распространяются от источника ко всем узлам шины[2]. *Логическая топология для арбитража — древовидная* иерархическая, «верховный арбитр» — корневой узел.

Архитектура сети

Архитектура IEEE 1394 позволяет организовывать сети, состоящие из одной или нескольких (до 1023) шин, причем не только последовательных. К шинам IEEE 1394 подключаются физические устройства, которые должны иметь по крайней мере один порт. К топологии сети относятся следующие понятия:

- *Сеть* — совокупность узлов, подключенных к одной шине или нескольким шинам, соединенным мостами. Все узлы сети имеют возможность взаимодействия друг с другом.

[1] В последующих ревизиях предполагается автоматическое исключение петель в «патологических» конфигурациях.

[2] Пакет, передаваемый на высокой скорости, не будет доставлен узлу, не поддерживающему эту скорость. Также доставке могут препятствовать низкоскоростные узлы, находящиеся по пути между источником и получателем пакета.

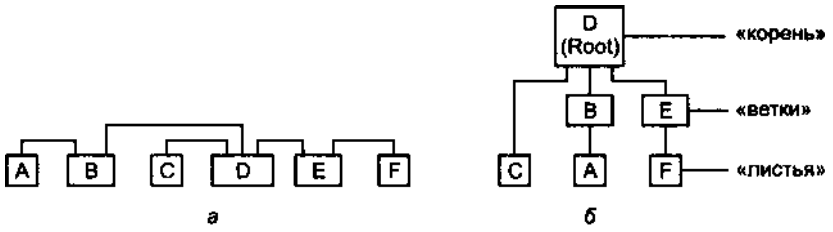


Рис. 18.1. Топология шины: а — физическая, *б* — логическая (для арбитража)

- *Шина* — совокупность узлов, связанных друг с другом кабельными сегментами. Все узлы шины используют ее общую (разделяемую) среду передачи, получая право на передачу путем арбитража. Подключение-отключение узлов вызывает реконфигурирование данной шины, в ходе которого узлы получают новые номера (физические адреса).
- *Модуль* (module) — физическое устройство («коробка»), подключаемое к шине и содержащее один или несколько узлов.
- *Узел* (node) — логический объект, имеющий уникальный физический адрес на шине. Узел имеет собственное адресное пространство, в котором обязательно присутствуют стандартные регистры управления и состояния (CSR), а также постоянная память (ROM) со стандартным набором структур, описывающих узел.
- *Блок* (unit) — часть узла, обеспечивающая его функциональность, в том числе работу памяти, ввод-вывод, обработку данных. Блоки имеют свои регистры и/или области памяти, отображенные на общее адресное пространство узла. Блоки функционируют относительно независимо и управляются своими драйверами.

Архитектура узла

В плане описания работы шины наибольший интерес представляет узел. Узел имеет явно выраженную трехуровневую структуру средств FireWire, к которой обращаются драйверы прикладного и системного ПО.

Физический уровень (Physical Layer, PHY) выполняет основные функции, связанные с подключением узла к шине:

- подключение узла к шине (механическое и электрическое);
- автоматическое конфигурирование шины и узла при инициализации;
- арбитраж при передаче данных;
- кодирование и декодирование сигналов состояния шины и потоков данных;
- предоставление сервисов канальному уровню;
- связывание сегментов сети в единую шину, если он имеет более одного порта IEEE 1394 (трансляцию сигналов между своими портами);
- предоставление питания по кабельной шине.

Физический уровень имеет несколько вариантов *физического интерфейса*:

- *кабельная шина 1394/1394a* с DS-кодированием, поддерживающая скорости S100, S200 и S400 на экранированной витой паре;
- *кросс-шина* (BackPlane Serial Bus, BPSB) с DS-кодированием, поддерживающая скорости S50 и S100 при связи узлов в пределах шасси;
- *кабельная шина 1394b* с кодированием 8B10B (так называемый бета-режим), поддерживающая скорости от S100 до S1600 и разные варианты кабелей, в том числе экранированную и неэкранированную (UTP-5) витую пару, пластиковое и стеклянное многомодовое оптоволокно.

Канальный уровень (Link Layer, LINK) из данных физического уровня формирует пакеты и выполняет обратные преобразования. При этом он задает (при передаче) и проверяет (при приеме) формат пакета и контрольные поля (CRC- коды). Канальный уровень обеспечивает асинхронный обмен узлов дейтаграммами (пакетами запросов, ответов и пакетами квитирования), а также передачу и прием изохронных потоков. Канальный уровень отвечает за адресацию — выявление и прием пакетов, предназначенных данному узлу.

Уровень транзакций (transaction layer) предоставляет приложениям сервисы для асинхронных обменов с регистрами и памятью любых узлов сети, состоящей из множества шин, объединенных мостами. Операции обменов включают *чтение*, *запись*, *блокированные операции* (чтение-модификация-запись). Операцией записи в специальный регистр узла можно вызвать *прерывание* для данного узла, при этом биты данного регистра будут нести информацию о соответствующих условиях прерывания. На уровне транзакций выполняется часть действий по обработке ошибок и организации повторных передач (канальный уровень только сообщает об обнаруженных ошибках).

Драйверы прикладного и системного ПО для организации асинхронных транзакций пользуются сервисами уровня транзакций. В плане обработки ошибок уровень транзакций предоставляет только уведомления об успехе или неудаче транзакции. В последнем случае организация повторов ложится на драйвер. Для изохронных передач (и потоковых асинхронных) драйвер пользуется сервисами канального уровня, который в данном случае обеспечивает лишь передачу пакетов, прием пакетов требуемых каналов с индикацией наличия или отсутствия ошибки в данных.

Управление шиной (bus management) затрагивает все вышеперечисленные уровни. Шина может иметь разную степень управляемости: полностью управляемая, частично управляемая (диспетчером изохронных ресурсов, необходимым, если есть узлы с изохронным обменом) и даже неуправляемая шина. Различные аспекты управления рассмотрены в 18.6.

Узел может быть вырожденным до простого кабельного концентратора — иметь только компоненты физического уровня. Тогда его многопортовый физический уровень будет выполнять функции повторителя, не нуждаясь в вышестоящих уровнях.

Интерфейс IEEE 1394 реализуется *аппаратно-программными средствами* устройства. Аппаратная часть FireWire обычно состоит из двух специализированных микросхем — трансивера физического уровня (*PHY Transceiver)* и моста

связи с микропроцессорной шиной *(LINK Chip)*. Интерфейс между ними стандартизован. Микросхема LINK выполняет все функции канального уровня и часть функций уровня транзакций; остальная часть уровня транзакций выполняется программно. Микросхема PHY выполняет сигнальное кодирование- декодирование данных, распознавание адресов, функции арбитража, а также трансляцию сигналов между своими портами. Уровень PHY достаточно автономен, все «общественно полезные» функции узла он может выполнять и при отключенных вышестоящих уровнях. Физический уровень может быть (но не обязательно) гальванически развязан с канальным уровнем. В бета-режиме (1394b) гальваническая развязка (более эффективная) возможна на уровне кабельного интерфейса. Гальваническая развязка необходима для предотвращения возникновения паразитных контуров общего провода, которые могут проявиться через провода защитного заземления блоков питания.

Физический и канальный уровни могут различаться в плане поддерживаемых скоростей передачи. Если многопортовый уровень PHY поддерживает более высокие скорости, чем LINK, то он способен транслировать высокоскоростные пакеты между своими портами. Однако скорость, на которой сам узел может общаться с остальными узлами шины, определяется самым слабым звеном в данной паре PHY-LINK. В этом случае она будет ограничиваться возможностями уровня LINK; эти возможности могут зависеть от организации узла. Для компьютера, подключаемого к 1394, поддерживаемая скорость LINK зависит от производительности шины, к которой подключен адаптер, и производительности его контролера памяти. Физический уровень для различных устройств практически одинаков, различия касаются поддерживаемых скоростей передачи, а в 1394b — и используемой среды передачи (разновидностей медных и оптических кабелей). Канальный уровень существенно зависит от прикладной части устройства — микропроцессора, на котором базируется устройство, и интерфейса подключения канального уровня.

Адресное пространство сети и узла

Каждому узлу выделяется адресное пространство размером 256 Тбайт (терабайт), которое является частью адресного пространства одной шины. Шин в системе может быть множество; все связанные шины с помощью *мостов* объединяются в общее адресное пространство размером 16 Эбайт (эксабайт — 2^{e4} байт). В IEEE 1394 используется 64-битный адрес, состоящий из номера шины, номера узла на данной шине и адреса внутри узла (рис. 18.2). Адресуемая единица — *квадлет* (32-битовое слово), адрес указывает на его старший байт[1]. Такой адрес используется для *асинхронных сообщений*, с помощью которых производятся обращения к регистрам, к памяти конфигурации, к областям памяти любого узла.

Адресное пространство узла делится на области, к части которых обращения со стороны других узлов выполняется чисто аппаратными средствами его контроллера шины, к другой части — с использованием программных средств.

[1] В IEEE 1394 принят порядок Big Endian (старший байт передается первым).

В адресное пространство узла, видимое со стороны шины, как правило, отображается часть пространства его системной памяти (ОЗУ). Единое (плоское) адресное пространство, объединяющее все узлы шины, позволяет узлам выполнять *прямой доступ к памяти* друг друга.

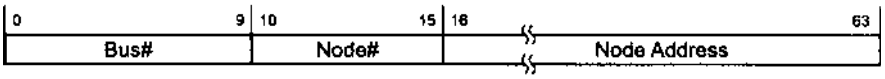


Рис. 18.2. Формат адреса IEEE 1394

18.3. Физический интерфейс

В первом варианте физического интерфейса (IEEE 1394 и 1394a) каждое кабельное соединение состоит из двух пар сигнальных электрических проводов и дополнительной пары проводов для подачи питания. Обе сигнальные пары используются для *двунаправленной* передачи сигналов, дифференциальных и линейных. По одной паре узел передает данные в последовательном коде, по другой — стробы. Этот режим получил название *DS-Mode* (DS означает Data-Strobe — данные-стробы), он обеспечивает простой механизм синхронизации приемника и передатчика при любой скорости обмена. Помимо передачи пакетов данных, *дифференциальные сигналы* используются для сброса, арбитража и конфигурирования. *Линейные сигналы* обеспечивают обнаружение фактов подключения-отключения устройств, сигнализацию скорости передачи данных и сигнализацию приостановки/возобновления работы (suspend/resume). Служебная сигнализация реализуется постоянным током, и для сигнализации скорости требуется различать несколько уровней напряжения. Из-за этого гальваническая развязка узлов на уровне кабельного интерфейса оказывается невозможной и ее при необходимости вводят в интерфейс PHY-LINK. Однако эта развязка спасает лишь от сигнальных помех, но не от высоких напряжений между схемными «землями» устройств.

В IEEE 1394b введен режим сигнализации *Beta-Mode*, в котором используются две встречные *однонаправленные* сигнальные линии. Примененный метод сигнального кодирования 8B/10B избавляет от постоянной составляющей сигнала; избыточность кодирования позволяет задействовать «лишние» символы для специальной сигнализации. Приемнику не требуется распознавать несколько уровней сигнала. Это позволяет использовать как электрическую, так и оптическую передачу, а для электрической передачи возможна полная гальваническая развязка приемников и передатчиков через импульсные трансформаторы. Порты 1394b могут быть универсальными «двуязычными» (bilingual), поддерживающими оба режима, или чисто бета-портами.

Кабели и коннекторы

Разъемы, используемые в кабельной шине IEEE 1394 (рис. 18.3), специально разработаны для обеспечения «горячего» подключения-отключения. Контакты цепей «земли» и питания (V6 и VP) длиннее других — они при подключении со

единяются раньше, а при отключении разъединяются позже сигнальных. На портах устанавливаются гнезда, на кабелях — вилки. Экранирующий кожух применяется как дополнительный контакт. Миниатюрные 9-контактные разъемы имеют две модификации:

- для бета-портов — с широким ключом на кожухе;
- для «двуязычных» портов — с узким ключом.

На вилках имеются соответствующие ключи; к бета-порту можно подключить кабель только с бета-вилкой, к «двуязычному» — с любой вилкой 1394b. Кабель с «двуязычной» вилкой может на противоположном конце иметь вилку любого типа (4-, 6- или 9-контактную).

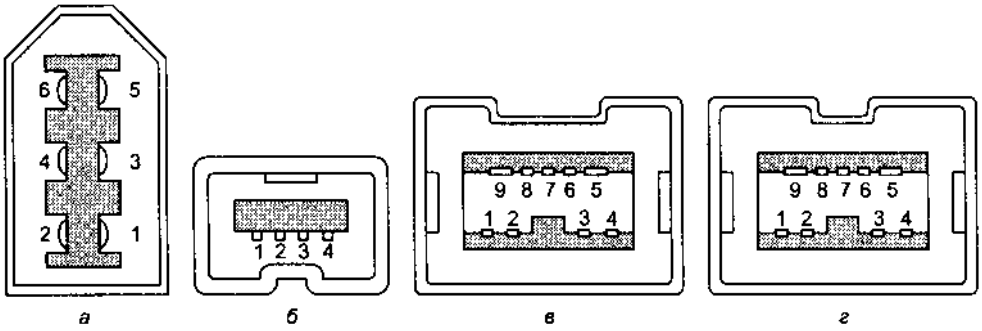


Рис. 18.3. Разъемы IEEE 1394 (размеры увеличены): а — 6-контактное гнездо, *б* — 4-контактное гнездо, *в* — 9-контактное гнездо бета-порта, г — 9-контактное гнездо «двуязычного» порта

Стандартный 6-проводный кабель 1394 содержит две экранированные витые пары для передачи сигналов (TPA и TPB) и дополнительно два провода для питания устройств. Все эти провода помещаются в общий экран. Сигнальные пары проводников соединяются перекрестно. При необходимости могут использоваться адаптеры-переходники с разъемами разных типов. Соединения контактов вилок кабелей иллюстрирует табл. 18.1. В IEEE 1394b для борьбы с помехами по земляным проводам предпринят ряд мер, ставших возможными с применением 9-контактного разъема.

Таблица 18.1. Соединительные кабели FireWire

Разъем А				Разъем Б			
4-конт.	6-конт.	9-конт.	Цепь	Цепь	9-конт.	6-конт.	4-конт.
–	1	8	VP	VP	8	1	–
Экран	2	6	VG	VG	6	2	Экран
1	3	1	TPB–	TPA–	3	5	3
2	4	2	TPB+	TPA+	4	6	4
3	5	3	TPA–	TPB–	1	3	1
4	6	4	TPA+	TPB+	2	4	2
–	–	5	TPA(R)	TPB(R)	9	–	–

продолжение

Таблица 18.2 (*продолжение)*

Разъем А				Разъем Б			
4-конт.	6-конт.	9-конт.	Цепь	Цепь	9-конт.	6-конт.	4-конт.
–	–	9	TPB(R)	TPA(R)	5	–	–
–	–	7	SC (не соединен)	SC (не соединен)	7	–	–
Экран	Экран	Экран	Экран	Экран	Экран	Экран	Экран

В IEEE 1394b для бета-режима введены новые варианты среды передачи и соответствующие типы разъемов (рис. 18.4), из которых для скоростей S200... S800 пригодны только STP и MMF:

- *UTP-5* (Unshielded Twisted Pair Cat 5) — неэкранированная витая пара категории 5 со стандартными коннекторами RJ-45, длина сегмента — до 100 м. Используются две пары проводов: контакты 1, 2 для TPB+ и TPB-; 7, 8 для TPA+ и TPA-. Можно применять как «прямые», так и перекрестные (crossover) кабели. Питание через кабель UTP-5 не предусматривается. Кабель UTP подключается через разделительные трансформаторы, обеспечивающие гальваническую развязку с напряжением выше 1 кВ.
- *POF* (Plastic Optical Fiber) и *HPCF* (Hard Polymer Clad Fiber) — оптоволоконные варианты с разъемами PN (дуплексные, диаметр стержня — 2,5 мм, шаг — 10,16 мм). Передатчики — светодиоды с длиной волны 650 нм:
  - *POF* — пара пластиковых волокон диаметром 1000 мкм со ступенчатым индексом преломления, это самый дешевый вариант оптической связи на малых расстояниях (до 50 м);
  - *HPCF* — пара твердых полимерных волокон диаметром 225 мкм с градиентным индексом преломления, длина — до 100 м.
- *MMF* (Multi Mode Fiber) — пара многомодовых стеклянных волокон 50/ 125 мкм, разъемы — малогабаритные дуплексные LC. Передатчики — коротковолновые лазерные диоды с длиной волны 830-860 нм, длина — до 100 м.
- *STP* (Shielded Twisted Pair) — экранированные витые пары с более высокими характеристиками передачи, чем в прежних версиях IEEE 1394. Кабели для бета-режимов (вплоть до S1600), обеспечивающие длину сегмента до **4,5** м, должны иметь на оболочке маркировку «1394b 2PR/25AWG 2C/ 22AWG». Более тонкие кабели обеспечивают длину до 2 м, они маркируются «1394b 2PR/30AWG 2C/26AWG».

Микросхемы PHY 1394b поддерживают любую среду передачи; соответствующие преобразования сигналов осуществляются в элементах PMD: RC-цепях для STP, трансформаторах для UTP и оптических трансиверах для POF, HPCF и MMF. Порт с электрическим интерфейсом в бета-режиме (для STP или UTP) несложно преобразовать в оптический. Для этого требуются оптический трансивер, поддерживающий требуемую скорость передачи, и несложные схемы преобразования уровней сигналов. При использовании трансиверов с низковольтным интерфейсом псевдо ЭСЛ (LV PECL) эти схемы содержат всего 8 резисторов, задающих смещение уровней сигналов трансиверов, и 4 разделительных конденсатора.

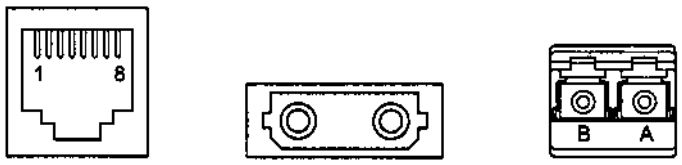


Рис. 18.4. Разъемы IEEE 1394b для бета-режимов: а — гнездо RJ-45 (для UTP), *б* — гнездо PN (для OF и HPCF), *в* — гнездо LC (для MMF)

Питание от шины

Для питания узлов постоянным током в кабелях IEEE 1394 предусмотрена отдельная пара проводов — V6 (общий провод, GND) и VP (положительный полюс питания) с напряжением 8-40 В при токе до 1,5 А. В 1394a диапазон напряжений сужен до 8-33 В. Узлы могут быть источниками, потребителями питания или не пользоваться питанием от шины; их отношение к питанию сообщается в пакете самоидентификации. В узлах-повторителях линии питания всех портов объединены. Узлы-источники питания подают напряжение на линию питания через ограничитель тока и диод, так что мощность, подаваемая от нескольких узлов, суммируется. При подключении узел сначала может потреблять от шины не более 1 Вт (1394a — 3 Вт), при этом обязательно должен быть включен уровень PHY. Уровни LINK и выше, как и прикладная часть устройства, могут потреблять дополнительную мощность. Однако эти уровни будут включаться только по команде от диспетчера шины или диспетчера изохронных ресурсов. Каждый узел контролирует питание от шины: при напряжении выше
7,5 В он устанавливает бит PS, сообщаемый в пакете самоидентификации. При падении уровня напряжения ниже 7,5 В физический уровень должен обнулить бит PS и уведомить об этом событии управляющее ПО.
В соответствии с классом питания возможны следующие конфигурации узла:

- *узел с автономным питанием* (self-power node) не питается от шины, но в зависимости от класса транслирует или не транслирует питание между портами;
- *поставщик питания* (power provider) питается самостоятельно и подает на шину питание с указанием минимальной мощности;
- *альтернативный поставщик питания* (alternate power provider) может питаться от шины, а также поставлять питание;
- *потребитель питания* (power consumer) — узел (только однопортовый), который питается от шины; до завершения конфигурирования мощность потребляет только PHY, а верхние уровни (LINK и выше), требующие дополнительной мощности, узел включает только по команде от диспетчера.

18.4. Конфигурирование

Конфигурирование шины происходит автоматически по включению питания, по подключению-отключению устройств, по инициативе какого-либо узла. Процесс конфигурирования выполняется только с участием уровней PHY —

вышестоящие компоненты могут быть отключены. Процесс состоит из трех последовательных этапов:

1. Сброс шины (bus reset) — приведение всех узлов в исходное (несконфигурированное) состояние.
2. Идентификация дерева — построение иерархической структуры шины.
3. Самоидентификация узлов — назначение физических адресов и сообщение ими своих свойств, относящихся к шине. На этом этапе также выбирается (не обязательно) узел-диспетчер изохронных ресурсов.

После этого шина переходит в состояние *Arbitration*, в котором она готова к выполнению асинхронных транзакций. Теперь может быть выбран диспетчер шины, который должен подать команду на включение канального уровня всех узлов (с учетом бюджета мощности питания от шины). При отсутствии диспетчера шины эту функцию исполняет диспетчер изохронных ресурсов.

С этого момента начинается регулярная работа в плане асинхронных транзакций. Для начала изохронного вещания узлы должны получить у диспетчера изохронные ресурсы — номера каналов и доступную полосу вещания.

Весь процесс конфигурирования занимает доли секунды. На время сброса и конфигурирования передача полезного трафика останавливается, все ожидающие транзакции сбрасываются. После конфигурирования могут измениться адреса узлов. Номера изохронных каналов, вещавших до сброса, могут и сохраниться — ранее вещавшие узлы имеют приоритет при выделении изохронных ресурсов, что позволяет минимизировать потери вещания.

Идентификация дерева

Изначально (по включению питания или сбросу) шина (совокупность соединенных узлов) имеет плоскую (неоформленную) структуру. В ходе инициализации первым делом автоматически (взаимодействием только уровней PHY) выполняется *идентификация дерева* (tree identification), в результате чего шина обретает форму перевернутого дерева (см. рис. 18.1). Ниже перечислены элементы дерева:

- *Корень* (root) является «верховным арбитром». Корень выбирается автоматически в зависимости от топологии шины; при необходимости стать корнем программно можно заставить любой узел.
- *Листья* (leaf) — конечные узлы, подключенные к шине только одним портом.
- *Ветки* (branch) — узлы, подключенные несколькими портами и находящиеся в иерархии между корнем и листьями.

И корень, и ветки являются промежуточными узлами, обеспечивающими трансляцию трафика. В первоначальной (плоской) структуре все порты узлов равноправны. После идентификации дерева порты дифференцируются в зависимости от их направленности в дереве:

- родительский порт (parent port), или p-порт, направлен в сторону корня (вверх);

♦ дочерний порт (child port), или с-порт, направлен в сторону листьев (вниз).

Родительским может быть только один из портов узла. Если на шине присутствует только два узла, то один из них (определяемый случайным образом) станет корнем с с-портом, другой — листом с р-портом. В процессе идентификации дерева участвуют только те порты, к которым подключены другие узлы (физический уровень способен определить состояние подключения).

Тип порта влияет только на распространение сигналов арбитража. Данные, принимаемые по одному порту, узел транслирует на все остальные порты, к которым подключены узлы-соседи, но с учетом скорости: высокоскоростные пакеты данных не транслируются на те порты, для которых установлена более низкая скорость работы. Данные, исходящие из узла, транслируются на все его порты, к которым подключены устройства и которые способны работать на скорости передачи пакета.

Самоидентификация узлов

После идентификации дерева выполняется *самоидентификация* (self identification) *узлов*, в результате которой все узлы шины получают уникальные адреса — физические идентификаторы (`phy_ID`). В процессе самоидентификации с помощью арбитража каждый узел последовательно получает право на передачу широковещательного пакета самоидентификации. В этом пакете содержатся идентификатор узла, число его портов, их состояние и назначение, скоростные возможности устройства и некоторые другие параметры. Устройство назначает себе идентификатор само, исходя из числа ранее «услышанных» им пакетов самоидентификации от других устройств. Первое устройство, которому предоставили право на передачу, получает `phy_ID = 0`; последним самоидентифицируется корень — у него будет максимальное значение `phy_ID` (не более 62). После посылки пакета самоидентификации узел обменивается со своим партнером-«родителем» сигналами, идентифицирующими максимальную скорость работы. Это позволяет непосредственным партнерам попарно согласовать свои скоростные возможности.

По информации, собранной из всех пакетов самоидентификацйи, любой узел может построить *карту топологии* (topology map) сети, а на ее основе и *карту* достижимых скоростей (speed map) обмена между любой парой узлов. Эти карты должен строить и публиковать узел-диспетчер (менеджер) шины, предоставляя доступ к ним любым узлам. Кроме того, информация пакетов самоидентификации используется для управления питанием от шины и для оптимизации трафика.

18.2. Передача данных

Шина IEEE 1394 поддерживает два типа передач данных:

♦ Асинхронные передачи без каких-либо требований к скорости и задержке доставки. Целостность данных контролируется CRC-кодом, гарантированную

доставку обеспечивает механизм квитирования и повторов. Асинхронная передача может быть направленной и широковещательной:

- *направленная асинхронная передача* адресуется конкретному узлу и выполняется с квитированием, что позволяет организовать гарантированную надежную доставку, причем протокол шины дает возможность узлам с помощью асинхронных транзакций обращаться к памяти (регистрам) друг друга в режиме прямого доступа (DMA), и поэтому узлы не нуждаются в памяти и процессорных ресурсах «третьих лиц»[1];
- *широковещательная асинхронная передача* адресуется всем узлам и выполняется без квитирования (и без гарантии доставки).

♦ Изохронные передачи с гарантированной пропускной способностью, но без обеспечения надежной доставки. Изохронные передачи ведутся широковещательно и адресуются через *номер канала*, передаваемый в каждом пакете. На шине может быть организовано до 64 изохронных каналов, передачи всех каналов «слышат» все устройства шины, но из всех пакетов принимают только данные интересующих их каналов. Целостность данных контролируется CRC-кодом, но квитирование и повторы не применяются.

Если на шине используются изохронные передачи, то все транзакции организуются в последовательность *циклов* — интервалов времени с номинальной длительностью 125 мкс. Начало каждого цикла отмечается широковещательным *пакетом начала цикла* (cycle start). Эти пакеты посылает узел, являющийся *мастером циклов* (как правило, это корневой узел). Право на передачу этого пакета мастер получает через арбитраж, используя свой высокий приоритет (см. далее). После пакета начала цикла каждый узел, которому выделен изохронный канал, имеет право передать по одному пакету (до прихода следующего пакета начала цикла). Для изохронных передач используется короткий зазор арбитража, так что асинхронные транзакции в изохронную часть цикла вклиниться не могут. После того как иссякнут изохронные пакеты данного цикла, выполняются асинхронные передачи, у которых имеют место более длинные зазоры арбитража.

Арбитраж

Арбитраж определяет, какому из узлов, запрашивающих передачу, предоставляется это право. Арбитраж обеспечивает гарантированную пропускную способность для изохронных передач и справедливое предоставление доступа узлам для асинхронных транзакций. Арбитраж на шине IEEE 1394 выполняется перед посылкой любого[2] пакета запроса (синхронного или изохронного) или ответа. Пакеты квитирования посылаются без арбитража — право на их передачу разыгрывать не надо, поскольку квитанцию посылает только тот единственный узел, которому адресовался подтверждаемый пакет запроса или ответа.

[1] Для сравнения: взаимодействие возможно между устройствами USB только через память хост-компьютера и лишь под управлением его процессора.

[2] Исключением является соединенная (concatenated) форма выполнения транзакций.

Арбитражем занимается физический уровень каждого узла шины. Арбитраж в отношении иерархии выполняется децентрализованно: им занимаются все узлы, «верховным» арбитром является корневой узел, автоматически выбираемый на этапе конфигурирования шины.

Физический уровень предоставляет канальному уровню следующие *сервисы арбитража*, перечисленные в порядке нарастания приоритета:

- *справедливый арбитраж* (fair arbitration service) — передача обычных асинхронных пакетов;
- *приоритетный арбитраж* (priority arbitration service) — передача пакетов начала цикла и приоритетных асинхронных пакетов;
- *немедленный арбитраж* (immediate arbitration service) — передача пакетов квитирования;
- *изохронный арбитраж* (isochronous arbitration service) — передача изохронных пакетов.

Приоритет в арбитраже на шине IEEE 1394 определяется длительностью *зазора арбитража* (arbitration gap) — временем, в течение которого узел наблюдает покой шины перед началом передачи запроса арбитража. Чем меньше этот зазор, тем больше шансов у узла получить право на передачу. Исходная схема арбитража 1394 усовершенствовалась дважды: в 1394a были введены механизмы ускоренного арбитража, а в стандарте 1394b с его дуплексными соединениями был введен новый механизм — BOSS-арбитраж. Все усовершенствования направлены на снижение непродуктивных затрат времени.

Организация циклов представлена на рис. 18.5, где изображена работа двух изохронных каналов (Ch#J и Ch#K) и передача асинхронных пакетов A и B. После пакета начала цикла каждый узел, которому выделены изохронные каналы, имеет право передать по одному пакету для каждого канала (до прихода следующего пакета начала цикла). Для изохронных передач используется короткий зазор арбитража, так что асинхронные транзакции с более длинным зазором в изохронную часть цикла вклиниться не могут. После того как иссякнут изохронные пакеты данного цикла, выполняются асинхронные передачи с более длинными зазорами арбитража. Когда наступает пора посылки следующего пакета начала цикла, мастер цикла, дождавшись освобождения шины, снова получает право доступа (пользуясь своим приоритетом, обусловленным его положением в корне дерева) и посылает следующий пакет начала цикла. Таким образом, длительность цикла может отклоняться от номинального значения 125 мкс. Отклонения длительности цикла от номинального не страшны, поскольку пакет начала цикла несет значение системного времени точно на момент фактической передачи этого пакета.

Если на шине не используются изохронные передачи, то мастер циклов может отсутствовать и пакетов начала цикла на шине не будет. В этом случае все время на шине может заполняться асинхронными передачами с их длинными зазорами арбитража.

Справедливый арбитраж для асинхронных передач предохраняет шину и ее узлы от возможных перегрузок. Для этого на шине устанавливается *интервал*

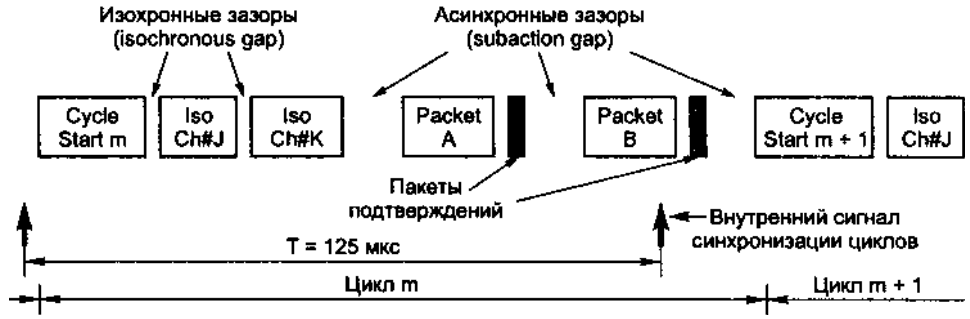


Рис. 18.5. Организация циклов

справедливости (fairness interval) — период времени, в течение которого узел имеет право послать лишь один асинхронный пакет запроса или ответа. Длительность интервала справедливости зависит от числа активных узлов (у которых включен канальный уровень) и от загруженности шины изохронным трафиком. Справедливость обеспечивает физический уровень, у которого имеется специальный *бит разрешения арбитража*. Физический уровень посылает запрос арбитража только при установленном бите разрешения. Этот бит изначально устанавливается после самоидентификации устройств. Далее, как только физический уровень (по запросу от канального) посылает запрос арбитража, бит сбрасывается. Повторно он установится только после того, как физический уровень будет видеть шину в покое в течение интервала зазора сброса арбитража (по умолчанию — 21 мкс). Этот интервал гораздо больше изохронного и асинхронного зазоров; такое длительное «молчание» всех устройств означает, что у них больше нет асинхронных пакетов, разрешенных к передаче в этом интервале справедливости. Таким образом, интервал справедливости — это время между соседними зазорами сброса арбитража, и его длительность заранее не определена.

Организация потоковых передач и изохронный обмен

Потоковые передачи могут быть изохронными и асинхронными. В обоих случаях используются пакеты одного и того же формата, но есть разница в организации:

- Изохронные передачи выполняются в специально выделенном периоде цикла, право на их передачу получается с коротким зазором арбитража. Для использования изохронных передач узел должен получить у диспетчера изохронных ресурсов номер канала и выделенную полосу пропускания шины (максимальное время передачи пакета).
- Асинхронные передачи выполняются в оставшееся время цикла. Для них у диспетчера изохронных ресурсов запрашивается только номер канала. Полоса пропускания асинхронного потокового канала не нормируется.

Получение номера канала и выделение полосы пропускания осуществляются обращениями узла к регистрам CHANNEL_AVAILABLE и BANDWIDTH_AVAILABLE диспетчера изохронных ресурсов. Если канал и полосу получить не удалось, узел может периодически повторять запросы. Когда изохронный обмен становится не нужным узлу, он должен освободить свою полосу и номер канала, чтобы этими ресурсами смогли воспользоваться другие устройства. Обмен управляющей информацией с диспетчером производится асинхронными транзакциями.

18.6. Управление

Шина IEEE 1394, обеспечивая равноранговые взаимодействия между узлами, нуждается в централизованном управлении некоторыми функциями. Управляющие функции могут брать на себя разные узлы шины; в зависимости от реализации тех или иных функций различают следующие варианты шины IEEE 1394:

- *Неуправляемая шина* должна иметь только корень для управления арбитражем. Корень, который становится «верховным арбитром», определяется на этапе идентификации дерева. Первоначальный кандидат на эту «должность» выбирается, исходя из топологии соединений, с возможным случайным розыгрышем этого права между двумя победителями предпоследнего тура. После завершения выборов корня производится самоидентификация (и назначение физических адресов) узлов, после чего шина становится готовой к асинхронным транзакциям между узлами. Впоследствии программным путем (через асинхронные сообщения по шине) возможно переназначение корня (с определением новой структуры дерева и адресов узлов).
- *Частично управляемая шина* в дополнение к корню должна иметь узлы, исполняющие роли мастера циклов и диспетчера изохронных ресурсов. Их работа обеспечивает возможность использования шины для изохронных передач.
- *Полностью управляемая шина* должна иметь узел-диспетчер шины, обеспечивающий дополнительные сервисы управления.

Диспетчер шины (bus manager) обеспечивает полное управление шиной. Им может стать любой узел, способный (и обязанный после избрания диспетчером) выполнять следующие функции:

- назначение (при необходимости) мастера циклов — если текущий корень не исполняет эту роль, то диспетчер шины выбирает на роль корня иной узел;
- управление питанием — подсчет баланса поставки и потребления мощности и разрешение работы узлов с учетом баланса (если диспетчера нет, то все узлы включит диспетчер изохронных ресурсов, но без учета баланса);
- построение и публикация карты топологии, вычисление и задание оптимального зазора арбитража для всех узлов (с учетом топологии);
- публикация карты скоростей, указывающей максимально возможную скорость передачи пакета между каждой парой узлов;

- оптимизация трафика шины (изменение полосы, выделяемой для изохронных передач).

Диспетчер шины может находиться в любом месте шины. Он выбирается из узлов-кандидатов на роль диспетчера изохронных ресурсов.

18.7. Применение

Шина IEEE 1394 широко применяется в устройствах различных классов, как связанных с компьютерами или являющихся их периферией, так и самостоятельных, работающих без подключения к компьютерам. Принципиальным преимуществом шины 1394 является ее идеология равноранговых взаимоотношений узлов и отсутствие необходимости в централизованном контроллере. Широкий круг применений и разные подходы к идеологии построения систем с использованием последовательной шины породили (и продолжают порождать) множество стандартов и спецификаций, часть из которых упоминается и бегло рассматривается в данной главе.

Шина 1394 в компьютерах

Шина IEEE 1394 (FireWire) является «родным» интерфейсом компьютеров Macintosh фирмы Apple. В PC-совместимых компьютерах поддержка IEEE 1394 была провозглашена в спецификации PC'99 (Personal Computer Design Guidelines 1999, http://www.microsoft.com/hwdev/desguid.htm). В современных версиях ОС поддерживаются спецификации OHCI, SBP-2, IEC61883 и др.

Порты шины IEEE 1394 выводятся на внешние разъемы системного блока, что позволяет подключать внешние устройства с помощью кабелей. Для подключения съемных устройств с интерфейсом IEEE 1394 (и USB 2.0), устанавливаемых пользователем в отсеки корпуса ПК, имеются отдельные спецификации (http://www.device-bay.org/):

- Device Bay — для съемных периферийных устройств, устанавливаемых в 5- или 3,5-дюймовые отсеки системного блока;
- CardBay — для съемных периферийных устройств, устанавливаемых в слоты конструктива PC Card блокнотных ПК (возможна совместимость с CardBus).

В IBM PC-совместимых компьютерах контроллеры IEEE 1394 чаще всего встречаются в виде карт расширения, но они уже встраиваются и в некоторые модели системных плат. Современные блокнотные ПК, как правило, имеют встроенный контроллер 1394.

Контроллер IEEE 1394 для PC, как правило, является устройством PCI, поскольку лишь шина PCI (и ее современные «родственники») способна обеспечить производительность обмена периферийного устройства с памятью, достойную шины FireWire. Для контроллера IEEE 1394 существует стандартная спецификация OHCI. Для работы с шиной используется протокол *SBP-2* (Serial Bus Protocol-2), широко применяемый Microsoft и Apple. В протоколе задействован механизм DMA, что позволяет снизить необходимое число преры

ваний центрального процессора. Часть области системного ОЗУ ПК отображается на адресное пространство узла 1394. В нем формируется связанный список команд и указателей на буферы данных. Адрес этого списка передается устройству, и оно исполняет требуемые команды и обмены данными, самостоятельно обращаясь к памяти ПК транзакциями IEEE 1394.

Шина 1394 для устройств хранения данных

Шину IEEE 1394 можно использовать как транспортное средство передачи пакетов команд и данных SCSI. Спецификация RBC (Reduced Block Commands) описывает сокращенный набор команд SCSI, относящихся к устройствам хранения, с использованием протокола SBP-2 (ftp.symbios.com/pub/standards/io/x3t10/ drafts/rbc/). Эта спецификация применяется для устройств хранения в ОС Windows, Mac OS и др.
Для дисковых устройств хранения (http://www.1394ta.org) имеются общая спецификация (disk general specification) и следующие расширения:

- AV/C MiniDisk subunit — расширение для мини-дисков;
- AV/C hard disk drive subunit — расширение для винчестеров;
- AV/C compact disk subunit — расширение для компакт-дисков;
- AV/C storage object descriptor subunit — абстрактное описание класса устройств хранения.

Шина 1394 для передачи и печати изображений

Для печати (и передачи) неподвижных изображений по IEEE 1394 есть несколько разных подходов со своими протоколами:

- 1394 Printer Working Group (PWG) Imaging Device Communication Specification — спецификация на использование транспортного протокола SBP-2 для принтеров, сканеров и прочих устройств ввода-вывода изображений, подключаемых к компьютерам. Из-за отсутствия единых спецификаций на форматы данных и команд для каждой модели принтера требуется свой драйвер.
- Протокол DPP (Direct Printing Protocol) предназначен для непосредственного соединения устройств ввода и вывода изображений — сканеров, принтеров, фотокамер. Протокол обеспечивает равноранговое соединение устройств и передачу масштабируемых изображений (http://www.1394ta.org/Technology/ Specifications/, http://www2.tokyoweb.or.jp/ieeel394pwgc/).

Шина 1394 для аудио- и видеоустройств

Интерфейс IEEE 1394 является общепринятым для современных цифровых устройств профессиональной и бытовой аудио- и видеотехники, которые используют эту шину и без участия компьютера. Помимо цифровых устройств, имеющих встроенные адаптеры 1394, допускается подключение к шине FireWire

и традиционных аналоговых и цифровых устройств (плееры, камеры, мониторы) через адаптеры-преобразователи интерфейсов и сигналов.

Для шины 1394 привлекательна возможность соединения устройств бытовой электроники в «домашнюю сеть», причем как с включением в нее PC, так и без. При этом стандартные однотипные кабели и разъемы 1394 заменяют множество разнородных соединений устройств бытовой электроники с PC. Разнотипные цифровые сигналы (сжатые видеосигналы, цифровые аудиосигналы, команды MIDI и управления устройствами, данные) мультиплексируются в одну шину, проходящую по всем помещениям. Используя одни и те же источники данных (приемники вещания, устройства хранения, видеокамеры и т. п.), можно одновременно в разных местах просматривать (прослушивать) разные программы с высоким качеством, обеспечиваемым цифровыми технологиями. Применение компьютера с адаптером 1394 и соответствующим ПО значительно расширяет возможности этой сети. Компьютер становится виртуальным коммутатором домашней аудиовидеостудии. Приложения для аудио- и видеоустройств используют логические «вилки» (plugs) и «розетки» (sockets), которые являются аналогами разъемов, применяемых в обычной аппаратуре. Вилки соответствуют выходам, розетки — входам соответствующих устройств. «Вставляя» эти «вилки» в «розетки», можно собрать требуемую систему.

Для аудио- и видеоустройств существует ряд спецификаций:

- DVC Blue Book — первый стандарт на аппаратные средства и протоколы, используемые в цифровых видеоустройствах, выпускаемых консорциумом Digital Video Consortium Devices. Этот стандарт должен учитываться при разработке ПО для совместимости со старыми устройствами.
- 1394 Digital Video Conferencing Camera Specification — определение функций, сервисов и набора регистров камер для видеоконференций без использования компрессии (http://www.1394ta.org/Technology/Specifications/).
- *HAVi* (Home AV Interoperability) — протоколы, обеспечивающие взаимодействие бытовой аудиовидеотехники с поддержкой PnP, как с ПК, так и без. Спецификации разработаны объединенными усилиями производителей бытовой техники Grundig, Hitachi, Matsushita, Philips, Sharp, Sony, Thomson и Toshiba (http://www.havi.org). Спецификации совместимы с современными устройствами, соответствующими спецификациям AV/C (см. далее), и ориентированы на следующее поколение бытовой техники и ее инфраструктуры. Набор команд для HAVi определен спецификацией HAVi CTS (Command Transaction Set).
- IEC 61883 — управление аудио- и видеоустройствами.
- AV/C (Audio/Video Compatibility) — обеспечение совместимости аудио- и видеоустройств с шиной IEEE 1394.
- Music LAN (MLAN) — спецификации для цифровых музыкальных инструментов (http://www.yamaha.co.jp/tech/1394mLAN/mlan.html).

Шина IEEE 1394 фигурирует в спецификациях OpenCable (http://www. opencable. com), описывающих интегрированные сервисы, предоставляемые с помощью сетей кабельного телевидения (в США): телевещание (аналоговое и цифровое),

мультимедийные сервисы (голосовая и видеосвязь), интерактивные приложения и т. п. В этой сети у пользователя устанавливается устройство STB (set-top box), подключаемое к TV-кабелю, декодирующее сигналы и обеспечивающее предоставление пользователю оплаченных услуг. STB имеет внешние интерфейсы для подключения аудио- и видеотехники: видеоинтерфейсы (композитный сигнал на RCA-разъемах, S-видео), аудиоинтерфейсы (аналоговые, SP/ DIF). IEEE 1394 является цифровым интерфейсом STB-устройств, через который подключаются цифровые аудио- и видеоустройства, DVD-плееры и устройства хранения данных.

Защита передаваемой информации

Одной из проблем цифровой передачи мультимедийной информации является защита авторских прав. Пользователь должен иметь возможность высококачественного воспроизведения принимаемых программ или приобретенных дисков, а их авторы (производители), в свою очередь, должны иметь возможность защитить свои права, по своему усмотрению вводя ограничения на цифровое копирование. К этой стороне шины относятся следующие спецификации:

- «5C» — механизм защиты информации при цифровой передаче (digital transmission copy protection mechanism), разработанный объединением 5 компаний: Sony, Matsushita, Intel, Hitachi и Toshiba. Механизм включает в себя методы шифрования данных и управления ключами. Технология защищена патентами, информация по которым предоставляется только легальным производителям оборудования. Реализация технологии не требует слишком сложной логики и больших вычислительных ресурсов. Лицензиями управляет организация 5C Digital Transmission Licensing Authority (http://www. dtcp.com).
- XCA Digital Transmission Copy Protection Mechanism — альтернативный вариант защиты, предложенный Zenith/Thomson, с использованием карт Smart Media в качестве ключевых элементов.
- OCPS (Open Copy Protection system) — открытая система защиты с использованием шифрования DES

К защите информации относятся также системы NDS, MRJ и др. В спецификациях AV/C есть субблок условного доступа (conditional access sub-unit), обеспечивающий дешифрование скремблированных видеоаудиопотоков (http://www. 1394ta.org/).

18.8. Открытый хост-контроллер

Открытый хост-контроллер (Open Host Controller, OHC) шины 1394 представляет собой реализацию канального уровня (LINK) шины IEEE 1394 с дополнительными средствами поддержки уровней транзакций и управления шиной. Для высокопроизводительного обмена данными OHC содержит контроллеры прямого доступа к памяти (DMA). Контроллер поддерживает все типы пакетов, передаваемых по шине 1394. Спецификация 1394 Open Host Controller

Interface Specification доступна в Сети (ftp://ftp.austin.ibm.com/pub/chrptech/ 1394ohci).
Со стороны шины 1394 *хост* — узел с контроллером OHC — выглядит как обычный узел шины, способный выполнять функции мастера циклов и диспетчера изохронных ресурсов. Контроллер позволяет хосту быть инициатором любых транзакций шины 1394 и отвечать на любые транзакции, адресованные узлу хоста. В адресном пространстве этого узла расположены архитектурные регистры CSR и память конфигурации; большая часть пространства доступна для обращений в виде обычных транзакций шины. Часть пространства узла может отображать пространство физических адресов памяти хоста. Часть обращений к хосту может отрабатываться исключительно аппаратными средствами контроллера; остальные обращения хост отрабатывает программно. Принимаемые пакеты запросов для программной отработки OHC своими каналами DMA помещает в буферы, размещенные в памяти хоста. Пакеты ответов программа размещает в других буферах, из которых OHC организует их передачу в шину, опять же с помощью каналов DMA.
Для *асинхронных транзакций* контроллер обеспечивает чтение пакетов из системной памяти хоста и их передачу в шину; пакеты, принимаемые из шины, контролер записывает в системную память. Обмен производится с помощью каналов DMA. Контроллер может функционировать и как шинный мост, аппаратно отрабатывая запросы транзакций чтения и записи шины 1394 как обращения к пространству памяти хоста.
Для *изохронных операций* OHC может исполнять роль мастера циклов, синхронизируясь от внутреннего генератора синхронизации или (не обязательно) от внешнего источника. Если OHC не исполняет роль мастера циклов, то он поддерживает синхронизацию внутреннего таймера циклов с таймером узла-мастера циклов (по приеме пакетов начала цикла). Для изохронных операций OHC имеет два контроллера DMA — для приема и для передачи данных. Каждый из этих контроллеров может поддерживать до 32 каналов DMA, называемых *контекстами DMA*. Передающий контроллер в каждом цикле может передавать данные из каждого контекста для связанного с ним изохронного канала. Принимающий контроллер способен в каждом цикле принимать данные в каждый контекст из связанного с ним канала. Кроме того, один из принимающих контекстов может быть настроен на прием данных из множества каналов.
По обнаружении *сброса на шине* OHC автоматически очищает все очереди асинхронных пакетов для передачи; прием пакетов не прерывается, но в потоке пакетов запросов появляется маркер, индицирующий факт сброса. Новый физический идентификатор узла (PHY ID), получаемый от PHY, контроллер записывает в соответствующий регистр. Контроллер возобновляет асинхронные передачи только по указанию программы, при этом повторное использование старых запросов в общем случае невозможно: физический идентификатор узла назначения может измениться. Изохронный прием и передача по сбросу не прекращаются — они возобновляются сразу по завершении инициализации.

Устройство контроллера OHC

Упрощенная структурная схема OHC приведена на рис. 18.6.

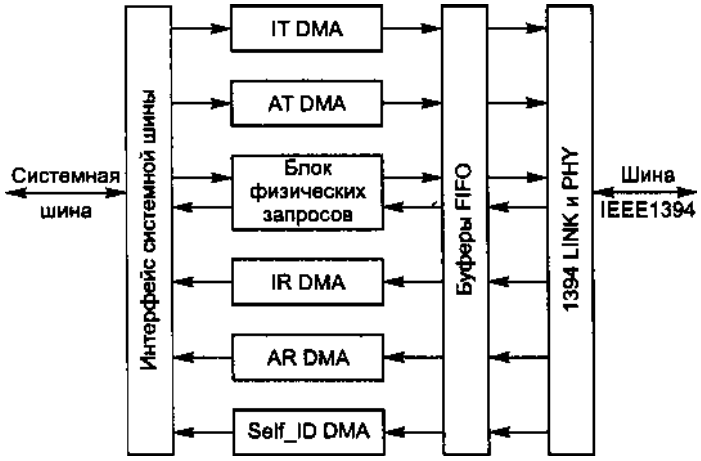


Рис. 18.6. Структура контроллера OHC 1394

Интерфейс системной шины (host bus interface) обеспечивает взаимодействие с контроллером в двух режимах:

- ведомый режим (PCI target) обеспечивает программный доступ к регистрам контроллера со стороны центрального процессора хоста;
- ведущий режим (PCI bus master) обеспечивает контроллеру возможность прямого доступа к системной памяти хоста — в этом режиме интерфейс системной шины должен выдерживать поток данных по крайней мере базовой скорости S100 (100 Мбит/с) с накладными расходами на организацию прямого доступа к памяти.

Контроллеры DMA обеспечивают обмен данными между шиной и системной памятью. В OHC имеются семь типов контроллеров DMA:

- Контроллер асинхронной передачи (AT DMA).
- Контроллер асинхронного приема (AR DMA).
- В блок физических запросов входят два контроллера:
  - контроллер приема аппаратнообрабатываемых запросов (physical receive);
  - контроллер ответов на аппаратнообрабатываемые запросы (physical response).
- Контроллер изохронной передачи (IT DMA).
- Контроллер изохронного приема (IR DMA).
- Контроллер приема пакетов самоидентификации (Self_ID DMA).

Каждый контроллер работает со своими *контекстами* — наборами регистров, управляющих работой канала и выборкой запросов из списков, расположенных в системной памяти. Контроллеры асинхронной передачи и приема имеют отдельные контексты для запросов ответов шинных транзакций. Контроллеры изохронного приема и передачи могут иметь до 32 контекстов каждый.

Канальный уровень OHC передает пакеты из FIFO-буферов передающих каналов и отдает в FIFO принятые пакеты с корректным адресом, предназначенные данному узлу. Уровень выполняет следующие действия:

- передает и принимает пакеты форматов IEEE 1394;
- генерирует соответствующие пакеты квитирования для принятых асинхронных пакетов, отрабатывая однофазный или двухфазный протокол повторов [7];
- выполняет функции мастера циклов;
- генерирует и проверяет корректность 32-битных CRC-кодов;
- обнаруживает пропуски пакетов начала цикла;
- взаимодействует с регистрами PHY;
- принимает изохронные пакеты (все время);
- игнорирует асинхронные пакеты во время изохронной фазы цикла.

Буферы FIFO, находящиеся между каналами DMA и канальным уровнем, выполняют промежуточную буферизацию данных, считываемых из системной памяти для передачи в шину и принимаемых из шины для записи в память. Буферы FIFO обеспечивают и выравнивание данных, побайтное для хоста и поквадлетное для шины 1394. При необходимости буферы FIFO вставляют байты-заполнители, выравнивающие данные до границ квадлетов. Переполнение (overflow) или переопустошение (underrun) буферов (по вине интерфейса системной шины и памяти), приводящее к потерям принимаемых или передаваемых пакетов, контролируется аппаратными средствами OHC.

Буферы могут «на лету» выполнять преобразование форматов представления квадлетов. Шина IEEE 1394 и, соответственно, канальный уровень работают с квадлетами, представленными в формате Big Endian (старший байт передается первым). Передача данных через хост-интерфейс может выполняться по выбору:

- в формате Big Endian, используемом на платформах фирмы Apple;
- в формате Little Endian (младший байт передается первым), используемом на платформах фирмы Intel.

Для поддержки функций управления OHC имеет 64-6итный *регистр глобально уникального идентификатора* (GUID, он же IEEE EUI-64), автоматически загружаемый из энергонезависимой памяти по сбросу контроллера (или однократно программируемый в самом контроллере).

Для выполнения функций диспетчера изохронных ресурсов контроллер имеет 4 *автономных регистра*, реализующих блокированные операции (*compare_ swap*) как со стороны шины, так и со стороны хоста.

Взаимодействие хоста и ОНС

Взаимодействие хоста и ОНС производится несколькими способами:

- Программные обращения процессора хоста к регистрам контроллера обеспечивают общее управление контроллером и узлом хоста, управление контроллерами прямого доступа и их контекстами, управление прерываниями и их идентификацию, доступ к автономным регистрам.
- При прямом доступе к памяти хоста (DMA) различают доступ по контекстным программам и доступ физических обращений к памяти:
  - *DMA по контекстным программам.* Структуры данных, расположенные в памяти хоста, описывают списки буферов данных. Контроллер автоматически последовательно обходит эти списки, передавая данные и обновляя в этих структурах информацию о состоянии выполнения. Каждый контекст DMA имеет набор регистров, через которые программа центрального процессора управляет контекстом;
  - *DMA физических обращений к памяти.* Контроллер аппаратно преобразует транзакции чтения и записи определенного диапазона адресов узла 1394 в транзакции чтения и записи определенного диапазона физической памяти. Для программы хоста эти обращения невидимы и при нормальной работе прерываний не вызывают.
- Прерывания центрального процессора хоста вырабатываются контроллерам по различным событиям: исполнению (или отказу) передачи или приема, завершению обработки контекстной программы, приему потока пакетов самоидентификации и т. п. Контроллер имеет регистры идентификации и маскирования событий, вызывающих прерывания.

18.9. Протокол SBP-2

Протокол SBP-2 описывает транспортировку команд, данных и информации состояния между устройствами, подключенными к шине IEEE 1394, с использованием расщепленных транзакций. Протокол соответствует требованиям *архитектурной модели SAM* (SCSI-3 Architecture Model), что позволяет реализовать шину SCSI в последовательном варианте. В протоколе имеются расширения (относительно SAM), обеспечивающие управление изохронными соединениями и передачу изохронных данных. Протокол описан в документе «Information technology — Serial Bus Protocol 2» (SBP-2) комитета X3T10. Здесь приводится только краткое изложение концепции SBP-2, подробно протокол и его использование описаны в [7].

При разработке SBP-2 преследовался ряд целей:

- обеспечить возможность инкапсуляции команд, данных и информации состояния выполнения для разнообразных наборов команд, как старых (определенных в SCSI), так и новых;

- предоставить инициатору возможность формирования большого набора выполняемых заданий без оглядки на ограничения, присущие целевым устройствам;
- предоставить инициатору возможность динамического добавления новых заданий целевому устройству без побочных влияний на исполнение ранее выданных заданий;
- реализовать поддержку различных уровней производительности и функциональности целевых устройств без влияния этих уровней на аппаратные и программные средства инициатора;
- использовать преимущества последовательной шины (функциональность, возможность изохронных обменов);
- обеспечить масштабируемость общей производительности системы; для этого протокол должен распределять контекст DMA от инициатора к целевым устройствам (при таком подходе способность обслуживания большого количества целевых устройств не будет зависеть от возможностей аппаратно-программных средств инициатора, то есть компьютера).

Организация взаимодействия устройств

Протокол SBP-2 определяет процедуры взаимодействия инициатора с целевыми устройствами. *Инициатор* (*initiator*) — устройство, заинтересованное в выполнении каких-либо действий целевым устройством и формирующее запросы на выполнение этих действий.

Целевое устройство (target) способно выполнять адресованные ему *команды*. Примером команд может быть чтение или запись данных устройством хранения, воспроизведение аудиотрека и т. п. Каждое целевое устройство имеет определенный набор поддерживаемых команд (систему команд).

Задание (task) — понятие, определяющее действия целевого устройства, связанные с выполнением команды. Для выполнения задания целевое устройство оперирует *контекстом задания*, в который входят, например, адреса и длина передаваемых данных, статус выполнения, отношения с другими заданиями. Время жизни задания начинается с момента, когда целевому устройству сигнализировано данное задание, и завершается, когда целевое устройство сигнализирует инициатору статус завершения. За время своей жизни задание использует ресурсы как инициатора, так и целевого устройства.

Для выполнения каждого задания инициатор формирует *запрос* в виде структуры данных в памяти узла шины. Запросы могут быть связаны в цепочки, каждая цепочка образует *набор заданий* (task set).

Целевое устройство само выбирает задания из памяти и выполняет их; при определенных условиях оно имеет право изменять порядок выполнения заданий в пределах набора. За *передачу данных*, необходимых для выполнения заданий, *отвечает целевое устройство*. Это как раз и обеспечивает масштабируемость системы: добавление новых периферийных устройств не будет вызывать перегрузку процессора инициатора (хост-компьютера). Состояние выполнения за

дания (успешное или с ошибкой) целевое устройство сообщает записью по адресу, указанному инициатором в запросе. Инициатор может динамически добавлять задания, а также управлять выполнением заданий (принудительно завершать).

Структура целевого устройства

Протокол SBP-2 используется для работы с целевыми устройствами, подключенными к шине IEEE 1394, то есть входящими в состав узла шины (см. 18.2). Блок узла содержит одно или несколько *логических устройств* (logical unit)[1], каждое из которых представляет модель устройства (например, устройство хранения, принтер и т. п.). Каждое логическое устройство имеет идентификатор LUN (Logical Unit Number — номер логического устройства). С каждым логическим устройством должен быть связан *сервер устройства* (device server), отвечающий за выполнение команд, посланных устройству. Потоковое устройство должно иметь еще и один или несколько *контроллеров потока* (stream controller). Сервер должен обслуживать один или несколько *наборов заданий* (task set) с командами, предназначенными для выполнения сервером устройства или контроллером потока.

Запросы

Целевые действия (например, чтение с диска, при котором данные передаются с носителя в системную память), определяются *запросами*, формируемыми инициатором и сигнализируемыми им целевому устройству. Запрос содержится в структуре данных, называемой *блоком запроса операции* (Operation Block Request, ORB). Запросы используются для следующих целей:

- получения доступа к ресурсам целевого устройства (*login requests*);
- транспортировки командных блоков, обычных и потоковых (*command block requests*);
- управления наборами заданий и освобождения целевых ресурсов (*management requests*);
- управления потоком изохронных данных *(stream control requests).*

Запросы команд и управления потоками могут собираться в *очереди* (наборы заданий).

Агенты целевого устройства

Агенты целевого устройства — это программные компоненты, занимающиеся приемом и отработкой запросов. Взаимодействие инициатора с агентами осуществляется через регистры агентов целевого устройства. Агенты целевого устройства разделяются на два основных типа:

[1] В [7] термин «logical unit» переводится как «логический блок», однако для технологии SCSI, в которой шина IEEE 1394 является одним из средств доставки, больше подходит «логическое устройство».

- агенты, обслуживающие *одиночные запросы*\
- агенты, обслуживающие *очередь запросов.*

Инициатор динамически добавляет новые запросы в очередь, а *выбирающий агент* (fetch agent) целевого устройства по мере освобождения своих ресурсов выбирает их из области памяти, указанной инициатором. Целевому устройству ради повышения производительности позволяется изменять порядок обслуживания запросов очереди. По завершении обслуживания запроса, успешном или аварийном, целевое устройство записывает блок состояния по адресу, указанному инициатором в данном запросе.

По назначению различают три типа агента целевых устройств:

- *Агенты управления* (management agent) принимают запросы подключения *(login),* управления заданиями *(task management)* и отключения *(logout).* Эти агенты отрабатывают только одиночные запросы. До того как инициатор успешно выполнит защищенное подключение *(secure login),* никакие другие запросы к целевому устройству выполняться не будут.
- *Агенты командных блоков* обслуживают основной поток целевых запросов (обычных и потоковых) в соответствии с правами, полученными при подключении. Эти агенты работает с очередями.
- *Агенты управления потоком* обслуживают изохронные операции. Агент управления потоком связывается с агентом, обеспечивающим передачу потоковых командных блоков. Этот агент управления работает с очередями.

Потоки

Поток в SBP-2 — это объект, базирующийся на возможностях изохронной передачи 1394. Поток использует ресурсы целевого узла, необходимые для передачи данных. В блоке-«слушателе» (listener) данные одного или нескольких изохронных каналов 1394 передаются в среду целевого устройства (пример — цифровые аудиоколонки). В передающем блоке (talker) данные из среды передающего целевого устройства переправляются в один или несколько изохронных каналов (пример — цифровой микрофон, аудио-CD).

Потоковые данные идентифицируются номером канала и позицией данных (порядком по времени поступления) в потоке. Потоки требуют управления скоростью с синхронизацией по времени или некоторым событиям. Из-за этих особенностей для полного управления потоком нужны два компонента — набор потоковых заданий и контроллер потока.

Во время изохронных передач возможны ошибки, поведение целевого устройства при ошибках может быть различным:

- игнорировать ошибки, продолжая работу;
- останавливаться при первой же ошибке и сообщать о ней;
- регистрировать ошибки, продолжая работу.

ГЛАВА 19

Интерфейс IDE - ATA/ATAPI и SATA

Интерфейс IDE предназначен для подключения устройств хранения данных, обладающих собственным контроллером. В настоящее время интерфейс ATA/ ATAPI является самым массовым интерфейсом устройств хранения данных, причем не только в мире PC-совместимых компьютеров. Пока что наибольшее распространение получил его «классический» параллельный вариант, ему на смену идут последовательные интерфейсы Serial ATA (SATA) и Serial ATA-II (SATA-II). Теперь параллельный интерфейс ATA/ATAPI стали называть PATA (Parallel ATA — параллельный интерфейс ATA).

Параллельный интерфейс ATA (Advanced Technology Attachment) был введен в конце 1980-х годов как интерфейс для подключения накопителей на жестких магнитных дисках к компьютерам IBM PC AT с шиной ISA. Интерфейс появился в результате переноса стандартного (для PC/AT) контроллера накопителя на жестком диске (Hard Disc Controller, HDC) ближе к накопителю, то есть создания *устройств со встроенным контроллером* (Integrated Drive Electronics, IDE). Для связи устройства с системной шиной ISA использовали лен точный кабель с *параллельным шинным интерфейсом*, получившим названия *ATA* и *IDE*, которые, фактически, являются синонимами. В этом интерфейсе используются сигналы шины ISA, часть из которых буферизовали на небольшой плате адаптера IDE, устанавливаемого в слот ISA, а часть направили прямо на разъем нового интерфейса.

При переносе регистровая модель HDC была сохранена из соображений совместимости. Поскольку стандартный контроллер AT позволял подключать до двух накопителей, эту возможность получил и интерфейс ATA. Однако теперь два накопителя стали означать и два контроллера, подключенных к одной интерфейсной шине. Чтобы сохранить программную совместимость со стандартным контроллером HDC, к которому подключено два накопителя, оба контроллера в новом интерфейсе должны располагаться в пространстве ввода-вывода по одним и тем же адресам, выделенным стандартному контроллеру HDC.

Интерфейс ATA предназначен для обмена с устройствами хранения *блоками* фиксированного размера — *секторами* по 512 байт. Адресация данных внутри устройств ATA имеет «дисковые корни»: для накопителей изначально указывали адреса цилиндра (cylinder), головки (head) и сектора (sector) — так называв-

мая *трехмерная адресация CHS*. Позже пришли к *линейной* адресации логических блоков (Logical Block Address, LBA), где адрес блока (сектора) определяется 28-битным числом. Трехмерная и 28-битная линейная адресация в ATA имеют предел емкости устройств в 136,9 и 137,4 Гбайт соответственно, что по нынешним меркам недостаточно. В современных версиях интерфейса линейную адресацию расширили до 48-битной, при этом предел адресации составляет 2^{48} = 281 474 976 710 656 секторов, или около 144 Пбайт (петабайт), то есть 144 115 188 075 855 360 байт. Устройство может поддерживать различные форматы адресации, причем формат адреса может меняться даже в соседних командах.

Принятая система команд и регистров, являющаяся частью спецификации ATA, ориентирована на блочный обмен данными с жесткими магнитными дисками — устройствами хранения с непосредственным доступом. Позже спецификацию расширили для иных устройств хранения:

- Спецификация ATAPI позволяет передавать *пакет*, содержащий *командный блок* (откуда и часть названия PI — Package Interface). Структура командного блока заимствована из SCSI, его содержимое определяется типом подключенного устройства: ленточного, оптического (CD, DVD), магнитооптического и т. п. ATAPI позволяет расширить границы применения шины ATA, введя всего лишь одну новую команду передачи управляющего пакета.
- Набор дополнительных команд CFA (Compact Flash Association) введен для устройств хранения на флэш-памяти. От обычных устройств хранения (с непосредственным доступом) флэш-память отличается сравнительно длительным стиранием данных перед перезаписью. Группа дополнительных команд позволяет более эффективно работать с этими устройствами (хотя возможен доступ к ним и обычными, «дисковыми» командами ATA).

Параллельный интерфейс ATA исчерпал свои ресурсы пропускной способности, достигшей 133 Мбайт/с в режиме UltraDMA Mode 6. Для дальнейшего повышения пропускной способности интерфейса (но, увы, не самих устройств хранения, которые имеют гораздо меньшие внутренние скорости обмена с носителем) было принято решение о переходе от параллельной шины к последовательному двухточечному интерфейсу Serial ATA (SATA). Цель перехода — улучшение и удешевление кабелей и коннекторов, улучшение условий охлаждения устройств внутри системного блока (избавление от широкого шлейфа), обеспечение возможности разработки компактных устройств, облегчение конфигурирования устройств пользователем.

Интерфейс SATA позволяет сохранить (и развивать) сложившуюся систему команд ATA/ATAPI, что обеспечивает преемственность и программную совместимость со старым ПО. Поначалу интерфейс SATA отличался только способом транспортировки данных и команд между контроллером и устройствами. Главная революция в организации обмена с устройствами хранения произведена в спецификации SATA II, в которой описан эффективный механизм обслуживания очередей — NCQ на базе механизма FPDMA. Для SATA II появилась новая спецификация контроллера — AHCI, которая меняет и идеологию взаимодействия (сохраняя команды), что выводит SATA на «профессиональный»

уровень интерфейса устройств хранения, почти не уступающий по возможностям интерфейсу SCSI.
Разработкой спецификаций ATA/ATAPI занимается технический комитет T13 (прежде — T10) Международного комитета по стандартизации в области информационных технологий (INCITS). Разработанные им спецификации оформляются в виде стандартов ANSI. Развитие интерфейса отражает история спецификаций, начавшаяся с ATA-1 (1994 г.). В 2005 году обсуждаются спецификации ATA/ATAPI-7 и ATA/ATAPI-8, в которые входят как параллельные шины (PATA), так и последовательные интерфейсы (SATA). В Сети можно найти спецификации параллельной шины ATA/ATAPI (http://www.t13.org) и последовательного интерфейса Serial ATA (http://www.serialata.org).
Подробное описание интерфейса ATA и программного взаимодействия с устройствами приведено в [8].

19.1. Устройства, адаптеры, контроллеры и интерфейсы IDE

Устройства с параллельным интерфейсом ATA/ATAPI подключаются к компьютеру через контроллер (или адаптер) интерфейса ATA или IDE. В спецификациях ATA средства сопряжения называются хост-адаптером. Контроллер (адаптер) может иметь один или более *каналов IDE* (шин ATA). К каждому каналу IDE (параллельной шине ATA) может подключаться до двух устройств IDE (ATA/ATAPI):

- *ведущее устройство* (master) — ПУ, в спецификациях ATA официально называемое *Device-O* (устройство 0);
- *ведомое устройство* (slave) — ПУ, в спецификациях официально называемое *Device-1* (устройство 1).

Хост-адаптер и устройства объединяются кабелем-шлейфом, соединяющим параллельно одноименные контакты интерфейсных разъемов. Регистры контроллеров обоих устройств оказываются в одних и тех же областях пространства ввода-вывода. Для выбора устройства, исполняющего текущую команду, используется бит выбора накопителя (DEV) в регистре номера устройства и головки (регистр DH). Запись в этот регистр воспринимается сразу обоими устройствами, на обращения к остальным регистрам реагирует только выбранное. Исполнять команду будет лишь выбранное устройство. Если бит DEV = 0, выбрано ведущее устройство, если DEV = 1 — ведомое. Выводить выходные сигналы на шину ATA имеет право только выбранное устройство. Такой механизм адресации приводит к тому, что, начав операцию обмена с одним из устройств (подав команду), хост-адаптер не может переключиться на обслуживание другого до завершения начатой операции. Из-за этого эффективность работы с двумя устройствами на одной шине оказывается низкой. Для устройств хранения характерно значительное время доступа: с момента подачи команды до начала передачи блоков данных проходит заметное время, которое могло бы использоваться для интерфейсного обмена с другим устройством (как в SCSI). Параллельно

могут работать только устройства, подключаемые к разным шинам ATA (каналам IDE). В спецификации ATA/ATAPI-4 введен механизм освобождения шины (перекрытие команд) и очереди команд, однако он не так эффективен, как в SCSI, и используется редко. Полностью этот недостаток ATA устранен только в спецификации Serial ATA II.

Адаптер интерфейса ATA (IDE) объединяет простейшие аппаратные средства: буферные формирователи для шины данных, дешифратор адреса и вспомогательную логику. Адаптер обеспечивает обращения процессора к регистрам устройств ATA, а также передачу блоков данных между устройством и системной памятью в режиме PIO (программный ввод-вывод). Адаптер позволяет также подключить шину ATA к каналу стандартного контроллера DMA, однако обмен данными по стандартному каналу малоэффективен, поскольку канал довольно медленный и из-за пересечения границ страниц неудобный (см. 5.1).

Контроллер интерфейса ATA (IDE) отличается наличием собственного высокопроизводительного и эффективного контроллера DMA. Популярной моделью является контроллер *PCI IDE Bus Master* (см. 19.4), который имеет два канала — управляет двумя шинами ATA. Контроллер PCI IDE позволяет организовывать передачу массива данных, расположенных в разрозненных страницах физической памяти. Однако он не имеет средств переключения контекста (см. далее), позволяющих эффективно организовать перекрытие и очереди команд.

Интерфейс SATA (Serial ATA — последовательный интерфейс ATA) является эволюцией своего предшественника — параллельной шины ATA. Переход на последовательный интерфейс позволяет отказаться и от шинной организации интерфейса: контроллер Serial ATA имеет индивидуальные порты для подключения каждого из устройств. При этом имеется *режим эмуляции пар «ведущий*-ведомый», в котором сохраняется программная совместимость с контроллером параллельного интерфейса ATA и его механизмом адресации устройств. Для более эффективной работы со множеством устройств имеется и режим *прямого подключения* (Direct Port Access, DPA), программно несовместимый с традиционным интерфейсом ATA (IDE). Контроллер и устройства Serial ATA-II имеют эффективный механизм FPDMA для обслуживания множества устройств и очередей команд. Промежуточный (по эффективности) режим контроллера SATA обеспечивает только полную независимость устройств: каждому устройству выделяется собственный контроллер DMA.

Большинство современных системных плат оснащено двухканальным контроллером PCI IDE с параллельным интерфейсом ATA. Этот контроллер может работать в режиме как традиционного адаптера (Legacy IDE) с фиксированными областями адресов, выделяемых устройствам, так и устройства PCI, в котором области адресов свободно перемещаются в пространстве. Новые платы имеют и интерфейс Serial ATA. Сосуществование параллельного и последовательного интерфейсов возможно, однако в традиционном режиме может работать только один из контроллеров (параллельный или последовательный). Это создает некоторые трудности на этапе определения подключенных устройств и загрузки ОС. Для эффективного взаимодействия с обоими контроллерами ОС должна

иметь драйверы контроллеров, работающие с ними как с полноценными устройствами PCI.

Новая программная модель контроллера Serial ATA — *AHCI* (см. 19.5) — принципиально меняет идеологию работы с устройствами, существенно упрощая задачи программного взаимодействия с ними. Интерфейс AHCI имеют, в частности, контроллеры SATA, интегрированные в хаб ICH6 современных чипсетов системных плат. На время переходного периода контроллеры с AHCI снабжают и традиционными программными интерфейсами, однако использование возможностей AHCI в полной мере дает только специальный драйвер.

Интерфейс ATA предназначен для подключения устройств внутри корпуса компьютера; это обусловлено ограничением на длину шлейфа и конструкцией разъемов (применяются дешевые IDC-коннекторы). Интерфейс Serial ATA в первой версии также ограничивался внутренним использованием. В интерфейсе Serial ATA II введена возможность подключения внешних устройств, ради чего разъемы были доработаны (для лучшей фиксации).

В Serial ATA-II введены и новые варианты топологии соединения устройств: группа устройств (например, дисковый массив) может подключаться *к мультиплексору* Serial ATA, который соединяется с одним портом хост-контроллера. Это требует усложнения механизма аппаратного переключения контекста DMA, в противном случае с устройствами, подключенными к одному порту, возможна только поочередная работа (как это было с устройствами одного канала ATA). Устройство (или массив устройств, подключенных к мультиплексору) может подключаться и к двум хостам через специальный *селектор* портов. При этом активным может быть только один порт (и хост), а второй служит запасным путем доступа к устройствам хранения.

Интерфейс ATA позволяет подключать устройства различных категорий, различающиеся как типом (жесткие диски, оптические, ленточные устройства и т. п.), так и «уровнем интеллекта» встроенного контроллера.

Первые дисковые накопители IDE относились к категории неинтеллектуальных устройств *(Non-Intelligent IDE).* Они не выполняли трансляцию номеров секторов — параметры их внешней геометрии совпадали с реальными. Команды идентификации устройства и установки параметров не выполнялись. Дефектные блоки, отмеченные в заводском списке, были видны пользователю. Низкоуровневое форматирование выполнялось непосредственно по команде, так что неудачное форматирование могло снизить производительность из-за нарушения оптимальных параметров чередования и смещения.

Позже появились более интеллектуальные устройства *(Intelligent ATA IDE).* Они способны выполнять расширенные ATA-команды — идентификацию устройства и установку параметров. Поддерживается трансляция геометрии, дефектные секторы скрываются от пользователя (до исчерпания резерва). Для ускорения обмена эти устройства поддерживают блочные режимы передачи Read Multiple и Write Multiple, а также высокоскоростные режимы обмена PIO и DMA. Низкоуровневое форматирование возможно только при установке внешней геометрии, совпадающей с реальной. Однако форматирование опять-таки

может «снести» заводскую оптимизацию, хотя более поздние устройства либо игнорируют стандартную команду форматирования трека, либо только заполняют по ней все секторы трека нулями, не выполняя низкоуровневого форматирования.

К следующей категории относятся современные устройства с зонным форматом записи *(Intelligent Zoned Recording IDE).* Поскольку они имеют различное количество секторов на разных треках (для повышения плотности хранения), трансляция геометрии является для них обязательной — спецификация ATA не предусматривает сообщения устройством способа разбиения на зоны и формата каждой зоны, так что обращаться к ним можно только по внешнему трехмерному (CHS) или линейному (LBA) адресу. Информация о зонном распределении хранится, как правило, в служебной области носителя. Эта информация используется микропрограммой контроллера устройства, и с помощью специальных программных средств можно добраться до не и, например, отключить дефектную зону (уменьшив доступный объем диска). Низкоуровневое форматирование по стандартной команде как таковое не выполняется. Такие устройства либо отвергают эту команду, либо выполняют ее фиктивно (только позиционируя головки), либо просто заполняют все секторы трека нулями.

Устройства IDE различаются также по интеллектуальности контроллера, которая выражается в наличии средств автоматического мониторинга внутренних параметров (S.M.A.R.T.), скрытия дефектных блоков, температурной коррекции системы позиционирования, управления энергопотреблением, управления акустическим шумом и пр. Важным параметром устройств является размер собственной буферной памяти (кэша) и эффективность алгоритмов кэширования. Алгоритмы замещения блоков в кэше современных дисков учитывают особенности работы в многозадачной среде — они отслеживают несколько потоков запросов. Производители устройств вводят различные усовершенствования, в основном направленные на повышение производительности. Эти усовершенствования постепенно входят в очередную версию стандарта ATA/ATAPI как дополнительные свойства (features). Номер версии стандарта, поддерживаемый устройством, исчерпывающей информации о возможностях устройства не дает: какие-то дополнительные свойства, введенные в указанной версии, могут не поддерживаться (на то они и дополнительные). Ниже перечислены наиболее интересные в настоящее время свойства интерфейсной части устройства:

- Поддержка режима UltraDMA и достижимая скорость (33, 66, 100, 133 Мбайт/с). Этот режим предпочтителен благодаря не только высокой скорости обмена, но и наличию CRC-контроля достоверности передачи по интерфейсу. В PIO и обычных (не-Ultra) режимах DMA никакого контроля нет.
- Объем буферной памяти, определяющий возможность повышения производительности за счет упреждающего чтения и внутреннего кэширования. Для записывающих устройств CD/DVD объем памяти определяет устойчивость к задержкам доставки данных (увеличение объема снижает риск неудачи записи).
- Поддержка очередей NCQ (для Serial ATA), определяющая эффективность работы с множеством запросов. Поддержка «обычных» очередей и перекры

тия команд ATA/ATAPI в традиционных системах (со стандартным контроллером PCI IDE) положительного эффекта практически не дает; устройств с этими возможностями выпускается мало.

Параметры, относящиеся к физическому носителю (объем, физическая скорость записи и считывания, время позиционирования и т. п.), в данной главе не рассматриваются.

19.2. Параллельный интерфейс ATA

Для устройств IDE существуют несколько разновидностей параллельного интерфейса ATA (IDE). Устройства ATA IDE, E-IDE, ATA-2, Fast ATA-2, ATA-3, ATA/ATAPI-4, ATA/ATAPI-5 и ATA/ATAPI-6 электрически совместимы. Степень логической совместимости довольно высока (все базовые возможности ATA доступны). Однако для полного использования всех расширений необходимо соответствие спецификаций устройств, хост-адаптера и его ПО. Приводимые далее описания опираются на спецификации ATA (до ATA/ATAPI-6 включительно).

Физический интерфейс

Параллельный интерфейс ATA представляет собой шину, в которой все сигналы соответствуют стандартной логике ТТЛ. В наиболее распространенном варианте интерфейса все его информационные сигналы передаются через 40-кон- тактный разъем, у которого ключом является отсутствующий на вилке и закрытый на розетке контакт 20.

Для соединения устройств применяется плоский многожильный кабель- шлейф, длина кабеля не должна превышать 0,46 м. Состав информационных сигналов интерфейса ATA приведен в табл. 19.1, вид кабеля — на рис. 19.1. В большинстве старых кабелей одноименные контакты всех разъемов соединяются своими проводами, и все коннекторы равноправны. Исключение составляют шлейфы, предназначенные для кабельной выборки устройства. Кабель должен соответствовать системе адресации, выбранной для обоих устройств.

Таблица 19.1. Интерфейс ATA (IDE)

Сигнал	Тип[1]	Контакт	Контакт	Тип[1]	Сигнал
RESET#	I	1	2	-	GND
DD7	I/O TS	3	4	I/O TS	DD8
DD6	I/O TS	5	6	I/O TS	DD9
DD5	I/O TS	7	8	I/O TS.	DD10
DD4	I/O TS	9	10	I/O TS	DD11
DD3	I/O TS	11	12	I/O TS	DD12
DD2	I/O TS	13	14	I/O TS	DD13
DD1	I/O TS	15	16	I/O TS	DD14
DD0	I/O TS	17	18	I/O TS	DD15

продолжение

Таблица 18.1 (*продолжение)*

Сигнал	Тип[1]	Контакт	Контакт	Тип[1]	Сигнал
GND	–	19	20	–	Ключ (нет штырька)
DMARQ	O TS[2]	21	22	–	GND
DIOW#/STOP[3]	I	23	24	–	GND
DIOR#/HDMARDY#/HSTROBE[3]	I	25	26	–	GND
IORDY/DDMARDY#/DSTROBE[3]	O TS[2]	27	28	I/O	SPSYNC/CSEL[7]
DMACK#	I	29	30	–	GND
INTRQ	O TS[2]	31	32	O OK	IOCS16#[8]
DA1	I	33	34	I, O[4]	PDIAG#/CBLID[3]
DA0	I	35	36	I	DA2
CS0#	I	37	38	I	CS1#
DASP#	I/O OK[5]	39	40	–	GND
+5 В (Logic)	–	41[6]	426	–	+5 В (Motor)
GND	–	43[6]	44[6]	–	Зарезервирован

[1] Тип сигнала для устройства: I — вход, 0 — выход, I/O — двунаправленный, TS — тристабильный, OK — открытый коллектор. Для хост-адаптера значения I и O имеют противоположный смысл.

[2] У старых устройств сигнал может иметь тип O К (при разнотипных сигналах на одной шине возможен конфликт).

[3] Сигналы, приведенные после символа /, используются только в режиме Ultra DMA (ATA-4).

[4] У ведущего устройства — вход, у ведомого — выход.

[5] У ведомого устройства — только выход.

[6] Контакты 41-44 используются только для миниатюрных дисков.

[7] Начиная с ATA-3 — только CSEL.

[8] Начиная с ATA-3 — зарезервирован.

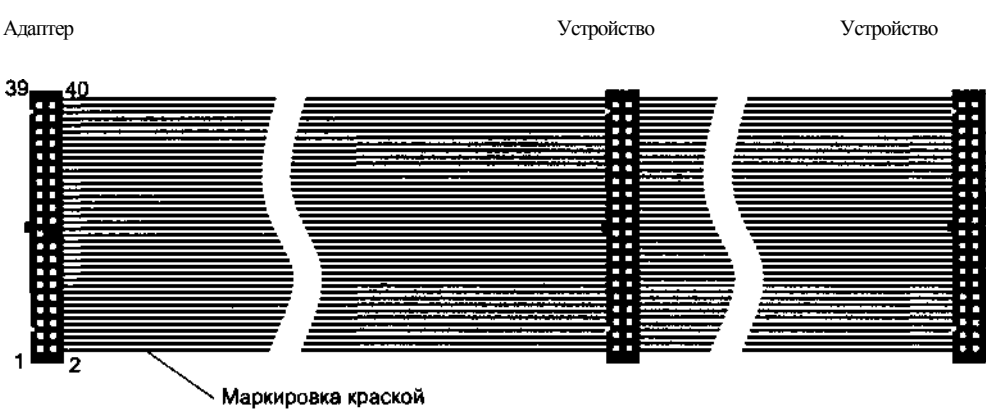


Рис. 19.1. Интерфейсный кабель ATA

Спецификация ATA «узаконивает» как 40-контактный интерфейсный разъем, так и 4-контактный разъем питания (рис. 19.2), но для малогабаритных устройств питание может подаваться по 44-проводному интерфейсному кабелю. Для малогабаритных внешних устройств существует довольно распространен

ный разъем HP 36, но в спецификацию ATA/ATAPI он не входит. Для устройств хранения на флэш-памяти используется коннектор, соответствующий спецификации CompactFlash Association. Для блокнотных ПК в стандарте име-

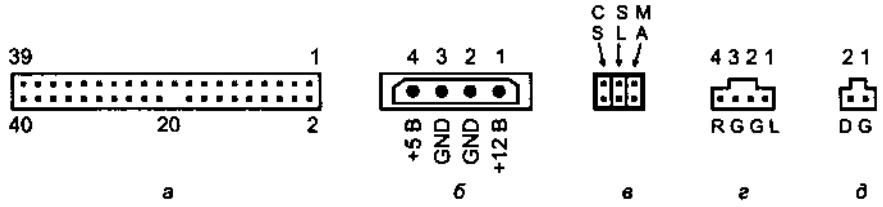


Рис. 19.2. Разъемы интерфейса ATA (вилки на устройствах): а — интерфейсный разъем, *б* - разъем питания, *в* — джамперы выбора устройства, г — аналоговый аудиовыход, *д* — цифровой аудиовыход

ется вариант интерфейса IDE на 68-контактном разъеме PCMCIA (PC Card), описанный в табл. 19.2.

Таблица 19.2. 68-контактный интерфейс ATA для PC Card (PCMCIA)

Контакт	Сигнал	Контакт	Сигнал
1	GND	26	
2	DD3	27	DA2
3	DD4	28	DA1
4	DD5	29	DA0
5	DD6	30	DD0
6	DD7	31	DD1
7	CS0#	32	DD2
8		33	
9	SELATA#	34	GND
10		35	GND
11	CS1#	36	CD1#
12		37	DD11
13		38	DD12
14		39	DD13
15		40	DD14
16	INTRQ	41	DD15
17	+5 В	42	CS1#
18		43	
19		44	DIOR#
20		45	DIOW#
21		46	
22		47	
23		48	
24		49	
25		50	

продолжение

Таблица 18.2 (*продолжение)*

Контакт	Сигнал	Контакт	Сигнал
51	+5 В	60	DMARQ
52		61	DMACK#
53		62	DASP#
54		63	PDIAG#
55	M/S#	64	DD8
56	CSEL	65	DD9
57		66	DD10
58	RESET#	67	CD2#
59	IORDY#	68	GND

Для устойчивой работы в режиме Ultra DMA рекомендуется применение *80- проводных кабелей*, обеспечивающих чередование сигнальных цепей и проводов схемной земли (GND). Такие кабели, требующиеся для режимов UltraDMA выше 2 (скорость выше 33 Мбайт/с), разводятся на специальные разъемы, имеющие 40-контактные гнезда с обычным назначением контактов, но ножевые контакты обеспечивают врезку 80 проводов. На 80-проводном кабеле в разъеме для подключения контроллера контакт 34 соединен с шиной GND и не соединен с проводом шлейфа; этим обеспечивается идентификация типа кабеля (CBLID). Провод шлейфа соединяет контакты 34 разъемов устройств, что обеспечивает прохождение сигнала PDIAG# от ведомого устройства к ведущему. При двухточечном соединении (контроллер — устройство) для режимов Ultra DMA Mode 3 и 4 можно использовать 40-проводный кабель (без среднего разъема).

До включения высокоскоростных режимов Ultra DMA должен быть определен факт присутствия 80-проводного кабеля, для чего есть несколько возможностей [6], [8]:

- *Определение типа кабеля через хост-контроллер*, для чего хост-контроллер должен иметь приемник сигнала CBLID#. Из-за некорректно работающего устройства 40-проводный кабель может ошибочно трактоваться как 80-про- водный.
- *Определение типа кабеля через устройство*, не требующее дополнительного приемника в контроллере (линия PDIAG#/CBLID# в хост-контроллере заземляется через конденсатор). Если на хост-контроллере нет конденсатора, то даже 40-проводный кабель будет идентифицироваться как 80-проводный, что опасно при передаче данных. Из-за некорректной работы ведомого устройства 80-проводный кабель может казаться 40-проводным.
- *Комбинированный метод определения типа кабеля* предполагает наличие на хост-контроллере и приемника сигнала CBLID#, и конденсатора (они друг другу не мешают). Решение о наличии 80-проводного кабеля принимается, только если это подтвердят оба метода. Возможная ошибка идентификации оказывается безопасной — при некорректном устройстве 1 на 80-проводном кабеле не будет включен высокоскоростной режим (наверное, это и к лучшему).

Назначение сигналов ATA

В документации на устройства могут быть указаны несколько различающиеся обозначения сигналов. Здесь приведены обозначения из стандарта ATA/ATAPI-4:

- RESET# (device reset) — сброс устройства (инвертированный сигнал сброса системной шины).
- DA[2:0] (device address) — три младших бита системной шины адреса, используемые для выбора регистров устройств.
- DD[15:0] (device data) — двунаправленная 16-битная шина данных между адаптером и устройствами.
- DIOR# (device I/O read) — строб чтения портов ввода-вывода.
- DIOW# (device I/O write) — строб записи портов ввода-вывода.
- IORDY (I/O channel ready) — готовность устройства завершить цикл обмена. Сигнал требуется при обмене в режиме PIO Mode 3 и выше.
- IOCS16# — разрешение 16-битных операций. Обращение ко всем регистрам, кроме регистра данных, всегда 8-битное. Начиная с ATA/ATAPI-3 не используется.
- DMARQ (DMA request) — запрос обмена по каналу DMA (необязательный).
- DMACK# (DMA acknowledge) — подтверждение DMA.
- INTRQ (interrupt request) — запрос прерывания, вырабатывает только выбранное устройство, когда у него имеется необслуженный запрос прерывания и выработка сигнала не запрещена битом nIEN в регистре Device Control.
- CS0# (chip select 0) — сигнал выбора регистра командного блока (command block registers). Для первого канала он вырабатывается при наличии на системной шине адреса порта ввода-вывода в диапазоне 1F0h — 1F7h (сигнал также называют CS1FX#).
- CS1# (chip select 1) — сигнал выбора регистра управляющего блока (control block registers). Для первого канала он вырабатывается при наличии на системной шине адреса порта ввода-вывода в диапазоне 3F6h — 3F7h (часто этот сигнал называют CS3FX#).
- PDIAG# (passed diagnostics) — сигнал о прохождении диагностики. Сигнал служит только для связи двух устройств и хост-адаптером не используется.
- CBLID# (cable assembly type identifier) — идентификация типа кабеля. В 80-проводной сборке контакт 34 на разъеме хост-адаптера соединяется с шиной GND, а контакты 34 разъемов устройств соединяются между собой и связи с разъемом хост-адаптера не имеют.
- DASP# (device active, slave present) — сигнал двойного назначения: индикатор активности устройства и присутствия ведомого устройства.
- SPSYNC/CSEL (spindle synchronization/cable select) — синхронизация шпинделя (отменен, начиная с ATA/ATAPI-3) или выборка кабелем. Сигнал CSEL позволяет устройствам определять свой адрес по положению на специаль

ном кабеле. Эта линия на хост-адаптере заземлена, и ведущее устройство получает заземленную линию, а ведомое — неподключенную.

В режиме Ultra DMA четыре линии получают новое назначение сигналов:

- STOP (stop Ultra DMA burst) — останов передачи пакета Ultra DMA.
- DDMARDY# (device Ultra DMA ready) — готовность устройства при приеме пакета Ultra DMA (управление потоком).
- DSTROBE (host Ultra DMA data strobe) — строб данных устройства при передаче пакета хосту. Данные передаются по обоим перепадам DSTROBE.
- HDMARDY# (host Ultra DMA ready) — готовность хоста при приеме им пакета Ultra DMA (управление потоком).
- HSTROBE (host Ultra DMA data strobe) — строб данных хоста при передаче пакета устройству. Данные передаются по обоим перепадам HSTROBE.

Для *блокнотных ПК* в 68-контактном разъеме (см. табл. 19.2) PCMCIA (PC Card) имеется ряд специфичных сигналов:

- SELATA# (select 68-pin ATA) — сигнал, которым хост идентифицирует режим использования разъема, PC Card (сигнал снят) или ATA (сигнал установлен, то есть низкий уровень).
- CD1# и CD2# (card detect) заземляются в устройстве — по этим сигналам хост определяет присутствие устройства.
- CS1# (device chip select 1) — выбор устройства, подается хостом на оба контакта (11 и 42), но устройство воспринимает только один из них.
- DMARQ, DMACK# и IORDY — эти сигналы не обязательны.
- M/S# (master/slave) — инверсия сигнала CSEL. Хост выдает сигналы M/S# и CSEL до подачи питания, устройство воспринимает лишь один из них.

Для «горячего» подключения разъем цепи GND обеспечивает более раннее соединение при подключении и более позднее при отключении. В устройстве сигналы CS0#, CS1#, RESET# и SELATA# подтягиваются к пассивному состоянию.

Подключение и конфигурирование устройств ATA/ATAPI

Устройства ATA/ATAPI перед подключением к шине должны быть корректно сконфигурированы и подключены к шлейфу нужного канала. Если к шине ATA подключено одно устройство, оно должно быть ведущим. Если подключены два устройства, одно должно быть ведущим, другое — ведомым.

ВНИМАНИЕ ----------

Ошибка в подключении разъемов ATA (переворот) обычно приводит к аппаратному сбросу компьютера, потому что сигнал RESET# (контакт 1) оказывается заземленным (контактом 40). Из-за этого при включении даже не появляются сообщения или звуковые сигналы POST.

Существует два способа задания адреса устройства — кабельной выборкой или явным указанием адреса на каждом из устройств. Режим кабельной выборки

включается перемычкой *CS* (Cable Select — кабельная выборка). В этом случае оба устройства на шине конфигурируются одинаково — в режим CS, а адрес устройства определяется его положением на специальном ленточном кабеле. Кабельная выборка будет работать, если она поддерживается и задана на всех устройствах канала, включая хост-адаптер, который обеспечивает заземление контакта 28. При этом способе задания адресов синхронизация шпинделей накопителей через тот же провод контакта 28 (как в старых RAID-массивах) исключается.

В первоначальном варианте 40-проводного шлейфа с кабельной выборкой (рис. 19.3) провод был 28 перерезан, так что контакт 28 (CSEL) для ведущего устройства заземлялся через хост-адаптер, а для ведомого оказывался неподключенным.

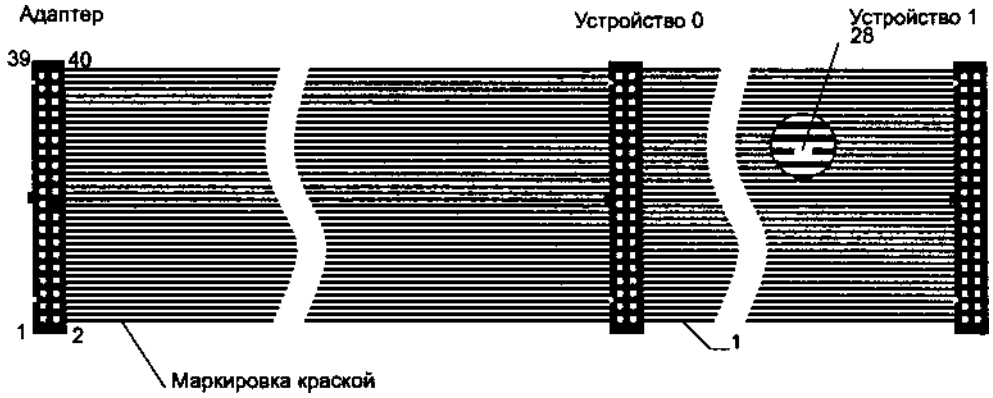


Рис. 19.3. Ленточный 40-проводный кабель интерфейса ATA с кабельной выборкой

Начиная с ATA/ATAPI-4 узаконили 80-проводный шлейф с кабельной выборкой, в котором для подключения устройства 1 (slave) определили средний коннектор (см. рис. 19.1). В среднем коннекторе контакт 28 либо не соединен с проводом, либо просто отсутствует.

Вполне понятно, что при кабельной выборке хост-контроллер подключать к среднему коннектору кабеля, изображенного на рис. 19.1, нельзя (как и к правому на рис. 19.3). Если номер устройства назначается явно, то для 40-провод- ного кабеля можно подключать устройства и хост-контроллер к любым коннекторам произвольно (но желательно избегать «висячих» концов). Современные 80-проводные шлейфы, используемые для режимов UltraDMA, имеют однозначное назначение коннекторов:

- Коннектор *хост-контроллера* расположен на конце шлейфа, у него контакт 34 заземлен и не соединен со шлейфом. Корпус коннектора должен быть *синим.*
- Коннектор *устройства 0* расположен на противоположном конце шлейфа, у него все контакты соединены со шлейфом. Корпус коннектора должен быть *черным.*

- Коннектор *устройства 1* (необязательный) расположен в середине шлейфа, у него контакт 28 не соединен со шлейфом. Корпус коннектора должен быть *серым.*

Заземление контакта 34 позволяет хосту распознать 80-проводный кабель. Если кабельная выборка не используется, то устройства 0 и 1 можно менять местами.

ИМАНИЕ ----------

Висящий (не подключенный к устройству) конец кабеля может быть источником ошибок передачи данных по интерфейсу. Во всех режимах обмена, кроме UltraDMA, эти ошибки НЕ КОНТРОЛИРУЮТСЯ и могут приводить к искажению и потере данных на носителе.

Более распространен режим явной адресации, при котором используется обычный «прямой» кабель (см. рис. 19.1). В этом случае перемычка CS не устанавливается, а адрес устройства задается перемычками, состав которых варьируется. В принципе, достаточно лищь указать устройству его номер (0 или 1), но в устройствах, разработанных до стандарта ATA, ведущему устройству «подсказывали» о наличии ведомого (по интерфейсу ATA оно могло бы определить это само по сигналу DASP#). Ниже перечислены комбинации джамперов, которые можно встретить на устройствах IDE:

- M/S (Master/Slave) — переключатель адреса (ведущее/ведомое). Если на шине присутствует одно устройство, оно должно быть сконфигурировано как ведущее. Если на шине два устройства — одно должно быть ведущим, другое ведомым. Иногда джампер обозначается как C/D (диск C/диск D), но для второго канала IDE такое название некорректно. Когда появились первые IDE- диски емкостью 1 Гбайт, для преодоления барьера в 504 Мбайт некоторые модели допускали конфигурирование в виде двух устройств (0 и 1) половинной емкости. В таком режиме на их ленточный кабель IDE второе физическое устройство подключать нельзя.
- SP (Slave Present), DSP (Drive Slave Present), Master but Slave is not ATA-compatible или Master but Slave uses only PDIAG-signal — устанавливается на ведущем устройстве для указания на присутствие ведомого. Если переключатель установлен, а ведомое устройство не подключено, POST сообщает об ошибке. Джампер применяется для дисков, не использующих сигнал DASP#.
- Single Drive — джампер, устанавливаемый на устройстве, если оно единственное на шине (встречается на дисках Western Digital). Устройство становится ведущим.
- ACT (Drive Active) — джампер, соединяющий линию DASP# с формирователем сигнала активности устройства. Устанавливается на устройстве 0, встречается редко.
- HSP — джампер, заземляющий линию DASP# (положение, взаимоисключающее ACT). Устанавливается на устройстве 1 для сигнализации о его присутствии (встречается редко).

Для полностью ATA-совместимых дисков правильно сконфигурированные устройства определяются автоматически. Современные контроллеры ATA позволяют подключать даже единственное устройство как ведомое — интерфейсные функции ведущего берет на себя контроллер.

Разобраться с джамперами старых устройств трудно, если нет документации, однако обширная база данных по разным моделям встроена в справочный файл утилиты Disk Manager. У современных устройств лишние джамперы упразднили, а существующие комментируются на наклейке (шильдике).

ВНИМАНИЕ

Следует учитывать, что перестановка джамперов часто воспринимается устройством только по включении питания. Кроме того, установка на один ленточный кабель двух разнотипных устройств, если они не являются устройствами ATA, часто невозможна.

Режимы передачи данных для устройств ATA

Программа общается с устройствами ATA через регистры, используя инструкции ввода-вывода IN и OUT. Для передачи данных с максимальной скоростью применяют программный доступ *PIO* к регистру данных инструкциями INSW/ OUTSW или обмен по каналу DMA. Тип обмена (PIO или DMA) определяется командой обращения. Программный доступ PIO обязателен для всех устройств, команды режима DMA устройствами могут не поддерживаться. Параметры различных режимов передачи приведены в табл. 19.3.

Таблица 19.3. Параметры режимов передачи

Режим передачи	Минимальное время цикла, нс	Скорость передачи, Мбайт/с	Интерфейс
PIO mode 0	600	3,3	ATA
PIO mode 1	383	5,2	ATA
PIO mode 2	240	8,3	ATA
PIO mode 3	180	11,1	E-IDE, ATA-2 (используется IORDY)
PIO mode 4	120	16,6	E-IDE, Fast ATA-2 (используется IORDY)
Singleword DMA Mode 0	960	2,08	ATA
Singleword DMA Mode 1	480	4,16	ATA
Singleword DMA Mode 2	240	8,33	ATA
Multiword DMA Mode 0	480	4,12	ATA
Multiword DMA Mode 1	150	13,3	ATA-2
Multiword DMA Mode 2	120	16,6	Fast ATA-2
Ultra DMA Mode 0	120[1]	16,6	ATA/ATAPI-4
Ultra DMA Mode 1	80[1]	25	ATA/ATAPI-4
Ultra DMA Mode 2	60[1]	33	ATA/ATAPI-4
Ultra DMA Mode 3	45[1]	44,4	ATA/ATAPI-5
Ultra DMA Mode 4	30[1]	66,6	ATA/ATAPI-5

продолжение ›

Таблица 19.3 (*продолжение*)

Режим передачи	Минимальное время цикла, нс	Скорость передачи, Мбайт/с	Интерфейс
Ultra DMA Mode 5	20[1]	100	ATA/ATAPI-6
Ultra DMA Mode 6	15[1]	133	ATA/ATAPI-7

[1] В пакете данных режима Ultra DMA за каждый такт передаются два слова данных, одно по фронту синхронизирующего сигнала, друге по спаду. Период следования синхросигналов равен удвоенному времени цикла.

Обмен в режиме PIO (Programmed Input/Output — программируемый ввод-вывод) выполняется в виде следующих друг за другом операций чтения или записи в пространстве ввода-вывода по адресу регистра данных. Готовность устройства проверяется перед началом передачи блока, после чего хост производит серию операций в темпе, который определяется выбранным режимом *PIO Mode 0-4*. Для каждого режима определены допустимые параметры временной диаграммы цикла обмена. Обмен PIO программно реализуется с помощью процессорных инструкций ввода-вывода — строк REP INS или REP OUTS с занесенным в регистр CX количеством слов (или байтов) в передаваемом блоке. Эти инструкции обеспечивают максимально возможную скорость обмена для конкретных процессора и системной шины. «Обуздать» процессор в соответствии с выбранным режимом входит в задачу адаптера ATA, который использует для удлинения цикла сигнал готовности шины (для ISA — IOCHRDY). Традиционные режимы 0, 1 и 2 имеют временные параметры, задаваемые только хост-адаптером. Для прогрессивных режимов ATA-2 (PIO Mode 3 и выше) устройство может затормозить обмен сигналом готовности IORDY. Программный обмен на все время передачи блока занимает и процессор, и системную шину.

Обмен в режиме DMA занимает исключительно шины ввода-вывода и памяти. Процессору требуется выполнить только процедуру инициализации канала, после чего он свободен до прерывания от устройства в конце передачи блока (этим могут воспользоваться многозадачные системы). Стандартные каналы DMA шины ISA для интерфейса ATA из-за низкой пропускной способности не применяются. Высокопроизводительные адаптеры ATA имеют собственные более эффективные контроллеры. Режимы обмена по каналу DMA бывают одиночными и множественными. В *одиночном режиме* (singleword DMA) устройство для передачи каждого слова вырабатывает сигнал запроса DMARQ и сбрасывает его по сигналу DMACK#, подтверждающему цикл обмена. В *множественном режиме* (multiword DMA) на сигнал DMARQ хост отвечает потоком циклов, сопровождаемых сигналами DMACK#. Если устройство не справляется с потоком, оно может приостановить его снятием сигнала DMARQ, а по готовности установить его снова. Множественный режим позволяет развить более высокую скорость передачи.

Режим *Ultra DMA*, появившийся в спецификации ATA/ATAPI-4, позволяет перешагнуть барьер в 16,6 Мбайт/с, свойственный традиционным режимам и используемому кабелю. При этом обеспечивается и контроль достоверности передачи данных по шине, чего не делалось ни в PIO, ни в стандартных режимах

DMA (а зря!). Стандартом ATA-4 было определено 3 режима Ultra DMA (0, 1 и 2), впоследствии ввели новые режимы; выбор режима осуществляется командой Set Features. В режимах Ultra DMA сигналы DMARQ и DMACK# сохраняют свое назначение, а вот смысл сигналов DIOR#, DIOW# и IORDY на время передачи пакета (Ultra DMA Burst) существенно меняется. В пакете данные на шине сопровождаются стробом, генерируемым источником данных, причем для синхронизации используются оба перепада сигналов. Это позволяет повысить пропускную способность шины до 33 Мбайт/с, не увеличивая частоту переключения сигналов сверх 8,33 x 10^6 $с^{-1}$ (этот предел для обычного кабеля достигается в режимах PIO Mode 4 и Multiword DMA Mode 2). Для более высоких скоростей требуется 80-проводный кабель. Передача в пакете может приостановиться, если приемник снимет сигнал готовности (DDMARDY# или HDMARDY#). Передача пакета может прекращаться по инициативе устройства (снятием сигнала DMARQ) или хоста (сигналом STOP). Противоположная сторона должна подтвердить окончание цикла сигналом STOP или снятием сигнала DMARQ соответственно.

Каждое переданное слово участвует в подсчете CRC-кода, который передается хост-контроллером в конце пакета. Подсчет ведется и источником данных, и приемником. При несовпадении принятого устройством кода с ожидаемым кодом фиксируется ошибка передачи, о которой устройство сообщает в конце исполнения команды. Способ сообщения об ошибке передачи зависит от типа выполнявшейся команды. Для команд READ DMA, WRITE DMA, READ DMA QUEUED и WRITE DMA QUEUED в регистре ошибок ER устанавливаются биты 7 (ICRC) и 2 (ABRT). Для пакетной команды REQUEST SENSE в случае ошибки в регистре состояния SR устанавливается бит 0 (CHK) и сообщается ключ состояния (Sense key) 0Bh (команда отвергнута). Для всех других пакетных команд в случае ошибки устанавливается бит CHK и сообщается состояние 04h (аппаратная ошибка), а в последующих командах REQUEST SENSE сообщается значение ASC/ASCQ 08h/03h (ошибка CRC при связи с логическим устройством). Получив сообщение об ошибке, хост должен повторить команду. Если ошибки появляются постоянно, хост должен снизить скорость обмена (вплоть до выхода из режима Ultra DMA).

Тип режима обмена определяется возможностями хост-адаптера (и его драйвера), устройств и кабеля, и для каждого устройства он ограничен минимумом из максимальных возможностей всех этих компонентов. Как правило, режимы устанавливаются системой автоматически, но пользователю дается возможность при необходимости «подрезать крылья» контроллеру настройкой BIOS Setup.

Правильный выбор режима обмена обеспечивает надежность и производительность. Все устройства поддерживают режим PIO Mode 0, в котором считывается блок параметров идентификации. В блоке имеются поля, описывающие режим обмена по умолчанию и более эффективные режимы обмена, поддерживаемые устройством. Командой Set Features можно изменить параметры режима. Иногда накопитель не обеспечивает надежной передачи данных в заявленном высокоскоростном режиме. Если данные начинают пропадать, первым делом следует перейти на более медленный режим обмена.

BIOS определяет режим обмена с каждым устройством с учетом ограничений, заданных в Setup. Старые диски, не сообщающие своих параметров, могут не работать с новыми режимами PIO. На одном ленточном кабеле (канале ATA) могут присутствовать устройства с разными режимами обмена — спецификация это допускает. Однако реально могут возникать аппаратные или программные ограничения. Некоторые чипсеты не позволяют независимо программировать режим обмена для устройств канала. В таком случае при подключении двух разных устройств (например, PIO Mode 1 и 3) обмен с обоими устройствами будет происходить со скоростью меньшего режима (PIO Mode 1). Поэтому не рекомендуется к одному каналу ATA (порту IDE) подключать быстрый винчестер и медленный привод CD-ROM. Иногда взаимозависимость режимов обмена двух устройств обусловлена ограниченным набором параметров конфигурации в BIOS. Быстрые режимы множественного обмена по DMA реализуются только драйверами ОС. «Глупый» драйвер может попытаться навязать медленный режим обоим устройствам канала, так что смешивать разные устройства не стоит и по этой причине.

19.3. Интерфейс Serial ATA

Интерфейс SATA (Serial ATA — последовательный интерфейс ATA) предназначен для замены традиционного параллельного (PATA) с сохранением регистровой модели подключаемых устройств и возможностей передачи данных в режимах PIO и DMA. При этом шинное подключение пары устройств к одному каналу ATA заменяется двухточечными соединениями *устройств* с *порта- ми* хост-контроллера (или концентратора). Программное взаимодействие с устройствами Serial ATA практически совпадает с прежним, набор команд соответствует ATA/ATAPI-5. Для полной программной совместимости контроллер SATA может эмулировать пары устройств (ведущее-ведомое) на одном канале, если такая необходимость возникнет. В то же время аппаратная реализация хост-адаптера Serial ATA значительно отличается от примитивного (в исходном варианте) интерфейса ATA. В параллельном интерфейсе ATA хост-адаптер был простым средством программного обращения к регистрам, расположенным в самих подключенных устройствах. В Serial ATA ситуация иная: хост-адаптер имеет блоки так называемых «теневых» регистров (shadow registers), совпадающих по назначению с обычными регистрами устройств ATA. Каждому порту соответствует свой набор регистров. Обращения к этим теневым регистрам вызывают процессы взаимодействия хост-адаптера с подключенными устройствами и исполнение команд.

Переход на последовательный интерфейс и двухточечные соединения в Serial ATA дает ряд преимуществ:

- каждое устройство получает монопольный канал связи с контроллером, что позволяет повысить производительность обмена с устройствами;
- исключаются ненужные протокольные взаимодействия ведущего и ведомого устройств параллельной шины и связанные с ними проблемы совместимости устройств;

- появляется возможность одновременной работы контроллера с несколькими устройствами с использованием механизма FPDMA и эффективной поддержкой очередей (NCQ);
- упрощается (для пользователя) конфигурирование устройств (не требуется выбор адреса);
- обеспечивается возможность полной поддержки горячего подключения/отключения;
- имеются перспективы повышения скорости обмена с устройствами (относительно базовой скорости 150 Мбайт/с);
- упрощаются и удешевляются кабели и разъемы;
- улучшаются условия охлаждения устройств — тонкий кабель не препятствует циркуляции воздуха в корпусе компьютера или массива устройств.

Помимо преимуществ последовательного двухточечного интерфейса, в SATA решена проблема адресации — введен режим LBA-48, появившийся и в последних версиях параллельного интерфейса (ATA/ATAPI-6). Наиболее эффективно возможности SATA используются в его естественном режиме работы, а не в режиме совместимости с параллельным интерфейсом ATA.

Все функции взаимодействия устройства и контроллера, выполняемые в параллельном интерфейсе при помощи множества управляющих и информационных линий, реализуются и в последовательном, но с использованием только двух встречных сигнальных линий. В стандарте рассматривается четырехуровневая модель взаимодействия хоста и устройства, где на верхнем (прикладном) уровне между хостом (процессором и памятью) и устройством SATA выполняется обмен командами, информацией о состоянии и хранимыми данными. Три нижестоящих уровня обеспечивают связь устройства и хост-контроллера по последовательному интерфейсу:

- *Транспортный уровень* конструирует *информационные структуры* (Frame Information Structure, FIS), которыми обмениваются контроллер и устройство, передает эти структуры канальному уровню и обеспечивает управление FIFO-буферами обмена с прикладным уровнем. Структуры, принятые от канального уровня, он разбирает на составные части и передает их прикладному уровню;
- *Канальный уровень* из информационных структур, представляемых потоками двойных слов, конструирует *кадры* (обрамляет структуры служебными примитивами, подсчитывает CRC для потоков данных транспортного уровня), выполняет кодирование 8B/10B, скремблирование и передачу кадров физическому уровню в виде битового потока. Принимаемые с физического уровня битовые потоки канальный уровень преобразует обратно в выровненные потоки двойных слов, проверяет корректность CRC и, освободив от служебных примитивов, передает их транспортному уровню. Со своим партнером (канальным уровнем противоположной стороны интерфейса) уровень обменивается подтверждениями успешного приема кадра и уведомляет об этих успехах свой транспортный уровень.

- *Физический уровень* принимает от канального данные кадра в параллельном 10-, 20-, 40-разрядном (или более) виде и преобразует их в сигналы последовательного интерфейса. Над последовательными данными, принятыми от партнера по интерфейсу, производятся обратные преобразования. Уровень выполняет инициализацию интерфейса при подключении и подаче питания, определяет состояние подключения устройства и успех согласования скоростей, передавая эту информацию канальному уровню. Дополнительно уровень может заниматься управлением энергопотреблением интерфейса, а также калибровкой приемопередатчиков (согласованием с линией).

Контроллеры SATA уже имеют свою историю: первые контроллеры (например, Intel 31244), разработанные по спецификации Serial ATA 1.0a, были нацелены на поддержку нового интерфейса подключения и старого программного интерфейса. Затем была принята спецификация нового программного интерфейса — AHCI, позволяющая реализовать основные преимущества SATA. Контроллер AHCI избавляет центральный процессор от рутинной обработки даже старых команд ATA/ATAPI, в которых используется режим обмена PIO. Возможность работы с новыми контроллерами по старым интерфейсам спецификацией AHCI не запрещается, но это уже другой механизм, никак не связанный с новым. Интерфейс AHCI реализован, например, в южном хабе ICH6 новых чипсетов системных плат фирмы Intel.

Физический интерфейс SATA

В первом поколении Serial ATA данные по кабелю передаются со скоростью **1,5** Гбит/с, что (с учетом кодирования 8B/10B) обеспечивает скорость 150 Мбайт/с (без учета накладных расходов протоколов верхних уровней). В дальнейшем планируется повышать скорость передачи, но в SATA-II новая скорость еще не вводилась. В интерфейс заложена возможность согласования скоростей обмена по каждому порту в соответствии с возможностями хоста и устройства, а также качеством связи. Хост-адаптер имеет средства управления соединениями, программно эти средства доступны через специальные регистры Serial ATA.

Стандарт SATA определяет новый однорядный двухсегментный разъем с механическими ключами, препятствующими ошибочному подключению. Сигнальный сегмент имеет 7 контактов (S1-S7), питающий (необязательный) — 15 (P1-P15); все контакты расположены в один ряд с шагом 1,27 мм. В стандарте определено и положение разъемов на корпусе устройства. Малые размеры разъема и малое количество цепей облегчают компоновку системных плат и карт расширения, а также позволяют применять однотипные разъемы как для обычных устройств (5" и 3,5"), так и для малогабаритных (2,5" и 1,8"). Назначение контактов приведено в табл. 19.4, вид разъемов — на рис. 19.4. Питающий сегмент используется на устройствах, предназначенных для установки в шасси. Обычные устройства могут получать питание от традиционного 4-контактного разъема ATA, которым снабжены типовые блоки питания ATX и AT.

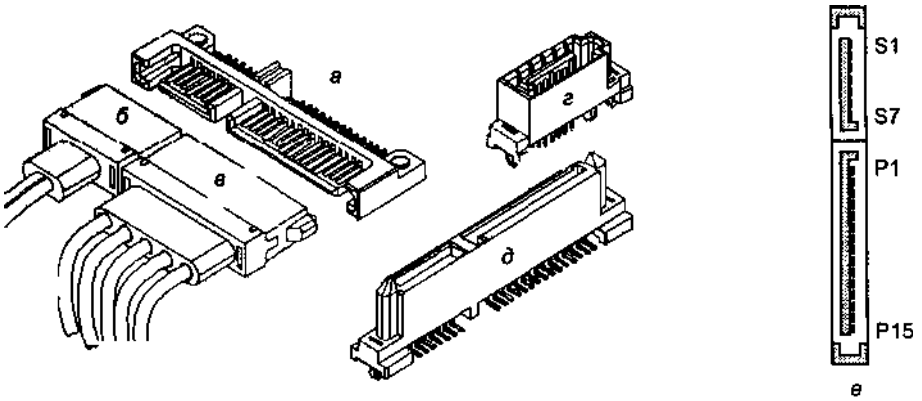


Рис. 19.4. Разъемы Serial ATA: а — полный разъем на устройстве, б — сигнальный сегмент кабельного разъема, *в* — питающий сегмент кабельного разъема, г — сигнальный сегмент разъема хост-адаптера, *д* — разъем хоста для непосредственного подключения устройства, в — расположение контактов на разъеме устройства

Таблица 19.4. Разъем Serial ATA

Контакт	Цепь	Назначение
S1	GND	Экран
S2	A+	Дифференциальная пара сигналов A; на хост-контроллере (HT+, HT-) – выход передатчика, на устройстве (DR+, DR-) – вход приемника
S3	A–	
S4	GND	Экран
S5	B–	Дифференциальная пара сигналов B; на хост-контроллере (HR+, HR-) – вход приемника, на устройстве (DT+, DT-) – выход передатчика
S6	B+	
S7	GND	Экран
P1	V33	Питание 3,3 В
P2	V33	Питание 3,3 В
P3	V33	Питание 3,3 В, предварительный заряд
P4	GND	Общий
P5	GND	Общий
P6	GND	Общий
P7	V5	Питание 5 В, предварительный заряд
P8	V5	Питание 5 В
P9	V5	Питание 5 В
P10	GND	Общий
P11	Резерв	
P12	GND	Общий
P13	V12	Питание 12В, предварительный заряд
P14	V12	Питание 12 В
P15	V12	Питание 12 В

Конструкция разъемов, предложенная в Serial ATA 1.0, оказалась не очень удачной — в ней не предусмотрена фиксация кабельной вилки в розетке устройства и контроллера (хотя для применения внутри корпуса компьютера, на которое была нацелена спецификация Serial ATA, это было терпимо). В Serial ATA II разъем доработали — ввели пружинные фиксаторы (на кабельной части) и соответствующие прорези на разъеме устройства или порта. Конструкции разъемов SATA и SATA II совместимы, но надежная фиксация обеспечивается лишь при подключении устройств (и контроллера) SATA II кабелем с вилками SATA II.

В SATA II определено несколько новых типов сигнальных разъемов и кабелей:

- Многопортовый (multi lane) внутренний разъем, по которому могут передаваться сигналы 2-4 портов SATA. Кабель с таким разъемом может иметь на другом конце аналогичный разъем, а может представлять собой пучок кабелей (fanout cable) с обычными однопортовыми разъемами на концах.
- Однопортовый внешний разъем SATA, полностью экранированный.
- Многопортовые (на 2-4 порта) малогабаритные внешние разъемы SFF 8470, которые могут быть как с ключами (выступы и прорези), так и без них. При этом вилки (на кабеле) без ключей не входят в слоты (на устройстве) с ключами.

Для обеспечения горячего подключения контакты разъемов имеют разную длину. В первую очередь соединяются (и в последнюю — разъединяются) контакты «земли» P4 и P12; затем остальные «земли» и контакты предварительного заряда конденсаторов в цепях питания P3, P7 и P13 (для уменьшения броска потребляемого тока), после чего соединяются основные питающие контакты и сигнальные цепи. На контакты предварительного заряда питание от соответствующих источников подается через резисторы сопротивлением 10-20 Ом. В шасси с «горячим» подключением факт подключения устройства может определяться посредством измерения сопротивления между контактами предварительного заряда и основными питающими контактами (например, P7 и P8).

Расширения SATA для систем хранения данных

Интерфейс Serial ATA «дорос» до использования в сложных системах хранения данных. Его производительность и эффективность высоки, малое число сигнальных проводов позволяет организовывать подключение множества устройств. С помощью SATA можно строить полностью избыточные системы хранения, содержащие массивы устройств (например, RAID), которые могут быть подключены и более чем к одному хост-компьютеру (но активным может быть только один хост). Пример системы хранения данных, иллюстрирующий различные элементы топологии, приведен на рис. 19.5.

В SATA-II вводится абстрактное понятие *концентратора* — средства подключения к хосту множества устройств SATA. У концентратора имеется *хост-интерфейс* и ряд *портов SATA* для подключения устройств. Типы хост-интерфейса разнообразны — PCI/PCI-X/PCI Express, Advanced Switching, InfiniBand, Ethernet (iSCSI), Fibre Channel, Serial ATA... Концентратор может являться

просто мостом, RAID-контроллером, коммутатором или мультиплексором портов. Хост-контроллер, подключенный к шине PCI или интегрированный в чипсет системной планы, также является концентратором (мостом между PCI и SATA).

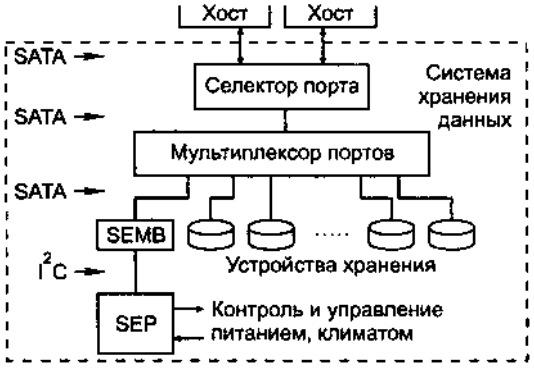


Рис. 19.5. Пример системы хранения данных

Новый (для ATA) элемент топологии — *мультиплексор портов* (port multiplier) — служит для подключения множества устройств SATA к одному порту хост-контроллера. Противоположное назначение имеет *селектор порта* (port selector) — этот элемент позволяет выбрать хост для работы с данным устройством или подсистемой хранения. Селектор порта обеспечивает избыточность путей доступа к устройству, но активным в данный момент времени может быть только один путь.

Подсистемы хранения, как правило, конструктивно оформляются в виде внешнего (по отношению к компьютеру) *блока храпения* (storage enclosure), в состав которого, помимо концентратора, устройств и, разумеется, блока питания, могут входить дополнительные *средства обслуживания* (enclosure management). В круг задач, решаемых этими средствами, могут входить поддержание климата, контроль питания, несанкционированного доступа и т. п. Эти задачи решает *обслуживающий процессор* (Storage Enclosure Processor, SEP), с которым хост должен иметь возможность взаимодействия. Чтобы не увеличивать номенклатуру интерфейсов ради связи хоста и SEP, в SATA II определена логическая топология интерфейсов, приведенная на рис. 19.5.

Связь хоста с SEP обеспечивает мост *SEMB* (Serial ATA Enclosure Management Bridge), который для хоста представляется как устройство ATA (Serial ATA), а для подключения SEP имеет последовательный интерфейс I^2C.

19.4. Адаптеры и контроллеры ATA

Адаптеры и контроллеры обеспечивают подключение физических устройств ATA и возможность взаимодействия с ними хоста. В данном разделе рассмат

риваются простейший адаптер, с которого началась история ATA, его современный вариант PCI IDE и два поколения контроллеров SATA.

Традиционный адаптер шины ATA

Простейший адаптер ATA содержит только буферы сигналов шины и дешифратор зоны адресов. Все регистры контроллера и схемы кодирования размещены в самом устройстве IDE. Шина ATA требует выделения системных ресурсов — двух областей портов ввода-вывода и линии прерывания; дополнительно может потребоваться канал DMA. Интерфейсу ATA первого канала выделили ресурсы, ранее использовавшиеся контроллером жестких дисков. Второму каналу назначили ресурсы альтернативного контроллера жестких дисков. Позже определили ресурсы еще для двух каналов (табл. 19.5). Традиционному контроллеру жестких дисков выделялся канал DMA3, но он является 8-битным, в то время как шина ATA требует 16-битного канала DMA. Производительности стандартных каналов DMA для шины ATA явно недостаточно.

Таблица 19.5. Системные ресурсы каналов ATA

Канал	CS0	CS1	IRQ
1	1F0h–1F7h	3F6h–3F7h	14
2	170h–177h	376h–377h	15 или 10
3	1E8h–1Efh	3Eeh–3Efh	12 или 11
4	168h–16Fh	36Eh–36Fh	10 или 9

На системных платах с шиной PCI типична установка двухканального адаптера, занимающего ресурсы каналов 0 и 1. В идеальном варианте двухканальные контроллеры имеют шины, полностью изолированные друг от друга буферными и логическими схемами. В самом дешевом варианте они используют общие буферы для линий данных и управляющих сигналов, а отдельные буферы применяются только для некоторых сугубо индивидуальных сигналов.

Современные системные платы оснащаются высокопроизводительными контроллерами шины ATA, обеспечивающими *прямое управление шиной PCI* при обмене с устройствами в режимах DMA и Ultra DMA. Прямое управление шиной повышает суммарную производительность компьютера в многозадачных и многопоточных операционных системах. Сами по себе режимы DMA не дают выигрыша в скорости обмена по шине ATA — только режимы UltraDMA Mode 1 и выше превосходят по скорости режим PIO Mode 4 (см. табл. 19.3). Однако обмен в режиме DMA значительно меньше загружает центральный процессор компьютера, и параллельно с дисковым обменом процессор может заниматься обработкой других потоков (задач). В однозадачных (и однопоточных) системах во время дискового обмена процессор все равно ничем другим не занимается, поэтому для них хорош и режим PIO Mode. Для реального использования режима прямого управления в операционной системе должен быть установлен специальный драйвер Bus-Master, соответствующий используемому контроллеру ATA (как правило, чипсету системной платы). Стандартный контроллер PCI IDE описан далее. Операционная система MS-DOS режимы DMA (и прямое

управление) не поддерживает. Для многозадачных ОС (Windows 9x/NT/2000, OS/2, Unix, Linux, NetWare...) драйверы могут входить в комплект поставки ОС или поставляться производителями системных плат (контроллеров ATA). И наконец, режим DMA должны поддерживать подключаемые устройства. Практически все современные устройства поддерживают режим Ultra DMA (или Multiword DMA), но если в паре с таким устройством к одному контроллеру подключено старое устройство, не поддерживающее этот режим, то прогрессивные режимы могут оказаться недоступными (по вине чипсета или драйвера) и для нового устройства.

ВНИМАНИЕ ----------

Режим UltraDMA привлекателен не только скоростью и разгрузкой процессора, но и контролем достоверности передач по шине ATA (правда, этот контроль корректно работает не со всеми драйверами).

Поскольку контроллеры ATA подключаются к 32-разрядной шине PCI, в них ввели возможность обращения к регистру данных ATA двойными словами. При этом за одну 32-битную операцию процессора и шины PCI по шине ATA последовательно передаются два 16-битных слова. Возможностью 32-разрядного доступа к регистру данных можно управлять через параметр IDE 32-bit Transfer (Enable/Disable) BIOS Setup.
Более сложные контроллеры (отдельные карты расширения) могут иметь собственные кэш-память и управляющий процессор. Они могут аппаратно поддерживать «зеркальные» диски и организовывать RAID-массивы ATA-дисков. Некоторые адаптеры позволяют соединять несколько физических дисков в один логический на уровне вызовов BIOS.
Адаптеры ATA одно время часто размещали на звуковых картах (для подключения CD-ROM). По умолчанию им назначают ресурсы канала 3 или 4. К этим каналам можно подключать винчестеры, но будет ли их там искать BIOS во время теста POST — вопрос. Современные версии BIOS позволяют хранить конфигурационные параметры четырех жестких дисков, более старые версии — двух. Четыре канала ATA физически позволяют подключить до восьми накопителей, но работа с ними лимитирована программными ограничениями.
В последнее время получили распространение переходные адаптеры, позволяющие подключать устройства ATA/ATAPI к шине USB. В случае шины USB 2.0 простота подключения внешнего устройства сочетается и с высокой скоростью передачи данных.

Контроллер PCI IDE Bus Master

Для шины PCI существует стандартный вариант интерфейса контроллера IDE (название «ATA» здесь не очень уместно, поскольку оно ориентировано на ISA-подобную шину). Спецификация «PCI IDE Controller Specification» появилась еще в 1994 году и описывала реализацию контроллера, совместимого с интерфейсом ATA. По сравнению с обычным контроллером у данного контроллера в блоке управляющих регистров имеется лишь один адрес (устаревший

регистр адреса устройства недоступен). Чуть позже была опубликована спецификация «Programming Interface for Bus Master IDE Controller», описывающая работу с устройствами в режиме DMA с прямым управлением шиной. Двухканальный контроллер является однофункциональным устройством PCI; четырехканальный контроллер представляется уже многофункциональным. Для контроллера определены два режима распределения ресурсов:

- в *режиме совместимости* (compatibility) каналам выделяются традиционные области адресов ввода-вывода и линии прерываний;
- в *естественном режиме PCI* (native-PCI) базовые адреса блоков регистров и линии прерывания задаются в регистрах конфигурационного пространства и могут произвольно перемещаться в любую область.

В режиме совместимости контроллер должен получить две линии запросов прерывания — IRQ15 и IRQ14; в естественном режиме PCI он как однофункциональное устройство PCI использует лишь одну линию запроса прерывания.

Если контроллер функционирует в режиме совместимости, то для ПО он «прозрачен» — его работа не отличается от работы традиционного контроллера ATA. Текущий режим и возможность его смены отражается в байте кода программного интерфейса, являющегося последним элементом идентификатора класса устройства.

Типовой контроллер в чипсетах большинства современных системных плат идентифицируется как PCI-устройство класса 01 подкласса 01 с кодом интерфейса 80h. Контроллер выглядит как расширение стандартного контроллера ATA, обеспечивающего доступ к регистрам устройств ATA/ATAPI по известным адресам. Расширение касается прямого управления шиной PCI, благодаря которому можно реализовать обмен данными с устройствами в режимах DMA. Контроллер позволяет использовать все доступные ему режимы обмена PIO (3,3-16,6 Мбайт/с), режимы DMA в стиле контроллера 8237A (2-16,6 Мбайт/с) и режимы UltraDMA (16,6-133 Мбайт/с). Все временные параметры настраиваются через конфигурационные регистры PCI, их состав может быть специфичным. Однако с этими регистрами должна иметь дело только процедура POST, устанавливая для каждого обнаруженного устройства ATA/ ATAPI оптимальные режимы PIO и DMA/UltraDMA. В эту настройку может вмешаться пользователь, установив какие-либо ограничивающие параметры в BIOS Setup. В итоге после конфигурирования (во время теста POST) при обращении программ к устройствам остается выбор лишь между PIO и DMA (если устройство поддерживает DMA). Для обмена в режиме PIO никаких специальных действий не требуется, программа просто выполняет чтение или запись в регистр данных командами REP INSW/OUTSW. Для обмена в режиме DMA требуется «зарядить» и запустить контроллер прямого управления, о чем и пойдет речь далее.

Двухканальный контроллер DMA имеет 16-байтный блок регистров, расположенный в пространстве портов ввода-вывода. Имена регистров имеют префикс BMI (Bus Master IDE) и окончание P (Primary) для первого канала, S (Secondary) — для второго, поэтому в описании будем использовать окончание «x» (любой канал).

- *Регистр команд BMICx*, доступный по чтению и записи, используется для запуска контроллера и задания направления передачи. Останов для текущего сеанса необратим (продолжение сеанса невозможно). Останов контроллера (сброс бита) обычно выполняют после выполнения команды ATA/ATAPI (по прерыванию). Преждевременный останов ведет к ошибке выполнения команды с соответствующим сообщением. Направление должно быть задано до запуска контроллера, изменять его «на ходу» нельзя.
- В *регистре состояния BMISx* содержится ряд признаков: активность канала, ошибка обмена по PCI, запрос прерывания от устройства, поддержка DMA устройствами 0 и 1, признак симплексного режима (каналы не могут работать одновременно).
- В *регистр BMIDPTx* заносят адрес таблицы дескрипторов областей физической памяти (Physical Region Descriptor, PRD), с которыми производится обмен данными. Контроллер способен при чтении памяти собрать поток данных из произвольного числа физических областей (gathered read), а при записи «разбросать» поток по этим областям (scatter write). Такая возможность встречалась еще в EISA-системах, она позволяет преодолевать барьеры на границах страниц, свойственные стандартным контроллерам DMA и страничной переадресации памяти. *Дескрипторы* в таблице PRD располагаются друг за другом, каждый дескриптор описывает физический адрес начала области, счетчик байтов и признак конца таблицы.

Каждая область может быть расположена в произвольном месте памяти (кроме памяти, отображаемой на шину ISA) и иметь произвольный размер, но не должна пересекать границы страниц размером 64 Кбайт. Это неестественное (для 32-разрядных систем) ограничение обусловлено тем, что в PCI IDE счетчики адреса — 16-разрядные, а старшие 16 бит адреса получаются от статических регистров страниц (как в EISA DMA). Таблица дескрипторов должна быть выровнена по границе двойного слова и тоже не должна пересекать границы страниц, имеющих размер 64 Кбайт. Число дескрипторов в таблице произвольно, последний должен содержать признак конца таблицы. Обмен начинается с области, описанной первым дескриптором; за ней идет область следующего дескриптора, и так далее до последнего. Контроллер остановится по исчерпании счетчика в последнем дескрипторе или по инициативе устройства, если в обмене должно участвовать меньшее число данных. Если устройству данных не хватит, оно укажет на ошибку при завершении команды.

Контроллер SATA Intel 31244

Микросхема Intel 31244 — один из первых контроллеров Serial ATA для шины PCI-X. Этот контроллер соответствует спецификации Serial ATA 1.0a. Контроллер может работать в одном из двух режимов:

- *Native PCI IDE*, он же *M/S* (Master/Slave) — режим совместимости со стандартным двухканальным контроллером PCI IDE, к каждому из каналов которого подключены два порта SATA. В этом режиме по сравнению со стандартным контроллером PCI IDE появляется новый блок регистров SATA

(см. 19.5). Все блоки регистров в режиме M/S отображены на пространство ввода-вывода, на их положение указывают регистры BAR конфигурационного пространства (используются 6 областей).

- *Direct Port Access (DPA)* — режим прямого подключения к устройству, естественный для SATA. Каждый порт SATA имеет свой контроллер DMA, что обеспечивает полную независимость устройств и возможность их одновременного использования. В этом режиме все регистры собраны в один блок и отображаются на пространство памяти, на начало блока указывает 64-бит- ный регистр BAR_10h.

Переключение режима *M/S-DPA* не может быть динамическим: после выбора режима необходима перезагрузка ОС для автоконфигурирования на PCI. Ресурсы (занимаемые адреса) контроллера в обоих режимах приведены в табл. 19.6.

Таблица 19.6. Ресурсы контроллера SATA i31244

BAR	Native PCI IDE	Direct Port Access
BAR_10h	Блок командных регистров 1-го канала	Базовый адрес блока регистров
BAR_14h	Блок управляющих регистров 1-го канала	Старшие байты адреса
BAR_18h	Блок командных регистров 2-го канала	
BAR_1Ch	Блок управляющих регистров 2-го канала	
BAR_20h	Регистры каналов DMA	
BAR_24h	Блок регистров SATA (SATA Superset Register)	

В *режиме M/S* контроллер с программной точки зрения выглядит почти так же, как и традиционный двухканальный контроллер PCI IDE (см. 19.4), с теми же форматами регистров, дескрипторов (PRD) и ограничениями (невозможность пересечения границ 64-килобайтных страниц). Отличие заключается в регистрах команд BMICx (в i31244 они называются SUICDCR0 и SUICDCR1), в которых появились два новых бита состояния: направление текущей операции FPDMA и активность FPDMA. Стандартные регистры SATA (SCR0-SCR3), в документации на контроллер называемые *SATA Superset Registers*, в режиме *M/S* располагаются в пространстве ввода-вывода.

В *режиме DPA* контроллер приобретает новый облик: каждый порт получает собственный набор регистров, включая регистры персонального контроллера DMA; все регистры располагаются в пространстве памяти и имеют разрядность 32 бита.

Общие регистры контроллера включают *регистры запросов и масок прерываний* SUPDIPR и SUPDIMR. В этих регистрах каждому порту отводится свой байт, в котором каждый бит отвечает за свое событие (запрос прерывания от устройства или особые ситуации в интерфейсе SATA).

Стандартные регистры ATA дополнительных комментариев не требуют. *Стандартные регистры PCI IDE* используются так же, как и в режиме M/S; для поддержки 64-битной адресации памяти они дополнены регистрами расширения адреса. *Стандартные регистры SATA* по назначению соответствуют регистрам SCR0-SCR3 спецификации SATA II. *Специальные регистры интерфейса* порта управляют физическим интерфейсом (например, расширением диапазона сиг

налов) и тестированием интерфейса. В исполнении команд обмена с устройствами они роли не играют.

Особого внимания заслуживают регистры, используемые для реализации режима FPDMA. Поскольку контроллер соответствует спецификации SATA 1.0a, полноценной поддержки NCQ (введенной в SATA II) он, к сожалению, не имеет. Для организации FPDMA контроллер предоставляет набор регистров, которые позволяют сформировать структуру *DMA Setup FIS*, посылаемую устройству. Эти регистры определяют 64-битный *идентификатор буфера DMA* — число, которое суммируется с базовым адресом для получения адреса таблицы PRD, используемой для данного обмена. В NCQ из этого идентификатора задействуется лишь 5-битный тег, расположенный в самых младших битах. Даже если устройство NCQ в ответной структуре FIS пошлет тег, команды с разными значениями тега не смогут ссылаться на разные таблицы PRD; следовательно, для запуска DMA потребуется программное вмешательство драйвера. Регистры позволяют задавать начальное смещение в буфере и счетчик байтов. Регистры обеспечивают возможность задания значения двух резервных полей в структуре FIS (слова 3 и 6), однако использование этих резервных полей в SATA II не определяется. Было бы интереснее иметь доступ к резервному байту слова 0, в котором передается идентификатор порта при использовании мультиплексора портов (см. 19.3), но этой возможности нет. Для структур *DMA Setup FIS,* приходящих от устройства, контроллер предоставляет возможность чтения элементов структур через соответствующие регистры.

Доставкой структур *DMA Setup FIS* к устройству управляет регистр SUPDDSFCSR, в нем задаются направление, признаки использования прерывания и разрешения на автоматическое выполнение режима FPDMA для команд с очередями, а также некоторые дополнительные функции.

Контроллер AHCI

Спецификация Serial ATA AHCI 1.0 Specification описывает интерфейс улучшенного хост-контроллера (Advanced Host Controller Interface, AHCI). Спецификация доступна на сайте фирмы Intel (http://developer.intel.com).

Спецификация AHCI относится к устройству PCI, являющемуся интерфейсом между системной памятью и устройствами SATA. Это устройство, в спецификации называемое *адаптером*[1] *хост-шины* (Host Bus Adapter, HBA), может поддерживать 1-32 порта SATA. К портам могут подключаться устройства с наборами команд ATA или ATAPI, с которыми поддерживаются режимы обмена PIO и DMA. Контроллер работает со структурами данных, расположенных в системной памяти; для памяти возможна 64-битная адресация. Структуры данных в памяти содержат общие поля для управления и сообщения состояния, а также списки команд и таблицы дескрипторов буферов данных для каждой команды. Эти структуры достаточны для того, чтобы по ним контроллер мог полностью управлять подключенными к нему устройствами SATA в плане

[1] По сложности это все же контроллер.

исполнения или команд. Поддержка списков команд (до 32 элементов на каждый порт) и NCQ через протокол FPDMA с очередями не является обязательной. Минимальная длина списка (по 1 элементу на порт) позволяет работать только с одиночными командами, без очередей. Длинный список позволяет существенно снизить нагрузку на центральный процессор при исполнении даже обычных команд (без очередей и даже с обменом в режиме PIO).

Контроллер AHCI является *устройством-функцией PCI*, при этом он может интегрироваться в чипсет ядра системной платы или подключаться к любой PCI- подобной шине (PCI, PCI-X, PCI Express, Hyper Transport...). Контроллер является мастером шины с мощным интеллектуальным механизмом DMA. Основное взаимодействие программ с устройствами в AHCI происходит через системную память, передача параметров команды однобайтными обращениями к регистрам устройства исключена.

Контроллер AHCI имеет следующие доступные программно *регистры*:

- Набор регистров в конфигурационном пространстве PCI (PCI-X) со всеми стандартными атрибутами.
- Набор регистров, отображенных на пространство памяти. В этот набор входят:
  - общие регистры управления контроллером;
  - специфические регистры;
  - управляющие регистры портов.

Структуры данных в системной памяти, связанные с каждым портом, включают:

- список команд (command list), состоящий из 1-32 слотов. Каждый слот (16-байт- ная структура) ссылается на *командную таблицу* (command table). За постоянной частью командной таблицы (размером 256 байт) размещается таблица PRD (переменного размера), ссылающаяся на произвольное (до 65 535) число буферов данных произвольного размера (до 4 Мбайт). Буферы могут описывать передачу суммарной длиной 4 Гбайт;
- буфер принимаемых структур FIS размером 256 байт.

Основой взаимодействия программы и устройства, подключенного к порту, является *список команд* — структура данных в памяти, определенная для каждого порта контроллера. Список состоит из последовательности *командных слотов* (command slot), в каждом из которых может размещаться любая команда ATA/ ATAPI. С каждым слотом связана своя *командная таблица*, содержащая FIS для подачи команды и таблицу дескрипторов буферов данных (PRD). Для команды PACKET в таблице присутствует и командный блок ATAPI, передаваемый устройству. В командном слоте указываются адрес командной таблицы, длина передаваемой структуры FIS, длина таблицы PRD, направление передачи, а также дополнительная управляющая информация. В слоте имеется поле для указания номера порта мультиплексора, к которому адресуется данная команда. Во время исполнения контроллер сообщает число успешно переданных дан

ных. Каждый дескриптор таблицы PRD задает базовый адрес и длину буфера, а также имеет бит I, позволяющий вызывать прерывания по отработке данного дескриптора.

С каждым портом контроллера связан свой *буфер принимаемых структур FIS*, в котором определены поля для структур *DMA Setup FIS* и *PIO Setup FIS*, для структур FIS регистров, а также для неизвестных типов структур FIS. В этих полях контроллер помещает копии всех приходящих структур FIS; по ним программа может следить за состоянием устройств.

Программа заполняет свободный слот списка команд для данного порта и информирует об этом контроллер установкой соответствующего бита в его регистре PxCI. Получив это уведомление, контроллер включает данный слот в свой план работы и, когда до него доходит очередь, считывает команду из памяти и передает ее устройству. Контроллер переходит к обслуживанию следующего заполненного слота только при условии равенства нулю битов состояния устройства (BSY = DRQ = ERR = 0). Это позволяет использовать список команд для любых команд: с очередями и без, с обменом FPDMA, DMA и PIO. Слот освобождается, когда для него контроллер получает структуру FIS, указывающую на обнуление битов BSY, DRQ и ERR, а для команд с очередями (NCQ) — на обнуление в регистре SAct бита, соответствующего данному слоту. Освобождение слота контроллер отмечает обнулением нужного бита в регистре PxCI. Для программы это означает доступность информации о состоянии и данных, полученных при чтении.

Контроллер обходит слоты по кругу. *Команда без очередей* не позволит контроллеру перейти к следующему слоту до ее завершения. Когда после получения команды устройству потребуется передача данных, оно пошлет контроллеру структуру *FIS DMA Setup* или *PIO Setup*, которая будет относиться к командной таблице текущего слота. Местоположение буферов данных в ОЗУ будет определяться таблицей PRD данной командной таблицы.

Команды с очередями (NCQ) сразу сбрасывают бит BSY, и контроллер переходит к следующему слоту. Когда для принятой команды устройству потребуется передача данных, оно пошлет структуру *FIS DMA Setup* с идентификатором буфера, содержащим тег команды. Тег должен совпадать с номером слота, из которого команда была запущена, — по нему контроллер считает требуемую таблицу PRD. Старые команды ATA с очередями, использующие биты REL, SERV и DRQ, контроллером AHCI не поддерживаются.

Команда сброса может посылаться в устройство при любом его состоянии.

Команды с обменом PIO контроллер исполняет с помощью своего контроллера DMA. Поддержка режима PIO может быть ограничена: контроллер может не поддерживать передач, в которых бит DRQ устанавливается несколько раз (передача идет несколькими блоками). Поддержка PIO с передачей одного блока обязательна — с ее помощью, например, передается блок параметров идентификации устройства.

Контроллер имеет специальную *поддержку устройств ATAPI*. Для команды PACKET в командной таблице имеется специальное поле командного пакета и признак команды ATAPI. Получив команду, устройство посылает структуру *PIO Setup FIS*, по которой контроллер передает устройству поле командного пакета (длину заказывает устройство). Кроме того, в регистре управления порта имеется бит ATAPI, отключающий управление общим индикатором активности устройств, подключенных к контроллеру (индикация ATAPI отличается от индикации ATA).

Для поддержки «*горячего» подключения* (необязательной) контроллер имеет средства обнаружения факта подключения устройства (cold presence detect) и сигнализации состояния замка блокировки съема устройства (interlock switch). Эти средства могут присутствовать не на всех портах.

Поддержка *разнесенного запуска* (staggered spin-up) устройств (необязательная) позволяет инициализировать интерфейсы портов не одновременно (по сбросу контроллера), а поочередно. Инициализация интерфейса порта вызывает запуск двигателя подключенного устройства; программное управление инициализацией позволяет уменьшить броски тока потребления при включении большого числа устройств.

Для *экономии энергопотребления* контроллер может иметь средства агрессивного управления режимом энергопотребления интерфейса (aggressive link power management). Если агрессивный режим включен, то когда кончается обработка всех запросов устройством и неактивны все слоты (обнуляются регистры PxSACT и PxCI), интерфейс порта переводится в режим *Slumber* (пониженное потребление).

Поддержка мультиплексоров портов (необязательная) в AHCI весьма ограничена, поскольку для каждого порта контроллера имеется только один набор регистров, отражающих состояние устройства. Это не позволяет использовать очереди команд одновременно для нескольких устройств, подключенных к порту через мультиплексор. Контроллер AHCI способен коммутировать информационные потоки только по командам (command based switching): командный слот адресует порт мультиплексора, к которому подключено целевое устройство. Команда должна выполняться целиком — устройство должно послать все связанные с ней структуры FIS до того, как следующей командой, возможно, будет проложен иной маршрут. Для команд, выполняемых без очередей, любой слот может использоваться для команды к любому порту мультиплексора. При заполнении слотов следует учитывать циклический порядок обслуживания слотов, чтобы обеспечить равномерность (или желаемые диспропорции) обращений к разным устройствам. Команды с очередями (NCQ) можно задействовать только для одного устройства, и пока все поставленные в очередь команды не будут обслужены, к другим устройствам команды подавать нельзя. Чтобы снять это ограничение, контроллер должен поддерживать коммутацию на основе FIS (FIS based switching) с учетом принятого поля PMP. Однако в AHCI 1.0 эта возможность не предусматривается. Обнаружение мультиплексора выполняется чисто программно: порт должен послать команду сброса в каждый порт, адресуясь к управляющему порту мультиплексора. Если мультиплексор имеется, он ответит соответствующей сигнатурой.

19.5. Программное взаимодействие с устройствами ATA/ATAPI и SATA

Общей задачей программы, управляющей устройством хранения (его драйвера), является подача команд устройству и передача блоков данных из устройства в системную память хоста или из памяти в устройство. В этом обмене всегда участвует центральный процессор хоста, посылающий команды в устройство и считывающий его состояние. В самой передаче данных, как правило, принимает участие контроллер DMA (хотя возможен и чисто программный обмен PIO).

Традиционный контроллер DMA (8237A) для работы с устройствами хранения малопригоден: его скорость передачи невысока; пересылаемый блок должен располагаться в непрерывной области физических адресов памяти, не пересекать границ 128-килобайтных страниц[1] и располагаться в первых 16 мегабайтах адресов ОЗУ.

Как уже отмечалось, современные контроллеры DMA имеют поддержку виртуальной памяти 32-разрядных процессоров: они способны при чтении памяти собрать поток данных из произвольного числа областей (gathered read), а при записи «разбросать» поток по этим областям (scatter write). Для них передача блока данных, расположенного в непрерывной области виртуальных (логических) адресов, разбивается на несколько передач в областях с непрерывными физическими адресами. При этом передача описывается *таблицей дескрипторов физических областей* (PRD). Иное название этой таблицы — *scatter-gather list*. Текущее состояние передачи определяется *контекстом DMA* — набором регистров, управляющих работой канала и задающих текущий адрес ячейки памяти для обмена данными, используя таблицу (или список) дескрипторов. Контроллер PCI IDE Bus Master, применяемый для современных параллельных интерфейсов ATA, имеет по одному контексту для каждого своего канала (шины ATA). Для эффективной многозадачной работы контроллер должен иметь множество контекстов, переключаемых аппаратно. Такие возможности имеют контроллеры SATA, из которых наиболее развитые — контроллеры с интерфейсом AHCI.

Протоколы взаимодействия определяют последовательность действий, которые должен выполнить драйвер для подачи команд и получения результатов. Протоколы зависят от выбора режима передачи данных (PIO, DMA, UltraDMA или FPDMA), а также от использования (или неиспользования) механизма перекрытия команд и очередей. Режим обмена PIO подходит только для однозадачных ОС; для многозадачных ОС (больший интерес представляет обмен по протоколу DMA и UltraDMA; наиболее перспективен режим FPDMA для устройств SATA-II. С точки зрения драйвера есть различия лишь между типами режимов; аппаратные нюансы режимов внутри типа влияют только на скорость передачи. Режим UltraDMA значительно отличается от обычного режима

[1] Это ограничение 16-битных каналов DMA связано с реализацией расширения адреса для 16-раз- рядного контроллера 8237A.

DMA необходимостью обработки возможных ошибок передачи по шине; в случае постоянных ошибок драйвер должен «понизить» режим UltraDMA (вплоть до перехода на традиционные режимы). Режим FPDMA отличается минимизацией программных действий драйвера и самой высокой эффективностью обмена.

В [8] подробно рассмотрены различные варианты протоколов программного взаимодействия:

- Традиционные протоколы (одиночные команды) работают с устройствами одной шины ATA поочередно без перекрытий (отключения устройств от шины) и очередей.
- В случае перекрытия команд и очередей устройства ATA/ATAPI могут освободить шину, если полученная команда требует длительных внутренних операций (например, позиционирования на носителе). Перекрытие команд (освобождение шины) допускается только для ограниченного (но достаточного) набора команд. Устройства могут поддерживать и *очереди команд*, но только для команд, допускающих перекрытие (эти свойства тесно связаны). Принятый механизм освобождения шины и продолжения выполнения команд малоэффективен (например, по сравнению с SCSI) — он требует привлечения хоста для обнаружения факта готовности устройства и переключений контекста. Готовность устройства определяется хостом путем полиига (периодического опроса состояния устройств), реализуемого хост-контроллером (аппаратно) или программой хост-компьютера.
- Технология естественных очередей в SATA-II (NCQ) позволяет существенно упростить протокол взаимодействия драйвера и устройств и повысить эффективность обмена данными. Для использования технологии NCQ хост-контроллер должен иметь контроллер FPDMA с аппаратным переключением контекста по сообщениям от устройства. Этой возможностью обладают далеко не все контроллеры SATA.
- При наличии интерфейса AHCI контроллер SATA (см. 19.4) обеспечивает реальную поддержку технологии NCQ, а также отработку обычных команд (без очередей) с минимизаций операций ввода-вывода, выполняемых процессором. Заметим, что технология NCQ в AHCI 1.0 полностью поддерживается только для случая одновременного и непосредственного (без мультиплексоров) подключения множества устройств к контроллеру. Программа в основном работает со структурами данных в памяти: командными таблицами и списками команд. Из регистров контроллера она оперативно пользуется только регистрами запросов прерываний, регистрами активности командных слотов (PxCI), активности запросов (PxSACT, лишь для команд NCQ) и регистрами с образами SR и ER (PxTFD).

Адресация блоков данных

В устройствах ATA/ATAPI минимальной адресуемой единицей данных является *логический блок*, как правило, размером 512 байт. Для традиционных дисковых устройств ATA блок является *сектором.* Для дисковых устройств применимы два способа адресации данных: трехмерная адресация (CHS) и линейная

адресация (LBA). Для иных устройств хранения, использующих интерфейс ATAPI, применима только линейная адресация.

Традиционная *адресация CHS* задает три координаты сектора устройства ATA:

- *Cylinder* — номер цилиндра, который задается регистрами CH и CL (старший и младший байты) и может принимать значение 0-65 535 (2^{16} - 1). Устройство может иметь до 65 536 (2^{16}) цилиндров (Cyl);
- *Head* — номер головки, который задается битами 3:0 регистра D/H и может принимать значение 0-15 (2^4 - 1). Устройство может иметь до 16 (2^4) головок в цилиндре (Heads Per Cylinder, HPC);
- *Sector* — номер сектора, который задается регистром SN и может принимать значение 1—255 (2^8 - 1), нулевой номер сектора не используется. Устройство может иметь до 255 (2^8 - 1) секторов на каждом треке (Sectors Per Track, SPT).

Здесь в координатах подразумевается *внешняя геометрия* — адреса, которые заносятся в командные регистры устройств. В системе CHS устройство ATA позволяет адресовать до 267 386 880 (65 536 x 16 x 255) секторов (блоков), что при размере сектора 512 байт дает 136 902 082 560 байт (около 137 Гбайт).

Линейная (логическая) адресация (Logical Block Addressing, LBA) гораздо проще — здесь адрес блока (сектора) определяется одним числом в диапазоне от 0 до N - 1, где N — емкость устройства хранения (в блоках). Предел адресуемого объема определяется разрядностью логического адреса и емкостью устройства:

- 28-битный адрес, изначально используемый для устройств ATA, позволяет адресовать до 2^{28} - 1 = 268 435 455 секторов, что при размере сектора 512 байт дает 137 438 952 960 байт (137,4 Гбайт, чуть больше, чем в CHS). Число секторов, доступных в режиме LBA, отражается в словах 61:60 блока идентификации устройства. По соглашению об адресации и сам линейный адрес LBA, и общее число адресуемых блоков N должны умещаться в 28 бит. Поскольку нумерация блоков начинается с нуля, номер блока 0F FF FF FF не используется;
- 32-битный адрес, используемый устройствами ATAPI, позволяет адресовать до 2^{32} - 1 = 4 294 967 295 секторов, что при размере сектора 512 байт дает
  2 199 023 255 040 байт (около 2,2 Тбайт). Доступная емкость устройства определяется посылкой пакета с SCSI-командой, соответствующей классу данного устройства;
- 48-битный расширенный адрес устройств ATA, введенный в ATA/ATAPI-6 и Serial ATA, позволяет адресовать до 2^{48} - 1 = 281 474 976 710 655 секторов — 144 115 188 075 854 848 байт (около 144 Пбайт). Число секторов, доступных в этом режиме (назовем его LBA-48), отражается в словах 103:100 блока идентификации устройства.

Попытки обращения к сектору с номером LBA, превышающим максимально возможный, или же с координатами C, H, S не вписывающимися в текущую геометрию, приведут к генерации ошибки IDNF или ABRT (обмена данными, естественно, не будет).

С устройствами ATAPI работа возможна только по линейному адресу LBA. Использование 32-битного адреса оказалось временным решением проблемы предела адресации для жестких дисков — существуют модели винчестеров, идентифицирующие себя как устройства ATAPI.

С устройствами ATA можно работать в разных режимах (CHS, LBA, LBA-48), но, естественно, с оглядкой на предел адресации выбранного режима. Возможность адресации LBA была заложена в самой первой спецификации ATA, стандарты ATA и ATA-2 допускают поддержку обоих режимов адресации (CHS и LBA). В стандартах, начиная с ATA-3, поддержка LBA для всех устройств обязательна. Для устройств ATA емкостью более 137,4 Гбайт обязательна поддержка LBA-48.

Для устройств ATA, поддерживающих оба режима адресации (CHS и LBA), конкретный режим определяется для каждой команды битом L (бит 6) регистра D/H; режимы могут чередоваться произвольным образом. Режим LBA-48 используется только с новыми командами обмена, что и позволяет отличать его от всех других.

Регистры устройств ATA

Каждое устройство ATA (и ATAPI) имеет стандартный набор регистров (табл. 19.7), состоящий из двух блоков:

- *регистры управляющего блока* (выбираются сигналом CS0#) используется для подачи сигналов сброса и разрешения прерываний от устройства, а также для получения информации о его состоянии;
- *регистры командного блока* (выбираются сигналом CS1#) служит для посылки команд устройству, чтения информации о его состоянии и передачи данных (через этот блок выполняется основное программное взаимодействие с устройством).

Таблица 19.7. Регистры контроллеров устройств ATA

Имя	Назначение (R — чтение, W — запись)	Адрес для канала №	
		1	2
	Control Block Registers — регистры управляющего блока	3FX	37X
AS	R: Alternate Status — альтернативный регистр состояния	3F6	376
DC	W: Device Control — регистр управления устройством	3F6	376
DA	R: Drive Address — регистр адреса (не используется)[1]	3F7	377
	Command Block Registers — регистры командного блока	1FX	17X
DR	R/W: Data — регистр данных	1F0h	170
ER	R: Error — регистр ошибок	1F1h	171
FR	W: Features — регистр свойств	1F1h	171

Имя	Назначение (R — чтение, W — запись)	Адрес для канала №	
		1	2
SC	R/W: Sector Count – регистр счетчика секторов	1F2h	172
SN (LBA_Low)	R/W: Sector Number – регистр номера сектора/LBA[7:0][2]	1F3h	173
CL (LBA_Mid)	R/W: Cylinder Low – регистр младшего байта номера цилиндра/LBA[15:8][2]	1F4h	174
CH (LBA_Hi)	R/W: Cylinder High – регистр старшего байта номера цилиндра/LBA[23:16][2]	1F5h	175
D/H (Device)	R/W: Device/Head – регистр номера устройства и головки/LBA[27:24][2]	1F6h	176
SR	R: Status – регистр состояния	1F7h	177
CR	W: Command – регистр команд	1F7h	177

[1] Рекомендуется, чтобы на сигнал чтения по этому адресу устройство не отвечало.

[2] Регистры сектора, цилиндра и головки в режиме LBA содержат указанные биты логического адреса.

В параллельном интерфейсе ATA блок регистров выбирается сигналами CS0# и CS1# от хост-адаптера, из которых активным (низкий уровень, «0») может быть только один (или ни одного). Внутри блока регистр адресуется сигналами DA2, DA1, DA0 (младшие биты адреса на системной шине), чтение и запись выполняются по сигналам DIOR# и DIOW#. В режиме DMA обмен данными происходит через *порт данных* (он в пространстве ввода-вывода не адресуется), при этом активны сигналы DMARQ (запрос) и DMACK# (подтверждение обращения к порту данных), а сигналы CS0# и CS1# неактивны. При обращении к регистрам сигнал DMACK# должен быть неактивным. В SATA программа работает с теневыми регистрами устройств, расположенными в хост-контроллере. Обмен данными между реальными и теневыми регистрами устройств, а также пересылка данных по DMA осуществляется передачей по последовательному интерфейсу. С программной точки зрения эти различия интерфейсов незаметны (и несущественны).

В табл. 19.7 приведены адреса регистров в пространстве ввода-вывода IBM PC-совместимого ПК для первичного (primary) и вторичного (secondary) каналов ATA. Заметим, что программное обращение хоста к любому регистру адресуется одновременно к обоим устройствам, подключенным к одной шине ATA. При этом реагировать на программное обращение будет только устройство, выбранное битом DEV регистра D/H.

Чтение регистров командного блока и альтернативного регистра состояния должно производиться только при нулевом значении бита BSY регистра состояния (это указывает на действительность и доступность их содержимого). *Запись в регистры* должна производиться лишь при BSY = 0 и DRQ = 0, кроме особо оговоренных случаев. Если устройство поддерживает управление энергопотреблением, в «спящем» режиме содержимое этих регистров недействительно и запись игнорируется, кроме особо оговоренных случаев.

Регистры, задающие адрес блока (SN, CH, CL, D/H), инициализируются *хостом* и сохраняют значение в случае успешного выполнения команды. В случае воз

никновения ошибки *устройство* помещает в них адрес, по которому случилась ошибка. До принятия спецификации ATA-2 считалось, что эти регистры должны модифицироваться и после успешного выполнения операции, отражая текущее значение адреса в носителе.

Форматы регистров устройств ATA приведены на рис. 19.6, назначение полей поясняется далее. Для устройств ATAPI те же регистры используются иначе, подробности см. в [8]. В ATA/ATAPI-6 регистры SN, CL и CH получили новое название — LBA Low, LBA Mid и LBA High, а регистр D/H назван Device. В данной книге в основном используются более привычные и короткие старые названия.

Регистр	7	6	5	4	3	2	1	0
DC	HOB					SRST	nIEN	0
DA	HiZ	nWTG	nHS[3:0]				nDS[1:0]	
ER		UNC	MC	IDNF	MCR	ABRT	TK0NF	AMNF
D/H	FUA	L	(1)	DEV	H[3:0] / LBA[27:24]			
SR	BSY	DRDY	DF	DSC	DRQ	CORR	IDX	ERR

Рис. 19.6. Регистры устройств ATA

В *режиме LBA-48* регистры CH, CL, SN и SC (или FR для команд с очередями) для устройств становятся *двойными* — они содержат 16-битные числа. Младшим байтом этих чисел является последнее значение, записанное в данный регистр, старшим — предыдущее записанное. Таким образом, загрузка 16-разрядных значений выполняется записью пар 8-разрядных значений. При выполнении команд с обычной адресацией (CHS или LBA-28) старшие байты этих регистров игнорируются (считаются нулевыми). Программное считывание данных регистров возвращает последнее записанное значение. Для чтения предыдущих значений (старших байтов) необходимо установить в единицу бит HOB регистра DC. Бит HOB автоматически сбрасывается при любой записи в блок командных регистров.

Альтернативный регистр состояния AS (для первого канала — адрес 3F6h, для второго — 376h) имеет те же биты, что и в основном регистре состояния (см. далее), но его чтение не приводит к каким-либо изменениям состояния устройства.

Регистр управления устройством DC (3F6h, 376h) служит для программного сброса обоих устройств одновременно и управления разрешением прерывания выбранного устройства. Запись в этот регистр возможна в любой момент. Программный сброс через регистр DC должен отрабатываться и в состоянии *Sleep*.

Назначение битов регистра DC:

- HOB (High Order Bits) — указание на считывание старших байтов (предыдущих значений) регистров CH, CL, SN, SC и FR при использовании 48-битной адресации (бит автоматически сбрасывается при любой записи в блок командных регистров);
- SRST (Software Reset) — программный сброс, действует все время, пока бит не сброшен (оба устройства на шине воспринимают программный сброс одновременно);

- nIEN (Interrupt Enable) — инверсный бит разрешения прерывания (при нулевом значении бита выбранное устройство может вырабатывать сигнал INTRQ через тристабильный выход).

Регистр адреса устройства DA (3F7h, 377h) использовался только в первой версии ATA для совместимости со старыми контроллерами. Регистр не входит в блок (он совпадает с диагностическим регистром состояния контроллера НГМД), поэтому рекомендуется, чтобы устройство ATA не отвечало на чтение этого регистра. Из-за несоблюдения этого требования могут возникать проблемы, если контроллер (адаптер) ATA и контроллер НГМД находятся на разных платах.

Регистр данных DR (1F0h, 170h) может использоваться как 8-битный или 16-бит- ный в зависимости от типа данных, передаваемых в текущей команде. Обращение к этому регистру происходит в режиме обмена PIO (когда сигнал DMACK# неактивен), при выполнении передач протокола PO (PIO Out) хост производит запись в этот регистр, при PI (PIO In) — чтение.

Регистр ошибок ER (1Flh, 17lh) хранит состояние выполнения последней операции или диагностический код. После завершения операции на наличие ошибки указывает бит ERR регистра состояния.

Назначение битов регистра ER:

- UNC (Uncorrectable Data Error) — неисправимая ошибка данных;
- MC (Media Changed) — смена носителя (после смены носителя первая команда обращения отвергается и устанавливается данный бит, после сброса бита следующие команды выполняются нормальным образом);
- IDNF (ID Not Found) — не найден идентификатор сектора;
- MCR (Media Change Requested) — индикатор запроса смены носителя (после обнаружения запроса смены носителя команды Door Lock возвращают бит ошибки ERR и бит MCR, причем бит MCR сбрасывается командами Door Unlock, Media Eject или сигналом аппаратного сброса);
- ABRT (Aborted Command) — команда отвергнута как недействительная или возникла иная ошибка;
- TK0NF (Track 0 Not Found) — по команде Recalibrate не удалось найти нулевой трек;
- AMNF (Address Mark Not Found) — не найден адресный маркер данных в заголовке сектора.

После выполнения любого сброса или команды Execute Device Diagnostic регистр ошибок содержит *диагностический код*. Трактовка битов, за исключением бита 2 (ABRT), может меняться в зависимости от исполненной команды.

Назначение регистра *свойств* FR (1F1h, 171h) зависит от команды. В случае команды Set Features через него задается код подкоманды. В случае команд чтения и записи с очередями в этом регистре задается число секторов для передачи (в LBA-48 он становится 16-разрядным). Некоторые старые устройства могут игнорировать запись в этот регистр. До принятия спецификации ATA-2

в него помещали значение рекомендуемого номера цилиндра для предварительной компенсации при записи.

Регистр счетчика секторов SC (1F2h, 172h) содержит число секторов, участвующих в обмене (кроме команд чтения и записи с очередями). Хост инициализирует этот регистр до подачи команды (нулевое значение соответствует 256 секторам). По успешном завершении операции доступа к данным регистр должен обнуляться. Если команда завершилась с ошибкой, в регистре оказывается число секторов, которые должны быть переданы для успешного завершения предыдущего запроса. Команды Initialize Device Parameters и Write Same могут переопределить значение регистра. В некоторых командах он используется для передачи иных параметров (см. 19.6). Для режима LBA-48 регистр стал 16-разрядным, что позволяет передавать одной командой до 65 536 секторов; старший байт этого регистра (SC_Exp) используется только в командах с суффиксом *Ext.*

Регистры номера сектора SN (lF3h, 173h) и *номера цилиндра* — младшего CL (1F4h, 174h) и старшего CH (1F5h, 175h) байтов — имеют двоякое назначение в зависимости от выбранной системы адресации (CHS или LBA). Они инициализируются хост-адаптером, а в случае ошибки при выполнении операции устройство помещает в них адрес, по которому встретилась ошибка. Для режима LBA-48 регистры стали 16-разрядным, старшие байты этих регистров (SN_Exp, CL_Exp и CH_Exp) используются только в командах с суффиксом *Ext.*

Регистр номера устройства и головки D/H (1F6h, 176h) служит не только для хранения части адресной информации, но и для выбора ведущего или ведомого устройства и метода адресации. С переходом на адресацию LBA-48 и SATA с отказом от модели ведущий-ведомый регистр теряет первоначальное назначение.

Назначение битов регистра D/H:

- FUA — управление надежностью и кэшированием операции (в SATA II);
- L — единичным значением указывает на применение режима адресации LBA, при нулевом значении бита используется режим CHS;
- DEV (Device) — выбор устройства (DEV = 0 — ведущее, DEV = 1 — ведомое);
- H[3:0] — номер головки в режиме CHS (при L = 0), LBA[27:24] — старшие биты логического адреса в режиме LBA (при L = 1);
- биты 7 и 5 вплоть до ATA-3 должны были быть единичными, в ATA/ ATAPI-4 их объявили устаревшими; в SATA II бит 7 заняли под флаг FUA.

Регистр состояния SR (1F7h, 177h) отражает текущее состояние устройства в процессе выполнения команд (занятость, готовность, наличие ошибок и др.). Чтение регистра состояния разрешает дальнейшее изменение его битов и сбрасывает запрос аппаратного прерывания.

Назначение битов регистра SR описано ниже:

- BSY (Busy) — устройство занято. Значение этого бита действительно всегда. При BSY = 1 устройство игнорирует попытки записи в командный блок регистров, а чтение этих регистров дает неопределенный результат. При BSY = 0 регистры командного блока доступны, в это время устройство не может ус

танавливать бит DRQ, изменять значение битов ERR и содержимое остальных командных регистров (могут меняться только значения битов IDX, DRDY, DF, DSC и CORR). Бит может устанавливаться на кратковременный интервал, так что хост может этого не заметить. Бит устанавливается:

- при сбросе устройства;
- по получении команды, если не устанавливается DRQ DRQ;
- между передачами блоков данных в режиме PIO и после них, пока не обнулен бит DRQ;
- во время передач данных в режиме DMA.

- DRDY (Device Ready) — устройство готово к восприятию любых кодов команд. Если состояние бита изменилось, оно не может вернуться обратно до чтения регистра состояния. При DRDY = 0 устройство воспринимает только команды Execute Device Diagnostic и Initialize Device Parameters, прекращая выполнение текущей команды и сообщая об этом флагом ABRT в регистре ошибок и флагом ERR в регистре состояния. Другие команды приводят к непредсказуемым результатам. Устройства ATAPI обнуляют бит по любому сбросу и команде Execute Device Diagnostic. Бит устанавливается устройством ATA, когда оно готово к выполнению всех команд. Устройство ATAPI устанавливает бит до завершения выполнения команд, за исключением команд Device Reset и Execute Device Diagnostic.
- DF (Device Fault) — индикатор отказа устройства.
- DSC (Device Seek Complete) — индикатор завершения поиска трека. В командах, допускающих перекрытие, бит называется SERV (Service Required) — устройство требует обслуживания.
- DRQ (Data Request) — индикатор готовности к обмену словом или байтом данных.
- CORR (Corrected Data Error) — индикатор исправленной ошибки данных.
- IDX (Index) — индекс, трактуется особо каждым производителем.
- ERR (Error) — индикатор ошибки выполнения предыдущей операции. Дополнительная информация содержится в регистре ошибок. Если установлен бит ERR, до приема следующей команды, программного или аппаратного сброса устройство не изменяет состояние этого бита, регистра ошибок, регистра количества секторов и регистров цилиндра, головки и номера сектора. Для команд Packet и Service бит называется CHK и служит признаком исключительной ситуации.

В стандарте ATA/ATAPI-4 для некоторых команд биты 4 и 5 могут иметь иное назначение, а биты 1 и 2 объявлены устаревшими.

Назначение регистра команд CR (1F7h, 177h) очевидно из названия. Устройство начинает исполнять команду сразу, как только ее код оказывается записанным в данный регистр (к этому моменту должны быть заданы все параметры команды). Команду можно записывать только лишь при готовности устройства (BSY =

- 0 и DRQ = 0); команду Device Reset для устройств ATAPI можно подавать неза

висимо от состояния битов BSY и DRQ, даже в состоянии *Sleep*. Список команд ATA приведен в 19.6.

Регистры Serial ATA

Регистры, расположенные в устройствах SATA, имеют свои так называемые теневые образы в хост-контроллере. Программное взаимодействие (чтение и запись) осуществляется с теневыми регистрами; связь теневых регистров с регистрами устройств обеспечивается кадрами, передаваемыми по последовательному интерфейсу SATA. Каждое устройство, подключенное к адаптеру Serial ATA, представляется тремя блоками регистров: управляющих, командных и SCR.

Первые два блока соответствуют одноименным блокам регистров ATA (см. раздел «Регистры устройств ATA»), блок регистров SCR — новый. Привязка адресов блоков регистров к адресному пространству хоста стандартом не регламентируется. В режиме эмуляции пар «ведущий-ведомый» (M/S) теневые регистры традиционных блоков располагаются по стандартным адресам ввода-вывода для ATA; блоки регистров SCR отображаются на пространство памяти. В режиме прямого подключения (DPP) все блоки регистров отображаются на память, каждому устройству (порту контроллера) отводится 512-байтный блок. Пример расположения регистров в пространстве памяти для контроллера i31244 (PCI-X SerialATA Controller) приведен в 19.4. В интерфейсе AHCI блоки командных и управляющих регистров упразднены, взаимодействие с регистрами устройств осуществляется через их образы в памяти.

В *блоке управляющих регистров,* как и в ATA, используется лишь один (AS для чтения, DC для записи). В *блоке командных: регистров* 8-битные регистры SC, SN, CL и CH позволяют задавать 16-битные значения. Их старшие байты задействуются только в командах с расширенной (48-битной) адресацией, назначение младших байтов сохранилось. В регистре D/H бит DEV требуется только при эмуляции пар устройств на одном канале, в «естественном» режиме он игнорируется.

Новый *блок регистров SCR* (Serial ATA Status and Control Registers) состоит из 16 смежных 32-разрядных регистров SCR0-SCR15, из которых пока определены лишь 5 (остальные зарезервированы).

Регистр SStatus (SCR0) — регистр текущего состояния интерфейса хост-адаптера, по которому можно определить, обнаружено ли устройство, согласована ли скорость и какая, в каком состоянии потребления находится интерфейс.

Регистр SError (SCR1) — регистр диагностической информации, относящейся к интерфейсу. В регистре представлены ошибки, накапливающиеся с момента последней очистки регистра.

Регистр SControl (SCR2) — регистр управления интерфейсом (инициализация, ограничение скорости, смена состояния энергопотребления).

Регистр SActive (SCR3) — регистр активности запросов (введен в SATA-II). Каждый бит отображает активность запроса с соответствующим тегом (бит 0 — tag = 0, бит 31 — tag = 1Fh). Драйвер устанавливает бит при постановке запроса в очередь, устройство сбрасывает его, завершив передачи данных, указанные

для данного запроса. Хост-контроллер сбрасывает все биты регистра по аппаратному и программному сбросу.
Регистр SNotification (SCR4) — регистр уведомления об асинхронных событиях (введен в SATA-II). Поле Notify используется для идентификации источника — устройств, сигнализирующих о событиях. Необходимость такого поля возникает при подключении множества устройств к одному порту через мультиплексор портов.

19.6. Система команд ATA/ATAPI и SATA

Стандарты ATA/ATAPI и SATA задают систему команд, передаваемых от хоста к устройству по параллельному (ATA, PATA) или последовательному (SATA) интерфейсу. Система команд, изначально ориентированная на фиксированные накопители на жестких магнитных дисках (НЖМД), впоследствии «обросла» различными дополнениями для поддержки смены носителей и иных типов носителей (CFA и малогабаритные флэш-карты). Благодаря всего лишь одной команде Packet система команд ATA получила возможность почти неограниченного расширения (ATAPI) за счет передачи командного пакета SCSI по интерфейсу ATA/SATA.

Команды доступа к данным ATA

Команды доступа к данным ATA предназначены для чтения и записи в устройствах хранения, логически соответствующих модели жесткого диска с его регистрами и посекторной адресацией (с размером сектора 512 байт). Для устройств иных классов, как уже отмечалось, имеется альтернативный способ доставки команд и параметров — команда Packet.
Команды чтения секторов в режимах обмена PIO и DMA позволяют считывать последовательно расположенные секторы, количество которых задано в регистре SC, а адрес начального сектора — в регистрах CH, CL, D/H и SN.
Команда верификации выполняет проверку возможности чтения, но не передает данные от устройства.
Команды записи секторов имеют версии с повторами (в случае ошибки) и без повторов. В команде записи с верификацией для каждого сектора после записи выполняется контрольное считывание. Для логической инициализации (очистки области) дисков имеется команда записи Write Same, которая позволяет содержимое 512 байтов, принятых от хоста, записать в группу секторов.
Команды блочного обмена (чтения и записи) отличаются от обычных команд (с обменом PIO) тем, что запросы прерывания вырабатываются не на каждый сектор, а на блок секторов. За счет сокращения числа прерываний, которые должен обслужить процессор, блочный режим в многозадачной системе позволяет повысить производительность на 30 %. Производительность обмена зависит от размера блока. Размер, оптимальный для устройства, может не быть оптимальным для ОС.

Команды с расширенной адресацией (с суффиксом Ext) используют двойные командные регистры CH, CL, SN и SC, в которых размещается линейный 48-битный адрес (см. 19.5). В этих командах регистр D/H требуется только для задания номера устройства.

Команды, выполняемые с *постановкой в очередь*, должны сопровождаться командой Service, которая служит для определения тега исполненной команды. В этих командах регистры FR и SC используются по-своему (см. 19.5).

Команды, выполняемые с *очередями NCQ* (эти команды введены в SATA II), опираются только на адресацию LBA-48. Они не требуют дополнительных команд для продолжения обслуживания, но до их подачи нужно инициализировать контексты DMA для используемых тегов. В этих командах специфично применение регистров FR и SC, а обработка ошибок выполняется через чтение журнала. В SATA II нет указаний на неприменимость данных команд к устройствам с пакетным интерфейсом (все вышеперечисленные команды применимы только для «непакетных» устройств).

Вспомогательное назначение имеют *команды поиска*. По команде поиска Seek устройство устанавливает головки на заданный цилиндр/трек и считывает идентификатор сектора.

Пакетный интерфейс ATAPI

Для подключения к интерфейсу ATA накопителей CD-ROM и стримеров (а также других устройств) недостаточно набора регистров и системы команд ATA. Для них существует аппаратно-программный интерфейс ATAPI (ATA Package Interface — пакетный интерфейс ATA). Устройство ATAPI поддерживает минимальный набор команд ATA, который неограниченно расширяется 16-байтным *командным пакетом.* Командный пакет посылается хостом в регистр данных устройства (в режиме PIO) по команде Packet. Структура командного пакета (блок дескрипторов) происходит от SCSI, об этом говорит схожесть драйверов для устройств с интерфейсами SCSI и ATAPI. При любой длине блока дескрипторов, которая определяется кодом команды (нулевым байтом пакета), передаваемый пакет имеет длину 16 байтов, но используется только указанное количество байтов. Классификация устройств совпадает с принятой в SCSI (см. 20.2), класс устройства сообщается им в начале блока параметров идентификации. Систему команд и структуру пакетов стандарт ATA/ATAPI не описывает, но для каждого класса устройств в SCSI определен стандартизованный набор команд с фиксированной структурой пакетов.

Интерфейс ATAPI может использоваться со стандартными адаптерами и контроллерами ATA и SATA — никаких модификаций аппаратного протокола (и транспортного для SATA) не требуется[1]. Для традиционного хост-адаптера

[1] В первое время после появления устройств ATAPI возникали проблемы их совместимости со сложными (по тем временам) контроллерами ATA. Эти контроллеры, имеющие собственный процессор, память и встроенное ПО (firmware), могли «не догадываться», что в регистр данных устройства помимо 512-байтных блоков данных можно записывать 16-байтный блок с командным пакетом.

поддержка ATAPI осуществляется чисто программно, в AHCI предусмотрен механизм доставки пакета с использованием контроллера DMA.

Инициализация, идентификация и конфигурирование устройств

Устройства ATA воспринимают три вида сброса в исходное состояние:

- *сброс по включению питания* (power on reset) обеспечивает самотестирование, запуск двигателя, проверку механики, установку параметров умолчания, сброс интерфейса и регистров в исходное состояние;
- *аппаратный сброс* (hardware reset), выполняемый по сигналу RESET#, обеспечивает самотестирование, установку параметров умолчания, сброс интерфейса в исходное состояние;
- *программный сброс* (software reset), выполняемый по установке бита SRST регистра DC, обеспечивает сброс интерфейса в исходное состояние.

Помимо этих трех видов сброса для сброса интерфейса *устройств ATAPI* в исходное состояние предназначена команда Device Reset.

Устройства SATA выполняют сброс по сигналу *COMRESET*, который может появиться как по инициативе хоста (перечисленные выше условия сброса), так и из-за событий на интерфейсе. Событиями, вызывающими сброс, являются подключение-отключение интерфейса и просто потеря сигнала. Для того чтобы устройство SATA по сигналу *COMRESET*, о котором хост может «не знать», «не забывало» изменения конфигурации, программно выполненные хостом, в SATA II есть специальное свойство (feature) сохранения программных параметров (см. далее).

После любого сброса или выполнения команды диагностики устройство в блоке командных регистров содержит *сигнатуру*, определяющую его тип:

- SC = 01h, SN = 01h, CL = 00h, CH = 00h, DH = 00h — устройства ATA;
- SC = 01h, SN = 01h, CL = 14h, CH = EBh, DH = 00h или 10h — устройства ATAPI (значение DH = 10h появляется после выполнения команды Device Reset устройством 1);
- SC = 01h, SN = 01h, CL = 69h, CH = 96h, DH = 00h — мультиплексор портов SATA;
- SC = 01h, SN = 01h, CL = 3Ch, CH = C3h, DH = 00h — мост SEMB с подключенным процессором SEP (SATA II);
- SC = FFh, SN = FFh, CL = FFh, CH = FFh, DH = FFh — мост SEMB с неподключенным процессором SEP (SATA II).

В системе команд имеются команды идентификации свойств и управления свойствами устройств, дающие возможность, в частности, обновлять встроенное ПО.

Команда идентификации Identify Device позволяет считать из контроллера блок из 256 слов, характеризующих устройство ATA (паспорт диска). Аналогичная команда Identify Packet Device предназначена для устройств ATAPI. Блок пара

метров может храниться как в энергонезависимой памяти устройства, так и на самом носителе в месте, недоступном для обычных обращений.

Команда установки параметров задает режим трансляции геометрии в системе CHS.

Команда установки свойств определяет особенности поведения устройств (в том числе и доступные режимы обмена и возможности SATA). После включения питания или аппаратного сброса установленные свойства заменяются принятыми по умолчанию, однако результат установки свойств можно и закрепить, подав специальную подкоманду (а можно и вернуться к заводским установкам).

Для управления конфигурацией устройства в ATA/ATAPI-6 введен новый набор свойств — *оверлей конфигурации устройства* (device configuration overlay). Оверлей представляет собой 512-байтную структуру, в которой описаны наиболее употребимые свойства устройства: поддерживаемые режимы Multiword DMA и Ultra DMA, максимальный LBA-адрес, поддерживаемые команды и свойства (S.M.A.R.T., защита, очереди, управление шумом, 48-битная адресация).

Команда задания параметров блочного режима передачи через регистр SC указывает число секторов, передаваемых с одним запросом прерывания.

Команда диагностики Execute Device Diagnostic, адресуясь всегда к ведущему устройству, выполняется одновременно обоими устройствами. О ее результате ведомое устройство сообщает ведущему (сигналом PDIAG#). Состояние обоих устройств (нормально, неисправно или отсутствует) определяется по *диагностическому коду*, который считывается из регистра ошибок ведущего устройства.

Фиктивная команда Nop, не изменяя содержимого регистров, позволяет считать информацию о состоянии устройства; специальная подкоманда сбрасывает всю оставшуюся очередь.

Команда загрузки микрокода Download Microcode позволяет модифицировать встроенное (firmware) ПО устройства. В зависимости от кода в регистре свойств загруженный микрокод будет действовать до выключения питания или постоянно. Загрузка некорректного микрокода может привести к выходу устройства из строя.

Журналы ошибок и событий

Журналы (log) ошибок и событий — это структуры данных о состоянии устройства, сохраняемые на носителе устройства в специальной зоне (не пользовательской). Они доступны в виде последовательности секторов. Журналы появились в спецификации ATA/ATAPI-5 первоначально для системы мониторинга состояния (S.M.A.R.T., см. далее). В ATA/ATAPI-6 введены общие команды чтения и записи различных журналов, а в SATA II специальные журналы определены для обработки ошибок при работе с NCQ и для счетчиков событий физического интерфейса. В стандартах описаны структуры данных нескольких журналов. Стандартные журналы со стороны хоста допускают только чтение, их заполняет устройство. Ряд номеров журналов отведен для специфических

журналов, заполняемых хостом (по усмотрению разработчика средств хоста) и устройством (по усмотрению разработчика устройств).

Мониторинг состояния — S.M.A.R.T

Для предупреждения о возможном отказе устройства служит технология S.M.A.R.T. (Self-Monitoring, Analysis and Reporting Technology — технология самонаблюдения, анализа и сообщений). Предсказуемые отказы (predictable failure) появляются в результате выхода параметров за некоторый порог. Отслеживаемые параметры: время разгона до заданной скорости, время позиционирования, процент ошибок позиционирования, «высота полета» головок, производительность (зависящая от числа повторов), количество использованных резервных секторов и др. Мониторинг может осуществляться двояко: в рабочем режиме (*on-line*) — одновременно с выполнением команд хоста (при возможном некотором замедлении), в автономном режиме (*off-line*) — в паузе между «полезными» командами, не снижая производительности. Если во время выполнения этой процедуры приходит внешняя команда, мониторинг прерывается на время исполнения команды, причем начало исполнения команды может задержаться на время до двух секунд. Значения отслеживаемых атрибутов сохраняются в служебной области носителя.

Для непакетных устройств имеется команда SMART, подкоманды которой задаются через регистр свойств (пакетные устройства используют для этих целей собственный протокол).

Работа со сменными носителями

Для накопителей со сменными носителями требуются специальные меры, предотвращающие:

- потерю данных при смене носителя (страхуют от попытки записи по каталогу прежнего носителя);
- потерю данных при изъятии носителя, когда кэш записи еще не выгружен;
- несанкционированную смену носителя.

Для накопителей со сменными носителями в ATA-2 предназначены команды загрузки и выгрузки, подтверждения смены носителя, блокировки и разблокировки устройства. Их реализация специфична для каждой модели устройства.

В ATA-4 принят иной набор команд для сменных носителей. Здесь может работать один из двух механизмов защиты от несанкционированной смены носителя, по переводу названий которых трудно судить об их реальном назначении:

- Механизм *уведомления о состоянии сменного носителя* (removable media status notification) применяется и к устройствам ATA, и к устройствам ATAPI. Этот механизм является предпочтительным, он дает хосту возможность полного управления носителем. Нажатие кнопки на накопителе смены носителя не вызывает, этот факт, а также состояние носителя могут определяться по команде Get Media Status, которую хост должен периодически посылать устройству. Смену носителя для устройств ATA вызывает только

команда Media Eject. Эта команда применима лишь к устройствам ATA, для устройств ATAPI предназначена пакетная команда Start/Stop Unit.

- Механизм *набора свойств сменного носителя* (removable media feature set) относится только к «непакетным» устройствам. Этот механизм позволяет устройству непосредственно отрабатывать нажатие кнопки смены носителя (если носитель не заблокирован), менять носитель и определять его состояние. Смена носителя блокируется командой Media Lock и разрешается командой Media Unlock; эти же команды используются для определения состояния чтением регистра ER.

В любой момент в устройстве может быть реализован только один из этих механизмов.

Поддержка флэш-памяти и компактных карт

Для запоминающих устройств на флэш-памяти в ATA-4 ввели команды, начинающиеся с аббревиатуры CFA (Compact Flash Association — ассоциация производителей компактных флэш-карт). Специфика этих устройств заключается в записи: запись обеспечивается лишь в предварительно стертые ячейки (секторы), хотя есть устройства, автоматически осуществляющие стирание при записи. Операция записи выполняется существенно медленнее чтения, скорость которого приближается к скорости обращения к динамической памяти. Операция стирания занимает еще больше времени. Поэтому ввели команды записи в предварительно стертые секторы, применение которых позволяет повысить производительность обмена с устройствами. Для стирания блоков предназначена отдельная команда, в которой указываются начальный адрес и количество стираемых секторов. Интерес представляет информация о состоянии сектора (стертый или нет, сколько раз выполнялась запись). Эту информацию можно получить по команде CFA Translate Sector.

Поскольку организация флэш-памяти значительно отличается от традиционных устройств с подвижными носителями, сообщения об ошибках этих устройств не вписываются в стандартные определения битов регистра ER. Устройства CFA могут иметь 8-разрядную внутреннюю организацию, и для разрешения 8-битного обмена в режиме PIO служат соответствующие подкоманды команды SET FEATURES. Обмен в режиме DMA для данного класса устройств не обязателен.

В ATA/ATAPI-6 введена поддержка адаптеров малогабаритных карт памяти (MMC, Smart Media, SD...). Адаптер представляет собой мост, транслирующий команды с интерфейса ATA (SATA) на специфический интерфейс карты. Для упрощения моста определен диапазон кодов команд, транслируемых мостом. Эти команды специфичны для каждого типа карт.

Управление энергопотреблением и шумом

Специальные команды управления энергопотреблением были заложены еще в первой спецификации ATA, в ATA/ATAPI-4 они уже считаются обязательны

ми. Различают следующие состояния, перечисленные в порядке возрастания энергопотребления:

- *Sleep* (сон) — «заснувшее» устройство потребляет минимум энергии, команды не воспринимаются, «разбудить» его может только сброс. Время «пробуждения» — не более 30 с;
- *Standby Mode* (дежурный режим) — устройство способно принимать команду по интерфейсу, но для доступа к носителю может потребоваться столь же большое время. В это состояние устройство может перейти как по команде, так и по таймеру (standby timer), отсчитывающему время от последнего запроса, полученного в состоянии ожидания или активном состоянии. Время срабатывания таймера программируется; он может быть отключен;
- *Idle Mode* (режим ожидания) — устройство способно сразу начать обслуживание обращений к носителю, но не слишком быстро, так как некоторые узлы отключены;
- *Active Mode* (активный режим) — устройство обслуживает все запросы за кратчайшее время.

В ATA/ATAPI-5+ ввели возможность перехода в состояние *Standby* по включении питания (*Power-Up In Standby*). При этом снижается пиковая нагрузка на блок питания в момент включения. Устройство может также поддерживать и специальную подкоманду Set Features для раскрутки шпинделя. Если эта подкоманда поддерживается, то ее необходимо выполнить до обращения к носителю; если не поддерживается — запуск произойдет по первой же команде, требующей обращения к носителю.

Устройства могут поддерживать *расширенное управление энергопотреблением* (Advanced Power Management, APM). При этом задается уровень *APM Level*, определяющий степень активности: 01h — минимальное потребление, FEh — максимальная производительность. Уровень выше 80h не позволяет устройству останавливать двигатель привода носителя (переходить в *Standby).*

В ATA/ATAPI-6 появилась новая возможность автоматического управления уровнем акустического шума (automatic acoustic management), издаваемого устройством при работе. Для этого вводятся специальные подкоманды Set Features, задающие уровень управления акустическим шумом (acoustic management level) в диапазоне от 0 (минимальные шум и производительность) до FFh (максимальная производительность без заботы о шуме). Управление не обязательно должно быть плавным — может быть всего несколько различимых градаций, и один и тот же способ снижения шума может применяться для ряда соседних значений уровня.

Защита данных

Начиная с ATA-3, в стандарт введена *группа команд защиты*, ее поддержка определяется по команде идентификации словом 128. Если защита поддерживается, то устройство должно отрабатывать все команды группы защиты. С точки зрения защиты устройство может находиться в одном из трех состояний:

- *Отперто* (unlocked) — устройство выполняет все свойственные ему команды. Устройство с установленной защитой можно отпереть только командой Security Unlock, в которой передается блок данных, содержащий правильный пароль. Для осложнения подбора пароля (его длина составляет 32 байта) служит внутренний счетчик неудачных попыток отпирания, по срабатывании которого команды отпирания будут отвергаться до выключения питания или аппаратного сброса;
- *Заперто* (locked) — устройство отвергает все команды, связанные с передачей данных и сменой носителя. Допустимы лишь команды общего управления, мониторинга состояния и управления энергопотреблением. Из команд защиты допустимы лишь команды стирания (Security Erase) и отпирания (Security Unlock). В это состояние устройство с установленной защитой входит каждый раз по включении питания;
- *Заморожено* (frozen) — устройство отвергает команды управления защитой, но выполняет все остальные. В это состояние устройство переводится командой Security Freeze Lock или автоматически по срабатыванию счетчика попыток отпирания устройства с неправильным паролем. Из этого состояния устройство может выйти только по аппаратному сбросу или при следующем включении питания. Срабатывание счетчика попыток отражается установкой бита 4 (EXPIRE) слова 128 блока параметров, бит сбросится по следующему включению питания или по аппаратному сбросу.

Устройство выпускается производителем с неустановленной защитой, то есть по включению оно будет отперто. Система защиты поддерживает два пароля: *главный* (master password) и *пользовательский* (user password) — и два уровня защиты: высокий (high) и максимальный (maximum). При *высоком уровне* защиты устройство можно отпереть любым из двух паролей. При *максимальном уровне* устройство отпирается только пользовательским паролем, а по главному паролю доступна только команда стирания, при выполнении которой *вся информация с носителя будет стерта.*

Команда защитного стирания имеет два варианта. При *обычном стирании* устройство заполняет двоичными нулями всю область данных, видимую пользователем. При *расширенном стирании* заполняется весь носитель, включая и области, ранее переназначенные устройством (из-за угрозы неисправимых ошибок). Кроме того, вместо двоичных нулей используется образец данных, заданный производителем (им заполняются новые устройства). После стирания защита снимается, но главный пароль сохраняется.

Команды защиты в ATA-4 дополнены ограничением максимального адреса, доступного пользователю (сообщаемого в блоке параметров идентификации). В ATA/ATAPI-5+ команды ограничения защищаются паролем. После включения питания можно ограничить максимальный адрес (но только один раз) или же ограничить доступ к команде ограничения адреса. Команды ограничения адреса неприменимы для устройств со сменными носителями.

Потоковое расширение команд

Традиционно главным качеством устройств хранения данных является способность достоверно хранить данные, а производительность является вторичной, хотя ее всячески стремятся повышать. Если при чтении (записи) блока данных происходит ошибка, то устройство автоматически выполняет серию повторных обращений к носителю, в результате чего чаще всего удается получить достоверный результат. Однако при этом возникает непрогнозируемая задержка доставки данных, которая крайне нежелательна для мультимедийных приложений, связанных с записью или чтением аудио- и видеоданных в реальном времени. В спецификацию ATA/ATAPI-7 предполагается ввести потоковое расширение системы команд (streaming feature set). *Потоком* называется непрерывная последовательность логических блоков, которые должны быть считаны или записаны с определенным допустимым временем. Для потоков вводятся новые команды чтения и записи в режимах PIO или DMA, а также новый журнал сообщений об ошибках потоковых команд. Потоковые команды позволяют управлять поведением устройства в случае обнаружения ошибок, а также «подсказывать» устройству правильную политику кэширования. Заметим, что ранее потоковое расширение в несколько ином виде (audio visual data features) планировалось ввести в ATA/ATAPI6, но в окончательную версию этого стандарта команды данного расширения не вошли.

ГЛАВА 20

Интерфейс SCSI

Интерфейс SCSI (Small Computer System Interface — системный интерфейс малых компьютеров, произносится «скази») предназначен для соединения устройств различных классов: памяти прямого (жесткие диски) и последовательного (стримеры) доступа, CD-ROM, оптических дисков однократной и многократной записи, устройств автоматической смены носителей информации, принтеров, сканеров, коммуникационных устройств и процессоров. Наиболее широко этот интерфейс используется для устройств и систем хранения данных.

Первоначально интерфейс SCSI был определен в виде *параллельной шины*, объединяющей *равноранговые устройства*. *Устройством SCSI* (SCSI Device) называется как *хост-адаптер,* связывающий шину SCSI с какой-либо внутренней шиной компьютера, так и *контроллер целевого устройства* (target controller), с помощью которого устройство подключается к шине SCSI. На шине возможно присутствие более одного хост-адаптера, что позволяет обеспечить разделение (совместное использование) периферии несколькими компьютерами, подключенными к одной шине SCSI. Интерфейс SCSI оптимизирован для многозадачной работы: после получения команды на время выполнения своих внутренних операций устройство может освобождать шину (и неоднократно), а затем восстанавливать соединение для передачи данных и информации о завершении. Во время освобождения шину могут занимать процессы ввода-вывода, связанные с тем же или другим устройством.

Интерфейс SCSI изначально ориентирован на эффективное обслуживание множества устройств. Применение цепочек команд позволяет разгружать хост даже при выполнении довольно сложных процедур, связанных с хранением и поиском данных. Правда, далеко не всегда эти возможности практически используются операционными системами и приложениями. Но и без них независимость работы устройств друг от друга и освобождение шины на время внутренних операций обеспечивают SCSI неоспоримые преимущества перед параллельным вариантом ATA в качестве интерфейса для мощных систем хранения данных. Однако с внедрением интерфейса Serial ATA II и механизма NCQ у SCSI появился серьезный конкурент.

20.1. Спецификации SCSI

Первая версия интерфейса, позже названная SCSI-1, была стандартизована ANSI в 1986 году (X3.131-1986). Это была 8-битная параллельная шина с максимальной частотой переключений 5 МТ/с[1], допускающая подключение до 8 устройств. Скорость передачи данных — около 2 Мбайт/с, режим передачи данных — асинхронный. Впоследствии (1991 г.) появилась спецификация SCSI-2, расширяющая возможности шины. Частота переключений шины *Fast SCSI-2* достигает 10 МТ/с, а *Ultra SCSI-2* — 20 МТ/с. Разрядность данных может быть увеличена до 16 бит — эта версия SCSI-2 называется *Wide* (широкая), а 8-бит- ную версию стали называть *Narrow* (узкая). 16-битная шина допускает включение 16 устройств. Стандарт SCSI-2 определял и 32-битную версию интерфейса, которая не получила практического применения. Появился синхронный режим передачи данных, введена дифференциальная версия интерфейса. Спецификация SCSI-2 определяет систему команд, которая включает набор базовых команд (Common Command Set, CCS), обязательных для всех ПУ, и специфических команд для периферии различных классов. Стандарт полностью описывает протокол взаимодействия устройств, включая структуры передаваемой информации.

Спецификация SCSI-3 — дальнейшее развитие стандарта, направленное на увеличение количества подключаемых устройств, расширение системы команд и поддержку технологии PnP. В качестве альтернативы параллельному интерфейсу SCSI-3 (SCSI-3 Parallel Interface, SPI) появляется возможность применения последовательного интерфейса, в том числе интерфейса Serial SCSI со скоростью 150 Мбайт/с и волоконно-оптического интерфейса Fibre Channel со скоростями 100 и 200 Мбайт/с. Спецификация SCSI-3 существует в виде широкого спектра документов, определяющих отдельные аспекты интерфейса на уровне физических соединений, транспортных протоколов и наборов команд. Их объединяет SAM — документ, описывающий архитектурную модель.

Верхний уровень модели SAM представляет собой набор общих команд SCSI-3 (SCSI-3 Primary Commands, SPC) для устройств различных классов, который дополняется набором команд соответствующего класса. Два нижних уровня представляют возможные *транспортные протоколы* с соответствующими спецификациями *физических соединений*:

- SPI (SCSI-3 Parallel Interface) — параллельная шина (разъемы, сигналы и транспортный протокол). Обеспечивает соединение небольшого (до 16) числа устройств с небольшим (до 25 м) удалением друг от друга.
- SBP (Serial Bus Protocol) — протокол последовательной шины, реализуемый интерфейсом IEEE 1394 (FireWire, см. главу 18). Обеспечивает подключение среднего (до 63) числа устройств с удалением друг от друга до 4,5 м при суммарной протяженности кабеля до 72 м[2].

[1] МТ/с — миллионов передач в секунду (Mega Transfer/sec, MT/s). Называть это тактовой частотой некорректно, поскольку тактового сигнала в шине нет.

[2] В IEEE 1394b для ряда типов кабелей допустима длина до 100 м.

- FCP (Fibre Channel Protocol) — протокол для интерфейса Fibre Channel с соответствующим физическим уровнем FC-PI и FC-FS (см. главу 21). Позволяет создавать большие (по числу узлов и протяженности) домены устройств SCSI, используется в крупных сетях хранения данных (SAN).
- SAS (Serial Attached SCSI) — устройства SCSI с последовательным интерфейсом, совместимым с интерфейсом SATA. Эта спецификация позволяет объединять значительное число устройств (до 16 384), расположенных на среднем (до 1 м) удалении друг от друга. Имеется возможность одновременного обмена между несколькими парами устройств.
- SSA-S3P (Serial Storage Architecture SCSI-3 Protocol) — транспортный протокол для использования транспортного и физического уровней (SSA-PH и SSA-TL) архитектуры SSA (Serial Storage Architecture). Архитектура SSA (фирменное решение IBM) обеспечивает объединение большого числа узлов подключения устройств через последовательный интерфейс. К особенностям архитектуры относятся различные варианты топологий с использованием избыточных связей, полнодуплексных соединений и коммутации пакетов, а также отсутствие издержек арбитража и одновременное выполнение конкурирующих заданий.
- iSCSI — транспортный протокол для доставки команд, данных и информации состояния в любые IP-сети, используя обмен IP-дейтаграммами. Позволяет объединять устройства, расположенные на практически неограниченном (в пределах Земли) удалении друг от друга.

20.2. Архитектурная модель SAM

Развитием идей, заложенных в шину SCSI, стала ее *архитектурная модель*, названная *SAM* (SCSI Architecture Model — архитектурная модель SCSI), введенная в SCSI-3 и развиваемая поныне. Модель SAM определяет многоуровневую структуру стандартов SCSI, в которой нижние уровни (средства доставки информационных блоков) разнообразны: параллельная шина, последовательные шины и интерфейсы, сетевые средства. В модели представлены система команд, общая для всех устройств, и расширяемый список наборов команд, свойственных устройствам различных классов.

Модель SAM определяет идеологию SCSI — клиент-серверные отношения между *инициаторами обмена* (initiator) и *целевыми устройствами* (target). Чаще всего в роли инициатора выступает хост-адаптер компьютера, а в роли целевых устройств (ЦУ) — периферийные устройства. Возможны комбинированные устройства, выступающие в роли и инициатора, и ЦУ. Инициатор обмена является *клиентом* — он обращается к целевому устройству с запросом выполнения операции ввода-вывода (команды). Целевое устройство выступает *сервером* — оно интерпретирует команду и управляет интерфейсом для выполнения всех операций, связанных с командой: доставкой данных, сообщением результата выполнения (состояния). В ряде случаев роли устройств меняются: например, при выполнении команды копирования *(Copy)* инициатор дает указание ведущему устройству копирования (*Copy Master)* на обмен данными, который мо

жет производиться и с другим ЦУ (для которого ведущее устройство копирования выступает в роли инициатора).

Инициатор и целевое устройство взаимодействуют через *порты SCSI*. Совокупность портов инициаторов и целевых устройств, связанных подсистемой доставки, называется *доменом SCSI* (SCSI domain). Функциональность домена ограничивается реализацией транспортного уровня и средствами физических подключений (двумя нижними уровнями модели). В SAM заложена возможность взаимодействия однопортовых и многопортовых устройств (как инициаторов, так и ЦУ) через один или несколько доменов SCSI.

Каждое целевое устройство может содержать одно или несколько независимых *логических устройств* (ЛУ) со своими номерами (Logical Unit Number, LUN), представляющих периферийные устройства или их части. Первоначально для задания номера логических устройств предназначалось 3-битное поле LUN (до 8), в параллельной шине SCSI-3 поле LUN расширили до 6 бит (до 64 логических устройств). Архитектурная модель SAM оперирует адресами, форматы которых могут поддерживать и большее число логических устройств (до 256 и даже 16 384), а помимо них и иерархические структуры (доступ через специальное ЛУ к подчиненным ему ЛУ).

Команды, задания и очереди

Логически завершенным актом взаимодействия инициатора и целевого устройства в SCSI является *задание* (task), в терминах SCSI-2 эквивалентом задания является *процесс ввода-вывода* (I/O process). Задание может включать в себя *одиночные команды* или *цепочки связанных команд* (linked command).

Цепочка команд — последовательность команд, в которых результаты выполнения предыдущей команды (например, адрес найденного блока при поиске по образцу) могут использоваться в качестве параметров в следующей команде (например, чтения). Все команды цепочки являются частью одного процесса. Команды не являются полностью независимыми — при относительной адресации последний блок, адресованный предыдущей командой, доступен для следующей. Так, можно исполнить команду *Search Data*, по которой на диске будет найден блок, содержащий информацию, совпадающую с эталоном поиска. Связав с ней команду чтения *Read*, можно прочитать этот блок или блок с указанным смещением относительно найденного.

В целевых устройствах могут организовываться *наборы заданий* (task set), в терминах SCSI-2 называемые *очередями* (queue). Задание запускается инициатором, исполнением задания управляет целевое устройство — оно организует передачу команд, данных и служебной информации по интерфейсу.

По завершении выполнения каждой команды ЦУ передает инициатору *байт состояния* ее выполнения (*Status*); по нему можно определить, является ли выполненная команда последней в цепочке (единственной), что будет признаком завершения задания. Задания друг от друга независимы — порядок их исполнения по отношению к порядку их поступления устройство может менять. Команды в цепочке (внутри задания) выполняются строго упорядоченно.

Задания ставятся в очередь с разнообразными атрибутами. Очередное задание можно поставить в очередь «по-честному», а можно «пропихнуть» вне очереди: задание с атрибутом *Head Of Queue* выполняется сразу после завершения текущего активного процесса. Задания с атрибутом *Simple* исполняются ЦУ в порядке, который оно считает оптимальным. Задания с атрибутом *Ordered* исполняются в порядке поступления, после выполнения всех ранее запущенных процессов. Изменение порядка выполнения заданий не касается порядка команд в цепочке, поскольку цепочка принадлежит одному заданию. Инициатор может удалить задание из очереди, сославшись на него по тегу.

Соединения

Для запуска процесса ввода-вывода инициатор *устанавливает соединение* с определенной частью выбранного целевого устройства (его LUN), к которой адресуется команда (или цепочка команд). Это соединение может удерживаться до окончания выполнения процесса, оставляя шину занятой, а может временно разрываться, освобождая шину для выполнения других процессов. Устанавливая соединение, инициатор предоставляет ЦУ *право на разрыв соединения* (disconnect), разрыв выполняется по усмотрению ЦУ. При этом оно занимает шину только на время активной передачи данных, освобождая ее на время своих внутренних операций (поиска данных). В это время по шине могут производиться обмены между теми же и другими устройствами, что очень выгодно позиционирует шину SCSI для работы с большим количеством устройств под управлением многозадачных и многопоточных ОС. Инициатор во время выполнения процесса может также *затребовать разрыв соединения*.

Состояния, исключения и асинхронные события

Когда логическое устройство (ЛУ) завершает (нормально или ненормально) выполнение команды, оно посылает байт состояния. Возможные состояния приведены в табл. 20.1.

Таблица 20.1. Байты состояния

Байт	Состояние	Значение
0	Good	Успешное завершение команды
2	Check Condition	Указание на особые условия, возникшие при выполнении команды
4	Condition Met	Запрошенная операция выполнена (для одиночных команд Search Data и Pre-Fetch)
8	Busy	Занято (невозможен прием команды)
10h	Intermediate	Успешное выполнение команды в цепочке
14h	Intermediate Condition Met	Запрошенная операция в цепочке команд выполнена
18h	Reservation Conflict	Попытка обратиться к ЛУ, зарезервированному другим ЦУ
22h	Command Terminated	Завершение текущего процесса по сообщению Terminate I/O Process или по асинхронному событию (отменено, начиная с SAM-2)

Байт	Состояние	Значение
28h	Task Set Full	Набор заданий заполнен, задание в очередь не поставлено
30h	ACA Active	Команда не принята из-за режима ACA
40h	Task Aborted	Задание снято по инициативе другого инициатора (SAM-2)

В процессе выполнения команды в устройстве могут возникнуть *особые условия,* или *исключения* (exceptions), о чем сообщается байтом состояния *Check Condition.* В этом случае в логическом устройстве для обработки исключительной ситуации все задания блокируются на время действий режима. Продолжение нормальной работы (разблокирование заданий) вызывается по-разному: по получению следующей команды, по автоматическому сообщению уточненного состояния или с помощью функций управления заданиями. Автоматическое сообщение уточненного состояния (autosense) поддерживается всеми транспортными протоколами, кроме старых параллельных, управляемых сообщениями и не допускающих передач информационных блоков (IU, см. 20.4).

В целевых устройствах могут возникать события, требующие *асинхронного уведомления,* — передачи блоков уточненного состояния, не связанных с текущими исполняемыми командами. Асинхронное уведомление требуется в случае возникновения различных событий: появления нового доступного устройства, запроса смены носителя оператором, неудачи физического выполнения кэшированной записи (когда инициатору ранее сообщили об успешном завершении команды) и ряда других.

Транспортный протокол должен обеспечивать возможность асинхронных уведомлений, но для целевых устройств возможность всегда можно запретить. Асинхронные уведомления посылаются на предварительно указанный порт инициатора. Альтернативой асинхронным уведомлениям является установка байта состояния *Check Condition* в ответ на исполнение команды.

Типы периферийных устройств

Для целевых устройств SCSI определена стандартная классификация типов (табл. 20.2). В сложное ПУ могут входить несколько однотипных логических устройств (со своими номерами LUN). По характеру обмена данными устройства разделяются на 2 класса — блочные (block device) с типами 0, 4, 5, 7 и потоковые (stream device) с типами 1, 2, 3, 9. Особенности устройств различных классов рассмотрены в [6], [8].

Таблица 20.2. Типы ПУ SCSI

Код типа	Назначение
00h	Direct-access device — устройства прямого доступа (накопители на магнитных дисках)
01h	Sequential-access device — устройства последовательного доступа (типичные накопители на магнитных лентах)
02h	Printer device — принтеры
03h	Processor device — процессоры (устройства обработки данных)

продолжение

Таблица 20.2 (*продолжение)*

Код типа	Назначение
04h	Write-once device — устройства однократной записи (некоторые оптические диски)
05h	CD-ROM device — накопители CD-ROM
06h	Scanner device — сканеры
07h	Optical memory device — устройства оптической памяти
08h	Medium Changer device — устройства смены носителей (jukebox)
09h	Communications device — коммуникационные устройства
0Ah–0Bh	Устройства класса ASC IT8 (Graphic Arts Pre-Press Devices — высококачественные устройства печати)
0Ch	Array controller device — контроллеры массивов накопителей
0Dh	Enclosure services device — сервисные устройства шкафов
0Eh	Reduced block command devices — блочные устройства с сокращенным набором команд (RBC)
0Fh	Optical card reader/writer device — устройства считывания и записи оптических карт (штрих-коды)
10h–1Eh	Зарезервировано
1Fh	Неизвестный тип или устройство отсутствует

Система команд SCSI

Система команд SCSI четко стандартизована, что обеспечивает высокий уровень совместимости устройств одного класса разных производителей и разных моделей. Классы устройств определены в соответствии с их функциональным назначением и командами, необходимыми для взаимодействия с ними. Каждое устройство должно поддерживать команды, обязательные для данного класса; есть и общие команды, обязательные для поддержки устройствами всех классов.

Для устройств хранения разных классов (магнитных дисков с произвольным доступом, ленточных с последовательным доступом, оптических и др.) в SCSI используется только линейная адресация логических блоков данных с одномерным адресом (Logical Block Address, LBA), что унифицирует взаимодействие с устройствами разных типов и объемов хранимой информации. Разрядность линейного адреса — 21, 32 или 64 бита.

Блок дескриптора команды (Command Descriptor Block, CDB) может иметь фиксированную длину (6, 10, 12 или 16 байтов) или же произвольную. Поле адреса логического блока (LBA) может иметь длину 21 бит для 6-байтных блоков, 32 бита — для 10- 12- и 16-байтных, 64 бита — для 16-байтных. В дескрипторе имеются признаки цепочки команд и реакции устройства на особые условия, возникающие при отработке команды (автоматическая или неавтоматическая).

Команды посылаются инициатором к логическому устройству. Любое логическое устройство обязано поддерживать команды получения информации об устройстве (*Inquiry*), о его уточненном состоянии (*Request Sense)* и готовности *(Test Unit Ready),* а для ряда классов обязательна и команда *Send Diagnostic.*

Эти команды требуются для конфигурирования системы, тестирования устройств, а также сообщений об ошибках и исключительных ситуациях.

Устройства могут поддерживать многофункциональные команды *Read Buffer* и *Write Buffer*. С их помощью выполняются различные диагностические функции (например, эхо — чтение возвращает результаты предыдущей записи), загрузка микрокода, запись журналов, управление коммуникационным протоколом экспандеров и специфические функции по усмотрению разработчика.

Команды для «полезного» обмена информацией соответствуют природе устройств (по классам), здесь мы их обсуждать не будем. Отметим лишь команды *Copy* и *Search Data*, иллюстрирующие общую «интеллектуальность» SCSI.

В SCSI возможно непосредственное копирование блоков данных с одного устройства SCSI на другое без промежуточного помещения этих данных в ОЗУ компьютера. Команда копирования *(Copy)* оперирует парой ЛУ, которые могут принадлежать как одному ЦУ, так и разным. Копирование возможно между *устройствами разных классов* — прямого и последовательного доступа (блочные и поточные). Новая команда *Extended Copy* позволяет выполнять операции копирования между *наборами источников* и *получателей*. Источниками и получателями данных могут быть как логические устройства (указанные диапазоны их логических блоков), так и сегменты данных, описанные соответствующими дескрипторами. Копируемые данные могут быть получены и клиентскими приложениями (командой *Receive Copy Results*, ссылающейся на идентификатор набора).

Для устройств хранения имеются команды поиска данных по образцу. В командах поиска *Search Data* данные ищутся сравнением указанного числа логических записей с эталоном. Логические записи определяются длиной, начальным логическим блоком и смещением внутри него. Можно потребовать попадания искомых данных в один логический блок. Кроме того, есть команда сравнения данных *(Compare)*, которая, как и команда *Copy*, оперирует с парой ЛУ.

Отличия ATAPI от SCSI

Система команд SCSI может применяться и для устройств с интерфейсом IDE (ATA), что описывает спецификация ATAPI. Здесь задействуются те же блоки дескрипторов команд (CDB), однако архитектурные возможности SCSI в ATAPI используются в усеченном варианте:

- нет средств адресации логических устройств;
- нет SCSI-функций управления заданиями;
- не поддерживаются дескрипторы переменной длины;
- устройство не может сообщить о специфических вариантах завершения команд *(BUSY, TASK SET FULL, RESERVATION CONFLICT* и т. п.), связанных с многозадачной природой SCSI;
- в режиме DMA невозможно передавать блоки данных с нечетным числом байтов, хотя в режиме PIO это возможно;
- перекрытие команд и очереди на практике не используются.

20.3. Хост-адаптер SCSI

Хост-адаптер является важнейшим узлом интерфейса, определяющим производительность системы SCSI. В его задачу входит передача данных между хостом (программой, исполняемой центральным процессором) и другими устройствами, подключенными к шине, по протоколам вышеописанных физических интерфейсов. Структуры передаваемых блоков данных и команды устройств стандартизованы, их описание приводится в [6], [8]. Однако архитектуры и программные модели адаптеров не стандартизованы (в отличие, например, от адаптеров ATA). Существует широкий спектр адаптеров, к простейшим можно подключать только устройства, некритичные к производительности. Такие адаптеры могут входить, например, в комплект поставки сканеров, а подключение к ним диска может оказаться невозможным. В простейших адаптерах весь протокол шины SCSI (последовательность фаз) может реализовываться центральным процессором хоста при минимальных аппаратных средствах. Высокопроизводительные адаптеры имеют собственный специализированный процессор, большой объем буферной памяти и используют высокоэффективные режимы прямого управления шиной для доступа к памяти компьютера. Адаптеры SCSI существуют для всех шин расширения (PCI, PCI-X, PCI Express, CardBus, PCMCIA, ISA, EISA, MCA, VLB), шин USB и FireWire и даже для LPT-порта. Ряд системных плат имеют встроенный адаптер SCSI, подключенный к одной из локальных шин. При выборе интерфейса, к которому подключается хост- адаптер, нужно учитывать производительность — интерфейс не должен стать узким местом при обмене с высокопроизводительными устройствами SCSI. Наибольшую эффективность имеют хост-адаптеры для шин PCI, PCI-X, PCI Express, CardBus. Конечно, за мощный адаптер для сервера приходится платить — его цена может превышать цену рядового настольного компьютера. Еще дороже хост-адаптеры со встроенными контроллерами RAID-массивов, которые содержат мощный RISC-процессор и большой объем локальной памяти.

Конфигурирование хост-адаптеров с точки зрения шины SCSI не отличается от конфигурирования других устройств. Современные адаптеры конфигурируются не джамперами, а программно. Утилита конфигурирования обычно входит в расширение BIOS, установленное на плате адаптера, и приглашение к ее вызову выводится на экран во время теста POST.

Как и всякая карта расширения, хост-адаптер должен быть сконфигурирован с учетом шины расширения, к которой он подключается. Системные ресурсы для шинного SCSI-адаптера включают:

- область памяти для расширения ROM BIOS, необходимого для конфигурирования устройств и дисковых функций (если в системе установлено несколько однотипных хост-адаптеров, для них используется ROM BIOS с одного адаптера, а разнотипные хост-адаптеры не всегда могут работать вместе);
- область разделяемой буферной памяти;
- область портов ввода-вывода (I/O port);

- запрос прерывания (IRQ);
- канал прямого доступ к памяти (для шин ISA/EISA), обычно используемый лишь для захвата управления шиной (bus mastering).

Всем устройствам SCSI, в том числе и хост-адаптеру, требуются специальные драйверы. Базовый драйвер дисковых устройств входит в BIOS хост-адаптера; он обычно эмулирует трехмерную адресацию дискового сервиса Int 13h. Расширения, например ASPI (Advanced SCSI Programming Interface — усовершенствованный интерфейс программирования для SCSI), загружаются отдельно. От драйверов в значительной степени зависит производительность устройств SCSI. «Умное» ПО способно эффективно загружать работой устройства, а иногда и «срезать углы» — выполнять копирование данных между устройствами без выхода на системную шину компьютера. Наиболее предпочтительны драйверы, работающие в режиме прямого управления шиной; их применение позволяет реализовать все преимущества SCSI в многозадачных системах.

20.1. Параллельные шины SCSI

Современные параллельные интерфейсы SCSI представляют собой реализацию транспортного уровня и уровня подключений архитектурной модели SAM. Поскольку параллельный интерфейс SCSI появился до принятия SAM, некоторые части модели в нем не реализованы из-за ограничений транспортного протокола и исторически сложившихся соглашений. В частности, в параллельных интерфейсах нет имен портов и устройств (имеются только имена логических устройств). Идентификаторы портов ограничены 4-битным значением (16 возможных идентификаторов), формат LUN ограничен 6-битным полем. Тем не менее, основная клиент-серверная идеология SAM полностью реализована во всех версиях параллельной шины. Каждому устройству назначается *идентификатор (SCSI ID*), уникальный на шине. Назначение идентификаторов производится статически (вручную или определяется положением устройства на шасси). Попытки автоматического (с использованием технологии PnP) назначения идентификаторов устройствам шины SCSI потерпели неудачу и прекратились.

Версии параллельной шины

Параллельный интерфейс SCSI существует в нескольких версиях, различающихся разрядностью шины, способами передачи сигналов и синхронизации. Физически «узкий» интерфейс SCSI представляет собой шину, состоящую из 18 сигнальных и нескольких питающих цепей. В «широком» варианте число сигнальных цепей увеличено. Для защиты от помех каждая сигнальная цепь имеет собственный обратный провод. На применяемых двухрядных разъемах контакты сигнальных и обратных цепей располагаются друг против друга. Это позволяет применять в качестве кабелей как витые пары проводов, так и плоские ленточные кабели, где сигнальные и обратные провода чередуются. С точ

ки зрения передачи сигналов на значительные расстояния (до 25 м) это наиболее правильный подход к кабельным соединениям.

По типу сигналов различают *линейные* (Single Ended, SE) и *дифференциальные* (differential) версии SCSI. Их кабели и разъемы идентичны, но электрической совместимости устройств нет. Значки для обозначения типа интерфейса стандартизованы (рис. 20.1).

Рис. 20.1. Обозначения интерфейсов: а — HVD, *б* — SE, *в* — LVD/SE, г — LVD, *д* — SAS

В широко используемой *линейной* версии каждый сигнал передается потенциалом с ТТЛ-уровнями относительно общего провода. Общий (обратный) провод для каждого сигнала тоже должен быть отдельным, что снижает перекрестные помехи. В SCSI-1 применяются передатчики с открытым коллектором, приемники на биполярных транзисторах. Высокий уровень при пассивном состоянии передатчиков обеспечивают пассивные терминаторы (см. далее). В SCSI-2 стали применять и передатчики с активным снятием сигнала (active negation). Схема с открытым коллектором для снятия сигнала просто «отпускает» линию, и ее потенциал возвращается в исходное состояние только под действием терминаторов. При активном снятии выходная схема передатчика кратковременно форсирует переход линии к потенциалу пассивного состояния, после чего «отпускает» линию; при этом создается иллюзия возможности работы без терминаторов. В SCSI-3 стандарт SPI предписывает использование интерфейсных схем КМОП (CMOS).

Дифференциальная версия для каждой цепи задействует пару проводников, по которым передается парафазный сигнал. В первой дифференциальной версии *(Diff*), позже названной высоковольтной (High-Voltage Differential, HVD), задействуются дифференциальные приемопередатчики, применяемые и в интерфейсе RS-485, что позволяет значительно увеличить длину кабеля, сохраняя скорость обмена. Высоковольтный дифференциальный интерфейс использовался в дисковых системах серверов, но в обычных PC распространения он не получил. Интерфейс HVD (но еще с названием Diff) появился в SCSI-2, а в стандарте SPI-3 (SCSI-3 1999 года) он уже упразднен, поскольку скорость Fast-80 и выше он не выдерживает.

Низковольтная версия (Low-Voltage Differential, LVD) дифференциального интерфейса позволяет работать на высоких скоростях при значительной длине шины (25 м для пары устройств, 12 м для шины).

Универсальные устройства LVD (multimode LVD) совместимы с устройствами SE благодаря возможности их автоматического переконфигурирования. Эти устройства распознают напряжение на линии DIFFSENS и по низкому уровню напряжения на ней способны переключаться из режима LVD в SE (с понижением

доступной скорости обмена). Контакт разъема, на который выводится эта цепь, в устройствах SE заземлен, что и обеспечивает автоматическое «понижение» режима всех устройств шины до SE при наличии на шине хотя бы одного устройства SE.

ВНИМАНИЕ----------

Устройства HVD электрически несовместимы ни с LVD, ни с SE. Подключение устройств HVD к шине с LVD или SE чревато их выходом из строя.

Информация по шине данных передается побайтно (пословно) посредством запросов (*REQuest*) и подтверждений (ACKnowledge). Каждый байт информации контролируется на нечетность, но контроль может быть отключен. В более поздних версиях шины для контроля достоверности используются CRC-коды, защищающие блоки данных и информационные блоки. Интерфейс допускает синхронную передачу данных, ускоряющую обмен (в SCSI-1 синхронного режима не было).

Скорость передачи данных определяется *частотой переключений шины данных*, измеряемой в миллионах передач в секунду (Mega Transfers per second, MT/s), и *разрядностью* шины. Изначально разрядность узкой (*narrow*) шины SCSI составляла 8 бит, а частота — до 5 MT/c. Широкий *(wide)* вариант шины имеет разрядность 16 бит. В первых версиях шины применялась классическая синхронизация данных. *ST* (Single Transition), при которой на действительность данных указывал спад синхронизирующего сигнала. Позже ввели двойную синхронизацию *DT* (Double Transition), при которой на действительность данных указывают и спад, и фронт синхронизирующего сигнала. В последних версиях для самых высоких скоростей ввели режим *одновременного переключения* (pace[1]), в котором данные и синхронизирующие сигналы переключаются одновременно. Особенности передачи данных в различных режимах рассмотрены далее.

Допустимые варианты сочетаний типов приемопередатчиков (HVD, SE, LVD), синхронизации (ST, DT) и скоростей передачи приведены в табл. 20.3. Здесь в качестве меры скорости (Fast-5, -10, ..., -160) указана максимальная частота передач по шине данных. Максимальная скорость передачи данных дана для узкой и широкой шины (через слэш).

Таблица 20.3. Варианты скоростей передачи (Мбайт/с) для параллельных интерфейсов SCSI

Скорость, тип	Async	Fast-5	Fast-10	Fast-20	Fast-40	Fast-80	Fast-160	Fast-320
HVD (ST)	+	5/10	10/20	20/40	40/80	–	–	–
SE (ST)	+	5/10	10/20	20/40	–	–	–	–
LVD ST	+	5/10	10/20	20/40	40/80	–	–	–
LVD DT	–	–	–/20	–/40	–/80	–/160	–/320 (pace)	–/640 (pace)
Стандарт	SCSI-1	SCSI-2	SCSI-2	SPI	SPI-2	SPI-3	SPI-4	SPI-5

[1] Дословно и образно можно перевести как «иноходь» — одновременный шаг и линий данных, и стробирующего сигнала.

СОВЕТ

При наличии устройств SE и LVD их следует подключать к разным шинам, поскольку совместное использование на шине этих устройств снижает допустимый скоростной режим до Fast-20.

Протокол параллельной шины

Сигналы шинного интерфейса

Назначение сигналов параллельной шины раскрывает табл. 20.4.

Таблица 20.4. Назначение сигналов шины SCSI

Сигнал	Назначение
BSY#	Busy — шина занята
SEL#	Select — выбор ЦУ инициатором (фаза Select) или инициатора целевым устройством (фаза Reselect)
C/D#	Control/Data — передача управляющей информации (низкий уровень) или данных (высокий уровень)
I/O#	Input/Output — направление передачи относительно инициатора: вводу (от ЦУ) соответствует низкий уровень. Используется и для различия прямой (Select) и обратной (Reselect) выборки: фазе Selection соответствует низкий уровень
MSG#	Message — признак фазы сообщения или DT-данных
DB[0:15]#	Data Bus — инверсная шина данных
P_CRCA (DBP)	Parity/CRC Available. В традиционных режимах — бит четности, дополняющий количество единичных битов DB[0:7] до нечетного. В фазе арбитража не действует. В режиме передачи групп этим сигналом ЦУ указывает на передачу заполнителя или CRC-кода. В режиме передачи информационных блоков этим сигналом ЦУ указывает на передачу последнего блока данных в потоке
P1 (DBP1)	Parity1. В традиционных режимах — бит четности, дополняющий количество единичных битов DB[8:15] до нечетного. В фазе арбитража не действует. В режиме DT paced источник данных фазой переключения этого сигнала указывает на действительность данных
TERMPWR	Terminator Power — питание терминаторов
ATN#	Attention — внимание (намерение инициатора послать сообщение)
REQ#	Request — запрос от ЦУ на пересылку данных
ACK#	Acknowledge — подтверждение передачи (ответ на сигнал REQ#)
RST#	Reset — сброс шины
DIFFSENS	Признак дифференциального (LVD) интерфейса: ниже 0,7 В — линейный интерфейс (SE); 0,9–1,9 В — низковольтный дифференциальный интерфейс (LVD); выше 2,4 В — высоковольтный дифференциальный интерфейс (HVD)

Адресация устройств и фазы шины

Каждое *устройство SCSI*, подключенное к шине, должно иметь свой уникальный *адрес*, назначаемый при конфигурировании. Для 8-битной шины диапазон значений адреса — 0-7, для 16-битной — 0-15. Адрес задается предварительной установкой переключателей или джамперов. Для хост-адаптера возможно программное конфигурирование. Адресация устройств на шине в фазах выборки

осуществляется через *идентификатор SCSI ID,* представляющий адрес в позиционном коде. Адрес определяет номер той линии шины данных, которая осуществляет выборку данного устройства. Устройство с нулевым адресом выбирается низким уровнем на линии DB0# (SCSI ID = 00000001), с адресом 7 — на линии DB7# (SCSI ID = 10000000). Значение идентификатора определяет *приоритет устройства* при получении доступа к шине. Понятия «адрес» и «идентификатор» часто путают, но это всего лишь две различные формы представления одного и того же параметра.

В любой момент обмен информацией по шине может происходить только между парой устройств, активным (на шине) может быть только один процесс ввода-вывода. Процесс запускает *инициатор обмена*, а *целевое устройство* (ЦУ) его исполняет. При использовании традиционного протокола (без передач информационных блоков) каждый процесс ввода-вывода один или несколько раз вызывает следующие последовательности фаз шины: из состояния *Bus Free* через фазу *Arbitration* переход к фазе *Selection* или *Reselection*. Далее следуют фазы передачи информации (*Command, Data, Status, Message).* Завершающей фазой является *Message In*, в которой инициатору передается сообщение *Disconnect* (освобождение шины с последующим продолжением) или *Task Complete* (завершение выполнения команды), после чего шина переходит в состояние покоя *Bus Free*. В протоколе с передачей информационных блоков вместо различных фаз *Command, Data, Status* используется лишь один тип *(Data).*

В фазе *Bus Free* шина находится в состоянии покоя — нет никаких процессов обмена; она готова к арбитражу.

В фазе *Arbitration* устройство может получить право на управление шиной. Дождавшись покоя шины *(Bus Free),* устройство вводит сигнал BSY# и свой идентификатор *SCSI ID.* Если идентификаторы выставили несколько устройств одновременно, то право на управление шиной получает устройство с наибольшим приоритетом, а остальные устройства отключаются до следующего освобождения шины. Устройство, выигравшее арбитраж, вводит сигнал SEL# и переходит в фазу *Selection* или *Reselection.*

В фазе *Selection* инициатор, выигравший арбитраж, вводит на шину данных результат логической операции ИЛИ от пары идентификаторов — своего и ЦУ, — сопровождая его битом четности. Инициатор устанавливает сигнал ATN#, указывая, что следующей фазой будет *Message OUT,* и снимает сигнал BSY#. Адресованное ЦУ отвечает сигналом BSY#, если на шине данных присутствует только пара идентификаторов (его и инициатора) и контроль четности не указывает на ошибку. На некорректные значения устройства отвечать не должны. Если за заданное время ЦУ не ответило сигналом BSY#, срабатывает тайм-аут, инициатор освобождает шину или вводит сигнал сброса RST#.

Фаза *Reselection* аналогична предыдущей, но ее вводит ЦУ. Эта фаза появляется в том случае, если ЦУ в ходе исполнения процесса отключалось от шины на время выполнения своих внутренних операций. По завершении внутренней операции это устройство, выиграв арбитраж, будет вызывать инициатор, который запускал процесс. Адресованный инициатор «вспоминает» о запущенном им процессе и отвечает сигналом BSY#. Условия ответа (только два активных

бита на шине данных с правильной четностью) и тайм-аут аналогичны предыдущей фазе.

В информационных фазах *Command, Data, Status* и *Message* по шине данных передаются байты информации (а не идентификаторы). В информационных фазах передачами управляет ЦУ, удерживая занятость шины сигналом BSY#. В это время инициатор может запросить посылку сообщения введением сигнала ATN#. На это ЦУ (когда сочтет возможным) введет фазу *Message OUT*, в которой инициатор и сможет передать данное сообщение. Между информационными фазами сигналы BSY#, SEL#, REQ# и ACK# должны оставаться в неизменном состоянии, меняться могут только состояния сигналов C/D#, I/O#, MSG# и шины данных. ЦУ может освободить шину, сняв сигналы MSG#, C/D#, I/O# и BSY#.

Сигналы ATN# и RST# могут порождать условия *Attention* и *Reset* соответственно, причем асинхронно по отношению к фазам шины. Эти условия могут привести к изменению предопределенного порядка фаз. Сигнал ATN# вводится инициатором во время любой фазы, кроме арбитража и состояния покоя шины. Сигнал RST# вводится в любой момент любым устройством, и по сигнализируемому им условию *Reset* все устройства должны немедленно освободить шину. В зависимости от настройки, принятой для всех устройств конкретной системы, возможно выполнение одного из двух вариантов сброса. «Жесткий» сброс переводит устройства в состояние, принятое по включению питания, сбрасывая все текущие процессы, очереди и т. п. В случае «мягкого» сброса после освобождения шины устройства пытаются завершить начатые операции, сохраняя текущие параметры.

При описании фаз не говорилось о временных задержках. Они определяются спецификацией так, чтобы возможный «перекос» — неодновременный приход сигналов, вызванный задержкой как в электронных схемах, так и в разных проводах кабеля, — не влиял на устойчивость протокола. В асинхронном режиме обмена на скорость передачи информации влияет и длина кабеля, поскольку изменения состояний участников обмена привязываются к сигналам, распространяющимся по кабелю с ограниченной скоростью. Если в широкой шине имеется пара кабелей (A и B, что на практике встречается редко), то в каждом из них используется своя пара управляющих сигналов (REQ#/ACK# и REQB#/ ACKB#), поскольку эти кабели могут иметь разную длину.

Арбитраж, захват и освобождение шины

Право на использование шины разыгрывается путем арбитража. Традиционный арбитраж основан на значениях идентификаторов устройств, определяющих их приоритет. Если в фазе *Arbitration* шину запрашивают несколько устройств одновременно, то право на управление получает устройство с наибольшим приоритетом. Приоритет связан с идентификатором следующим образом:

- для «узкой» шины наибольший приоритет имеет устройство с адресом 0, для устройств с адресами 1...7 приоритет убывает;
- для «широкой» шины наибольший приоритет имеет устройство с адресом 8, приоритет последовательно убывает для устройств с адресами 9... 15, 0...7.

В SPI введена возможность *справедливого арбитража* (fairness arbitration), обеспечивающая равноприоритетный доступ к шине. Для реализации справедливого арбитража устройство должно иметь специальный *регистр справедливости* (fairness register), в котором содержится информация о неудачных попытках арбитража устройств с более низким приоритетом. Устройство не должно пытаться захватывать шину, если его регистр справедливости имеет ненулевое значение.

По протоколу арбитража различают *нормальный арбитраж*, начинающийся с фазы покоя шины, и *ускоренный арбитраж*, начинающийся с сообщения *QAS Request* (Quick Arbitration and Selection Request — запрос ускоренного арбитража и выбора). Для ускоренного арбитража применяется особый способ освобождения шины целевым устройством. Нормальный арбитраж обязателен для всех устройств, QAS-арбитраж не является обязательным, он разрешается при согласовании протокола. Справедливый арбитраж обязателен для QAS-арбитра- жа, для нормального арбитража справедливость (если поддерживается) разрешается при согласовании протокола.

Информационные фазы

В информационных фазах по шине данных передаются собственно данные и управляющая информация: команды, сообщения, информация состояния. Для передачи управляющей информации (*Command, Status, Message)* всегда используется только 8-битная разрядность. Передача данных (и информационных блоков) выполняется на предварительно согласованной разрядности (8 или 16 бит). Для информационных фаз применяются три режима обмена, различающихся способом синхронизации:

- асинхронный режим;
- синхронный режим;
- режим одновременного переключения.

Асинхронный режим

Асинхронный режим используется для передачи управляющей информации (*Command, Status, Message*), а также данных до согласования более прогрессивных режимов. В этом режиме передача каждого байта сопровождается взаимо-

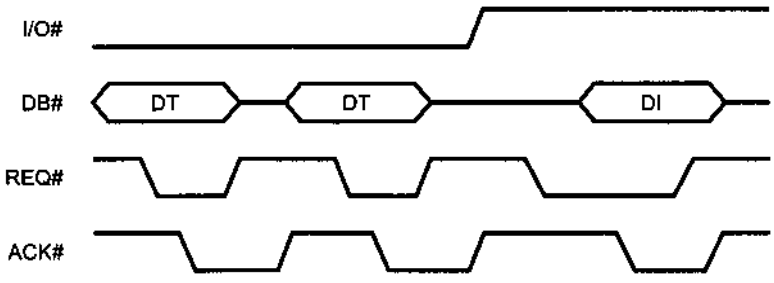


Рис. 20.2. Временные диаграммы асинхронного обмена (DI — данные от инициатора, DT — данные от ЦУ)

связанной парой сигналов REQ#/ACK#: изменение состояние одного сигнала происходит после обнаружения смены состояния другого. В этой связке существенную роль играет задержка распространения сигналов в кабеле. Данные считаются действительными после введения сигнала REQ# (передача к инициатору) или ACK# (передача от инициатора, рис. 20.2). Достоверность информации контролируется битами четности, которые формируются источником данных одновременно с самими данными. Асинхронный обмен поддерживается всеми устройствами для всех фаз передачи информации.

Синхронный режим (ST и DT)

Синхронный режим применяется в фазах передачи данных *Data OUT* и *Data IN* по предварительной «договоренности» устройств. Синхронный режим существенно ускоряет обмен, поскольку здесь отсутствует взаимная блокировка сигналов REQ# и ACK#, сдерживающая темп передачи при существенных задержках распространения сигналов в кабеле (и ограниченном быстродействии интерфейсной части устройств). Синхронный режим имеет две разновидности синхронизации: ST (одиночная) и DT (двойная).

Временные диаграммы передач в режиме *одиночной синхронизации* (ST) приведены на рис. 20.3. В фазе *ST Data IN* (рис. 20.3, *а)* ЦУ передает инициатору серию данных, сопровождаемых стробами REQ#, в темпе, ограниченном установленными временными параметрами выбранного скоростного режима. Инициатор фиксирует принимаемые данные после отрицательного перепада сигнала REQ#, но его ответы (сигналом ACK#) достигают целевого устройства с некоторым опозданием (из-за задержек в кабеле). Как только отставание числа принятых сиг-

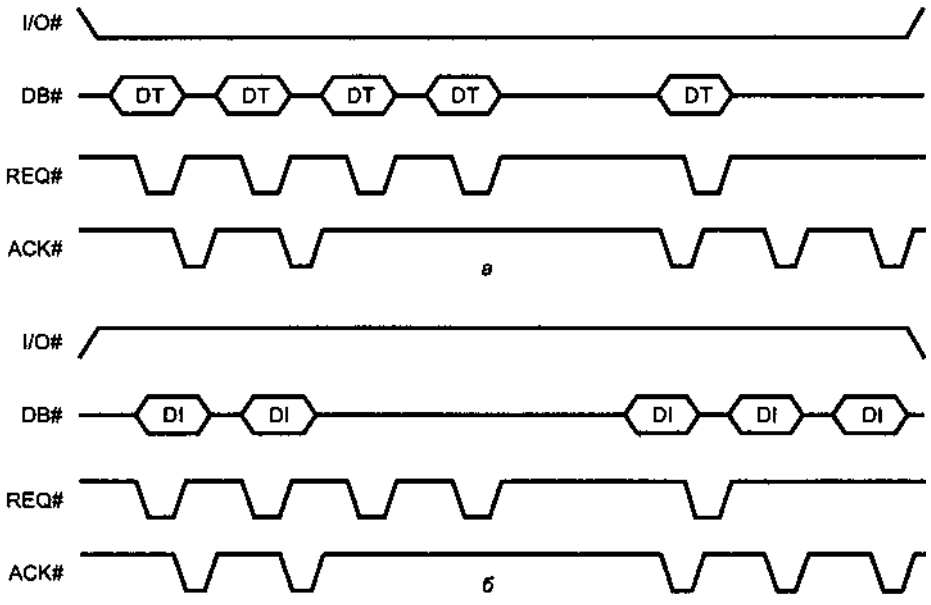


Рис. 20.3. Временные диаграммы синхронного обмена: а - фаза ST Data IN;
б - фаза ST Data OUT

налов ACK# от числа посланных REQ# достигает оговоренного предельного значения (в данном примере — 2), ЦУ приостанавливает обмен до прихода очередного подтверждения ACK#. Операция считается завершенной, когда число принятых подтверждений совпадет с числом посланных запросов. Достоверность информации контролируется битами четности (как и в асинхронном режиме). Фаза *ST Data OUT* выглядит аналогично (рис. 20.3, *б*).

В режиме *двойной синхронизации* (фазы *DT Data In* и *DT Data Out*) на действительность данных указывает *смена состояния* сигнала REQ# или ACK#, а не только спад. Режим двойной синхронизации применяется исключительно в 16-раз- рядных шинах с интерфейсом LVD. В режиме *DT* контроль достоверности осуществляется с помощью 32-разрядных CRC-кодов, которыми снабжаются *группы данных* (data group) и *информационные блоки* (Information Unit, IU). Какие именно структуры передаются в фазе данных — группы или информационные блоки, — определяется при согласовании протокола обмена.

Группа данных представляет собой последовательность слов данных, не имеющую каких-либо заголовков, определяющих их длину. При передаче *информационных блоков* их длина и положение поля CRC известны заранее и источнику, и получателю информации (по содержимому блока см. далее).

Режим одновременного переключения

Режим одновременного переключения (paced mode) предназначен для передачи информационных блоков с максимально уплотненной временной диаграммой. Этот режим используется только в 16-разрядной шине с интерфейсом LVD на скоростях Fast-160 и Fast-320. Параметры режима (скорость и допустимое отставание сигнала ACK# от REQ#) предварительно должны быть согласованы. Кроме того, до передач данных в обоих направлениях должны выполняться «тренировочные» посылки (training pattern), по которым устройства могут подстроить некоторые параметры интерфейсных схем. Во время передачи в режиме одновременного переключения стробирующие сигналы (REQ# или ACK#) меняют свое состояние одновременно со сменой информации на шине данных. Каждый импульс этих сигналов соответствует передаче 32-битного слова.

Информационные блоки

Многообразие типов информационных фаз (команда, состояние, данные, сообщения), различимых с помощью сигналов параллельной шины, плохо соответствует современным реализациям протокола, работающим посредством высокоскоростных блочных передач с CRC-контролем. С целью сокращения этого многообразия в SPI-4 введен протокол обмена информационными блоками, обязательный для устройств, работающих в режиме одновременного переключения. *Информационный блок* (IU) представляет собой последовательность двойных (32-битных) слов, последним из которых является CRC-код (iuCRC). Информационные блоки передаются в фазах данных в определенных последовательностях. В SPI определены информационные блоки нескольких типов. Первым всегда передается блок *IU L_Q,* задающий *связку L_Q* (логическое устройство и тег задания), к которой относится последующий блок, а также тип,

длину и направление передачи этого блока. За ним следует теперь уже известный блок *IU Command* (команда или функции управление заданиями), *IU Data* (данные), *IU Data Stream* (блок данных потока) или *IU Status* (байт состояния или уточненное состояние и список протокольных ошибок). Использование потоков данных позволяет после одного блока *IU L Q* передавать серию блоков данных (уменьшаются протокольные накладные расходы).

Управление шинным интерфейсом — сообщения

Для управления интерфейсом в SCSI определена *система сообщений* (message system), которыми обмениваются инициатор и ЦУ. Обмен сообщениями происходит в фазах *Message IN/OUT*. Сообщения могут быть однобайтными, двухбайтными и расширенными (произвольной длины). Передачей сообщений (как и всеми передачами информации) управляет целевое устройство. Сообщения обеспечивают различные стороны управления интерфейсом:

- установление, завершение и разрыв соединения;
- запуск и принудительное завершение процессов;
- сброс устройства;
- управление указателями;
- обработку ошибок;
- согласование режимов обмена.

Согласование протокола и режимов

Режим передачи, применяемый в фазах данных, а также параметры протокола, относящиеся к любым фазам шины, подлежат согласованию. До предварительного согласования все обмены выполняются в асинхронном режиме с 8-битной разрядностью. Согласование режимов выполняется с помощью сообщений:

- *SDTR* (Synchronous Data Transfer Request) — запрос согласования параметров синхронной передачи *(Synchronous Negotiation*);
- *WDTR* (Wide Data Transfer Request) — запрос согласования разрядности передач;
- *PPR* (Parallel Protocol Request) — запрос согласования параметров протокола, синхронного обмена и разрядности (параметры протокола относятся к потоковым передачам, ускоренному арбитражу, использованию информационных блоков, двойной синхронизации и некоторым другим свойствам).

Согласование выполняется для каждой пары портов (инициатора и целевого устройства). Каждый порт проверяется, чтобы определить, было ли согласование с каждым из возможных партнеров.

Управление режимами портов

В системе команд SCSI имеются команды установки *(Mode Select)* и опроса *(Mode Sense)* режимов работы портов и логических устройств, определяющих поведение (свойства) устройства на шине. Каждой группе свойств соответствует своя *страница* (page) или *подстраница* (subpage).

- *Страница управления отключением-подключением* (disconnect-reconnect mode page) определяет условия разрыва и возобновления соединения при исполнении команды.
- *Страница управления логическим устройством* (logical unit control mode page) пока несет только идентификатор протокола.
- *Страница управления режимом* (port control mode page) порта задает таймаут синхронного обмена. Для детального управления режимом имеется ряд подстраниц, позволяющих узнавать возможности порта, согласованные параметры режимов, подстраивать параметры передатчиков и сохранять тренировочные параметры (для режима одновременного переключения).

Процессы ввода-вывода на шине SCSI

Процесс ввода-вывода (I/O process) на шине SCSI заключается в начальном установлении соединения между инициатором и целевым устройством, за которым следует 0 или более последующих восстановлений того же соединения. Все эти соединения относятся к выполнению одного *задания* — одиночной команды или цепочки команд.

На шине процесс (задание) однозначно идентифицируется связкой I_T_L: SCSI-идентификаторами инициатора (I), целевого устройства (T) и логического устройства (L). Если целевое устройство может работать с очередями, то для полной идентификации требуется еще и тег очереди (Q), тогда процесс идентифицируется связкой I_T_L_Q. Способ сообщения идентификатора процесса зависит от протокола шины: в традиционном варианте используются сообщения *Identify*, в новых версиях идентификатор передается в информационных блоках *(L_Q)*.

Каждый процесс ввода-вывода описывается набором из трех *указателей*: для *команды*, *состояния* и *данных*. Устройства SCSI (инициатор и ЦУ) обеспечивают сохранение одного или нескольких наборов указателей (в зависимости от числа конкурирующих процессов, с которыми может оперировать устройство). Указатели задают местоположение очередного байта команды, состояния и данных во внутренней памяти устройства.

В любой момент времени на шине выполняется лишь один (текущий) процесс. Инициатор имеет *текущий* (активный) *набор указателей* (только один), в который копируется сохраненный набор указателей для текущего процесса. Когда ЦУ отсоединяется от шины, информация о текущем процессе ввода-вывода содержится в сохраненном наборе указателей. При повторном соединении ЦУ может потребовать у инициатора скопировать сохраненный набор в текущий и продолжить выполнение команд данного процесса ввода-вывода. Штатным завершением выполнения каждой команды SCSI является передача *байта состояния* (status byte) от ЦУ к инициатору.

Выполнение команд в традиционном варианте протокола

Традиционный вариант протокола (информационные блоки запрещены) основан на использовании фаз *Message* для идентификации процесса (L Q) и фаз

Command и *Status* для передачи команды и состояния. Традиционный протокол позволяет передавать только 6-битный идентификатор LUN и однобайтный тег очереди.
Инициатор запускает процесс, получив доступ к шине (фаза *Arbitration)* и выбрав целевое устройство (фаза *Selection).* С этого момента выбранное ЦУ удерживает шину в занятом состоянии и управляет фазами шины. В процессе исполнения команд ЦУ управляет модификацией указателей в инициаторе, посылая ему соответствующие сообщения.
Выбрав целевое устройство, инициатор (запросив фазу *Message OUT* сигналом ATN#) передает сообщения *Identify* для *установления соединения* с логическим устройством, имеющим требуемый идентификатор LUN. В этом сообщении указывается и возможность временного разрыва соединения по инициативе ЦУ. Если используется очередь, то вслед за *Identify* должно посылаться и сообщение, определяющее отношение данной команды к существующим очередям и тег задания. Инициатор может посылать и дополнительные сообщения для управления параметрами соединения. Если инициатор не запрашивает посылку сообщения *Identify*, подразумевается, что в установленном соединении LUN = 0 и тег очереди передаваться не будет.
Далее в фазе *Command* ЦУ принимает от инициатора команду. Передачи данных для этой команды ЦУ организует в фазах *Data IN* (к инициатору) и *Data OUT* (от инициатора). В фазе *Status* ЦУ передает инициатору байт состояния выполнения команды (см. табл. 20.1). Если выполнена одиночная команда или последняя команда в цепочке, ЦУ посылает сообщение *Task Complete* и освобождает шину (фаза *Bus Free).* На этом выполнение процесса завершается.
Если выполнена не последняя команда в цепочке, ЦУ посылает сообщение *Linked Command Complete* (требование продвижения указателей на следующую команду), шина остается занятой, процесс продолжается.
В случае возникновения особых условий ЦУ передает состояние *Check Condition*, и инициатор выполняет соответствующую процедуру обработки этой ситуации.
Если ЦУ решило освободить шину до окончания выполнения команды, оно посылает сообщение *Disconnect* (разрыв соединения) и освобождает шину. Целевое устройство, временно освободившее шину, для завершения процесса должно восстановить соединение. Для этого ему требуется получить доступ к шине (фаза *Arbitration*), вызвать инициатора данного процесса (фаза *Reselection)* и послать сообщение *Identify,* содержащее номер ЛУ, а затем сообщение *Simple Queue Tag,* содержащее тег задания. По этому сообщению инициатор возвращает набор указателей, соответствующий данной связке I_T_L_Q, в активный набор, и выполнение процесса продолжается, как было описано выше. Если ЦУ хочет отсоединиться, когда часть данных уже передана (например, головка диска дошла до конца цилиндра и требуется время на позиционирование), оно сначала посылает сообщение *Save Data Pointer*, затем — *Disconnect*. После повторного соединения передача данных возобновляется с точки, определенной последним сохраненным значением указателя. Если происходит ошибка или исключение, ЦУ может повторить обмен данными, послав сообщение *Restore*

Pointers или отсоединившись без сообщения *Save Data Pointers.* С этого момента выполнение процесса продолжается вышеописанным способом.

Для *управления заданиями* инициатор использует соответствующие сообщения. Если требуется управление заданиями того устройства, которое в данный момент занимает шину, инициатор подает сигнал ATN# и в последующей фазе *Message Out* передает сообщения управления заданиями. Инициатор может вынудить ЦУ освободить шину, послав ему сообщение *Disconnect*. Еще он может запросить аппаратный сброс устройства, а также прекращение и сброс процессов (всех или выборочно).

Если требуется управлять заданиями другого устройства, инициатор, получив доступ к шине, выбирает это устройство, подает сигнал ATN# и посылает необходимые сообщения.

Здесь мы не рассматриваем различные ситуации, приводящие к отклонениям от нормальной последовательности событий интерфейса. К ним относятся некорректные соединения со стороны инициатора, выбор несуществующего ЛУ, неожиданные выборки инициатора, округление параметров, реакция на асинхронные события и т. п.

Протокол с передачей информационных блоков

Протокол с передачей информационных блоков характерен для версий интерфейса с двойной синхронизацией. Применение этого протокола предварительно согласуется. При разрешенном использовании информационных блоков функции, прежде выполнявшиеся передачей сообщений, информации состояния и команд, выполняются путем передачи блоков в фазах *DT Data.* Общая идея организации выполнения заданий на шине SCSI сохраняется, но появляется ряд особенностей.

После получения доступа к шине (нормальным или ускоренным арбитражем) связь инициатора и ЦУ (I_T) устанавливается, как и в традиционном протоколе. Целевое устройство управляет процессом ввода-вывода, посылая блоки *IU* L_Q, в которых указываются тип, направление и длина последующего передаваемого информационного блока. Для *идентификации логического устройства и тега* требуется передача информационного блока *IU L_Q,* в котором поле LUN имеет размер 8 байт (согласно модели SAM), а тег — 2 байта. Команды, данные и информация состояния передаются с помощью соответствующих информационных блоков (*Command, Data, Data Stream, Status*).

Фазы сообщений используются ограниченно: для арбитража (*QAS Request*), освобождения шины *(Disconnect),* сигнализации о некоторых ошибках *(Initiator Detected Error*; *Message Parity Error*; *Message Reject*), неготовности сообщения *(No Operation)* и согласования протокола *(PPR),* режима *(SDTR)* и разрядности (*WDTR*).

Физический и электрический интерфейсы Кабели и разъемы

Шинное соединение устройств SCSI обеспечивается кабелями-шлейфами или цепочечным соединением устройств. Внешние устройства имеют по два разъе-

ма, с помощью которых и организуется цепочечное соединение. Внутренние устройства имеют по одному разъему и соединяются кабелем-шлейфом, на котором устанавливаются несколько разъемов. При необходимости кабели могут сращиваться через специальные переходные разъемы, причем только концевые; Т-образные ответвления кабелей недопустимы[1]. На шасси, в которые устанавливаются SCSI-устройства, шина организуется на кросс-плате. Более сложные конфигурации соединений обеспечивают устройства-экспандеры (см. далее).

В качестве шлейфа используется плоский (ленточный) или витой (каждая пара проводов скручена) кабель. Разъемы на шлейф устанавливаются наколкой (IDC-разъемы). Для внешних соединений, как правило, применяется круглый кабель, состоящий из витых пар. У всех разъемов кабеля контакты одноименных цепей соединяются «один в один».

Допустимая длина кабеля зависит от версии интерфейса, скорости и числа устройств (табл. 20.5). Наибольшую дальность (25 м) обеспечивают дифференциальные интерфейсы при соединении пары устройств. Для варианта SE длина 3 м допустима только при малом (до 4) числе устройств. При подсчете суммарной длины кабеля следует учитывать возможность использования одного порта хост-адаптера одновременно для внешних и внутренних подключений и, в случае такого подключения, суммировать длины внутренних и внешних кабелей.

Таблица 20.5. Максимальная длина кабелей SCSI

Скорость, тип	Async	Fast-5	Fast-10	Fast-20	Fast-40	Fast-80...160	Fast-320
HVD (ST)	Точка–точка – 25 м, шина – 12 м					–	–
SE (ST)	6 м	3 м	3 м/1,5 м	–	–	–	–
LVD	Точка–точка – 25 м, шина – 12 м						20 м/10 м

В настоящее время ассортимент разъемов, применяемых в устройствах SCSI, довольно широк, что иногда заставляет использовать переходные адаптеры. Разъемы различаются как по числу, так и по форме и размеру контактов (о назначении контактов см. далее). Практически все разъемы двухрядные, и раскладка цепей рассчитана на чередование сигнальных и обратных проводов. Исключение составляют разъемы DB-25, у которых число «земляных» контактов меньше, чем сигнальных, и экзотические трехрядные DB-50. Ниже описаны применяемые типы разъемов:

- IDC-50 — разъемы для соединения внутренних устройств (как в ATA, где применяются 40-контактные разъемы IDC-40). Разъемы имеют квадратные штырьковые контакты с шагом 0,1" (2,54 мм), пластмассовый корпус без кожуха и дополнительных фиксаторов (рис. 20.4, *а)*. На устройствах устанавливают *вилки* (IDC-50M), на ленточных кабелях — *розетки* (IDC-50F).
- CX-50 — разъемы типа Centronics, аналогичные применяемым в принтерах (но 50-контактные). Разъемы имеют пластинчатые контакты с шагом 0,085"

[1] Допускается длина отвода до 10 см, сюда входит длина проводника от ответвления до входа микросхемы приемопередатчика.

(2,16 мм) и внешний металлический кожух (рис. 20.4, *б)*. Применяются для соединения внешних устройств. На корпусе устройства (и SCSI-адаптера) устанавливают *розетки* (CX-50F), на кабелях — *вилки* (CX-50M). Разъемы фиксируются проволочными скобами, установленными на розетке и входящими в выемки на корпусе вилки. Часто называются внешними разъемами SCSI-1 (SCSI-1 External).

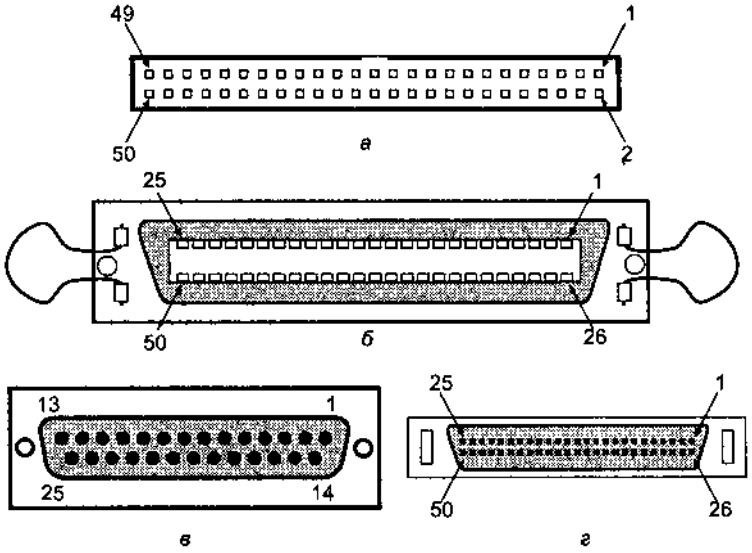


Рис. 20.4. Разъемы 8-битного устройства SCSI: а — IDC-50F; *б* - CX-50F; *в* - DB-25F; г - HD-50F

- DB-25 — разъемы с круглыми штырьковыми контактами в металлическом кожухе D-образной формы (как на LPT-порте компьютера). На устройстве устанавливается *розетка* (DB-25F), на кабеле — *вилка* (DB-25M); фиксация выполняется с помощью винтов (рис. 20.4, *в)*. Применяются на некоторых внешних устройствах (например, Zip).
- HD-50, они же MiniD50 (рис. 20.4, г), — разъемы со штырьковыми контактами в металлическом кожухе D-образной формы, с высокой плотностью контактов (High Density) — с шагом 0,05[м] (1,27 мм). На устройстве устанавливается *розетка* (HD-50F), на кабеле — *вилка* (HD-50M); фиксация выполняется с помощью защелок (клипсов). Часто называются внешними разъемами SCSI-2 (SCSI-2 External).
- HD-68, они же MiniD68 — аналогичные разъемы, но с 68 контактами. На устройстве устанавливается *розетка* (HD-68F или MiniD68F), на кабеле — *вилка* (HD-68M или MiniD68M). Внешние разъемы фиксируются с помощью клипсов или винтов, внутренние — только на трении. Часто называются разъемами SCSI-3, в настоящее время наиболее широко используются для «широкого» интерфейса. На рис. 20.5 показан внешний разъем (слева изображена клипса, справа — резьбовая букса).

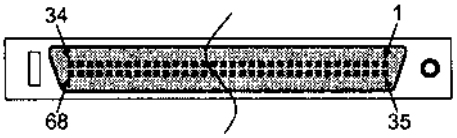


Рис. 20.5. Разъемы 16-битного устройства SCSI HD-68F

- VHDCI-68 — внешние разъемы с особо высокой плотностью (Very High Density Connector, VHDC), контакты в стиле Centronics с шагом 0,8 мм. Применяются нечасто, иногда их ошибочно называют разъемами SCSI-4 или SCSI-5.
- MCX (Micro-centronics) — разъемы в стиле Centronics, но в миниатюрном варианте. Наиболее распространены разъемы MCX-68 и MCX-80, более известные как SCA.
- SCA (Single Connector Attachment — подключение устройства одним разъемом) — разъем предназначен для подключения дисков, устанавливаемых в шасси с возможностью «горячей» замены (или, по крайней мере, легкой, через лицевую панель). В настоящее время распространена спецификация SCA-2 на разъемах MCX-80 (рис. 20.6). На устройстве устанавливается *вилка* (MCX- 80F), на шасси — *розетка* (MCX-80M). Помимо интерфейсных сигналов на разъем выводятся шины питания, а также сигналы конфигурирования устройства (идентификатор, режимы и т. п.). На боковых направляющих имеются дополнительные контакты заземления. Конфигурационные джамперы устанавливаются не на устройстве, а на шасси (или на плате адаптера).

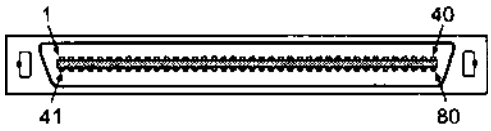


Рис. 20.6. Разъем для устройства SCSI с «горячей» заменой SCA-80

Для версии Narrow SCSI использовались разъемы, изображенные на рис. 20.4, для Wide SCSI — на рис. 20.5. Для устройств с «горячей» заменой применяют миниатюрный D-образный разъем SCA-2, общий для питания и сигнальных цепей (см. рис. 20.6).
Ассортимент кабелей SCSI довольно широк:

- *А-кабель.* Стандартный для 8-битного интерфейса, 25 пар проводов. Для внутренних устройств используется плоский ленточный кабель, для внешних — круглый кабель, состоящий из 25 витых пар в общем экране:
  - внутренний А-кабель SCSI-1 и SCSI-2 имеет разъемы с низкой плотностью контактов IDC-50 (розетки, см. рис. 20.4, а);
  - внешний А-кабель SCSI-1 имеет разъемы Centronics-50 (CX-50M, см. рис. 20.4, *б);*
  - внешний А-кабель SCSI-2 имеет разъемы MiniD50M (HD-50M, см. рис. 20.4, *в).*

- *B-кабель*. Малораспространенный 16/32-битный расширитель SCSI-2.
- *P-кабель*. 8/16-битный кабель с 34 парами проводов, снабжен улучшенными миниатюрными экранированными разъемами. Применяется в интерфейсах SCSI-2/3, в 8-битном варианте контакты 1-5, 31-39, 65-68 не используются:
  - внутренний P-кабель SCSI-3 имеет разъемы HD-68M без фиксаторов;
  - внешний P-кабель SCSI-3 имеет разъемы MiniD68M с фиксаторами;
  - внешний P-кабель SCSI SPI-2 имеет разъемы с особо высокой плотностью VHDCI-68M (иногда такой кабель ошибочно называют кабелем SCSI-4 или SCSI-5).
- Q-*кабель*. 68-проводное расширение до 32 бит (в паре с P-кабелем) аналогичной конструкции. Реально Q-кабель так и не использовался, в спецификации SCSI SPI-3 он уже не упоминается.
- *Mac SCSI*. Кабель с разъемами DB-25P (см. рис. 20.4, г) — 8-битный, стандартный для Macintosh (назначение контактов см. далее), используется на некоторых внешних устройствах (Iomega ZIP Drive). Встречается иная раскладка цепей, если 25-контактный разъем установлен на хост-адаптере.

Существуют также различные варианты кабелей-переходников (с разнотипными разъемами) и адаптеров. Адаптеры представляют собой печатную плату или монолитную конструкцию с разнотипными разъемами. У адаптеров, соединяющих шины разной ширины, может присутствовать терминатор (должен быть отключаемым!) старшего байта (см. далее). Адаптер для подключения SCA к обычной шине имеет стандартный разъем питания, а также набор джамперов, задающих конфигурацию устройства.

Назначение контактов разъемов кабелей приведено в [1], [5], [6], [8]. Неудобство вызывает система нумерации контактов, которая различна для внешних и внутренних разъемов. Однако физическая раскладка проводов на разъеме и в плоском кабеле одинакова, сигнальные линии (прямые) перемежаются обратными проводами (заземленными в устройствах SE).

Терминаторы

Каждая физическая шина SCSI должна оканчиваться терминаторами, устанавливаемыми на обоих ее концах. Терминаторы могут быть как внутренними (установленными внутри контроллеров и периферийных устройств SCSI), так и внешними — маленькими блоками, устанавливаемыми на разъем кабеля или дополнительный разъем последнего устройства. Терминаторы шины SCSI должны решать две задачи: избавить линии шины от отражений сигналов с ее концов за счет согласованной нагрузки и обеспечить требуемый уровень сигнала пассивных линий.

Из сказанного становится понятно, что пренебрежение правилами установки терминаторов может «сойти с рук», когда шина не очень длинная или когда устройств мало (скажем, контроллер и один винчестер), а скорость обмена невелика.

Наиболее популярны устройства SE и LVD, они совместимы: устройство LVD/ SE будет работать в режиме SE. Для режимов SE и LVD различаются как способ передачи по сигнальным линиям, так и режим терминации. Существует несколько разновидностей терминаторов:

- *Пассивные терминаторы SE* (рис. 20.7, *а)* имеют импеданс 132 Ом, что плохо согласуется с ленточным кабелем шины. Эти терминаторы пригодны лишь для «обычного» интерфейса SCSI (скорость передачи до Fast-5). Для режимов Fast-10 и выше они непригодны.
- *Активные терминаторы SE* (рис. 20.7, *б)* имеют импеданс 110 Ом, что позволяет их использовать на более высоких скоростях (Fast-10, Fast-20). Их «активность» заключается лишь в наличии внутреннего источника опорного напряжения (ИОН) +2,85 В, питающегося от линий TermPWR. Микросхемы активных терминаторов имеют и электронные ключи, включенные последовательно в каждую линию. Ключи управляются общим сигналом, позволяющим включать-отключать терминатор.
- *Терминаторы FPT SE* (FPT означает Forced Perfect Terminator) — улучшенный вариант активных терминаторов с диодными ограничителями выбросов, применяемые в высокоскоростных версиях интерфейса SE.
- *Терминаторы для LVD* (рис. 20.7, *в)* имеют дифференциальный импеданс 105 Ом (линейный — 150 Ом). Здесь два источника опорных напряжений обеспечивают смещение 112 мВ между прямым и обратным проводами (в их пассивном состоянии).
- *Универсальные терминаторы LVD/SE* сочетают в себе свойства активных терминаторов SE и дифференциальных терминаторов LVD, а также имеют схему определения режима и цепи коммутации каждого провода (прямого и обратного) шины SCSI на соответствующие терминирующие цепи.

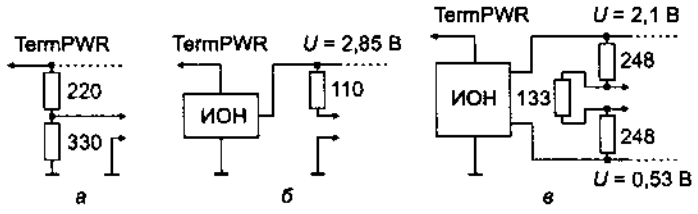


Рис. 20.7. Терминаторы SCSI: а — SE (пассивные), б — SE (активные), *в* — LVD

Универсальные терминаторы LVD/SE, как и остальные устройства, определяют режим работы шины по линии DIFSENSE. В старых устройствах SE контакт разъема, соответствующий этой линии, был заземлен. В устройствах LVD на этот контакт выведен потенциал 1,3 В, в устройствах HVD — выше 2,1 В. В терминаторе имеются компараторы, сравнивающие сигнал этой линии с эталонами, и логика, переключающая режим терминатора (если обнаруживается HVD, терминатор отключает все свои цепи).

По исполнению терминаторы могут быть как *внутренними* (размещенными на печатной плате устройства), так и *внешними* (устанавливаемыми на разъемы

кабеля или устройства). Внутренние терминаторы на каждом устройстве могут быть включены или выключены. В старых устройствах (SCSI-1) для включения терминаторов нужно было установить набор перемычек или вставить в специальную «кроватку» сборку резисторов. Активные терминаторы включаются-выключаются перестановкой одного джампера или даже бесконтактно — программно при конфигурировании устройства. Возможно даже автоматическое включение терминатора (если таковая возможность поддерживается устройством и разрешена при конфигурировании). Внешние терминаторы выглядят как разъемы с небольшой крышкой, под которой смонтирована их «начинка». Несмотря на внешнюю простоту, они имеют ощутимую цену — терминатор для Ultra-Wide SCSI стоит 10—15 долларов. Внешние терминаторы устанавливаются и снимаются только вручную.

Внутренние терминаторы (или, по крайней мере, панелька для их установки) имеются практически во всех устройствах, интерфейс которых не является интерфейсом LVD. В устройствах с интерфейсом LVD терминаторы, как правило, отсутствуют в целях экономии: когда на шину устанавливается несколько устройств, терминатор используется лишь в последних. Однако при подключении одного устройства экономия на цене устройства незаметна, а вот расходы на приобретение терминатора вполне ощутимы.

ИМАНИЕ----------

Отсутствие терминаторов на устройствах с LVD не означает отказа от правил терминации!

Экспандеры

Чисто шинная физическая топология не всегда удобна при построении систем с большим количеством устройств. Для расширения топологических возможностей шины могут применяться *устройства-экспандеры*, объединяющие пару сегментов шин в единый SCSI-домен. *Сегмент шины* (bus segment) — это набор электрических проводников с парой терминаторов на концах, объединяющий порты устройств. Домен может состоять из множества сегментов, соединяемых друг с другом экспандерами. Домен может быть гомогенным (с одинаковым типом приемопередатчиков для всех сегментов) или гетерогенным (с разнотипными приемопередатчиками). Для построения гетерогенного домена применяются гетерогенные экспандеры (LVD/SE). Применение экспандеров позволяет расширять топологические возможности домена:

- ♦ увеличивать число соединяемых устройств, если оно ограничивалось возможностями кабельной шины (см. табл. 20.5 и комментарии к ней);
- ♦ преодолевать ограничения на расстояние между устройствами: сегменты с устройствами можно соединять «ненаселенными» (двухточечными) сегментами, для которых допустима большая длина;
- ♦ бороться со смещением нулевого потенциала (экспандеры электрически разрывают сигнальные цепи);

- обеспечивать динамическое подключение-отключение целых фрагментов домена;
- объединять устройства с разнотипными приемопередатчиками без деградации общей производительности домена.

Простой экспандер (simple expander) не имеет собственных SCSI-идентификаторов и для портов инициаторов и ЦУ, расположенных в разных шинах, он полностью прозрачен. Экспандер не является источником запросов арбитража или сообщений. Экспандер может быть подключен к шине в любом месте, допустимом для подключения обычного устройства. Для большинства сигнальных линий экспандер выполняет лишь ретрансляцию с учетом шинного протокола, внося свою лепту в общую задержку распространения сигналов. Сигнал RST# с любой стороны транслируется на противоположную независимо от остальных сигналов на любой стороне. Экспандер не объединяет линий TermPwr со своих противоположных сторон и не связывает (электрически или логически) сигналы DiffSens. При смене режима приемопередатчиков (LVD/SE) на одном сегменте экспандер вызывает сброс (RST#) противоположного сегмента.

Коммуникативный экспандер (communicative expander) обладает дополнительными возможностями, предоставляемыми поддерживаемым им *протоколом ECP* (Expander Communication Protocol):

- предоставление инициатору возможности определения топологии домена (местоположения экспандеров) и адресации экспандеров;
- управление параметрами портов (margin control) экспандера;
- поддержка многопортовых экспандеров и селективное управление (разрешение/запрет работы) дальними портами;
- сообщение сохраненных тренировочных параметров (для режима одновременного переключения);
- сообщение возможностей экспандера.

Коммуникативный экспандер после шинного сброса работает как простой экспандер до тех пор, пока инициатор не разрешит работу протокола ECP. Впоследствии инициатор может запретить работу протокола ECP.

При построении SCSI-домена с использованием экспандеров должны выполняться следующие правила.

- Каждый сегмент шины должен удовлетворять требованиям к шине (длина, число устройств, характеристики кабеля и т. п.) и иметь терминаторы на обоих концах.
- Если два шинных сегмента соединяются с помощью экспандера или промежуточного сегмента (с парой экспандеров), то эта связующая часть должна поддерживать режим передачи не хуже, чем максимальный общедоступный для этой пары сегментов. К режимам передачи относятся скорость и разрядность.
- Максимальная задержка распространения сигнала между парой устройств в домене не должна превышать 400 нс.

- Число устройств в домене не должно превышать предела адресации (8 устройств при наличии «узких» шин или 16 устройств при их отсутствии).
- Кольцевые соединения недопустимы.

Примеры правильных вариантов топологии домена приведены на рис. 20.8, *а* и *б*. Из второго правила следует, что у гетерогенного домена должно быть мощное ядро, а понижение режима (монотонное) разрешается только в сторону периферии. В ошибочной конфигурации, приведенной на рис. 20.8, в, это правило нарушено.

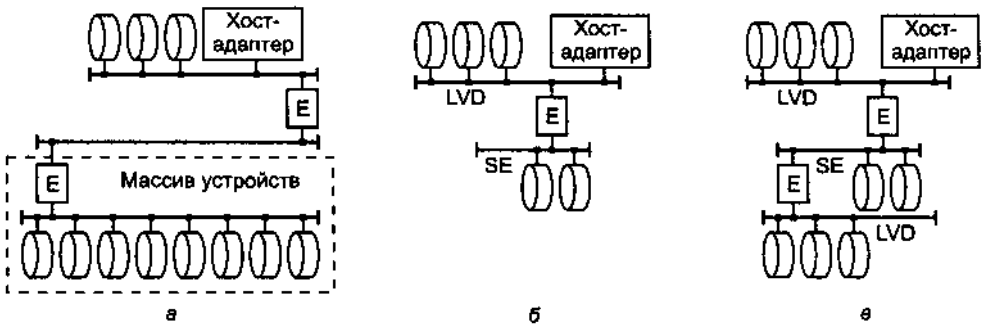


Рис. 20.8. Примеры топологий SCSI-доменов с использованием экспандеров: а, *б* — правильные, *в* — неправильная

Подключение устройств к шине

Подключение устройств к шине SCSI относительно несложно, но имеются нюансы при смешении разнотипных устройств на одной шине. Пропускная способность шины SCSI, «освоенная» компьютером, определяется, естественно, возможностями хост-контроллера. Шина SCSI обеспечивает хорошую совместимость устройств с параллельными интерфейсами разных поколений, «узких» и «широких», но зачастую одно старое устройство способно свести на нет возможности новых устройств, подключенных к шине. По типу интерфейса совместимыми являются только SE и LVD.

ВНИМАНИЕ ----------

Смешивать устройства LVD с HVD на одной шине нельзя!

Устройства LVD можно использовать на одной шине с SE, но при этом все устройства перейдут в режим SE и шина не сможет работать в режиме Ultra2, свойственном устройствам LVD. Интерфейс LVD, являясь дифференциальным, требует, чтобы каждый обратный провод (положительный сигнал) приходил на вход своего приемника; в версии SE все обратные провода на устройстве соединялись вместе и подключались к шине GND. Если на шине с устройствами LVD имеется хотя бы одно устройство SE, то линия DIFFSENS оказывается заземленной и все устройства LVD переходят в режим SE. При конфигурирова

нии устройство LVD может быть принудительно переведено в режим SE установкой джампера «Force SE».
Если на шине присутствуют устройства Ultral60 и Ultra2 (или еще ниже), то шина будет работать в самом «низком» из этих режимов. Принудительно запретить режим Ultral60 (понизить до Ultra2) можно джампером «Disable U160».
Подключение «узкого» устройства к «узкой» шипе — самая простая задача, поскольку здесь обычно встречаются лишь два типа разъемов (не считая Mac SCSI) — внешний (типа Centronics) и внутренний. Устройства должны быть сконфигурированы (см. выше), каждому должен быть назначен уникальный (на шине) идентификатор SCSI ID, формально — любой в диапазоне 0-7. Длина шины не должна превышать допустимого предела, на обоих концах шины (и только там!) должны быть установлены и включены терминаторы. На линию TERMPWR должно подаваться питание (чаще от хост-адаптера), что можно проверить, замерив напряжение на соответствующих контактах разъема.
Подключение «широкого» устройства к «широкой» шине может оказаться несколько сложнее, поскольку здесь больше разнообразия в разъемах. Из-за этого может потребоваться применение переходных адаптеров от одного типа разъема к другому. Также могут возникнуть сложности с подключением терминаторов, особенно для устройств LVD, среди которых внутренние терминаторы встречаются редко, а внешние могут занимать отдельный разъем на кабеле. Идентификаторы устройств можно задавать в диапазоне 0-15.
Подключение «узкого» устройства к «широкой» шине требует переходного адаптера с 68-контактного на 50-контактный разъем. Старший байт в этом адаптере не должен терминироваться, если подключаемое устройство не крайнее на шине. Если же устройство крайнее, то в адаптере старший байт должен терминироваться, кроме того терминатор должен быть установлен на самом устройстве. Выбор положения устройства (крайнее или промежуточное) может определяться имеющимся адаптером. Идентификаторы устройств должны устанавливаться в диапазоне 0-7 для всех устройств, поскольку из-за того что идентификаторы 8-15 для узкого устройства невидимы, процедура арбитража не может работать нормально (см. выше). Так как все «узкие» устройства — это устройства SE, линия DIFFSENS оказывается заземленной и все устройства LVD переходят в режим SE. Существуют, однако, адаптеры-мосты, позволяющие при подключении устройства SE остальным устройствам оставаться в режиме LVD. Определить режим можно, замерив напряжение на 16-м контакте 68-контакт- ного разъема (и на 46-м — 80-контактного).
Подключение «широкого» устройства к «узкой» шине также требует специального адаптера, и на «широком» устройстве следует установить джампер «Disable Wide». Дополнительно может потребоваться терминация старшего байта и относящихся к нему управляющих линий, чтобы обеспечить на них надежное пассивное состояние («висящие» входы восприимчивы к помехам). Некоторые версии встроенного микропрограммного обеспечения позволяют работать устройствам и без дополнительных терминаторов. Идентификаторы всех устройств должны быть в диапазоне 0-7 (по тем же соображениям, что и в предыдущем случае).

Рассмотрим различные конфигурации подключения устройств к контроллеру SCSI (рис. 20.9). Контроллер может быть расположен на карте расширения, устанавливаемой в слот PCI или ISA, или же встроен в системную плату. Устройства, подключаемые к нему, могут быть как внутренними (разного рода дисковые и ленточные устройства), так и внешними (те же, а также сканеры и другие периферийные устройства). Терминаторы расставляются, исходя из конкретных условий.

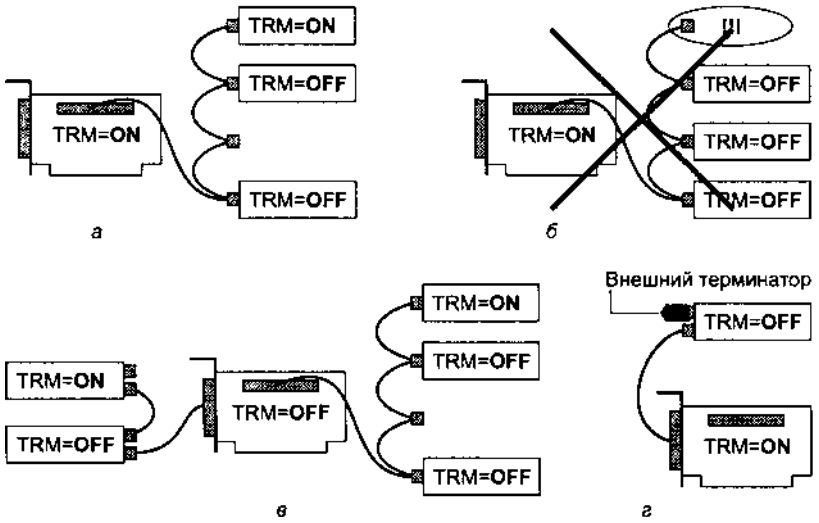


Рис. 20.9. Подключение устройств к карте контроллера SCSI: а, е, *г* — правильно; *б* — неправильно

Правила подключения довольно просты:

- концы кабельного шлейфа не должны висеть в воздухе (как на рис. 20.9, *б)*\
- на устройствах, подключенных к концам шлейфа, должны быть включены внутренние терминаторы (на рисунке обозначено как TRM = ON) или же установлен внешний терминатор (рис. 20.9, г);
- на промежуточных устройствах терминаторы должны быть отключены (TRM = = OFF).

Если контроллер SCSI смонтирован на дополнительной интерфейсной карте, то разъем, к которому подключаются внешние устройства, располагается довольно близко к внутреннему, так что длинной линии между ними нет. В этом случае терминация внешнего разъема хлопот не доставляет: когда используется только внутреннее (рис. 20.9, *а)* или только внешнее (рис. 20.9, г) подключение, терминатор на контроллере *включают*. Когда используется и внешнее, и внутреннее подключение (рис. 20.9, в), терминатор на контроллере *отключают*.

Если используется внутреннее подключение, а внешние устройства подключаются не все время, то приходится переключать терминатор контроллера в соответствии с текущей конфигурацией. В старых контроллерах приходилось вскры

вать системный блок и переставлять джамперы. В новых контроллерах вскрытия не требуется — терминаторы включаются/отключаются программно (утилитой SCSI Setup) или даже автоматически. Если по какой-либо причине переключать терминатор контроллера не хочется, можно его отключить и пользоваться внешним терминатором, устанавливая его на внешний разъем (снаружи корпуса компьютера), когда внешние устройства не подключены.

Когда контроллер SCSI установлен на системной плате, он имеет лишь один разъем, к которому подключается кабель-шлейф. Если требуется только внутреннее или только внешнее подключение (рис. 20.10, *а* и *б*), то терминатор на контроллере *включают*. Если используется и внутреннее, и внешнее подключение (рис. 20.10, в), терминатор на контроллере *отключают*. Если применяется универсальный кабель-шлейф с внутренними и внешним разъемами (как на рис. 20.10, в), но внешних устройств нет, то терминатор на контроллере должен быть *отключен*, а на внешнем разъеме должен быть установлен *внешний терминатор*.

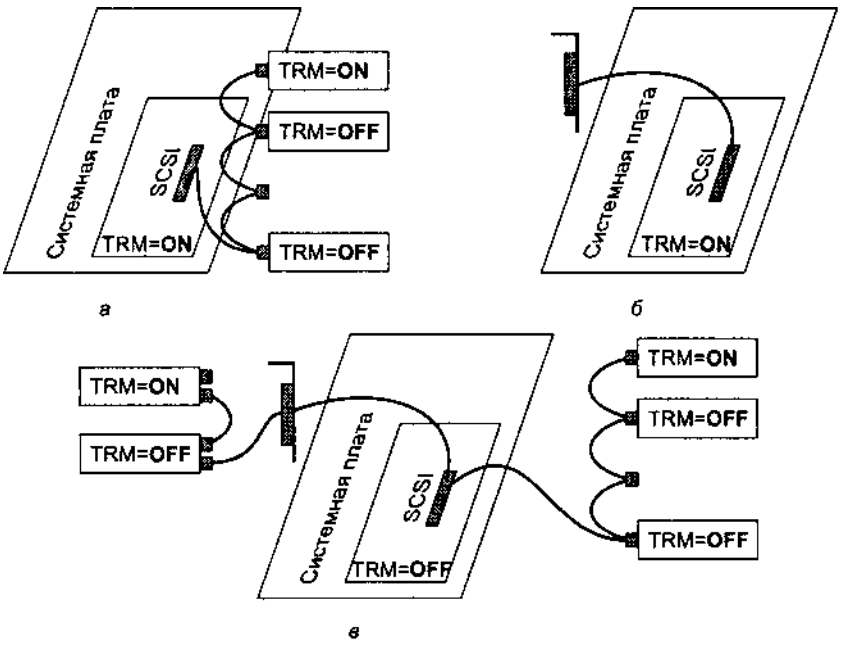


Рис. 20.10. Подключение устройств к интегрированному контроллеру SCSI

Кабели и терминаторы могут входить в комплект адаптеров SCSI или системных плат со встроенным контроллером SCSI, а могут приобретаться отдельно. То, что входит в стандартный комплект поставки, не всегда подходит для конкретного применения. Так, в комплекте с системной платой CT-6BTS, имеющей контроллер Ultra-Wide SCSI, поставляются универсальный «широкий» (wide) шлейф (такой, как на рис. 20.10, в, но с меньшим количеством внутренних разъемов) и «узкий» внутренний. Для подключения только внутренних

дисководов Wide SCSI такого комплекта недостаточно — нужен внешний терминатор! Можно, конечно, отрезать часть шлейфа, идущую от разъема для подключения контроллера к внешнему разъему, но жалко!

Контроллеры Wide SCSI обычно имеют также разъемы для подключения обычных («узких») устройств. На той же системной плате CT-6BTS помимо 68-кон- тактного разъема Wide SCSI имеется и 50-контактный — для обычных устройств. Узкий (8-разрядный) интерфейс можно рассматривать как подмножество широкого (16-разрядного), у которого задействована только младшая половина шины данных. В простых одноканальных контроллерах (как на этой плате) контакты узкого разъема запараллелены с частью контактов широкого. При этом можно объединять широкие и узкие устройства, для чего терминаторы на контроллере разделены на две половины: терминаторы младшего байта (TrmL) и старшего байта (TrmH) могут управляться независимо. Рис. 20.11, а и б, иллюстрирует корректные способы смешанного подключения (устройства с терминаторами на концах шлейфов подразумеваются). На рис. 20.11, *в* приведена некорректная схема — здесь в линиях младшего байта и сигналов управления оказываются три терминатора, что ведет к перегрузке передатчиков. Избежать перегрузки можно лишь не устанавливая терминатора на конце какого-либо шлейфа, но «висячий» конец приведет к отражениям. Заметим, что штатными кабелями из комплекта поставки платы (именно они изображены на рис. 20.11, *в)* корректно смешанное подключение выполнить нельзя.

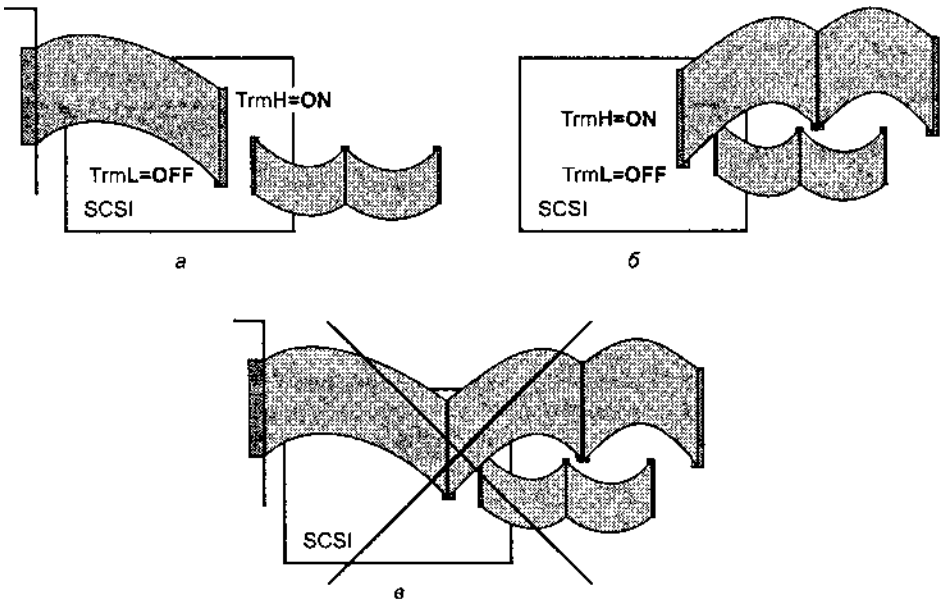


Рис. 20.11. Подключение узких и широких устройств SCSI: а, *б* — правильно; *в* — неправильно

Если следовать вышеприведенным правилам (и не превышать разрешенные длину и число подключений), то при исправном оборудовании шина SCSI

будет работать надежно, как ей и полагается. Если правила нарушать, то возможны варианты. Есть определенные модели контроллеров и устройств, для которых мелкие «шалости» с терминаторами «сойдут с рук». Так, может быть прощен (или почти прощен) висящий конец без терминатора (если он не очень длинный). Устройства могут работать (ОС — загружаться, диски — читаться), но, возможно, с не всегда заметными сбоями. Правда, если используется ОС Windows NT, то, заглянув в журнал регистрации событий (event log), можно увидеть «букет» красных фонариков, связанных с устройствами SCSI. «Пышность» этого букета будет зависеть от тяжести нарушений и «норова» устройств. Есть модели, придерживающиеся «строгих правил», и при нарушении терминации устройства работать вообще не будут. Как поступать в каждом конкретном случае, наверное, дело инсталлятора — на «лишний» терминатор или кабель другой конфигурации может просто не хватить денег. Но теория, увы, такова.

Конфигурирование устройств

Все устройства на шине должны быть согласованно сконфигурированы. Для них требуется программно или с помощью джамперов установить перечисленные ниже основные параметры:

- *Идентификатор устройства* (SCSI ID) — адрес 0-7 (для Wide SCSI 0-15), уникальный для каждого устройства на шине. Обычно хост-адаптеру, который должен иметь высший приоритет, назначается адрес 7 (8 для Wide SCSI, если все устройства 16-битные). Позиционный код, используемый для адресации, обеспечивает совместимость адресации 8- и 16-битных устройств на одной шине. Ряд версий BIOS считает загрузочным только устройство с SCSI ID - 0.
- *Контроль четности* (SCSI Parity). Если хотя бы одно устройство не поддерживает контроля четности, он должен быть отключен для всех устройств на шине. Контроль четности, особенно для дисковых устройств, является необходимым средством защиты от искажения данных при передаче по шине.
- *Включение терминаторов* (Termination). В современных устройствах применяются активные терминаторы, которые могут включаться одним джампером или программным сигналом.

ИМАНИЕ ----------

Правильная установка терминаторов крайне существенна — отсутствие/избыток терминаторов может привести к неустойчивости или неработоспособности интерфейса.

- *Питание терминаторов* (Terminator Power) должно быть включено джампером (или программно) хотя бы на одном устройстве.
- *Согласование скорости синхронного обмена* (SCSI Synchronous Negotiation). Режим синхронного обмена, обеспечивающий высокую производительность, включается по взаимному согласию устройств. Если хотя бы одно устройство на шине его не поддерживает, рекомендуют запретить согласование на хост-адаптере. Если обмен будет инициирован целевым устройством, под

держивающим синхронный режим, «нормальный» хост-адаптер поддержит этот режим. Целевому устройству можно запретить запрос синхронного режима специальным джампером, который может называться «Enable TI-SDTR» (Target Initiated Synchronous Data Transfer Request Negotiation).

- *Разрешение отключения* (Enable disconnection). Устройствам можно отключаться от шины при неготовности данных во время длительных операций с носителем, что весьма эффективно в многозадачном режиме при нескольких ПУ на шине. В случае одного устройства отключение приводит только к дополнительным затратам времени на повторное соединение.
- *Согласование ширины шины данных* (Wide Negotiation) тоже выполняется по протоколу шины, исходя из возможностей обоих участников обмена. Целевому устройству можно запретить запрос 16-битного режима специальным джампером, который может называться «Enable TI-WDTR» (Target Initiated Wide Data Transfer Request Negotiation).
- *Запрет 16-разрядного режима* (Disable wide) позволяет подключить «широкое» устройство к «узкой» шине.
- *Принудительное переключение в линейный режим* (Force SE) позволяет перевести устройство LVD в режим SE независимо от состояния линии DIFFSENS.
- *Запрет синхронизации по обоим фронтам* (Disable U160) позволяет принудительно перевести устройство Ultra3 SCSI в режим Ultra2.
- *Старт по команде* (Start on command), или *запрет автоматического запуска шпиндельного двигателя* (Disable Auto Spin up). При установке этого параметра запуск двигателя устройства выполняется только по команде от хост-адапте- ра, что позволяет снизить пик нагрузки блока питания в момент включения. Хост будет запускать устройства последовательно.
- *Задержанный старт* (Delayed Start) в сочетании с джамперами выбора задержки позволяет автоматически запускать двигатель через указанный интервал после подачи питания (разным устройствам задают различные значения задержки).

20.5. Устройства SCSI с последовательным интерфейсом — SAS

Последовательный интерфейс для подключения устройств SCSI (Serial Attached SCSI, SAS) разработан на основе физического интерфейса Serial ATA. Однако устройства SAS используют свой транспортный протокол (SSP), который отвечает общей идеологии SCSI (см. 20.2). Для подключения устройств SAS к компьютеру служит *хост-адаптер* (HBA) с интерфейсом SAS, к которому можно подключать как устройства SAS, так и устройства SATA — их физические интерфейсы совместимы. Возможные варианты подключения устройств приведены на рис. 20.12.

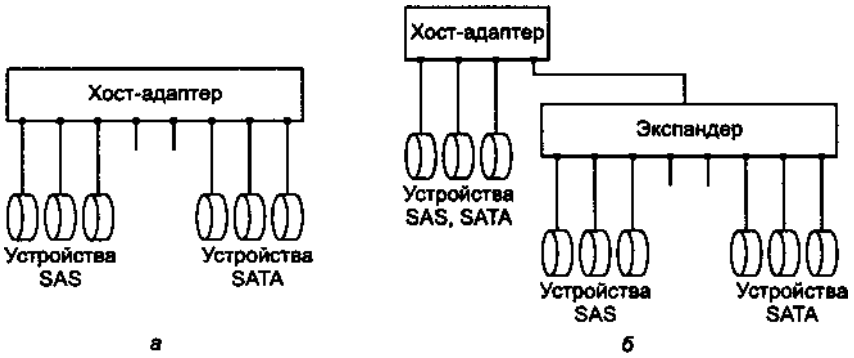


Рис. 20.12. Подключение устройств SAS и SATA к хост-адаптеру: а — прямое, *б* — через экспандер

При прямом подключении к хост-адаптеру (рис. 20.12, *а)* устройства SAS и SATA взаимодействуют с хостом по своим транспортным протоколам (SSP и «родной» протокол SATA соответственно). При использовании экспандеров (рис. 20.12, *б)* число подключаемых устройств может превышать число портов хост-адаптера.

В SAS используется три транспортных протокола:

- SSP (Serial SCSI Protocol) — поддержка *устройств SAS* (дисковых, ленточных и иных устройств SCSI с последовательным интерфейсом);
- STP (Serial ATA Tunneling Protocol) — организация туннелей для взаимодействия с дисковыми устройствами Serial ATA[1] (этот протокол используется, если устройство SATA подключается к экспандеру);
- SMP (Serial Management Protocol) — управление экспандерами SAS (промежуточными интерфейсными устройствами), обеспечивающими связь конечных устройств (инициаторов и целевых устройств).

Устройства, порты и соединения SAS

Подключение устройства SAS обеспечивает *трансивер Phy* — приемник и передатчик последовательного интерфейса. Приемник и передатчик работают на одинаковых скоростях (G1 — 1,5 Гбит/с, G2 — 3 Гбит/с), прием и передача идут независимо друг от друга (полный дуплекс). Каждый трансивер в устройстве имеет собственный идентификатор (номер).

Устройства SAS содержат порты. *Порт* — это группа трансиверов (от 1 до 128) с одинаковыми SAS-адресами, подключенных к другой группе трансиверов с одинаковыми (но другими) SAS-адресами. «Узкий» (narrow) порт содержит один трансивер, «широкий» — два и более трансиверов. Порты определяются (конфигурируются) на этапе инициализации устройства.

[1] Возможна также поддержка устройств ATAPI, хотя двойное преобразование протоколов вряд ли целесообразно.

SAS-адрес — это глобально уникальный 64-битный идентификатор[1], который присваивается каждому SAS-порту и каждому SAS-экспандеру.

В SAS определено два класса устройств: конечные устройства (end device) и устройства-экспандеры (expander device).

Конечные устройства — это устройства SAS, являющиеся инициаторами или/ и целевыми устройствами SCSI (или SATA).

Примером конечного устройства является SAS-диск, у которого два трансивера всегда имеют отдельные SAS-адреса, — устройство двухпортовое (для диска «широкий» порт не требуется).

Другой пример — хост-адаптер, содержащий 8 трансиверов, в котором возможны разные варианты конфигурации:

- один SAS-адрес на все трансиверы — это адаптер с одним, потенциально «очень широким» портом;
- два SAS-адреса, каждый на группу из четырех трансиверов — двухпортовый адаптер (удобен для подключения четырехканальными кабелями);
- восемь SAS-адресов — восьмипортовый адаптер, к которому можно подключить до 8 устройств SAS и SATA.

Устройства-экспандеры служат для объединения конечных устройств в сложных конфигурациях. Экспандер имеет собственный SAS-адрес для управления его функциями по протоколу SMP. Экспандер может содержать и внутренние устройства SAS (со своими SAS-адресами), подключаемые к его портам с *виртуальными трансиверами*. Эти устройства могут, например, использоваться для управления блоком (питание, климат, защита и т. п.). Каждый трансивер экспандера имеет собственный идентификатор (номер), уникальный в пределах экспандера. Порты экспандера могут служить для подключения инициаторов и целевых устройств SAS, а также других экспандеров. По этим портам будут передаваться кадры любых протоколов (SSP, STP, SMP). Экспандер может (не обязательно) содержать *мосты STP/SATA* (один или несколько), позволяющие к портам экспандера (узким) подключать устройства SATA. По этим портам будут передаваться только кадры SATA.

Для *выполнения задания SCSI* (передачи команды, данных или сообщения состояния) необходимо установить *SSP-соединение* (SSP-connection) — временную связь между трансиверами инициатора и целевого устройства. Соединения устанавливаются также и для обмена с устройствами SATA (STP-соединение), и для управления экспандерами (SMP-соединения). Установление соединения — характерная черта SAS, обусловленная идеологией SCSI (в SATA понятие соединения отсутствует).

Для установления соединения устройство посылает запрос, указав SAS-адрес желаемого партнера. Экспандеры, выполняя маршрутизацию запроса, прокладывают *путь* (pathway) — определяют набор физических связей (links) между инициатором и целевым устройством соединения. Этот путь сохраняется до

[1] Его формат совпадает с именем порта (Port_Name) Fibre Channel.

закрытия (или разрыва) соединения, и по нему передаются кадры и примитивы — как по выделенному двухточечному соединению — со скоростью, доступной всем участкам пути.

Соединения адресуются к портам, но, устанавливаются между трансиверами. Широкий порт (с *N* трансиверами) может устанавливать соединения одновременно с *N* различными портами. Два широких порта могут устанавливать друг с другом до *N* соединений. SAS-диски всегда представлены парой «узких» портов, «широкие» порты имеют только хост-адаптеры и RAID-контроллеры.

Топология домена и маршрутизация

Домен SAS в простейшем случае состоит из непосредственно соединенных друг с другом портов инициатора и целевого устройства SSP. Если используются экспандеры, то в домене появляются еще и порты инициаторов и целевых устройств SMP, необходимые для конфигурирования домена (экспандеров). Если используются и устройства SATA, то, соответственно, появляются порты-инициаторы и целевые устройства STP. Возможны сложные конфигурации с несколькими доменами (рис. 20.13).

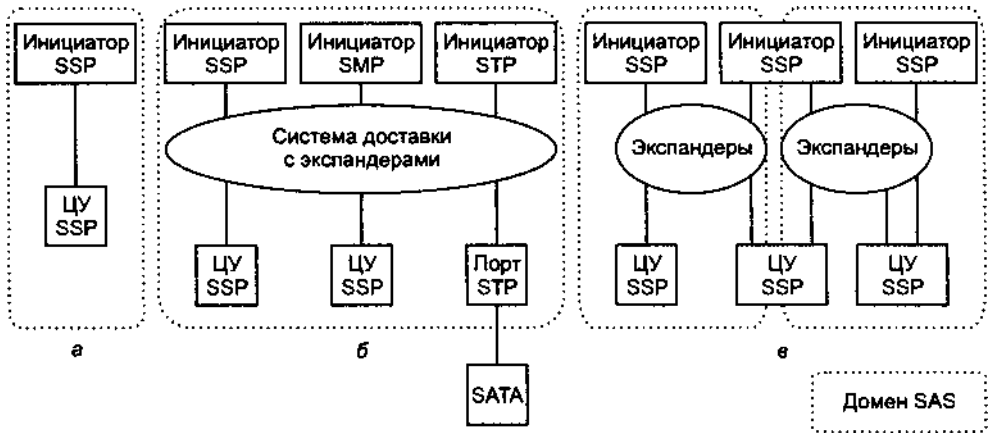


Рис. 20.13. Логическая структура доменов SAS: а — простейшая двухточечная, *б* — *с* экспандерами, *в* — *с* разделяемыми устройствами (инициатором и ЦУ)

Экспандеры по назначению разделяются на два типа:

- из E-экспандеров (edge expander) собираются *наборы E-экспандеров* (edge expander device set), обеспечивающие подключение конечных устройств;
- F-экспандеры (fanout expander) служат для объединения множества наборов E-экспандеров и конечных устройств.

Набор E-экспандеров представляет собой древовидную структуру экспандеров. Набор может вырождаться и в один экспандер. Каждый набор может содержать до 128 SAS-адресов (устройств-экспандеров и портов конечных устройств).

В одном домене SAS может присутствовать не более одного F-экспандера, с помощью которого объединяются до 128 наборов. Если F-экспандеров нет, то в домен может объединяться не более двух наборов. Конечные устройства могут подключаться и к F-экспандерам, и к E-экспандерам (рис. 20.14). Между любыми устройствами (и конечными, и экспандерами) возможны соединения «широкими» интерфейсами, петлевые соединения запрещены, множественные пути отсутствуют.

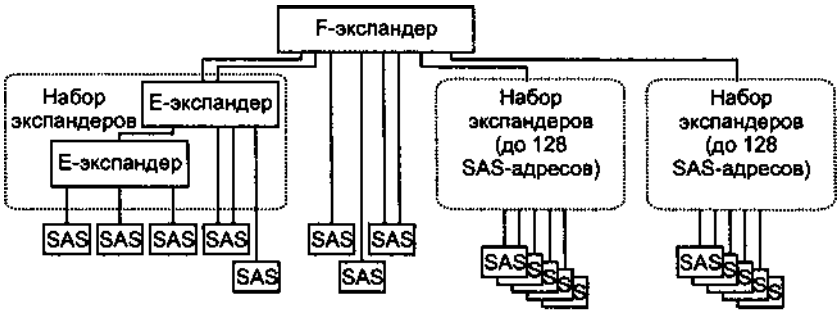


Рис. 20.14. Домен SAS

Маршрутизацию в домене SAS выполняют экспандеры. Выполняется она при установлении соединений (SSP, STP или SMP), внутри соединения обмен между инициатором и целевым устройством происходит по проложенному маршруту. Для маршрутизации экспандеры должны быть предварительно сконфигурированы: заполнены их таблицы маршрутизации. Этим занимается устройство-инициатор протокола SMP. Экспандер может быть *самоконфигурируемым* — у него имеется внутренний инициатор SMP, который заполнит таблицы (свои и других экспандеров).

Приняв запрос соединения от какого-то трансивера (кадр *OPEN),* экспандер по SAS-адресу назначения, содержащемуся в этом кадре, ищет свой трансивер, в который нужно послать данный запрос.

Для каждого трансивера экспандера известны SAS-адрес, тип и допустимая скорость подключенного устройства (эта информация получается на этапе идентификации, завершающем инициализацию физического интерфейса). Если к трансиверу подключено конечное устройство, то к нему выполняется прямая маршрутизация: в этот трансивер посылаются запросы с SAS-адресом, совпадающим с адресом подключенного устройства. Если к трансиверу подключен другой экспандер, то сначала выполняется табличная маршрутизация. В экспандере имеются таблицы маршрутизации, в которых указываются SAS-адреса, доступные для каждого его трансивера. Если для пришедшего запроса в таблицах не найден трансивер, в который его следует послать, то используется субтрактивная маршрутизация. В E-экспандере может присутствовать *субтрактивный порт* (набор из одного или нескольких трансиверов, к которым подключены трансиверы с одинаковыми SAS-адресами). В этот порт посылаются запросы, для которых не удалось найти трансивер при прямой или табличной

маршрутизации. Субтрактивный порт может быть подключен только к порту другого экспандера, и этот порт должен задействовать только табличную маршрутизацию. F-экспандеры не используют субтрактивную маршрутизацию. Они могут служить средством объединения множества наборов E-экспандеров и обычно поддерживают большие размеры таблиц для табличной маршрутизации.

В E-экспандерах различают *нисходящие* (downstream) трансиверы, использующие прямую и табличную маршрутизации, и *восходящие* (upstream) трансиверы, использующие субтрактивную маршрутизацию. На границе набора E-экспандеров может находиться корневой экспандер, восходящий (субтрактивный) порт которого подключается к F-экспандеру или к другому набору E-экспандеров.

Архитектурная модель SAS

Архитектурная модель SAS состоит из набора уровней:

- Физический уровень (physical layer) определяет коннекторы, кабели и электрические параметры приемопередатчиков.
- Phy-уровень (Phy layer) определяет последовательную передачу данных (кодирование 8B/10B) и специальную «внеполосную» (OOB) сигнализацию для служебных целей.
- Канальный уровень (link layer) определяет примитивы, адресные кадры и соединения. На канальном уровне решаются задачи идентификации подключенных устройств, выполнения их сброса, управления соединениями. Для каждого протокола (SSP, STP и SMP) канальный уровень определяет свои правила обмена кадрами и примитивами.
- Уровень порта (port layer) является прослойкой между канальным и транспортным уровнями, обеспечивающей установление и разрыв соединений в портах.
- Транспортный уровень (transport layer) определяет структуры кадров и транспортные сервисы для протоколов SSP, STP и SMP.
- Прикладной уровень (application layer) для протокола SSP определяет процедуры выполнения команд SCSI согласно архитектурной модели SAM. Для протокола SMP прикладной уровень определяет функции, необходимые для идентификации устройств, выяснения топологии домена и управления экспандерами. Для протокола STP прикладной уровень SAS не вводит каких-либо особенностей по сравнению с SATA.

Физический уровень SAS

Физический уровень SAS (physical layer) определяет коннекторы и кабели, а также электрические спецификации приемопередатчиков. Эти спецификации совместимы с SATA, но различаются в некоторых деталях. В SATA заданы характеристики передатчиков и кабелей, которые косвенно задают характеристики приемников; в SAS требования к приемникам заданы явно. В SATA допускается связь между устройствами как по постоянному, так и по переменному току; в приемнике SAS должны присутствовать разделительные конденсаторы

(в передатчике — не обязательно), так что связь имеется только по переменному току. В SATA возможно применение синхронизации с расширением спектра (SSC), в SAS расширение спектра не применяется.

Для SAS используются кабели с волновым сопротивлением 100 Ом, в качестве трансиверов используются те же компоненты, что и для современных последовательных интерфейсов Fibre Channel, Gigabit Ethernet, XAUI, InfiniBand, 1394b, PCI Express.

За основу разъема для внутреннего исполнения (рис. 20.15, а, табл. 20.16) взят коннектор Serial ATA, в котором в сигнальной секции добавлены контакты для вторичного физического интерфейса (secondary physical link). В питающей секции резервный (в SATA) контакт P11 использован для индикации готовности устройства. Подключаемые кабели могут содержать один или два физических интерфейса. Определен и разъем кросс-шины для непосредственного подключения устройств SAS с двумя физическими интерфейсами (устройства SATA с этим разъемом несовместимы из-за L-образных ключей).

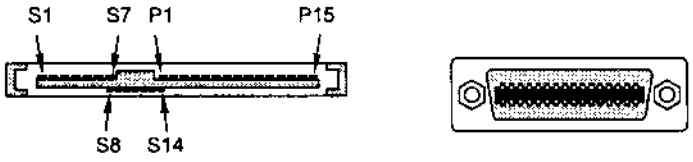


Рис. 20.15. Разъемы устройств SAS: а - внутренний, *б* — внешний

Таблица 20.6. Разъем устройства SAS

Контакт	Цепь	Назначение
S1	GND	Экран
S2	RP+	Дифференциальный вход первичного трансивера Phy
S3	RP–	
S4	GND	Экран
S5	TP–	Дифференциальный выход первичного трансивера Phy
S6	TP+	
S7	GND	Экран
S8	GND	Экран
S9	RS+	Дифференциальный вход вторичного трансивера Phy
S10	RS–	
S11	GND	Экран
S12	TS–	Дифференциальный выход вторичного трансивера Phy
S13	TS+	
S14	GND	Экран
P1	V33	Питание 3,3 В
P2	V33	Питание 3,3 В
P3	V33	Питание 3,3 В, предварительный заряд
P4	GND	Общий

продол.

Таблица 20.6 *(продолжение)*

Контакт	Цепь	Назначение
P5	GND	Общий
P6	GND	Общий
P7	V5	Питание 5 В, предварительный заряд
P8	V5	Питание 5 В
P9	V5	Питание 5 В
P10	GND	Общий
P11	Ready LED	Светодиодный индикатор готовности/активности
P12	GND	Общий
P13	V12	Питание 12 В, предварительный заряд
P14	V12	Питание 12 В
P15	V12	Питание 12 В

Для SAS-1.1 фирма HP предлагает использовать внутренний 4-контактный коннектор, на котором питание и индикация не предусмотрены, но введены контакты для дополнительных интерфейсов. Этот тип разъемов определен в стандарте SFF-8484.

Для внешних соединений в SAS приняты 4-канальные кабели и 32-контактные разъемы (см. рис. 20.15, *б),* у которых сигналы сгруппированы по направлениям (группа входов и группа выходов); питание и индикация не предусмотрены.

Протокол SSP

Протокол SSP обеспечивает выполнение заданий устройствами SCSI. Для работы протокола SSP требуется установление соединения SSP. Запросить соединение SSP может как инициатор, так и целевое устройство. После установления соединения SSP трансиверы инициатора и целевого устройства обмениваются между собой кадрами SSP. После обмена кадрами по взаимному согласию партнеров соединение закрывается. Канальный уровень для SSP обеспечивает *гарантированную доставку* кадров: на прием каждого кадра трансивер посылает примитив *ACK* (положительное подтверждение) или *NAK* (ошибка CRC-контроля). Кадры неверной длины игнорируются.

Канальный уровень обеспечивает *управление потоком* на основе кредитов: для того чтобы передать кадр, передатчик должен иметь ненулевой *кредит*. Кредит ему дает партнер по соединению посылкой примитива *RRDY*. При установлении соединения кредиты обнуляются, получение примитива *RRDY* увеличивает кредит на 1, посылка кадра — уменьшает на 1. Примитивом *CREDIT BLOCKED* устройство предупреждает об отсутствии кредита.

Кадр SSP состоит из примитива *SOF*' двойных слов данных и примитива *EOF*. Обмен выполняется в полнодуплексном режиме: кадры могут передаваться в обоих направлениях. Примитивы-подтверждения и примитивы кредитов могут внедряться в поток двойных слов данных, передаваемых во встречных направлениях.

В протоколе SSP используются блокированные и неблокированные передачи кадров. В случае *блокированной передачи кадров* (interlocked frames) очередной

кадр посылается только после получения подтверждения (*ACK* или *NAK)* в приеме предыдущего. Блокированные передачи используются для кадров *COMMAND, TASK, XFER_RDY, RESPONSE*; темп этих передач сдерживается временем оборота между источником и приемником.

Для кадров данных *(DATA)* используется *неблокированная передача* (noninterlocked frames): очередной кадр может быть послан без ожидания подтверждения предыдущих (но подтверждения передаются, хотя приходят позже). Здесь темп потока не зависит от времени прохождения данных между источником и приемником. Потоковые передачи данных аналогичны синхронной передаче в параллельных версиях SCSI.

В протоколе SSP используется формат кадра, заимствованный из Fibre Channel. Кадр начинается с 24-байтного заголовка, за которым может следовать от 0 до 1024 байтов информационного блока. При необходимости кадр выравнивается с помощью байтов-заполнителей (0-2); завершается кадр 4-байтным полем CRC. Для кадров определено 5 типов: *TASK, COMMAND, RESPONSE, DATA* и *XFER_RDY*. Кадр *TASK* требуется для управления заданиями (в Fibre Channel для этих целей используется кадр *COMMAND).*

Для команд SCSI, не требующих передачи данных, инициатор посылает кадр *COMMAND*, по исполнении команды устройство посылает кадр *RESPONSE.*

Для выполнения команд записи (передача данных от инициатора) целевое устройство после получения кадра *COMMAND* посылает кадр *XFER_RDY* (готовность к приему), на который инициатор отвечает одним или несколькими кадрами *DATA.* Посылка *XFER_RDY* и ответная посылка кадров *DATA* может производиться несколько раз, пока не будут переданы все данные для этой команды (или не возникнет особое условие). По исполнении команды целевое устройство посылает кадр *RESPONSE.*

Для выполнения команд чтения целевое устройство после получения кадра *COMMAND* посылает один или несколько кадров *DATA*, пока не будут переданы все данные для данной команды (или не возникнет особое условие). По исполнении команды целевое устройство посылает кадр *RESPONSE.*

Для двунаправленных команд целевое устройство организует доставку данных по своему усмотрению, запрашивая данные записи кадрами *XFER_RDY*и посылая данные чтения по своей инициативе.

Протокол SMP

Протокол SMP служит для управления экспандерами и определения топологии домена.

Для протокола SMP канальный уровень работает проще, чем SSP. Соединение SMP может запросить только инициатор. В каждом соединении передаются только два кадра — запрос от инициатора *(SMP_REQUEST)* и ответ на него от адресованного устройства (*SMP_RESPONSE*), — после чего обменом примитивами *CLOSE* соединение закрывается. Внутри соединения SMP используются только примитивы *SOF* и *EOF*, обрамляющие кадры. Подтверждать прием кадров *(ACK* или *NAK)* не нужно. Соединения SMP работают в полудуплексном режиме. Управление потоком не требуется:

открытие соединения является неявным кредитом для посылки единственного кадра инициатором и целевым устройством.
Инициаторами SMP могут быть как конечное устройство SAS, так и самоконфигурируемым экспандеры. Кадр SMP состоит из 4-байтного заголовка, поля данных (0-1024 байт, дополненных до целого числа двойных слов) и 4-байтного поля CRC. В заголовке кадра указывается код функции SMP, которую запрашивает инициатор протокола SMP. С помощью функций SMP определяется топология домена, конфигурируются экспандеры и осуществляется управление их портами.

Протокол STP

Протокол STP обеспечивает туннель между инициатором SATA (представленным хост-контроллером SAS) и мостом STP/SATA, к которому подключено устройство SATA. В SATA хост и устройство взаимодействуют непосредственно, без установления соединений. Протокол STP обеспечивает устройству SATA, подключенному к мосту STP/SATA, и иллюзию непосредственного соединения с хост-контроллером SATA — правда, с дополнительными задержками. Эту иллюзию создает STP-соединение между портом инициатора протокола STP (хост- адаптера SAS) и портом моста STP/SATA в экспандере, к которому подключено устройство SATA. Если устройство SATA подключено непосредственно к хост-адаптеру SAS, то протокол STP не используется (хост-адаптер с данным устройством работает по обычному протоколу SATA).
Канальный уровень STP построен с учетом протокола SATA, который несколько проще протокола устройств SAS. Устройства SATA никогда не передают кадры во встречных направлениях одновременно. В связи с этим в соединении STP используется полудуплексный режим: в одном направлении передаются двойные слова данных, во встречном — примитивы *R_IP*. Кадры SATA по конструкции аналогичны кадрам SAS (4-байтный заголовок, поле данных и CRC- код), но поле данных (SATA FIS) может иметь длину до 8192 байт.
Обмен кадрами и примитивами при установлении соединения STP аналогичен установлению соединения SSP; этот обмен выполняется между хостом и мостом и не выходит на порт подключения устройства SATA. Мост подтвердит установление соединения, только если порт устройства SATA находится в покое. Закрытие соединения STP также происходит невидимо для устройства SATA, и оно также возможно только в состоянии покоя. Внутри открытого соединения кадры и примитивы передаются по сквозному пути между трансивером инициатора и трансивером устройства SATA. Передача кадров SATA в соединении STP аналогична протоколу SATA.
Устройства SATA, в отличие от SAS (и всех устройств SCSI), рассчитаны на работу лишь с одним хостом. Целевое устройство протокола STP (мост STP/ SATA) может работать и с несколькими инициаторами. Чтобы защитить устройство SATA от работы с несколькими инициаторами, используется механизм *привязки* (affiliation). Мост устанавливает соединение с первым инициатором, который это соединение запрашивает, и привязывает этот мост (фактически, устройство SATA) к данному инициатору. Данная привязка удерживается, пока не произойдет одно из следующих событий:

♦ выключается питание моста;

- соединение закрывается специальным примитивом *CLOSE* (*CLEAR AFFILIATION*), посланным данным инициатором;
- управляющими функциями протокола SMP (*PHY CONTROL)* выполняется сброс трансивера, к которому подключено устройство SAS;
- физический интерфейс устройства SATA начинает последовательность сброса.

Инициатор, не привязанный к данному устройству, на запрос соединения STP будет получать отказ.

В SATA определены примитивы запроса режимов пониженного энергопотребления интерфейса — *SATA_PMREQ_P* (режим *Partial)* и *SATA_PMREQ_S* (режим *Slumber).* Эти запросы в STP не поддерживаются.

Определение структуры домена

Структуру домена (дерево экспандеров и конечных устройств) определяет специальное клиентское приложение-«разведчик» (discover), функционирующее на конечном устройстве SAS или в каком-либо экспандере. В процессе инициализации интерфейсов (из кадра *IDENTIFY)* для каждого своего трансивера устройству-разведчику становится известным SAS-адрес и тип (конечное устройство, E-экспандер или F-экспандер) подключенного к нему устройства SAS. К непосредственно подключенному устройству разведчик может обращаться сразу, до конфигурирования экспандеров, используя прямую маршрутизацию. К обнаруженному экспандеру разведчик первым делом посылает SMP-запрос *REPORT GENERAL*, по которому определяет число трансиверов экспандера и его свойства (самоконфигурируемость). Далее он посылает этому экспандеру запросы *DISCOVER* на каждый трансивер, определяя наличие и свойства подключенных к нему устройств. По полученным ответам разведчик заполняет элементы таблиц маршрутизации этого экспандера (если он не является самоконфигурируемым). Затем, пользуясь этими записями в таблицах, экспандер может провести соединения SMP от разведчика к каждому из обнаруженных партнеров данного экспандера. Эти соединения нужны для посылок запросов *REPORT GENERAL*, а затем и *DISCOVER* к обнаруженным экспандерам. Таким образом разведчик раскрывает структуру уровень за уровнем, начиная от ближних к себе устройств и добираясь до всех конечных устройств домена. По ходу разведки он конфигурирует экспандеры, заполняя их таблицы маршрутизации. По окончании этого процесса весь домен становится прозрачным для установления любых соединений между любыми парами устройств. Разведчик (инициатор протокола SMP) получает список всех доступных устройств (их SAS-адресов). Целевым устройствам этот список не нужен — им приходится устанавливать соединения только с ранее вызвавшими их инициаторами.

Заниматься определением топологии могут и несколько инициаторов протокола SMP. Однако независимо от того, какой инициатор выполняет определение топологии, в конце этого процесса все таблицы маршрутизации будут содержать те же записи на тех же местах.

Информация о структуре домена считается действительной до тех пор, пока не получен широковещательный примитив *BROADCAST (CHANGE).* Этот примитив посылает экспандер, обнаруживший изменение конфигурации (подключение-отключение устройств).

ГЛАВА 21

Интерфейс Fibre Channel

Fibre Channel (FC) — это интерфейс высокоскоростных коммуникаций между компьютерами и периферийным оборудованием, широко используемый в сетях хранения данных (Storage Area Network, SAN). Интерфейс Fibre Channel имеет «*канальные» черты* интерфейсов устройств хранения:

- высокую пропускную способность;
- малые задержки доставки;
- высокую надежность передачи.

Вместе с ними в интерфейсе Fibre Channel уживаются и *черты сетевых технологий*:

- возможность подключения большого количества устройств;
- большие расстояния;
- наличие средств управления и диагностики.

История Fibre Channel началась в 1988 году, с 1994 года это стандарт ANSI, которым занимается комитет NCITS T11. В интерфейсе используется последовательная передача данных в виде пакетов. Первоначально интерфейс предназначался только для оптоволокна (fiber), позже ввели медный кабель и в связи с этой универсальностью изменили написание названия на европейский манер («fibre» вместо «fiber»). В интерфейсе имеется возможность выбора топологии соединений: двухточечное соединение, кольцевая топология (FC-AL) и топология с коммутационной фабрикой (FC-SW). Среда передачи в основном определяется требуемым расстоянием между соединяемыми устройствами: медный кабель — для коротких (до 60 м) дистанций, многомодовое оптоволокно — для средних (до 500 м), одномодовое оптоволокно — для дальних (до 10 км). Наиболее широко FC применяется для устройств хранения в качестве транспортного средства протокола SCSI. Здесь основные преимущества — высокая скорость, большие расстояние и возможность подключения большего, чем в традиционном интерфейсе SCSI, числа устройств. В настоящее время широко используются скорости 100 и 200 Мбайт/с (1,0625 и 2,125 Гбит/с). Поскольку SCSI применяется для различных классов устройств (не только устройств хранения), Fibre Channel тоже можно задействовать для разнообразных соединений. Это могут быть соединения процессорных блоков, принтеров (без встроенных принт- серверов), мультимедийные соединения.

Интерфейс Fibre Channel позволяет использовать общую физическую среду передачи для общепринятых верхнеуровневых (Upper Layer Protocol, ULP) и сетевых протоколов, а также канальных технологий. Здесь под *каналами* подразумеваются относительно постоянные (статичные) соединения, устанавливаемые, как правило, между компьютерами и периферийным оборудованием (устройствами хранения данных). При этом число компьютеров и устройств относительно небольшое (десятки). Типичным примером использования каналов является интерфейс SCSI. *Компьютерные сети* — это объединение значительного количества узлов, между которыми обмены информацией непредсказуемы, так что организовывать каналы между какими-то парами узлов нецелесообразно. Из сетевых протоколов наиболее широкое применение получил IP, который традиционно используют на технологиях Ethernet, Token Ring и FDDI. Построение сетей на основе Fibre Channel само по себе особых преимуществ не дает (слишком дорогое оборудование), но при использовании FC для устройств хранения позволяет в ряде случаев отказаться от дополнительной инфраструктуры локальной сети.
Несмотря на значительную допустимую длину интерфейсных кабелей (до 10 км), сеть Fibre Channel все-таки является локальной. «Островки» Fibre Channel могут объединяться в большие сети хранения данных с помощью различных магистральных сетей: волоконных магистралей с волновым мультиплексированием, SONET/SDH, ATM и IP-сетей. По этим сетям передаются кадры Fibre Channel; при этом используются различные механизмы отображения, зависящие от сервисов, предоставляемых магистралью.
Более подробную информацию об интерфейсе Fibre Channel можно найти в [8]. Здесь мы ограничимся его кратким описанием.

21.1. Топология и типы портов

Интерфейс FC допускает различные варианты топологии (рис. 21.1):

- Двухточечное (Poin-to-Point) соединение — самый простой вариант выделенного интерфейса для соединения двух узлов (например, компьютера и устройства хранения). Это самый дешевый и высокопроизводительный вариант соединения Fibre Channel, применяемый в тех случаях, когда увеличение числа устройств не предвидится.
- FC-AL — арбитражное кольцо (Arbitrated Loop, AL). Объединение до 126 узлов без какого-либо дополнительного оборудования (каждый узел играет роль повторителя). Среда передачи разделяется между всеми узлами, каждому из которых в среднем достается лишь часть пропускной способности интерфейса. В любой момент времени возможна передача данных только для одной пары узлов, передача происходит на номинальной скорости интерфейса. Адреса узлов назначаются автоматически во время инициализации кольца. Кольцо может быть организовано с применением относительно простых концентраторов-хабов (hub), напоминающих концентраторы FDDI или Token Ring. Все узлы кольца работают на одной скорости и, как правило, должны использовать однотипные кабели. При большой длине кольца

задержки в кабеле и узлах приводят к увеличению времени арбитража — для получения доступа сигнал запроса арбитража должен сделать полный круг. По некоторым оценкам, кольцо в плане пропускной способности эффективно при числе узлов до 30 и длине до 100 метров; при дальнейшем увеличении размеров заметно снижение эффективной производительности. Однако если требуется большое пространственное разнесение устройств, топология FC-AL сохраняет работоспособность (пусть и со сниженной производительностью) и при полном числе узлов (126), удаленных друг от друга по кольцу на 10 км.

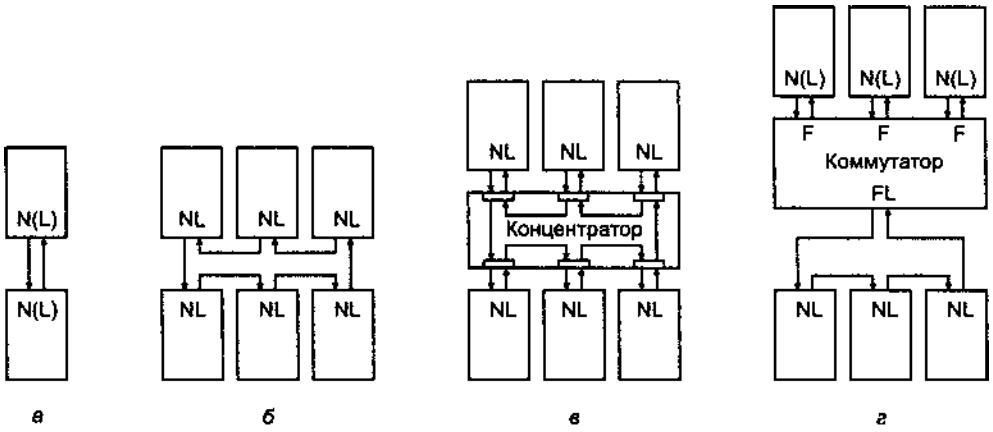


Рис. 21.1. Варианты топологии: *а* — двухточечная топология, *б* — физическое кольцо, в — логическое кольцо, *г* — смешанная топология

- FC-SW — коммутационная фабрика[1] (Switch Fabric, SW). Объединение до 2^{24} узлов с помощью одного или нескольких связанных между собой коммутаторов, обеспечивающих неблокируемые соединения между любыми парами узлов. Среда передачи разделяемой не является, одновременно возможен обмен данными между несколькими парами узлов. Узлы могут работать на разных скоростях и подключаться разнотипными кабелями. Фабрика обеспечивает назначение идентификатора естественного адреса (*S_ID)* каждому узлу, к ней подключаемому. Фабрика выполняет функции серверов группового вещания, широковещания и псевдонимов, а также средства поддержки классов передач и сервера каталогов. Соединение через фабрику обеспечивает подключение самого большого (для FC) числа узлов с самой высокой суммарной пропускной способностью.
- Смешанная топология — фабрика, к которой подключаются конечные узлы и кольца. Для реализации этой топологии фабрика должна иметь порты FL (см. далее). Кольца и порты FL могут быть *частными* (private) и *публичными*

[1] Слово *fabric* можно перевести и как «структура», и как «архитектура» (связанных портов), однако для удобства остановимся на простой транслитерации — *фабрика*.

(public). Публичные кольца и порты имеют связь с фабрикой, частные — нет, причем частный порт FL может находиться и в публичном кольце.

Порты FC различаются по назначению, среде передачи и поддерживаемым скоростям. По назначению можно выделить:

- *N-порт* — порт конечного узла (например, компьютера или дискового накопителя);
- *F-порт* — порт фабрики (коммутатора), к которому можно подключить конечный узел;
- *E-порт* — порт для соединения коммутатора с другим коммутатором, этот порт не обязательно должен поддерживать стандарт FC (на усмотрение производителя оборудования для фабрики), поскольку для подключения конечных узлов не используется;
- *G-порт* — универсальный порт фабрики, который может выступать и как F-порт, и как E-порт;
- ML-, *FL-* и *GL-порты* — порты узлов и коммутаторов, способные работать арбитражным кольцом.

21.2. Архитектура стандарта Fibre Channel

Интерфейс FC описывается многоуровневым набором протоколов:

- FC-0 — спецификации среды передачи, сигналов, приемников и передатчиков.
- FC-1 — кодирование символов с обнаружением ошибок (8B/10B), управление каналом.
- FC-2 — разбиение информационных блоков верхнеуровневых протоколов на *последовательности* (sequence) и *кадры* (frame), определение формата кадра, управление последовательностями и обменами, управление потоком, классы сервиса, подключение/отключение (login/logout), топологии, сегментация и обратная сборка блоков данных.
- FC-3 — уровень-прослойка, позволяющий включать дополнительные функции. При необходимости здесь могут выполняться сжатие и шифрование данных, аутентификация и прочие специфические функции (на усмотрение разработчиков). Пока что уровень только определяет варианты использования множества портов для одного узла:
  - *Striping* — использование связки портов для умножения пропускной способности;
  - *Hunt group* — возможность группе портов отвечать на псевдонимы адреса, снижает шанс получения ответа занятости порта узла;
  - *Multicast* — групповая передача, доставка одной передачи множеству портов назначения.
- FC-4 — отображение протоколов верхних уровней (ULP mapping) в обмены по FC: SCSI, IP, HIPPI (High Performance Parallel Interface), ATM-AAL5,

IPI-3 (Intelligent Peripheral Interface) — интерфейс для дисковых и ленточных устройств, SBCCS (Single Byte Command Code Sets) и др.

«Лицо» Fibre Channel для вышестоящих протоколов в основном определяют его уровни FC-0...FC-2, описанные в спецификации FC-PH, а затем в FC-FS. Спецификации Fibre Channel существуют в виде набора большого числа документов, часть из которых оформлена как стандарты ANSI.

Основополагающим является документ *FC-PH* (Fibre Channel — Physical and Signaling Interface, ANSI X3.230-1994), в котором определены уровни FC-0, FC-1 и FC-2. Дополнительные возможности этих уровней определены в FC- PH-2 (ANSI X3.297-1997). В новом виде эти спецификации определены в документе *FC-FS* (Fibre Channel — Framing and Signaling, 2003 г.), ведется работа над новым документом FC-FS-2.

Документ *FC-AL* (Fibre Channel — Arbitrated Loop, ANSI X3.272-1996) определяет поведение устройства в кольцевой топологии (арбитражном кольце). Этот вариант топологии появился уже после публикации FC-PH, поэтому и стал отдельным стандартом. Его развитие — стандарт FC-AL-2 (1999 г.) и проект FC-AL-3.

21.3. Среда и скорости передачи

Интерфейс FC допускает многообразие скоростей и сред передачи. *Полной скоростью* (full speed) считается 1,0625 Гбит/с, что при кодировании 8B/10B и с учетом накладных расходов на заголовки и обрамление кадров дает скорость передачи данных 100 Мбайт/с. Помимо полной определен ряд более низких скоростей — *half speed*, *quarter speed* и *eighth speed* (50, 25 и 12,5 Мбайт/с, или 531, 266 и 133 Мбит/с соответственно), а также двойная и четырехкратная скорости — *double speed* и *quadruple speed* (200 и 400 Мбайт/с, или 2,126 и 4,252 Гбит/с). Для любой среды передачи чем ниже скорость, тем больше достижимая дальность.

В чисто кольцевой топологии (без концентраторов) применяются одиночные кабели (один коаксиальный, пара проводов или одно волокно); в остальных топологиях требуется дуплексный кабель (два коаксиальных, две пары или два волокна). Каждый дуплексный кабель является перекрестным: он соединяет выход передатчика Tx (на одном конце) и вход приемника Rx (на другом конце).

В *электрическом интерфейсе* используются сигналы с уровнем ECL или PECL (ЭСЛ или псевдо-ЭСЛ), наиболее распространен разъем DB-9. Интерфейс позволяет использовать различные типы кабелей: коаксиальный (75 Ом), твинаксиальный и экранированную витую пару. Электрический интерфейс применяется для скоростей до 200 Мбайт/с (на 200 — только коаксиальный). Максимальное расстояние для полной скорости — 60 м (толстый телевизионный кабель).

Оптический интерфейс используется для любых скоростей; в качестве передатчиков применяют светодиоды, коротковолновые и длинноволновые лазеры. Наиболее распространен разъем SC-дуплекс, встречаются и более современные

малогабаритные разъемы. Для полной скорости применение одномодового волокна обеспечивает дальность до 10 км, многомодового волокна 50/125 мкм — до 500 м, а 62/125 мкм — 175 м. Для работы с коротковолновыми лазерами в FC-0 определен механизм *безопасности открытого волокна* (Open Fibre Control system, OFC): передатчик порта, не получающего сигнал от партнера, вместо нормального сигнала посылает короткие импульсы (безопасной мощности). Этот механизм реализуется не во всех портах.

Ассоциация FCA (Fibre Channel Association) приняла спецификацию интерфейсного адаптера MIA (Media Interface Adapter), обеспечивающего подключение оптоволокна к электрическому порту. В этом адаптере оптический трансивер питается от напряжения +5 В, подаваемого через запасной контакт разъема DB-9.

21.4. Адресация и подключение узлов

В FC используется трехбайтный идентификатор узла, динамически назначаемый при конфигурировании. Этот идентификатор фигурирует в качестве адреса источника (S_ID) и получателя кадра (D_ID). Адрес узла назначает фабрика процедурой подключения *Fabric Login*: в случае двухточечного соединения эта процедура не проходит, и узлы выбирают себе пару произвольных (разных) адресов. Для кольца физический адрес AL_PA (младший байт идентификатора) назначается автоматически во время инициализации кольца (см. далее). Два старших байта адреса назначаются фабрикой, а если кольцо с фабрикой не связано, они нулевые.

Для глобально уникальной идентификации в FC требуются 64-битные идентификаторы имени (Name_Identifiers). Эти идентификаторы используются в отображении верхнеуровневых протоколов, но в кадрах как средства маршрутизации не фигурируют.

Прежде чем начать обмены с другими узлами, узел должен пройти процедуру подключения (Login). В FC определено две процедуры подключения: к фабрике (*Fabric Login*) и к узлу (*N_Port Login).* Подключения действуют до последующей инициализации или до явного отключения по протоколу *N_Port Logout.*

Попытку подключения к фабрике (*Fabric Login)* должны выполнять все порты, за исключением частных портов кольца (private FL-Port). В случае наличия фабрики процедура подключения дает информацию о параметрах фабрики, включая классы сервиса и кредит для межбуферного управления потоком, а также назначает (или подтверждает) идентификатор узла. Здесь же устанавливается значение тайм-аута для процедур обработки ошибок, определяемое максимальным временем оборота по фабрике (порядка единиц секунд). В случае отсутствия фабрики на подключение может ответить N-порт, что будет признаком двухточечной топологии.

До начала обменов с портом узла необходимо к нему подключиться, выполнив процедуру *N_Port Login.* Процедура подключения дает информацию о параметрах узла, включая классы сервиса и кредит для сквозного управления потоком. В случае двухточечной топологии при этом устанавливается и межбуферный кредит.

21.5. Арбитражное кольцо — FC-AL

Арбитражное кольцо (FC-AL) — распространенный вариант использования интерфейса FC, не требующий дорогостоящей коммутационной фабрики. В кольцо может объединяться большое число узлов, размер кольца ограничен временем оборота сигнала по кольцу (тайм-аут кольца — 15 мс). При задержке передачи в каждом узле в 6 слов и расстоянии между узлами кольца 10 км в кольцо может собираться до 134 узлов. Активными участниками (источниками и получателями кадров) в кольце может быть не более 126 узлов; если узлов больше, часть окажется в *безучастном режиме* (nonparticipating mode) — они будут только транслировать битовый поток.

Все узлы кольца собираются в замкнутую цепочку и транслируют приходящие кадры дальше по кольцу. Синхронизация передатчика каждого узла автономна, а для компенсации расхождения частот синхронизации используются межкадровые слова-заполнители, часть из которых может периодически отбрасываться или вводиться дополнительно при трансляции.

Кольцо FC-AL может быть построено на звездообразной и древовидной физической топологии, более удобной в обращении (см. рис. 21.1, в). Для этого применяются *концентраторы* (хабы). Простейший концентратор содержит набор связанных в кольцо портов (приемопередатчиков). При этом порт включает в кольцо подключенное устройство, лишь получив от него сигнал (примитив сброса). Порт, на который со внешнего разъема не поступает сигнал, замыкается коммутатором порта. Таким образом обеспечивается обход отказавших (отключенных) узлов. Более сложные концентраторы обеспечивают дистанционное управление портами и сообщают их состояние. Без концентраторов кольцо становится уязвимым — при отказе одной линии или устройства все кольцо оказывается неработоспособным.

До начала нормальной работы кольцо должно быть *проинициализировано* — каждому узлу назначен физический адрес AL_PA. Этот адрес становится младшим байтом адресных идентификаторов (D_ID и SI_ID).

Арбитраж и открытие соединений

В FC-AL право доступа разыгрывается арбитражем, а не передачей маркера (как в FDDI и Token Ring). Узел запрашивает право доступа (передачи), посылая в кольцо *примитив ARBx*, где *x* — его физический адрес в кольце (Arbitrated Loop Physical Address, AL_PA). Если до узла по кольцу дойдет посланный им примитив *ARBx*, значит, он выиграл арбитраж и получил доступ. После этого, послав примитив *OPN* (open) к узлу назначения, он устанавливает двухточечное соединение с требуемым узлом. При этом все остальные узлы просто транслируют битовые потоки. Для разрыва соединения узел-участник соединения посылает примитив *CLS* (close). В примитиве *OPN* указываются тип соединения и необходимые параметры.

Запрашивать арбитраж узел может только убедившись, что шина не занята. При одновременной попытке доступа победит узел, у которого номер AL_PA

меньше. Этот узел захватывает управление кольцом на неограниченное время, в чем и проявляется канальная (а не сетевая) сущность FC-AL. Остальные узлы получат шанс доступа только после того, как данный узел освободит шину. Необязательный *алгоритм справедливого доступа* (access fairness algorithm) дает другим узлам право на доступ к шине, прежде чем узлу — на повторную попытку доступа.

Инициализация кольца

Инициализация кольца начинается с посылки последовательности примитивов инициализации *LIP* (Loop Initialization Primitive). Эту последовательность начинает посылать любой узел по включении питания или по обнаружении ошибки в кольце (потере синхронизации своим приемником). Последовательность примитивов *LIP* распространяется по кольцу и заставляет каждый узел генерировать *LIP*.

Далее должен быть выбран узел, который станет мастером инициализации и управлять назначением адресов. Если кольцо подключено к порту фабрики, то мастером станет этот порт.

Третий этап инициализации — назначение физических адресов (AL_PA). Назначение происходит в четыре этапа: сначала адреса занимают порты фабрики (если они есть), затем порты, имевшие действительные адреса ранее, затем порты, претендующие на фиксированные адреса (если они еще свободны), — и, наконец, оставшиеся порты разбирают оставшиеся адреса. Завершая инициализацию, мастер информирует все узлы о начале нормальной работы.

21.6. Протокол FCP — Fibre Channel для SCSI

Протокол *FCP* (Fibre Channel Protocol) для SCSI предназначен для взаимодействия устройств SCSI через FC, используя сервисы FC-PH (или FC-FS). Протокол позволяет любому порту FCP представлять устройство SCSI, являющееся инициатором или целевым устройством. Портом FCP называют любой порт узла (N-порт или NL-порт), способный выполнять транзакции FCP. *Операции ввода-вывода* SCSI протокол отображает *FC-обменами*; блоки с командами, данными и состоянием устройств SCSI — *информационными блоками* (IU). *Операция ввода-вывода* в протоколе FCP соответствует термину *задание* архитектурной модели SAM. Архитектура FC позволяет между любой парой портов FCP одновременно поддерживать до 65 535 обменов (16-битный идентификатор), в каждом обмене может быть до 256 последовательностей. Эти возможности превышают потребности устройств SCSI при многозадачной работе.

Для выполнения операции ввода-вывода SCSI клиентское приложение запрашивает FCP-сервис *Execute command*. Операция может состоять из одной команды или цепочки связанных команд SCSI. Программный запрос содержит всю информацию, необходимую для выполнения команд: адрес устройства хранения, адреса буферов данных, собственно команду (цепочку команд).

Отрабатывая запрос, инициатор посылает блок *IU FCP_CMND* с командой: идентификатор LUN (8-байтный), управляющие флаги, описания буферов и собственно CDB — дескриптор команды SCSI. Этот блок инициирует начало обмена. Когда устройство, получившее команду SCSI, ее интерпретирует и станет готовым к передаче данных для этой команды, оно посылает блок *IU FCP_ XFER RDY* с дескриптором той части данных, которую следует переслать. При выполнении *команды записи* в устройство инициатор в ответ на блок *FCP_ XFER_RDY* посылает заказанный блок данных *FCP_DATA*. При выполнении *команды чтения* целевое устройство вслед за блоком *FCP_XFER_RDY* посылает объявленный блок данных *FCP_DATA.* Обычно на каждый запрос *FCP_XFER_ RDY* посылается по одному блоку данных *FCP_DAT A*, но этот механизм управления потоком может быть отключен (если используется иной).

После передачи всех данных устройство посылает блок *IU FCP_RSP,* содержащий информацию о состоянии выполнения команды SCSI, а при ненормальном завершении — и расширенную информацию о состоянии (*SCSI REQUEST SENSE).* На этом выполнение команды завершается. Если команда в операции является последней (или единственной), то на этом завершается и обмен. Если команда не является последней в цепочке, а состояние, переданное инициатором в блоке *IU FCP_RSP,* позволяет продолжать выполнение, инициатор посылает блок *IU FCP_CMND* для следующей команды с тем же идентификатором последовательности. Эта команда будет выполняться вышеописанным образом.

Для того чтобы начать работу по протоколу FCP, процесс одного порта FCP должен установить связь с процессом другого порта FCP, выполнив *процедуру входа PRLI* (Process Login). Процедура PRLI выполняется канальным уровнем (FC-2). Эта процедура устанавливает *связи пар образов процессов* (image pair) и позволяет инициатору и целевому устройству согласовать параметры протокола FCP.

Для разрыва связей пар образов служит *процедура выхода PRLO* (Process Logout). После разрыва связи взаимодействие пары портов (N-портов или NL-пор- тов) по протоколу FCP становится невозможным.

Список литературы

1. *Гук М.* Аппаратные средства IBM PC: Энциклопедия. — 2-е изд., доп. — СПб: Питер, 2001.
2. *Гук* М. Процессоры Pentium II, Pentium Pro и просто Pentium. — СПб: Питер Ком, 1999.
3. *Гук М., Юров В.* Процессоры Pentium 4, Pentium III, Athlon и Duron. — СПб: Питер, 2001.
4. *Гук* М. Аппаратные средства локальных сетей: Энциклопедия. — СПб: Питер, 2000.
5. *Гук М.* Дисковая подсистема ПК. — СПб: Питер, 2001.
6. *Гук* М. Аппаратные интерфейсы ПК: Энциклопедия. — СПб: Питер, 2002.
7. *Гук М.* Шины PCI, USB и FireWire: Энциклопедия. — СПб: Питер, 2005.
8. *Гук М.* Интерфейсы устройств хранения: Энциклопедия. — СПб: Питер, 2006. (планируется).
9. *Кулаков В.* Программирование на аппаратном уровне: Специальный справочник. — 2-е изд. — СПб: Питер, 2003.
10. *Рудометов Е.*, *Рудометов В.* Материнские платы и чипсеты. — 3-е изд. — СПб: Питер, 2002.

Алфавитный указатель

1-9

1000BaseT, 752
100BaseT4, 752
100BaseTX, 752
10BaseT, 751, 752
144 pin SO DIMM, 384
168-pin Buffered DIMM, 381
168-pin Registered DIMM, 382
168-pin Unbuffered DIMM, 382
16-битное обращение, 829
200 pin SO DIMM, 385
28F008SA, 411
28F016SA, 411
28F032SA, 412
29EE010, 415
29EE011, 415
29LE010, 415
29LE011, 415
29VE010, 415
29VE011,415
32-битное обращение, 829
3D, 583
3D RSX, 706
3dfx, 590
3DNow!, 273, 330
3DNow!E, 331
3D-акселератор, 638, 798
3D-звук, 705
3D-Mbiuib, 653
3GIO, 807
72 pin SO DIMM, 384 8086, 279
80-проводной кабель, 936
8253/8254, 127
88 pin DRAM cards, 386

A

A3D, 706 AC-3, 704
ACCESS.Bus, 611 AC-Link, 716
ACPI, 145, 179, 313 Active Mode, 975
Ad Lib, 730 ADC, 692
Additional ROM BIOS, 175
ad-hoc, 758
ADPCM, 697, 742
ADSL, 748
ADSR, 700
AFG, 724
AGP, 251, 582, 593, 629, 798, 799, 802
AGP Clock, 222
AGP Pro, 807
AHCI, 931
AIMM, 373
AMD, 330
AMD K6, 330
AMD K6-2, 330
AMD K6-III, 331
AN, 567
APIC, 278, 312
APM, 143, 301, 975
Arbitration, 910, 991, 998
ARBx, 1031
ARLL, 446
AS, 604
ASCII-код, 651, 652 ASK IR, 860, 861
ATA, 240, 427, 548, 927, 931, 938, 940, 941, 950, 961, 969, 971, 975 ATA/ATAPI, 542 ATAPI, 970 Athlon, 331
Athlon system bus, 332
ATX, 231
ATX 12V, 80
Audio-CD, 488, 692
Auto HALT PowerDown, 301
Auto Refresh, 347
Autorun, 539
AWE32, 733
A-кабель, 1002

B

bay, 38 BD, 498

BEDO, 351
Be11_103,
742
Bell_212A, 742
Beta-Mode, 906 Bi-Di, 677, 838, 839 Big Endian, 23, 151 Big Real Mode, 269 BIOS, 33, 147, 162, 173, 224, 295 32-разрядные вызовы, 172 адресация, 161 видеорежимы, 634 видеосервис, 631 восстановление, 226 модули расширения, 176 носитель, 224 область данных, 173 обновление, 225, 227 положение для 386+, 295 преодоление ограничений, 550 прерывания, 169 сервисы, 170 системный модуль, 160 стандартные драйверы дисков, 531 функции, 160 BIOS DATA AREA, 173 BIOS Int 08h, 456 BIOS Int 10h, 631,633 BIOS Int 13h, 456, 531 BIOS Int 14h, 843 BIOS Int 15h, 134, 683 BIOS Int 16h, 651, 680 BIOS Int 17h, 680 BIOS Int 19h, 138, 167 BIOS Int 1Ah, 133 BIOS Int 9h, 650 BIOS Setup, 222 BIOS shadow, 105 BIOS32, 172 bit shift translation, 535 BitBLIT, 581 Bi-tronics, 825 Bluetooth, 864 BMICx, 953 BMIDPTx, 953 BMISx, 953 BNC-разъем, 612 Boot Block, 405, 411 Boot ROM, 757 Bootable CD-ROM, 539 bootstrap loader, 138 BPI, 446 BSP, 306 BTB, 265, 295 Bulk Erase, 405, 410 Bus Free, 991, 998 bus mastering, 125 Bus Mouse, 652
bus reset, 888
Bus-Master, 950
B-кабель, 1003

C

Caddy-Type, 503
Card Bus Type II, 818
Card Bus Type III, 818
CardBus, 820, 821
CAS Latency, 353
CATV, 749
CAV, 500
CBR, 347
CD, 291, 492
CD Extra, 495
CD Plus, 495
CD Single, 490
CD-Changer, 503
CD-DA, 490
CD-E, 491
CD-R, 492, 505
CDRAM, 370
CD-Recorder, 505
CD-ReWriter, 505
CD-ROM, 505, 506, 539, 839
CD-RW, 491, 492, 505
CD-WORM, 491
Celeron, 319, 323
Centronics, 71, 676
Centronics-36, 834
CFLSH, 313
Check Condition, 983, 998
chirp-sequence, 888 CHS, 420,
535, 928, 961 CLV, 500
CMOS обнуление, 230
память конфигурации и часы, 129 питание, 228 потеря параметров, 230 программирование таймера, 134 разрушение информации, 230 сброс параметров и пароля, 229 CMOS RTC,
129
CMOS Setup, 138, 244, 245, 543, 603, 774
CMOV, 313
COAST, 397
Command, 991, 998
command block requests, 925
Command Complete, 998
Compact PCI, 47
Compare, 985
composite video, 618
COM-порт, 842, 844, 851, 852, 855
continuity module, 365
conventional memory, 106
Cooler, 85

Coppermine, 322 Copy, 985
Core Speed, 221 Covox, 690
CPU Clock, 221 CPUID, 311
CR, 677 CRC, 446 CRT, 594
CS, 939
Ctrl+Alt+Del, 139 CVID, 300
CX-50, 1000 CX8, 312

D

DAO, 508 DAT, 692 Data,
991 DATA, 802 Data OUT,
998 DB-15S, 682 DB-25, 834,
1001 DB25P, 843 DB9P, 843
DDC1, 611 DDC2, 611
DDC2AB, 611 DDC2B, 611
DDIM, 177 DDMA, 126, 710,
783 DDO, 550
DDR SDRAM, 357, 383 DDR
SRAM, 395 DDR2 SDRAM,
358 DDRII SRAM, 395 DE,
312 Deep Sleep, 302 Defrag,
556 Deshutes, 321 Device-0,
929 Device-1, 929 DFP, 615
DIB, 320
Digital Ready, 713 Digital
Versatile Disk, 496 DIM, 781
DIME, 582 DIMM, 380
DIMM-168, 372 DIMM-184,
373 DIMM-240, 373, 383
DIP-переключатель, 68
Direct Port Access, 954
DirectCD, 507 Dirty SRAM,
397 Disconnect, 991, 998
Disk Manager, 550 DiskOnChip, 518
DivX, 580 DLT, 515
DMA, 33, 124, 125, 126, 154, 852, 942
DMA Setup FIS, 955, 957
DMI, 177
DMTF, 177
Dolby Digital, 704, 708
Dolby Surround Pro Logic, 704
dot pitch, 595
DotClock, 625
double speed, 1029
Doze Mode, 253
DPA, 954
DPCM, 697
DPMS, 602,611
DRAM, 340, 342, 350, 351, 408
DrvSpace, 556
DS3D, 707
DSD, 694
DS-Mode, 906
DSP, 697, 712
DTS, 313, 709
Duron, 332
DVD, 496, 497, 498
DVD+R, 500
DVD+RW, 500
DVD-Audio, 500, 708
DVD-R, 500
DVD-RAM, 499, 500
DVD-ROM, 498, 500
DVD-RW, 499, 500
DVD-Video, 500, 708
DVD-плеер, 580
DVI, 615

E

EAX, 707 EBDA, 173 ECC, 374, 446
ECC-On-Simm, 374 ECC-Optimized, 374
ECHS, 535
ECP, 126, 677, 830, 839, 1006
ECP+EPP, 839
ECU, 244
ED, 450
EDD, 533
EDO, 350
EDRAM, 370
EEPROM, 399, 404
eighth speed, 1029
El Tori to, 539
EMS, 106
EMU8000, 733
END, 814

Energy Star, 602
Enhanced Virus Protection, 263
EOF, 880
EOI, 119
EPP, 828, 839
EPP 1.7, 830
EPR, 439
EPROM, 399, 402
ESCD, 103
Ethernet, 751
EVC, 612
Execute command, 1032
Execute Disable Bit, 263, 320
Expansion ROM, 793
ExpressCard, 822
Extended ASCII Keystroke, 651
Extended Copy, 985
External LoopBack, 856
EZ135, 483
EZ-BIOS, 551
EZFlyer, 483
E-порт, 1028

F

Fabric Login, 1030 Fake Parity, 374
Fan, 85
Fast Centronics, 126, 677, 833, 839 Fast SCSI-2, 979 FAT, 441
Fax Group_III, 743
FC-AL, 1029 FC-FS, 1029 FCI, 445 FCP, 1032 FC-PH, 1029
FCP-сервис, 1032
FC-обмен, 1032
FDC, 455 FDC AT, 455
FDD, 447
Fibre Channel, 428, 430, 1025, 1028, 1031
FIFO, 159, 922
FireWire, 900, 912
Flash BIOS, 224
Flash File, 406,411
Flash-BIOS, 147
Floptical, 481
Flow Control, 848
FL-порт, 1028
FM, 445
FM Music Synthesizer, 688 FM-синтезатор, 702 FORMAT, 451 Fps, 589 FPU, 272, 312 FRAM, 415
FRC, 304 FSB Clock, 221 Full On Mode, 253 FXSR, 313 F-порт, 1028

G

G.723.1, 764
G.729A, 764
game port, 681
gap, 446
GART, 592, 804
Gate A20, 99, 100, 161, 646
GDDR SDRAM, 360
GDDR2 SDRAM, 360
GDR3 SDRAM, 361
General MIDI, 738
G-list, 478
GL-порт, 1028
GPA Card, 373
GUS, 709
G-порт, 1028

H

half speed, 1029
Handshaking, 848 HAVi, 918 HD, 450
HD-50, 1001
HD-68, 1001
HDA, 458
HDPT, 533
HDSL, 748
HEL, 707
Hewlett Packard Bi-tronics, 825
hibernate, 145
High Color, 565
host bridge, 787
Hot Plug, 797
Hot-Plug Controller, 797
HPCF, 908
HP-GL, 673
HPM, 350
HP-SIR, 860
HST, 743
Hunt group, 1028

I

I/O APIC, 120 I/O Redirection Table, 120 I/O Space, 792
I/O(x)APIC, 122 i16450, 843 i16550, 843 I2C, 611 I2S, 729

i8042, 646 i8250, 843 IBM PC, 20 IBM PC/AT, 20 IBM PC/XT, 20 IDC, 70 IDC-50, 1000
IDE, 150, 240, 427, 927, 931
Identify, 997, 998 IDLE, 802
Idle Mode, 975 IDT, 277
IEEE 1284, 826, 837 IEEE 1284.3, 834 IEEE 1394, 428
IEEE 1394-1995, 900 IEEE 1394a, 901 IEEE 1394b, 901
IN, 941 IN/OUT, 828 Industrial PC, 46 Infra Red, 860 INIT, 295 initiator, 924 Inquiry, 984
Instantly Available PC, 145
Int 0Eh, 533
Int 10h, 633, 854
Int 13h, 534, 541, 987
Int 14h, 850, 854
Int 16h, 854
Int 19h, 540
Int 76h, 533
Int 9h, 650
INTA, 116
Intelligent ATA IDE, 931 Intelligent Zoned Recording IDE, 932 intensity stereo, 698
Interrupt Acknowledge, 116 IPL, 167
IP-телефония, 763 IR Connection, 860 Ir TP, 861 IrCOMM, 862 IR-Connector, 863
IrDA, 860 IrDA 1.1, 860 IrDA FIR, 861
IrDA HDLC, 861 IrDA SIR, 860 IrLAN, 862 IrLAP, 861 IrLMP, 861 IrOBEX, 862
IRQ 15, 533 IRQ 6, 533 IRQ1, 645, 647, 649
IRQ11,851 IRQ12, 647 IRQ4, 851 ISA, 222 iSCSI, 428 ISO 9660, 506 ISW, 399
IU Command, 996 IU Data, 996 IU Data Stream, 996

J

jack sense, 725 Jaz, 482
joint stereo, 698 Joliet, 506 jumperless, 69, 136

K

K56flex, 743
Katmai, 322 KBC, 646 KNI, 322

L

Laptop, 43 Large Disk, 535 LBA, 420, 535, 928 LCD, 604 LED, 607 legacy card, 110 LF, 677
LH-порядок следования байтов, 23
LIM EMS, 106
LIMDOW, 484, 486
Line In, 727
Line Out, 727
LINK Chip, 905
LIP, 1032
Little Endian, 23, 151 LLF, 446
Logical Parity, 374 login, 926 login requests, 925 logout, 926 look aside, 341 look through, 341 Low-Profile PCI, 798
LPC, 408 LPCM, 708
LPT-порт, 678, 823, 824, 825, 839
LR, 604
LS-120, 481
LSB, 24
LTO, 515
LVD, 988

M

Mac SCSI, 1003
Magic Packet, 757
management requests, 925
Matrix Storage, 433
MBR, 168, 438
#MC (исключение), 312
MC'97, 716
MCA, 313
MCE, 312
MCX, 1002
MDRAM, 370, 621
Memory Mapped I/O, 26, 792
MESI, 290
Message, 991, 997
Message IN/OUT, 996
Message OUT, 991, 998
MFG, 724
MFM, 445
Mic In, 727
Micro-centronics, 1002
Micron, 414
microPC, 48
MIDI, 682, 689, 735
MIDI Sync, 738
MIDI Time Code, 738
MIDI-In, 735
MIDI-Out, 735
MIDI-Thru, 735
MIDI-клавиатура, 689
MIDI-контроллер, 736
MIDI-секвенсор, 737
Mini PCI Express, 820
Mini PCI Specification, 819
Miniature Card, 530
MiniD50, 1001
MiniD68, 1001
MIPmap, 588
MLP, 708
MMC, 525, 526
MMF, 908
MMX, 272, 313
Mobile Rack, 482
Mode 1, 797
Mode 2, 781, 797
Mode Select, 996
Mode Sense, 996
mouse, 652
MP3, 699
MPEG, 575
MPEG audio, 708
MPEG-1,576, 698
MPEG-2, 577, 699
MPEG2 Layer 3, 699
MPEG-4, 577
MPEG-плеер, 579
MPR II, 604
Mps, 589
MPU-401, 730, 731 MRH, 459
MS Mouse, 655 MSB, 25
MSCDEX, 539 MSI, 785
MSR, 294,311,312, 487 MT-32, 731 MTC, 738 MTH, 369
MTRR, 293,313 Mts, 589
Multicast, 1028 Multimedia CD-ROM, 505
MultiMediaCard, 525 Multi Read CD-ROM, 505
MultiRead CD-RW, 492
Multisession CD-ROM, 505

N

N_Port Login, 1030 N_Port Logout, 1030 NAP, 364
Narrow SCSI-2, 979 Native PCI IDE, 953 NetBurst, 325
NI, 636
Nibble Mode, 825 NIC, 753
NL-порт, 1028 NMI, 112, 277
No Snoop, 780 non Parity, 373
Non-Intelligent IDE, 931
normal state, 300 Notebook, 43
NTSC, 570 NV Storage, 398
NVRAM, 400 N-порт, 1028

O

OBP, 399 off, 144
OFF-Line, 677 on, 144 On Line, 841 OnliNet Basic, 94
OnNow PC, 145 OPN, 1031
OUT, 941 overclocking, 222
overdrive, 315

P

P&D-A, 612 P&D-A/D, 615

P5, 258
P6, 258, 306, 321 packet writing, 508 PAE, 312 PAL, 570 Palm-top, 45
Parallel Port FIFO Mode, 826
Parity, 374
Parity Generator, 374 PAT, 313
patch, 702 PC, 19, 97
PC Card, 410, 798, 818, 820, 821
PC Card Type I, 821
PC Card Type II, 821
PC Card Type III, 821
PC Card Type IV, 821
PC Mouse, 655
PC portable, 43
PC/104, 48
PC/PCI, 126,710
PC/PCI DMA, 783
PCI, 115, 136, 186, 222, 755, 774, 779, 783, 793, 795, 802
PCI BIOS, 793
PCI Bridge, 787
PCI Bus Master, 772
PCI Concurrency, 769
PCI Express, 807, 811
PCI IDE, 951
PCI IDE Bus Master, 930
PCI Peer Concurrency, 769
PCL, 672
PCM, 693
PCMCIA, 410, 820
PD/CD, 513
PDA, 45
PDN, 364
Peer-to-Peer bridge, 787 Pentium, 303 Pentium 4, 324 Pentium II, 321, 323 Pentium III, 323 Pentium Pro, 321 PGE, 313
PHY Transceiver, 904 PIC Mode, 120 PIO, 149, 154, 942 PIO Mode, 942 PIO Mode 0, 150 PIO Mode 0-4, 942 PIO Mode 4, 150 PIO Setup FIS, 957 PIROM, 316, 323 P-list, 478 PLL, 299
PnP, 136, 173, 837, 855
POF, 908 Port Bar, 45
POST, 33, 137, 162, 293, 774 POST Card, 162 PostScript, 673 Power Management, 786 PowerChute, 94
Power-Up In Standby, 975 PPR, 996
Prefetchable Memory, 792
PRLI, 1033
PRLO, 1033
PRML, 464
PROM, 399
PS/2, 239, 825, 839
PS/2 Type 1, 838
PS/2-Mouse, 656
PSE, 312
PSE-36, 313
PSN, 313
PXI, 47
P-кабель, 1003

Q

QAS Request, 993 QD, 450
QDR SRAM, 395 QDRII SRAM, 395
quadruple speed, 1029 quarter speed, 1029 Quick Start, 324 Q-кабель, 1003

R

Radio Frequency Modulator, 618
RADSL, 748 RAID, 431
RAID-контроллер, 433 RAM, 27, 334 RAM shadowing, 105 RAMDAC, 622 RCA, 618
RDRAM, 361, 362, 364, 383, 621
Read Buffer, 985
Receive Copy Results, 985
refresh, 346
Relaxed Ordering, 780
Remote Wake Up, 178
Request Sense, 984
Reselection, 991
RESET#, 294
Restore Pointers, 999
RGB, 560
RGB Analog, 610
RGB-интерфейс, 609
RIMM, 373, 383
RJ-11,744

Sound Blaster Pro, 732
SparQ, 483 SPCI, 818,
822 speaker, 128
Speaker Out, 727 SPI,
979
Split Completion, 781
Split Response, 781
SPP, 677, 824, 839 SPP-
порт, 823 SRAM, 340,
392, 398 SRC, 714
SRS 3D Sound, 706 SS,
313 SSA, 428 SSE, 273,
313 SSE2, 273,313, 324
SSE3, 273
SSP-соединение, 1015 standby,
144 Standby Mode, 253, 975
Status, 991, 998 STBY, 364 STN,
605
Stop Grant, 302, 324 STP, 813,
908
stream control requests, 925
Striping, 1028 Super 7, 801
Suspend, 254, 144 Suspend Mode,
253 S-VHS, 618 S-Video, 618
SyJet, 483
Symmetric I/O Mode, 120 Synch
Frame, 885 Synchronous
Negotiation, 996

T

TabletPC, 45 Tag SRAM, 396,
397 TAO, 508 Task Complete,
991 task management, 926 TCO
92, 604 TCO 95, 604 TCO 99, 604
Test Unit Ready, 984 TFT LCD,
605 Tiny TP, 861 TLB, 295 TOC,
506 Toslink, 728 TrackBall, 652
Tray-Type, 502 True Color, 565,
610 TruSurround, 706
TSC, 312 TSR, 141 TV-тюнер, 573 TWAIN,
660 TXT, 567
Type 1 parallel port, 825
Type 3 DMA parallel port, 825
Type IA, 819
Type IB, 819
Type II PC Card, 820
Type IIA, 820
Type IIB, 820
Type IIIA, 820
Type IIIB, 820

u

UADSL, 748 UART, 843, 850 UART mode,
731 UC, 291 UDF, 506, 510 Ultra DMA, 942
Ultra SCSI-2, 979 UMA, 103, 621 underrun,
508 Unreal Mode, 269 upgrade, 199 Up-
plugging, 816 UPS, 93, 94, 95
USB, 650, 872, 873, 875, 879, 880, 881, 884, 891, 896
USB Link, 899
USB-IF, 872
UTP-5, 908
UV-EPROM,
399

V

V.17, 743 V.21, 743 V.22, 743 V.22bis, 743
V.23, 743 V.27ter, 743 V.29, 743 V.32, 743
V.32bis, 743 V.32fast, 743 V.34, 743 V.34+,
743 V.90 (x2), 743 VAFC, 626 VBE, 635
VDSL, 748 VESA, 560,611
VESA Advanced Feature Connector, 626
VESA DDC, 611
VESA Feature Connector, 626

RJ-45, 751 RLL, 445 Rock Ridge, 506
ROM, 335
ROM BIOS, 33, 138, 147, 160, 224
ROM shadowing, 105
Romeo, 506
root, 441
ROR, 347
RPC, 501
RPL, 169
RS-232C, 842, 844, 845, 846, 852, 854
RS-422, 844
RS-422A, 854
RS-423A, 854
RS-485, 854
RSL, 362
RTS/CTS, 848

S

S.M.A.R.T., 479, 973
S/PDIF, 728
SACD, 690, 694, 708
SActive, 968
SAM, 980
SAO, 508
SAS, 428, 1013
SAS-адрес, 1015
SATA, 240, 428, 542
SATA Superset Registers, 954
saturating arithmetic, 272
Save Data Pointers, 999
SB16, 732
SBA, 799
SBP-2, 916
SCA, 1002
ScanDisk, 556
scatter-gather list, 959
SControl, 968
SCR, 968
Scratch EEPROM, 316, 323
SCSI, 240, 428, 552, 978, 986, 1003
SCSI ID, 987
SCSI-2, 979
SCSI-3, 979
SD, 525, 528
SDDS, 709
SDP, 813
SDRAM, 352, 361, 382 SDSL, 748
SDTR, 996 Search Data, 985 SECAM, 570 SECC, 321
Secure Digital, 525, 528 secure login, 926 Selection, 991, 998
Self Refresh, 347 Self-Healing Driver, 756 SEMB, 949 Send Diagnostic, 984 SEP, 313, 435
Serial Interface, 842 Serial Mouse, 655, 852 Serial Port, 842 SError, 968 Setup, 162, 839 SFF PCI, 818
SGRAM, 370, 621 shadow RAM, 105 shadow ROM, 105 shared interrupts, 122 shared slot, 68
Sharptooth, 331
Silicon Storage Technology, 414
SIMD, 272
SIMM, 379
SIMM 30-pin, 379
SIMM 72-pin, 379
SIMM-30, 372, 379
SIMM-72-pin, 372
SIMMSaver, 375
SIMMVerter, 375
Simple Queue Tag, 998
SIPP, 372, 378
Sleep, 301, 968, 975
Sleep Mode, 347
slow refresh, 348
SLR, 515
Slumber, 958
Small PC Card, 821
smart refresh, 348
SmartMedia, 518
SmartMedia Card, 522
SmartVoltage, 412
SMBus, 316, 324
SMI, 112, 297, 873
SMM, 269, 297, 874
SMP, 304, 305
SMRAM, 102, 297, 298
SNotification, 969
SO DIMM, 373, 386
SO DIMM-144 pin, 385
SO RIMM, 373
Socket F, 329
SOF, 880
Soft Sector, 450
Software Configured, 136
Soft-модем, 747
Sound Blaster, 732
Sound Blaster 16, 732
Sound Blaster AWE32, 733
Sound Blaster Live!, 733
Sound Blaster Live! 24bit, 734

VESA Media Channel, 627
VGA Auxiliary Video Connector, 626
VGA Palette Snooping, 790
VHDCI-68, 1002
VHS, 576
VID, 300
Video Capture, 572 Virtual Wire Mode, 120 VME, 47, 312 VP&D, 615 VRAM, 370, 621 VRM, 84, 184 VT-100, 854 VT-52, 854

W

Wake On LAN, 178, 756 WARM, 513
Wavetracing, 706 WB, 288, 291 WC, 291
WDTR, 996 WfM, 178 Wide SCSI-2, 979
widget, 724
Windows-акселератор, 582 Win-модем, 747 workstation, 19 WORM, 513 WP, 292
WRAM, 370, 621 Write Buffer, 985 WSM, 411 WSS, 732 WT, 288, 291
WT Music Synthesizer, 688 WT-синтезатор, 703 WYSIWYG, 560
WYS1WYP, 560

X

XA-Ready CD-ROM, 505 XBIOS, 550, 551 XDRAM, 366 Xeon, 319, 323 XFER, 422 XMS, 107
XON/XOFF, 848, 849

Y

Y/C, 618 YUV, 618

Z

ZBR, 466 Zip, 481
ZipCD, 482 Z-буфер, 586, 591

A

А-кабель, 1002
абсолютная координата точки, 657
авария, 276 автозагрузчик, 435
автоконфигурирование параметров, 245 устройств, 791 автономный регистр, 922 автоповтор, 649
автопредзаряд, 352 агент
выбирающий, 926
командных блоков, 926
управления, 926 целевого устройства, 925 агрегирование
сигнальных линий, 815 адаптер, 32, 66, 375 беспроводной сети, 759 графический, 559 дисплейный, 559, 619 для шины AGP, 628 PC1, 628 PCI-E, 628 интерфейса ATA, 930 конфигурирование, 758 ЛВС, 838 флэш-карты, 519 хост-шины, 955
адаптивная точка, 885 адаптивное кэширование, 468 аддитивный метод синтеза звука, 701 АДИКМ, 697, 742
адрес, 22, 24, 787 базовый, 283, 839, 843 виртуальный, 109 дальний, 277 линейный, 152, 280 логический, 152, 280 нулевой, 896 порта, 856 уникальный, 897 устройства, 990
физический, 26, 32, 109, 152, 281 эффективный, 279 адресация линейная, 420, 928, 961 логическая, 961 памяти, 803 трехмерная, 420, 928 адресный пакет, 534
аккумулятор, 83

акселератор, 798
активная конфигурация, 876
активная матрица, 605
активное устройство, 661, 772
активные колонки, 689
активный теплоотвод, 85
активный терминатор, 1004
акустический шум, 472
алгоритм справедливого доступа, 1032
альтернативный вариант интерфейса, 876
альтернативный поставщик питания, 909
альтернативный регистр состояния, 964
альфа-блендинг, 587
альфа-буфер, 587
аналоговая звуковая карта, 712
аналоговый интерфейс, 504
антенна, 760
антибликовое покрытие, 603 АПД, 844 АПД-АКД, 852 апертура AGP, 593, 804 апертурная решетка, 596
аппаратная поддержка окон, 581 аппаратная часть USB, 872 аппаратное прерывание, 112, 156, 170, 260, 278, 456, 628, 731, 771, 841, 843, 857 аппаратный RAID-контроллер, 433 аппаратный принт-сервер, 675
аппаратный протокол управления потоком, 848
аппаратный сброс, 971 арбитраж, 306, 902 зазор сброса, 914 изохронный, 913 немедленный, 913 нормальный, 993 приоритетный, 913 справедливый, 913, 993 ускоренный, 993 арбитражное кольцо, 1031
арифметика с насыщением, 272 арифметико-логическое устройство, 20 архитектура IEEE1394, 902 USB, 872
  асимметричная, 406 наборно-ассоциативная, 289 процессора, 263
  симметричная, 406 Хабовая, 188
архитектурная модель PCI Express, 810 SAM, 923, 980 SAS, 1018
асимметричный модуль памяти, 388
асинхронная передача, 65, 894
  направленная, 912 широковещательная, 912
асинхронная статическая память, 393
асинхронная точка, 885 асинхронная транзакция, 920 асинхронное сообщение, 905 асинхронное уведомление, 983
асинхронность, 196 асинхронный интерфейс, 65 асинхронный обмен, 842
асинхронный режим, 993 асинхронный список, 894 атмосферная перспектива, 586
атрибут блока, 523 символа, 567
  страницы, 523 аудио-CD, 707
аудиоакселератор, 713 аудиовизуальный объект, 577 аудиокодек, 52, 687, 695, 714, 715, 763 аудиокомпакт-диск, 488
аудиообъект
  естественный, 578
  синтетический, 578
аудиосистема, 52
аудиоэффекты, 696
АЦП, 692

Б

базовая инструкция, 274 базовая память, 99
базовая система ввода-вывода, 33, 224
базовое соединение электрического интерфейса, 815 базовый адрес, 283, 839, 843, 851 байт
  как единица передачи, 22 состояния, 524, 981, 997 байтный режим ввода, 826
байт-ориентированный режим, 847
банк, 215
  микросхем, 371 памяти, 337, 371 физический, 338
барабанный сканер, 659
барьер
  2.1 Гбайт, 549
  3,2 Гбайт, 549
  33,8 Гбайт, 550
  4,2 Гбайт, 549
  528 Мбайт, 534
  7,9 Гбайт, 550 8,4 Гбайт, 550 ATA, 549
  BIOS Int 13h, 548
безопасность
  открытого волокна, 1030
  передачи данных, 760

безусловный переход, 259 безучастный режим, 1031 Бернулли диски, 483 бесконфликтность каналов DMA, 135 памяти и портов ввода-вывода, 134 прерываний, 135 беспроводная локальная сеть, 758 беспроводная мышь, 656 беспроводная радиопередача, 61 беспроводная связь, 758 бета-порт, 906 бета-режим, 906 библиотека, 435 билинейная фильтрация, 588 бит, 22
данных, 847
разрешения арбитража, 914 четности, 847 битовый образ, 672 бит-ориентированный режим, 847 блок, 27, 720 IEEE 1394, 903 внешнего интерфейса, 626 данных, 401
дескриптора команды, 984 диска, 490 завершения, 326 запроса операции, 925 интерфейса монитора, 626 информационный, 418, 1032 командный, 928
командных регистров, 962, 968
копирование, 581 логический, 420, 524, 960 питания, 75 внешний, 83
импульсный двухтактный, 74
импульсный однотактный, 73
мощность, 78
с бестрансформаторным входом, 73 с трансформаторным входом, 72
предварительной обработки, 326
системный, 37, 38
страничной трансляции адресов, 281
управления, 20
управляющих регистров, 962, 968
физический, 525 фиксированного размера, 927 хранения, 434, 949 электроники, 463 блокированная передача кадров, 1021 блокнотный компьютер, 43, 938 болванка диска, 492, 511 бочка, 601 бренд-индекс, 315 будильник, 133
буквопечатающий принтер, 661 буфер FIFO, 922
ассоциативной трансляции, 284 двухпортовый, 158 канальный, 366 клавиатурный, 649 кольцевой, 158 комбинирования записи, 287 однопортовый, 158 опустошение, 159 отложенной записи, 287 переполнение, 159 принимаемых структур FIS, 957 с дисциплиной обслуживания FIFO, 159 буферизация данных, 158 буферированная запись, 364 буферная память, 335, 467 быстрая запись, 799, 801 быстрая смежная транзакция, 778

B

варистор, 93
ВВОД-ВЫВОД
данных, 731
дешифрация, 110
инструкции, 154
карта
разрешенных обращений, 155
распределения портов, 110 программируемый, 154 программный обмен PIO, 149 пространство, 110 символов, 851 вводная зона, 489 ведомое устройство, 929 ведомый контроллер, 119 ведомый режим, 921 ведущее устройство, 779, 783, 929 ведущий контроллер, 119 ведущий режим, 921 вектор прерывания, 106, 116, 117, 276 векторная инструкция, 273 с плавающей точкой, 275 целочисленная, 275 векторное устройство вывода изображений, 561 вентиль, 277 вентилятор, 78, 85, 87 верификация записи, 452 версия SCSI дифференциальная, 988 линейная, 988 низковольтная, 988 вертикальная развертка, 561 верхний уровень, 979 верхняя граница адресуемой памяти, 101

верхняя память, 99
ветка, 910 вибрато,
701 видео
Int 10h, 631
графические режимы, 634 дисплей,
594 кодек DVI, 575 Indeo, 575 JPEG,
575 M-JPEG, 575 MPEG, 575 общие
параметры подсистемы, 636 поток, 577
производительность подсистемы, 639 с
очень низкой скоростью потока, 579
телевизионный сигнал, 569
видеоизображение, 560 видеоинтерфейс, 617
видеокомпонент, 626 видеооверлейная
плата, 572 видеопамять, 103, 335, 564, 620,
627 видеосервис BIOS, 560, 633
видеосигнал, 618 видеосистема, 559
видеочипсет, 619
визуальный естественный объект, 578
визуальный синтетический объект, 578
вилка, 1000
виртуальная локальная сеть, 751
виртуальная машина, 261 виртуальная
память, 29, 262, 279 виртуальная реальность,
609 виртуальная страница, 282 виртуальный
адрес, 109 виртуальный диск, 29, 431
виртуальный канал, 811 виртуальный
трансивер, 1015 включение питания, 254
терминаторов, 1012 вложенное прерывание,
117, 278 внеполосная линия, 802
внеполосная подача команд, 803 внешнее
устройство, 425 внешний адрес, 475
внешний блок питания, 83 внешний вывод,
724 внешний диск, 839 внешний ИК-
адаптер, 863 внешний интерфейс, 626
внешний кэш, 396 внешний массив RAID,
433 внешний модем, 745, 852 внешний
терминатор, 1004, 1010 внешняя геометрия
диска, 467, 961 внешняя заглушка, 857
внешняя память, 27, 335, 410, 418
внутреннее прерывание, 169 внутреннее
устройство, 425 внутренний RAID-
контроллер, 433 внутренний модем, 746
внутренний приемопередатчик, 862
внутренний счетчик микросхемы, 347
внутренний терминатор, 1004 внутренний
цикл, 409 внутренняя память, 27
внутренняя шина, 625 внутрикадровое
сжатие, 574 волновая таблица, 703
воспроизведение, 688, 695 восходящий
порт, 874 восходящий трансивер, 1018
время
доступа, 28, 344, 394, 421, 491
исполнения, 782 наработки на отказ,
471 обмена данными, 465 ожидания,
465, 470 перехода, 470 поиска, 421,
465, 470 сканирования, 660 хранения,
400 вспомогательная шина, 365
встроенная сервометка, 462 вторичная
шина, 788 вторичный кэш, 286, 325
вторичный процессор, 306 вторичный
раздел, 439 вход данных, 844
управляющих сигналов, 844
входной порт, 736 входной поток, 720,
723 входной преобразователь, 724
выбирающий агент, 926 выборка
кабельная, 938 мгновенных значений,
692 вывод видеоизображения, 572
внешний, 724 на принтер, 680 текста,
581 телетайпный, 635 выводная зона,
489 выделенная сервоповерхность, 462
выделенный интерфейс, 56 вызов
процедуры, 259 вызывающая часть
драйвера, 531 выключение
компьютера, 254 высокая память, 99
высокая скорость, 872, 882
высокочастотная секция, 612
выходной порт, 736

выходной поток, 720, 723 выходной преобразователь, 724

Г

газоплазменная панель, 607 гальваническая развязка, 62, 844, 855 гамма-коррекция, 599, 622 гарантированная доставка кадров, 1020 гармоника, 700 генератор тона, 725 географическая нумерация, 773 геометрия внешняя, 467 линейная, 467 трехмерная, 467, 552 гермоблок, 458 гибкий магнитооптический диск, 481 гиперконвейер, 265 гиперпотоковая технология, 307 гиперпотоковость, 328 главная загрузочная запись, 168, 438 главная шина, 788 главный загрузчик, 438 главный мост, 768, 787 главный пароль, 976 глиссандо, 701 глобально уникальный идентификатор, 922 глобальное состояние системы, 179 глубина цвета, 659 головка, 444, 927 записи-считывания, 449 магнитная, 458 магниторезистивная, 459 парковка, 460 голосовое сообщение, 737 голосовой модем, 742 горизонтальная развертка, 561 «горячая» замена, 812 «горячее» подключение, 812, 958 готовность данных, 827 графика трехмерная, 583 графическая модель, 585 графический адаптер, 559, 798, 806 интеллектуальный, 580 символьный режим, 567 текстовый режим, 567 трехмерные функции, 581 графический акселератор, 582, 625 графический конвейер, 584 графический контроллер, 623 графический примитив, 580 графический сопроцессор, 260, 582 графический чипсет, 619 группа данных, 995 команд защиты, 975 грязная строка, 288

Д

дальний адрес, 277 данные потока, 722 продвижение, 266 упакованные, 272 дата и время, 133 датчик нулевого цилиндра, 449 двигатель шпинделя, 444 двойная независимая шина, 320 двойная синхронизация, 59, 995 двойное сканирование, 606 двойное слово, 23 двунаправленная конечная точка, 876 двунаправленная передача, 906 двунаправленный канал сообщений, 879 двунаправленный порт, 825 двухканальная память, 337 двухканальный контроллер DMA, 952 двухпортовый буфер, 158 двухролевое устройство, 896 дежурный источник, 80 декартова система координат, 583 декодер, 579 дельта-ИКМ, 697 дельта-кадр, 574 дельта-сигма-АЦП, 694 демодулятор, 741 дескриптор конечной точки, 893 области физической памяти, 953 передач, 893 прерываний, 277 страницы, 283 дефрагментация, 442 джампер, 68 джойстик, 53, 681 диагностический код, 965, 972 диагностический режим, 656 диагностическое сообщение, 163 дигитайзер, 50, 657 динамик, 128 динамический диапазон воспринимаемый ухом, 690 сканирования, 659 динамическое исполнение, 320 динамическое предсказание переходов, 265 динамическое реконфигурирование, 900 диск автоматический запуск, 539 Бернулли, 483 болванка, 511 виртуальный, 29, 431 высокой плотности, 498 гибкий, 450 дефрагментация, 442 жесткий, 438

диск (*продолжение*) загружаемый, 539 загрузочный, 437
загрузочный образ, 541
закрытие, 510 закрытый, 495
записываемый, 499 запись, 507
«на лету», 511 с образа, 511
инициализация, 547
конфигурирование, 555
кэширование, 29 логический, 437, 439, 441, 538
магнитооптический,483
многосеансовый, 494 несущий, 443 обслуживание, 556
оптический, 488 правила обращения, 494 привод, 502
причины отказов, 557 разделы, 438 системный, 31, 34, 437
стирание, 509 физический, 538
фиксированный, 533
форматирование, 510
фрагментация, 442 цвет, 493
дискета, 450, 533 DD, 450 ED, 450 HD, 450 QD, 450 SD, 450
защищенная от копирования, 451
дисковый интерфейс Mitsumi, 503 Panasonic, 503 Sony, 503 дисковый компрессор, 443 дисковый накопитель, 444, 481 дисковый сервис, 533 диспетчер изохронных ресурсов, 66
шины, 915 дисплей, 49, 594
DSTN, 605
коммутатор интерфейса, 637 плоский, 606
дисплейный адаптер, 619, 798 дисплейный интерфейс, 638 дифференциальная версия SCSI, 988 дифференциальный сигнал, 906
длительность пакетных циклов чтения, 336 док-станция, 45
домен SAS, 1016
SCSI, 981
иерархии, 808
коллизий, 750
широковещательных кадров, 751
дополнение к стандарту IEEE 1394a, 901 IEEE 1394b, 901 дополнительная задержка, 359 дополнительная память, 100 допустимая частота развертки, 598
достоверность обмена, 779 хранения данных, 471 доступ
линейный, 517
последовательный, 420
прямой, 420, 519 драйвер, 35, 426, 904 USB, 873
самоизлечивающийся, 756 хост-контроллера, 873 древовидная топология, 902 дублирование информации, 63

Е

евромеханика, 47
единица распределения пространства, 442
емкость
неформатированная, 468 памяти, 421
форматированная, 468 естественный аудиообъект, 578 естественный режим PCI, 952

Ж

жесткий диск, 168, 438
живое видео, 569
жидкокристаллическая панель, 604 ЖК-дисплей, 607, 640

З

завершение
обработки аппаратного прерывания, 119
работы операционной системы, 142 заголовок блока, 490
канального уровня, 813
сектора, 446 загрузка
с нестандартного устройства, 168
удаленная, 757 загрузочная дискета, 167 загрузочное устройство, 167
загрузочный диск, 437

загрузочный процессор, 306
загрузочный сектор, 441
загрузчик, 34, 167, 441
задание, 924, 981, 997, 1032
задержанный старт, 1013
задержка дополнительная, 359
доставки данных, 158 доступа
к шине, 782 первой фазы
данных, 782 программная, 310
заземление, 89 зазор
арбитража, 913 закрытая
сессия, 495 закрытие диска,
510 сессии, 509 закрытый
диск, 495 заливка, 580
замедление процессора, 301
замкнутая система, 461
зануление, 90 запирание, 76
записываемый диск, 499
запись
адреса, 828 аудио, 687, 695
буферированная, 364 в регистр, 963
данных, 646 команд, 646
высокоприоритетная, 803 данных, 828
линейная, 514 многократная, 491 на
оптический диск, 507 наклонно-строчная,
515 низкоприоритетная, 803 обратная,
288 однократная, 491 отложенная, 364
отправленная, 790 сквозная, 288
запоминающее устройство масочное
постоянное, 401 однократно
программируемое, 401
репрограммируемое, 401 запоминающее
ядро, 362 запрет
автоматического запуска шпиндельного
двигателя, 1013 синхронизации по обоим
фронтам, 1013 запрос, 924, 925
ввода-вывода, 878, 892 одиночный,
926 от источника прерываний, 116 по
положительному перепаду, 114
прерывания, ИЗ
запросчик, 781, 808 запуск
разнесенный, 958 защита
от записи, 452 от копирования,
451, 500 страниц, 283 защитное
заземление, 844 защищенный
режим, 262 звук тональный, 699
шумовой, 699 звуковая
диагностика, 163 звуковая карта,
709 Digital Ready, 713 PCI, 710
аналоговая, 711, 712 дочерняя,
726 интерфейсы, 726 цифровая, 713
звуковой канал PC Speaker, 128
звуковой модуль, 737 звучание
обволакивающее, 704
объемное, 704 зернистость,
595 знакогенератор, 568,
623 знакоместо, 567
знакосинтезирую
щий принтер,
661 зона, 431
вводная, 489
выводная, 489
данных, 489
хранения, 524
зонная запись,
466

И

ИБП, 93 интерфейс управления, 94
планирование включения и выключения,
94 подключение, 95 программная
поддержка, 94 телеметрия, 94 игольчатый
принтер, 661 игровой порт, 681
идентификатор SCSI ID, 991 буфера
DMA, 955 глобально уникальный, 922
процессора, 315 расширенный, 523
уникальный 128-битный, 523 устройства,
152, 772, 780, 987, 1012 идентификация
дерева, 910 источника, 122 логического
устройства, 999

идентификация (*продолжение)*
монитора
параллельная, 611 последовательная, 611
питания, 530 избыточный контроль
функционирования, 304 излучение электромагнитное,
604 изменение последовательности инструкций, 266
изохронная операция, 920 изохронная передача, 65,
876, 884, 894 изохронный арбитраж, 913 ИК-адаптер,
863 ИКМ, 693
импульсно-кодовая модуляция, 693
индекс трека, 489 индексный маркер,
444 инициализация, 851 портов, 836
потока данных, 156 инициатор
взаимодействия, 924
обмена, 980, 991 транзакций,
768 инструкция
64-битных режимов, 275
базовая, 274 векторная, 273
вызова процедуры, 259
линейная, 259 общего
назначения, 274 операндная
часть, 259 операционная
часть, 259 передачи
управления, 259 перехода,
259
с плавающей точкой x87, 275 скалярная,
273 интеллект графического адаптера, 580
интеллектуальная регенерация, 348
интервал справедливости, 914 Интернет,
761 интернет-телефония, 763
интерполяционное разрешение, 659
интерфейс
10BaseT/100BaseTX, 751
AC-Link, 716
AGP, 629
CardBus, 821
Centronics, 676, 680
DFP, 615
DMI, 177
DVI, 615
HDA Link, 722
IrDA, 239
MIDI, 735
RGB Analog, 610
S/PDIF, 728
SATA, 930, 944
интерфейс (продолжение)
S-VHS, 618 S-video, 618 Toslink, 728
USB, 875 VP&D, 615 Y/C, 618 YUV, 618
аналоговый, 504 асинхронный, 65 ввода-
вывода, 821 внешних устройств, 239
внутренних устройств, 239 выделенный,
56 ИРПР, 680 ИРПР-М, 676 клавиатуры,
239, 644 линии, 741 монитора, 626
мыши, 239, 653 НГМД, 453, 454
оптический, 1029 памяти, 821
параллельный, 58, 823 периферийный,
33, 56 последовательный, 58, 428, 842
асинхронный, 847 инфракрасный, 860
токовая петля, 855 прикладного уровня,
651 разделяемый, 56 системного уровня,
32 системной шины, 326, 921
специализированный, 56 узкий,987
универсальный, 56 управления, 94
устройства, 516, 876, 879 физический,
903 интерфейсная карта, 66
информационная структура, 945
информационный блок, 995, 1032
инфракрасная связь, 860 инфракрасный
порт, 61 ИРПР, 680 ИРПР-М, 676
искажение бочка, 601 трапеция, 601
исключение, 156, 259, 276, 983
искусственная реверберация, 696
искусственное эхо, 696 исполнение
по предположению, 266 с изменением
последовательности инструкций, 266
спекулятивное, 266

исполнитель транзакции, 781, 808
исполнительное ядро, 326 источник бесперебойного питания, 93
дежурный, 80

К

кабель, 840, 859
SCSI, 1002
USB, 873, 886
USB 1.x, 886
USB 2.0, 886
UTP, 908
интерфейса НГМД, 454 перекрестный, 752 подключения принтера, 677, 678
прямой, 751 кабельная выборка, 938
кабельная сеть, 750 кабельная шина 1394/1394a, 904 1394b, 904 кабельное исполнение PCI Express, 817 кабельное телевидение, 749 кабельный модем, 749
кадр, 490, 716, 720 Ethernet, 750 Fibre Channel, 1028 USB, 880 дельта, 574 ключевой, 574 передача, 753 прием,753
калибровка, 660 канал, 25, 720, 1026 DMA, 124, 125, 839 IDE, 929 RDRAM, 362 ввода, 25 виртуальный, 811 вывода, 25 клиентский, 879 коммуникационный, 877 логический, 737 обратный, 826 потоковый, 879 прямой, 826 сообщений
двунаправленный, 879
основной, 879 состояния, 25 управления, 25 канальная адресация, 831 канальное сообщение голосовое, 737 управляющее, 737 канальный буфер, 366
канальный уровень, 810, 904, 922, 945
карта
3,3 В, 794
5 В, 794
CompactFlash, 520 MMC, 525
PC Card, 798 PCI, 773, 823 PCI-X, 794 PCMCIA, 522 SD, 525, 528 SmartMedia Card, 522 без джамперов, 69, 136 ввода-вывода, 530 достижимых скоростей, 911 звуковая, 709 интерфейсная, 66 памяти, 517
разрешенных портов ввода- вывода, 155
расширения, 38, 66 топологии, 911 универсальная, 794 каскадное соединение контроллеров, 118 кассетный жесткий диск, 483 каталог дескрипторов страниц , 282 кэша, 341 качество обслуживания, 811 сведения лучей, 600 квадлет, 905 квантование, 692
квитирование, 64, 848 кинескоп, 594
клавиатура, 49, 642 автоповтор, 649 интерфейс, 644, 645, 651 прерывания, 650 русификация, 650 скан-код, 648 типы, 642
клавиатурный буфер, 649, 650 клавиатурный флаг, 650 класс трафика, 811 кластер определение, 442 потерянный, 442 клиент, 980
клиентский канал, 879 клиентское программное обеспечение, 873 ключ защиты от записи, 452
электронный, 853
ключевой кадр, 574
когерентность кэша, 287

код
диагностический, 965, 972
избыточный, 447
контрольный, 446, 490
нажатия и отпускания клавиш, 650
программный, 259
Рида - Соломона, 447
с исправлением ошибок, 64
символа, 567
системный, 438
цвета, 566
циклический избыточный, 64 кодек, 574 кодер, 698 колонки, 688 активные, 689 пассивные, 689 кольцевой буфер, 158
кольцо
арбитражное, 1031 публичное, 1027
частное, 1027 команда, 523, 721, 775, 924
RLC, 831 активации, 352 без очередей, 957
блочного обмена, 969 верификации, 969
диагностики, 972 загрузки микрокода, 972
задания параметров блочного режима передачи, 972 записи, 352, 1033
портов ввода-вывода, 779
сектора, 969 защитного стирания, 976 идентификации, 971
конфигурационного чтения, 780
конфигурационной записи, 780
обращения
к конфигурационным регистрам, 770 к памяти, 770, 780 к портам ввода-вывода, 770 подтверждения прерывания, 780
поиска, 970
приостановки устройства, 889 рисования, 580 с обменом PIO, 957 с очередями, 957 с расширенной адресацией, 970 сброса, 957
текстовая, 671 установки
параметров, 972
свойств, 972
фиктивная, 972
чтения, 352
команда (*продолжение)*
портов ввода-вывода, 779 сектора, 969 командная таблица, 956 командный байт, 646 командный блок, 928 командный пакет, 970 командный слот, 956
комбайн, 506
комбинированное устройство, 874
коммуникативный экспандер, 1006
коммуникационное устройство, 22, 32, 54, 899
коммуникационный канал, 877 коммутатор, 684, 750, 809 коммутация клавиатуры, 685 консольных устройств, 685 мониторов, 686 мыши, 685
композитный видеосигнал, 570, 618 компрессия RLC, 831 «на лету», 514 компьютер, 19
блокнотный, 43 инструментальный, 46
конфигурирование, 244 малогабаритный, 43
настольный, 38 промышленный, 46 компьютерная сеть, 1026 конвейер AGP, 801 графический, 584
конвейеризация, 264 конвейерная транзакция, 801
конвейерное выполнение операций, 364
конвейерно-пакетная статическая память, 395
конвертор VGA-TV, 571 частот выборок, 884
конечная точка, 808, 809, 875, 879 конечное устройство, 1015 консоль, 31, 37
конструктивная реализация PCI Express, 815
контекст, 922 DMA, 920, 959 задания, 924
контроллер, 32, 720, 721 APIC, 121 DMA, 921 PCI IDE, 951 SATA, 946
асинхронного приема, 921
асинхронной передачи, 921
атрибутов, 621 ведомый, 119

контроллер (*продолжение*) ведущий, 119 изохронного приема, 921 изохронной передачи, 921 интерфейса AC-Link, 715 ATA, 930 клавиатуры, 644 локальный, 120, 121 накопителя, 445, 455, 463 памяти, 347, 365 потока, 925 прерываний, 113, 120 приема пакетов самоидентификации, 921 принтера, 662 прямого доступа, 124 расширенный, 121 устройства хранения, 426 целевого устройства, 978 шины, 730 ЭЛТ, 620 контроль ошибок, 217 четности, 1012 контрольная сумма, 340 контрольный код, 446, 490 конфигурационное пространство, 808 конфигурационный регистр, 771, 798 конфигурация устройства, 876 конфигурирование COM-порта, 851 LPT-порта, 839 SCSI, 986, 1012 адаптера, 136, 758 жесткого диска, 438, 554 предварительное, 839 концентратор, 434 Ethernet, 750 Fibre Channel, 1031 SATA, 948 концепция хранимой программы, 20 копирование блока, 581 корень, 910 корневой каталог, 441 корневой комплекс, 808 корневой узел, 723 корневой хаб, 872 корпус ATX, 40 baby-AT, 39 big-tower, 40 desk-top, 39 midi-tower, 40 mini-tower, 40 slim, 41 tower, 39
низкопрофильный, 41
коррекция ошибок, 743 Котельникова теорема, 692 коэффициент умножения, 221 кредит, 1020 критическая секция, 117 критическая частота слияния мельканий, 562 кросс-плата, 46 кросс-шина, 904 курсор, 568 кэш, 29, 334, 341, 397 MESI, 290 MTRR, 293 WB, 288 WT, 288 внешний, 396 вторичный, 286 грязная строка, 288 данных, 325 инструкций, 325 каталог, 341 когерентность, 287 модифицированная строка, 288 наборно-ассоциативный, 289 напряжение питания, 397 несекторированный, 287 первичный, 286 политика записи, 288 полностью ассоциативный, 290 попадание, 341 промах, 341
прямого отображения, 289
секторированный, 287
сквозная запись, 288 строка, 287 тег, 287 трассы, 325
третьего уровня, 286
управление, 292 упреждающее чтение, 289
функционирование, 341 циклы слежения, 290 чистая строка, 288 кэширование, 29, 340, 468

Л

лазерный принтер, 666 лента
4-миллиметровая, 515 8-миллиметровая, 516 линейная адресация, 420, 928, 961 линейная версия SCSI, 988 линейная запись, 514 линейная инструкция, 259 линейная модуляция, 693 линейная организация памяти, 565 линейный адрес, 152, 280, 533, 535 линейный доступ, 517

линейный сигнал, 906
линиатура, 668
линия запросов прерываний, 250, 839, 851
лист, 910
листопротяжный сканер, 659 ловушка, 276 логическая адресация, 961 логическая операция, 152, 624 логическая передача, 780 логическая топология USB, 875 арбитража, 902 передачи данных, 902 логические параметры, 535 логический адрес, 152, 280, 533 логический блок, 420, 524, 960 логический диск, 437, 538 логический канал, 737 логический номер, 524 логический привод, 538 логический процессор, 307 логический суб-блок, 810 логическое устройство IEEE 1394, 925 PCI, 186 SAM, 981 USB, 875 логическое форматирование, 554 ложный вектор прерывания, 118 локальная память, 798 локальная сеть, 751 локальный контроллер, 120, 121

M

магнитная головка, 458 магнитное суперразрешение, 487 магнитооптический диск, 483 магниторезистивная головка, 459 максимальное время исполнения, 782 малогабаритный компьютер, 43 маломощный порт, 890 манипулятор, 852 мышь, 652 трекбол, 652 маркер транзакции, 881 маршрутизатор, 751 маршрутизация, 788 маска теневая, 595 щелевая, 596 маскируемое прерывание, 113, 277 масочное постоянное запоминающее устройство, 401 мастер порта AGP, 800 циклов, 912 шины, 26, 151, 772, 783
математический сопроцессор, 260, 272
материнская плата, 38, 66 матрица активная, 605 пассивная, 605 светодиодная, 607 ячеек, 604 матричный принтер, 661 машина фон Неймана, 20
медиа-инструкция, 275 межкадровое сжатие, 574 метка времени, 885 метод доступа, 420 модуляции, 445
растровый, 560 механическое разрешение, 659 микро PC, 48
микроархитектура, 325 NetBurst, 325 процессора, 264 микрокадр, 880 микроконтроллер интерфейса клавиатуры, 646
мыши, 652 микросхема RAMDAC, 622 асинхронной памяти, 397 второго поколения, 411, 412 синхронной памяти, 397
микшер, 688, 718, 724 минута, 490
многоголосный синтезатор, 701
многозадачная система, 142
многократная запись, 491
многопроцессорная система, 305
многосеансовый диск, 494
многотембровый синтезатор, 701
множественный режим DMA, 942
мобильный компьютер, 329
мобильный процессор, 324 МОД, 483 модель, 585
модельно-специфический регистр, 271
модем,54, 741 Soft-модем, 747 Win-модем, 747 внешний, 745 внутренний, 746 голосовой, 742 для выделенных линий, 749 для шины PCI, 746 кабельный, 749 конструкция, 745 модернизация, 745 модифицированная частотная модуляция,445

модуль
IEEE 1394, 903 звуковой, 737 памяти асимметричный, 388 второго поколения, 382 количество микросхем, 387 симметричный, 388 параметризованный, 723 расширения BIOS, 633 управления мощностью, 725 модульная архитектура кодеков, 723 модульная система, 47 модульный синтезатор, 702 модулятор, 741 модуляция, 445 импульсно-кодовая, 693 линейная, 693
плотностно-импульсная, 694
частотная, 445 монитор, 560, 595, 617 зернистость, 597
излучение, 603 компьютерный, 595 монохромный, 595
настройка
геометрии, 600 цветов, 599 плоский экран, 603 полоса пропускания, 598 размер диагонали экрана, 596 сведение лучей, 600 синхронизация, 601 телевизионный, 595 цветной, 595, 596 цветовая температура, 599 цифровое управление, 601 мониторинг состояния системного блока, 184 монофонический синтезатор, 701 монохромный монитор, 595 мост, 755, 905 PCI, 136, 769, 772, 787 PCI-Express-PCI, 809 STP/SATA, 1015 главный, 787 непрозрачный, 789 одноранговый, 787 прозрачный, 789 северный, 188, 192, 194 южный, 194 мощный порт, 890 мультикомпьютерная система, 309 мультимедиа, 687 мультимедийное устройство, 51 мультиплексор, 836, 931, 949 мультипроцессорный сервер, 333 мультиядерный процессор, 307, 328
мышь, 50, 652, 852 PS/2, 656 беспроводная, 656
интерфейс, 653
неисправности, 654
оптическая, 652
переходники, 653
последовательная, 655

H

набор, 289
E-экспандеров, 1016
Sct#l, 648
Set#2, 648
Set#3, 648
заданий, 924, 925, 981 свойств сменного носителя, 974 указателей, 997 наборно-ассоциативная архитектура, 289 надежность, 471 обмена, 779 передачи, 64 транзакций, 813 наклонно-строчная запись, 515 накопитель внешний, 838 дисковый, 827
на гибком магнитном диске, 240, 447 на жестком магнитном диске, 457 на магнитной ленте, 420 наладонный компьютер, 45 направленная асинхронная передача, 912 напряжение, 89 изоляции, 63 питания ядра, 247, 300 настольный компьютер, 38 начальная загрузка, 138 НГМД, 447 вертикальная запись, 448 датчики, 449 интерфейс, 453, 454 контроллер, 455 перпендикулярная запись, 448 программирование, 456 смена носителя, 453 совместимость, 457 фантомные каталоги, 457 чистка головок, 457 не чересстрочная развертка, 561 неблокированная передача кадров, 1021 неисправности COM-порта, 856 нелинейный монтаж аудиозаписей, 696 немаскируемое прерывание, 112, 277 немедленный арбитраж, 913 неплановая сеть, 758 непредсказуемый отказ, 479 непрозрачный мост, 789

несущий диск, 443 неуправляемая
шина, 915 неформатированная
емкость, 468 НЖМД быстродействие,
470 гарантийный срок, 471 геометрия
внешняя, 470 физическая, 469
дефектные секторы, 476
достоверность, 472
инициализация и самотестирование, 474
интерфейс, 467, 469
надежность, 471, 479
нормальный режим работы, 475
обработка ошибок, 476
отказ, 479
параметры, 468
пластины, 458
позиционирование головок, 460
свипирование, 479 скрытие дефектов, 476
термокалибровка, 478 низкая скорость, 872, 882
низковольтная версия SCSI, 988
низкопрофильный корпус, 41 низкоуровневое
форматирование, 475 низкочастотная секция,
613 нисходящий порт, 874 нисходящий
трансивер, 1018 номер байта, 524 блока, 524,
525
входа контроллера прерывания, 786 головки,
532 канала, 912 контроллера, 737 логический,
524 начального сектора, 532 страницы, 524, 525
устройства, 772, 773 функции, 772 цилиндра,
532 шины, 772 нормальный арбитраж, 993
носитель сменный, 421 фиксированный, 421
нота, 700 ноутбук, 43
нулевая конечная точка, 879
нулевой адрес, 896 нумерация
устройств, 896

О

обволакивающее звучание, 704
область DOS, 106
область (*продолжение*)
данных, 441, 442, 523
переменных BIOS, 106
обмен
асинхронный, 842 блочный,
149 в режиме DMA, 942
PIO, 942 по опросу
готовности, 156 по
прерываниям, 156 полинг,
157
программно-управляемый, 149 синхронный, 842
сообщениями, 152 обновление микрокода, 296
обозначение микросхем, 409 обработчик
аппаратного прерывания, 531 обратная запись, 288
обратная связь, 885 обратный канал, 826
обращения к памяти
последовательные, 369 произвольные, 369
к регистрам устройства, 153 обслуживающий
процессор, 435, 949 общий запрос прерывания,
116 объединение одноранговых устройств, 768
объект
аудиовизуальный, 577
визуальный, 578
естественный, 578
синтетический, 578 объем
видеопамяти, 637 кэшируемой памяти, 397
памяти тегов, 396 стандартного диска, 490
установленной оперативной памяти, 102 хранимой
информации, 28 объемное звучание, 704 оверлей
конфигурации устройства, 972 ограничения
операционных систем, 553 ограничитель
перенапряжения, 92 одиночная команда, 981
одиночная синхронизация, 994 одиночная
транзакция, 780 одиночный запрос, 926 одиночный
режим DMA, 942 одноголосный синтезатор, 701
однозадачная система, 141 однократная запись, 491
однонаправленная сигнальная линия, 906
однонаправленный потоковый канал, 879
однопортовый буфер, 158 одноранговое
взаимодействие устройств, 152

одноранговый мост, 787
ОЗУ, 334 октава, 700 октет,
22
оперативная память, 21, 105, 334
конфигурирование, 216 установка, 215
оператор, 702
операционная система, 34, 141
операция
ввода-вывода, 1032
логическая, 152 физическая,
152 определение типа кабеля
комбинированный метод, 936 через
устройство, 936 через хост-контроллер, 936 опрос
состояния, 851 оптическая мышь, 652 оптическая
перспектива, 586 оптический диск, 488
оптический интерфейс, 1029 оптическое
разрешение, 659 организация прямого доступа к
памяти, 153 ортогональная система координат,
583 ОС, 34
освещенность, 584 основная память,
334 основной канал сообщений, 879
основной тон, 700 особое условие,
983 отказ, 276
внутреннего кэша, 390
непредсказуемый, 479
памяти, 338
предсказуемый, 479, 973 отключение
FS/LS-устройств, 889 HS-устройств, 889
открытая среда передачи данных, 760 открытый
хост-контроллер, 893 отложенная запись, 364
отложенная транзакция, 779, 782, 790
отправленная запись, 790 отражение, 586
отрицательное напряжение, 859
отрицательное подтверждение, 813
отсек, 38, 423 отсчет, 720 охлаждение,
85 вентилятор, 85 процессора, 83, 85
радиатор, 85
холодильник Пельтье, 86 очередь, 802, 925
буферов, 894
дескрипторов передач, 893
очередь (*продолжение*)
заданий, 981 запросов,
926 команд, 960 ошибка
ввода-вывода, 880
исправимая, 813
неисправимая не фатальная,
813 фатальная, 813
опустошения буфера, 159
передачи, 847 переполнения
буфера, 159

П

пакет, 720, 751, 808, 901, 928 DLLP,
810 RDRAM, 364 TLP, 810, 813
адресный, 534 данных, 878, 881
записи, 353 маркер, 881 начала
цикла, 912 переменной длины, 506
фиксированной длины, 507 чтения,
353 пакетная запись, 781 пакетная
регенерация, 346 пакетная
транзакция, 775, 778, 780 пакетный
режим, 407 пакетный цикл, 336
память, 20, 27 CMOS, 228 DRAM,
342, 343 ECC, 339 MTH, 369
NVRAM, 400 SDRAM, 352 SIMM,
379 SIPP, 378 SMRAM, 102 VRAM,
621 базовая, 99 банк, 215 буферная,
335 быстродействие, 336 верхняя,
99, 103 верхняя граница, 101
видеопамять, 335 виртуальная, 29,
108, 279 внешняя, 27, 335, 410, 418,
422 внутренняя, 27, 418
возможности кэширования, 291
время доступа, 336, 345 высокая, 99
генератор четности, 339

память (*продолжение*) двусторонний модуль, 379
динамическая, 342, 343, 348, 350, 351 асинхронная, 348, 352 временные параметры, 345 модули,372 синхронная, 352 длительность цикла, 336 дополнительная, 100 допускающая предвыборку, 792 достоверность, 338 исправление ошибок, 339 количество циклов перепрограммирования, 400 команды, 352 контроль ошибок, 217 четности, 338 кэширование, 109, 334, 341 односторонний модуль, 379 оперативная, 21, 27, 105, 334 основная, 334 отказ, 338 отображаемая, 106 полупостоянная, 335 постоянная, 21, 27, 334 предоставляемая пользователю, 106 применение, 370 программирование, 402 производительность, 336 разрядность, 337 распределение, 99 расширенная, 100, 107 регенерация, 346 режим системного управления, 102 с интерфейсом I2C, 415 с последовательным доступом, 28 с произвольным доступом, 27, 335 с прямым доступом, 28 сбой, 338 стандартная, 99, 106, 279 статическая, 392 асинхронная, 393 конвейерно-пакетная, 395 синхронная, 394 твердотельная, 419 тегов, 287, 396 теневая, 105 тестирование, 390 устойчивость к электромагнитным воздействиям, 400 ферроэлектрическая, 400 флэш, 335, 404 хаб-конвертор, 369 энергонезависимая, 335, 398, 400, 402 масочная, 401
однократно программируемая, 401
программирование, 399
память (*продолжение)*
репрограммируемая, 401
свойства, 404 стирание, 402
панель
газоплазменная, 607
жидкокристаллическая, 604
панорамирование, 581
параграф, 24
параллельная идентификация
модулей, 376 монитора, 611
параллельная шина, 427, 978
параллельный интерфейс, 58
ATA, 927 SCSI, 987
параллельный матричный принтер, 663
параллельный шинный интерфейс, 927
параметризованный модуль, 723 параметры логические, 535 физические, 535 парковка головок, 460 пароль
главный, 976 пользовательский, 976 паспорт диска, 475, 971 пассивная кросс-плата, 46 пассивная матрица, 605 пассивное устройство вывода, 661 пассивные колонки, 689 пассивный переходник, 797 пассивный терминатор, 1004 патч, 703
Пельтье холодильник, 86 первичная шина, 788 первичный кэш, 325 данных, 286 инструкций, 286 первичный процессор, 306 первичный раздел, 439 передача асинхронная, 65, 894 данных, 738, 803, 924 двунаправленная, 906 изохронная, 65, 876, 884, 894 кадров, 753 блокированная, 1021 неблокированная, 1021 логическая, 780 массивов данных, 877 пакетная, 341 периодическая, 894 по шине USB, 893 с гарантированной доставкой, 894 с постоянной мгновенной скоростью, 65 с постоянной средней скоростью, 65

передача (*продолжение*) синхронная, 65
сообщений, 786, 789 управляющая, 877
перезагрузка «теплая», 139 «холодная»,
139 переименование регистров, 266
переключение задач, 261, 267 режима, 839
в защищенный из реального, 295 в
реальный из защищенного, 295 перекос, 59
перекрестный кабель, 752 переменная, 21
переполнение стека, 285 пересылка сообщений,
770 переход безусловный, 259 условный, 259
переходная плата, 41 переходной адаптер, 736
периодическая передача, 894 периферийное
устройство, 22, 826, 872 периферийный
интерфейс, 33, 56 периферийный контроллер
прерываний, 113 перо, 657
персональный компьютер, 19
перспектива атмосферная, 586
оптическая, 586 печать
содержимого экрана, 681 ПЗУ, 335
пиксел, 561 питание от интерфейса,
859 процессора, 83 терминаторов,
1012 устройств, 424 через COM-
порт, 859 питающее напряжение,
794 ПК, 19
планирование включения и
выключения, 94 периодических
транзакций, 894 планшет, 657
планшетный плоттер, 669
планшетный сканер, 51, 659 плата
видеооверлейная, 572
материнская, 38, 66
переходная, 41
системная, 38, 66 плеер
DVD, 580 MPEG, 579
плоская модель памяти, 152, 263
плоский дисплей, 615 плоский
экран, 603
плотностно-импульсная модуляция, 694
плотность записи, 450 плоттер, 50, 660, 669
интерфейс, 670 планшетный, 669 рулонный,
670 поверхность головки накопителя, 457
повторитель, 750 поддержка
мультиплексоров портов, 958 традиционного
интерфейса клавиатуры и мыши, 873 устройств
ATAPI, 958 подключение «горячее», 958 сканера,
838 «узкого» устройства к «узкой» шине, 1008 к
«широкой» шине, 1008 устройства, 888
безопасность, 897 питание, 897
поддерживаемая скорость, 897
работоспособность, 897 удобство, 897
«широкого» устройства к «узкой» шине,
1008 к «широкой» шине, 1008 подсветка,
605 подсистема хранения, 434
подстраница, 996 подстройка частоты
кадров, 885 подтверждение прерывания,
116 подчиненная шина, 788
позиционирование головок, 421, 460
поиск портов, 836 поле атрибутов, 568
данных, 446, 490 полинг, 118, 151, 157
политика записи кэша, 288
полифонический синтезатор, 701 полная
скорость, 872, 882, 1029 полнодуплексный
режим, 751 полностью ассоциативный
кэш, 290 полностью управляемая шина,
915 положительное напряжение, 859
положительное подтверждение, 813
полоса, 431 пропускания, 598 частот, 563
Полубайтный режим ввода, 826
полупостоянная память, 335 полутон, 700

пользовательский пароль, 976
пользовательское приложение, 141
поляризационный фильтр, 603
понижение уровня сигналов, 60 порог
заполнения, 159 опустошения, 159 порт,
750, 1014 AGP, 218, 251,806 MIDI, 689
MIDI-In, 736 MIDI-Out, 736 MIDI-Thru,
736 PCI, 808 SATA, 948 SCSI, 981
ввода-вывода, 21, 25
восходящий, 874
входной, 736 выходной,
736 данных, 963
инфракрасный, 61
маломощный, 890
мощный, 890
нисходящий, 874
с прямым доступом к памяти, 825
субтрактивный, 1017 транзитный, 736
портаменто, 701
последовательная идентификация мониторов,
611 параметров, 382 последовательная мышь,
655 последовательная передача управления, 21
последовательность
информационных блоков, 1028 обнаружения
подключения и сброса, 888 опроса устройств
загрузки, 140 сигналов, 846 согласования, 833
транзакций, 780 шагов теста POST, 162
последовательный интерфейс, 58, 428, 842, 847,
860
послойное смещение секторов, 466
поставщик питания, 909 постоянная
память, 21, 334 потенциал, 855
потерянный кластер, 442 поток, 720, 926,
977 аудио, 577 видео, 577 входной, 720,
723 выходной, 720, 723 данных, 848
инструкций, 261 потоковое расширение,
273, 471, 977
потоковый канал, 879
потоковый режим, 656
потребитель питания, 909
ППЗУ, 399
право на разрыв соединения, 982
предварительное конфигурирование, 839
предсказание переходов, 265 предсказуемый
отказ, 479, 973 преобразователь входной,724
выходной, 724 прерывание, 151, 754
аппаратное, 112, 156, 170, 260, 278 вложенное,
117, 278 внешнее, 277 внутреннее, 169, 276
клавиатуры, 650 конфигурирование, 115
конфликты, 123 ложное, 118 маскируемое,
113, 277 на шине USB, 877 немаскируемое,
112, 277 по низкому уровню, 114
программное, 260, 276, 534 разделяемое, 122
таблица назначений, 114 префикс, 267
привилегия, 262 привод DVD+RW, 500
логический, 538 позиционирования, 449 с
подвижной катушкой, 460 соленоидный, 460
физический, 531 шаговый, 460 шпинделя, 458
привязка, 1023 прием кадра, 753 признак
отпускания клавиши, 648 прикладное
программное обеспечение, 35 принтер, 50,
660, 839 буквопечатающий, 661 драйвер, 673
интерфейс, 674 лазерный, 666 матричный
знакосинтезирующий, 661 игольчатый, 661
параллельный, 663 разделяемый, 675 режим
печати, 671 светодиодный, 666 струйный, 665
сублимационный, 666

принтер (*продолжение)* твердокрасочный, 666
термический, 664 термодиффузионный, 666
управляющие коды, 671 фотопринтер, 669
принтерная мини-сеть, 684 принт-сервер, 675
принудительное переключение в линейный режим, 1013
приоритет устройства, 991 приоритетный арбитраж, 913
проблемы больших дисков, 546 пробуждение по сети, 178 сигнализация, 875 проверка на четность, 63
проводная оптическая связь, 62 программа, 755 ввода-вывода, 150 инициализации, 33 программатор, 402
программирование, 398, 402 программная модель PCI Express, 811 процессоров Intel, 267 программная область, 489 программная часть USB, 873 программное прерывание, 260, 534 программный «намек», 265
программный код, 259 программный обмен, 149
программный протокол управления потоком, 849
программный сброс, 971
прогрессивная развертка, 561
продвижение данных, 266
прозрачность, 586
прозрачный мост, 789
производительность дисплейного адаптера, 639
промежуточный слой, 488
промышленный компьютер, 46
пропускная способность шины, 884
просвечиваемость, 586
простой экспандер, 1006
пространство
ввода-вывода, 32, 110, 808
конфигурационное, 808
памяти, 279, 808 сообщений, 808 протокол ECP, 1006
PC/PCI DMA, 783 PCI, 781
доступа, 861 запроса сеанса, 896 объектного обмена, 862
протокол (*продолжение*)
согласования режимов, 752 роли хоста, 896 скорости, 888
управления соединением, 861
процедура входа, 1033 выхода, 1033 процесс ввода-вывода, 981, 997 процессор, 258, 267 80286, 98 8086/88, 98 AMD, 330 Athlon, 211,331 Celeron, 319, 322 Duron, 211, 332 K6-2, 330 K6-III, 331 P6, 306, 320 Pentium, 306 Pentium 4, 204, 324 Pentium II, 321 Pentium Pro, 321 Xeon, 319, 323
аппаратные прерывания, 260
архитектура, 263 ввода-вывода, 151
возможность установки, 200
вторичный, 306 для сокетов 5 и 7, 330 загрузочный, 306 замедление, 301 защищенный режим, 268
идентификация, 201, 311
исключения, 259, 275
конвейеризация, 264
конфигурирррвание, 201 логический, 307 микроархитектура, 264
мобильный, 324 модель, 315
мультиядерный, 307, 328
напряжение питания, 201
обеспечение совместимости, 310
обновление микрокода, 296
обслуживающий, 949 основные характеристики, 316 охлаждение, 83
первичный, 306 питание, 83
поколение, 314 потоковое расширение, 273 прерывания, 275
применимость, 200 программная модель, 267 программные прерывания, 260

процессор (*продолжение)*
реальный режим, 268 сброс,
294 семейство, 314
сигнальный, 697
скалярный, 265
совместимость, 310
степпинг, 315
суперскалярный, 265
температурный режим, 86
тип, 315
умножение частоты, 299 установка, 199 частота
ядра, 201 число выводов, 316 шестого поколения, 320
эксклюзивность, 317 прямое объявление скорости,
885 прямое управление шиной, 125, 151, 754, 950
прямой доступ к памяти, 124, 150, 771, 783, 906 по
инициативе устройства, 150 хоста, 150 по
контекстным программам, 923 физических
обращений, 923 прямой кабель, 751 прямой канал,
826 ПУ, 826, 827 публичное кольцо, 1027 путь, 1015
p
рабочая станция, 333 рабочий
магнитный слой, 458
равноранговые устройства, 978
радиальное смещение, 466
радиатор, 85 развертка
вертикальная, 561
горизонтальная, 561 не
чересстрочная, 561 определение
частоты, 562 прогрессивная, 561
чересстрочная, 561 разгон, 222
раздел, 437 вторичный, 439
описатель, 438 первичный, 439
расширенный, 439 разделяемое
прерывание, 122 разделяемый
интерфейс, 56 разделяемый слот,
68, 219, 794 размер диагонали
экрана, 596 зерна, 597
разнесенный запуск, 958
разомкнутая система, 449
разрешающая способность, 662
разрешение, 636
интерполяционное, 659
механическое, 659 оптическое,
659 отключения, 1013 принтера,
668 разрыв соединения, 982
разрядность RAMDAC, 637
адреса, 267 видеопамяти, 638
данных, 267 преобразователей,
622 шины, 337, 989 разъем, 840,
859 Centronics, 71 COM-порта,
844 DFP, 615 D-type, 69 DVI, 616
IDC, 70 P&D, 615 P&D-A, 615
RCA, 618 SCSI, 1003 USB, 875,
886 видео, 613 вилка, 843
питания, 424 накопителей, 79
системной платы, 79 розетка, 682
унифицированный 34-контактный, 454
район, 523
распределенная регенерация, 346
растеризация, 584 растрирование, 668
растровый метод вывода изображений, 560
растровый режим, 672 расширение BIOS, 628
LPT-порта, 825 PCI-X, 774 расширенная область
данных BIOS, 173 расширенная память, 100
расширенная таблица разделов, 439
расширенное управление энергопотреблением,
975 расширенный ASCII-код, 651 расширенный
дисковый сервис, 533 расширенный
идентификатор, 523 расширенный контроллер,
121 расширенный набор регистров, 829
расширенный раздел, 439 расширенный хост-
контроллер, 893

расщепленная транзакция, 780, 883
реальный режим, 268, 277
реверберация, 696 регенерация
изображения, 564 интеллектуальная, 348 пакетная, 346 памяти, 346, 364 распределенная, 346 с пониженной частотой, 348 скрытая, 347 регистр, 33 PAT, 294 автономный, 922 адреса устройства, 965 базового адреса, 792 битовой маски, 624 в пространстве ввода-вывода, 792 ввода-вывода, отображенный на память, 792 глобально уникального идентификатора, 922 данных, 646, 824, 965 двойной, 964 диагностики POST, 162 запросов, 954 защелка, 623 команд, 411, 646, 953, 967 конфигурационный, 798 масок прерываний, 954 модельноспецифический, 271 номера сектора, 966
устройства и головки, 966 цилиндра, 966
общего назначения, 271 ошибок, 965 палитр, 621 переименование, 266 понятие, 25 свойств, 965 сегментный, 270 системного назначения, 271 состояния, 411, 825, 953, 966 справедливости, 993 счетчика секторов, 966 указатель, 271 управления, 825 управляющий, 365, 798, 964 устройства ATA, 962 флагов, 271 регистроориентированное устройство, 830 регулятор напряжения, 184 режим
DIME, 592, 800
DMA, 592, 800
ECP, 827 EPP, 826
LBA-48, 964
режим (*продолжение*)
PCI, 801
PIC, 120
асинхронный, 993 байт-ориентированный, 847 безучастный, 1031 бит-ориентированный, 847 быстрого страничного обмена, 348 виртуального процессора 8086, 268 виртуальных проводов, 120 восстановления BIOS, 226 двойной синхронизации, 995 диагностический, 656 использования поля данных, 490 «нереальный», 269 обмена, 972 одиночной синхронизации, 994 одновременного переключения, 989 опроса, 118, 656 пакетный, 407 парковки, 895 передачи ATA, 941 полнодуплексный, 751 полубайтного обмена, 825 пониженного потребления, 355 потоковый, 656 приостановки синхронизации, 355 прямого подключения, 930 развертки, 636 растровый, 672 реальный, 268, 277 саморегенерации, 355 символьный, 567 симметричный ввода-вывода, 120 синхронный, 994 системного управления, 269, 297 сканирования, 636 совместимости, 269, 826, 952 страничного обмена, 348 страничный, 407 текстовый, 567 технологический, 475 чтения, 402 шины, 526
эмуляция пар «ведущий-ведомый», 930
рекордер, 504 рендеринг, 584, 586 ресивер, 729
ресэмплинг, 714 Рида - Соломона код, 447
розетка, 1000 рулонный плоттер, 670 рулонный сканер, 659 русификация клавиатуры, 650 ручка уровня, 724 ручной сканер, 658 ряд, 338

С

сабвуфер, 704
самоидентификация узлов, 911
саморегенерация, 355 сбой
памяти, 338 сброс, 294
аппаратный, 137, 971 маски
запроса, 119 на шине, 920
по включению питания, 971
программный, 971 счетчика, 133
таймера, 134 флага, 133 шины, 888
светодиодная матрица, 607
светодиодный принтер, 666
свипирование, 479 свопинг, 30
связка клиентских каналов, 879
связь
беспроводная, 758
инфракрасная, 860 образов
процессов, 1033 обратная, 885
через COM-порт, 853 через
LPT-порт, 838 сеанс, 896
северная часть чипсета, 186
северный мост, 192 северный хаб,
188, 192 сегмент, 24, 262, 366,
1005 сегментация, 262, 267
сегментная модель памяти, 98
сегментный регистр, 270
секвенсор, 702
сектор, 287, 444, 446, 524, 927, 960
диска, 490 заголовок, 446
загрузочный, 441 поле данных, 446
структура, 446 формат, 446 секунда,
490 секция
высокочастотная, 612
низкочастотная, 613 селектор
входа, 724 порта, 931, 949
сегмента, 280 сервер, 329, 980
приложений, 435 резервного
копирования, 435 устройства,
925 файлов, 435
сервис BIOS, 850 арбитража, 913 драйвера,
35 сервометка, 461 сервопривод, 461
сериализация, 311
серийный номер процессора, 312, 315
сессия закрытие, 509 понятие, 494
формирование, 509 сетевой адаптер, 750,
753, 839 сетевой фильтр, 89 сеть
Ethernet, 751 IEEE 1394, 902 беспроводная,
758 кабельная, 750 компьютерная, 1026
локальная, 751 неплановая, 758
принтерная, 684 с инфраструктурой, 758 с
расширенной инфраструктурой, 759
хранения данных, 436 сжатие, 443 данных,
743 изображений
внутрикадровое, 574
межкадровое, 574
информации, 573 сигнал,
750
возобновления работы, 889
данных, 835
дифференциальный, 906
композитный, 570 линейный,
906 понятие, 57 состояния,
835 управления, 835 шины
USB, 806 электрический, 60
сигнализация
о пробуждении, 875 прерываний,
301, 811 событий управления
энергопотреблением, 812
сигнальный процессор, 697, 712
сигнатура
процессора, 311, 315
устройства, 971 символ,
567
символьный режим, 567 симметричная
мультипроцессорная обработка, 304, 305
симметричный модуль памяти, 388

симметричный режим ввода-вывода, 120
синтезатор, 688 многоголосный, 701
многотембровый, 701 модульный, 702
монофонический, 701 одноголосный, 701
полифонический, 701 с аналоговым
управлением, 739 синтетический
аудиообъект, 578 синхронизатор, 625
синхронизация, 221 двойная, 59
запросов и ответов, 877 общая, 326 от источника,
60, 327 по спадам стробов, 327 потока данных, 156
процессоров, 299 совмещенная, 610 синхронная
динамическая память, 352 синхронная пакетная
статическая память, 394 синхронная передача, 65
синхронная точка, 885 синхронная флэш-память,
407 синхронное соединение, 877, 884 синхронный
обмен, 842 синхронный режим, 994 система
автоматического конфигурирования, 769
водяного охлаждения, 86 замкнутая, 461 команд,
267 ATA, 969 SCSI, 984 координат декартова, 583
ортогональная, 583 сферическая, 583
многозадачная, 142 многопроцессорная, 305
модульная, 47 мультикомпьютерная, 309 объемного
звучания, 704 однозадачная, 141 операционная, 34
привилегий, 267 прямого доступа, 437 с
избыточным контролем функционирования, 304
сообщений, 996 управления
потреблением, 83
энергопотреблением, 891
файловая, 31, 437, 506 хранения
данных, 435
системная плата, 38, 66 выбор, 256
конструктивы, 231 конфигурирование,
199 подключение внешних
интерфейсов, 239 внутренних
устройств, 239 органов лицевой панели,
234 поиск неисправностей, 241
синхронизация, 221 установка, 233
системная поддержка CD-ROM, 539
джойстика, 683 последовательной
мыши, 655 системная шина, 32, 147
системное сообщение, 738 системные
ресурсы, 134, 757 системный блок, 37,
38 системный диск, 31, 34, 437
системный код, 438 системный модуль
ROM BIOS, 160 системный таймер, 127,
133 скалярная инструкция, 273
скалярный процессор, 265 сканер, 51,
658, 838, 839 барабанный, 659
листопротяжный, 659 планшетный, 51,
659 рулонный, 659 ручка, 51 ручной, 51,
658 сканирование, 561 скан-код, 643,
648, 651 сквозная запись, 288 скорость
вращения, 458, 469
высокая, 872, 882
записи, 422 низкая,
872, 882 обмена, 28
передачи данных, 422, 490
полная, 872, 882, 1029
синхронного обмена, 1012
сканирования, 660 считывания,
422 скрытая регенерация, 347
слово, 671 двойное, 23
одинарное, 23 ответа, 722
учетверенное, 23 слой записи,
488 промежуточный, 488
считывания, 488

слот, 38, 68, 199
1,321 2, 323
3,3 B, 794
5 B, 794 AGP, 806 AGP Pro, 807 ISA, 1009 PC Card, 821, 822 PCI, 1009 PCI-X, 794 PCMCIA, 768 A, 211, 331 внутренний, 68 разделяемый, 68, 219, 794 расширения, 68 типы, 200 универсальный, 794
служебная секция, 523
смена носителя, 453
смещение адреса, 24
сектора
послойное, 466 радиальное, 466 снятие
сигнала запроса прерывания, 117 событие возобновления, 891 определение, 275 приостановки, 891 слежения, 301
совместимость накопителей, 457
программного обеспечения, 310
процессоров, 310 совмещенная
синхронизация, 610 согласование скорости обмена, 1012 ширины
шины данных, 1013 согласованное
видение памяти, 109 согласованность
данных, 306 соединение
PCI Express, 808, 810
синхронное, 877 сокет, 68, 199 370, 202, 203 423, 204 462, 211 478, 205
603, 208, 209
604, 208, 209
754,214 775, 207
939, 214
940, 214
A, 331 F, 329
сокет (*продолжение*)
ZIF, 68
A, 211 типы, 200 соленоидный привод, 460 сообщение, 190 асинхронное, 905
голосовое, 737 диагностическое, 163
канальное, 737 системное, 738
управляющее, 737 сопроцессор, 260 ввода-вывода, 260 графический, 260
математический, 260, 272 состояние
заморожено, 976 заперто, 976 отперто, 976
покоя, 778 потребления процессора, 180
устройств, 179 сна глобальное, 180 уровня производительности, 181 спекулятивное
исполнение, 266 специализированный
интерфейс, 56 специальная поддержка
графического адаптера VGA, 790
специальный регистр интерфейса, 954
специальный цикл, 780 спецификация AGP Pro, 807 ATX12V, 80 DVD+R, 500 DVD+RW, 500 PC'99, 711
быстродействия, 216 на отображаемую память, 106 на расширенную память, 107
СПИСОК
дескрипторов буферов, 721 кадров, 894
команд, 956 соединений, 724 сплиттер, 747
справедливый арбитраж, 913, 993
спрайт, 581
средства
обслуживания, 435, 949 управления энергопотреблением, 364
энергосбережения, 355 стадия передачи данных, 881 состояния, 881 установки, 881

стандарт ATX, 231
Be11_103, 742
Be11_212A, 742 IEEE
1394-1995, 900 V.17,
743 V.21, 743 V.22,
743 V.22bis, 743 V.23,
743 V.27ter, 743 V.29,
743 V.32, 743 V.32bis,
743 V.32fast, 743 V.34,
743 V.34+, 743 V.90,
743
на модуляцию, 742 стандартизованные свойства устройств, 791 стандартная память, 99, 279, 349 стандартная скорость обмена, 847 стандартный заголовок, 791 стандартный параллельный порт, 824 стандартный регистр ATA, 954 PCI IDE, 954 SATA, 954 старт по команде, 1013 старт-бит, 847 статическая память, 392 статический электрический потенциал, 603 статическое предсказание переходов, 265 стек, 285 стекер, 435
степень отражения, 586
стереоочки, 53, 608
стереопара кадров, 608
стирание обычное, 976
расширенное, 976 стоп-бит, 847 страница, 523
видеопамяти, 566
управления
логическим устройством, 997 отключением-подключением, 997 режимом, 997 страничная трансляция адресов, 263, 267, 282 страничный режим, 407 стример, 420 стробирование в режиме 1x, 804 в режиме 2x, 804 в режиме 4x, 804 в режиме 8x, 804
строка
грязная, 288 действительная, 287 недействительная, 287 чистая, 288
струйный принтер, 665 структура
данных PCI, 793 субблок
логический, 810 электрический, 810 субиерархия, 808 субканал, 489, 490 субкод, 490
сублимационный принтер, 666
субноутбук, 43
субтрактивный метод синтеза звука, 701
субтрактивный порт, 1017
суперконвейерная архитектура, 265
суперскалярный процессор, 265
сферическая система координат, 583 схема кодирования, 445 счетчик байтов, 780
съемный винчестер, 482

T

таблица атрибутов страниц, 294 волновая, 703
дескрипторов прерываний, 277 страниц, 282
физических областей, 959
командная, 956 параметров
дискет, 533 жестких дисков, 533
перенаправления прерываний ввода-вывода, 120
переопределения графических адресов, 592, 804 переходов, 265, 295 путей, 506 разделов, 438
размещения файлов, 439, 441
содержимого, 489 таймер
CMOS RTC, 134 задержки, 782 системная поддержка, 133 системный, 127 тайминг, 355 такт ожидания, 778 тактовая частота, 58, 247, 326, 375, 638 твердая копия, 660
твердокрасочный принтер, 666
твердотельная память, 419

тег
инициатора транзакции, 780
потока, 720 страницы, 287 тексел, 587
текстовая команда, 671
текстовые данные, 671
текстовый режим, 567
текстура, 587
телевизионный интерфейс, 638
телевизионный монитор, 595
телеметрия, 94 телетайпный вывод, 635 телеуправление, 94 тело
сообщения, 781 теневая маска, 595
теневая память, 105 теорема
Котельникова, 692 теплоотвод, 85
терменвокс, 701 терминал, 854
терминатор FPT SE, 1004 LVD, 1004 LVD/SE, 1004 SCSI, 1003
активный, 1004 внешний, 1004
внутренний, 1004 пассивный, 1004
универсальный, 1004 термодиод, 304
термодиффузионный принтер, 666
термокалибровка, 478 термоконтроль, 303 термомонитор, 303 термопринтер, 664 тесселяция, 585 тестирование
параллельного порта, 839
последовательного порта, 856
тетрада, 23
технологический режим, 475
технология CrossFire, 591 SLI, 590
xDSL, 747 тип
видеопамяти, 638
дисковода, 456 передачи, 876 синхронизации, 885
ток, 855
токовая петля, 735, 855 тональный звук, 699 тонкопленочная технология, 459
топология Ethernet, 751 USB
логическая, 875 физическая, 874
древовидная, 902 физических соединений, 902 шинная, 902
точка
адаптивная, 885 асинхронная, 885
доступа, 758, 760 конечная, 808, 809, 875, 879 синхронная, 885 транзакция, 775
завершение по тайм-ауту, 779 конвейерная, 801 на шине USB, 878, 881 нормальное завершение, 779 одиночная, 780 отказ, 779
отключение, 779 отложенная, 779, 782, 790
пакетная, 778, 780 повторение, 779
прекращение, 779 расщепленная, 780, 883
сигнализации прерывания, 301 смежная, 778 шинная, 892 транзитный порт, 736
трансивер Phy, 1014 виртуальный, 1015
восходящий, 1018 нисходящий, 1018
транслятор транзакций, 883 трансляция
логических адресов, 804 параметров вызова, 535 с помощью LBA, 535 сдвигов, 535 транспортный протокол, 979
транспортный уровень, 861, 945
трансформация, 584 трапеция, 601 трек, 444, 489, 738 трекбол, 50, 652, 852 тремоло, 701
трехмерная адресация, 420, 928
трехмерная геометрия, 552
трехмерный звук, 705 трилинейная фильтрация, 588

У

уведомление
асинхронное, 983
о состоянии сменного носителя, 973

удаленная загрузка, 757 удаленное пробуждение, 889
удаленный прямой доступ к памяти, 437 удельная
стоимость хранения единицы данных, 28 узел, 723 IEEE
1394, 903 с автономным питанием, 909 узкий интерфейс,
987 указатель, 24 данных, 997 инструкций, 259 команды,
997 состояния, 997 умножение частоты, 299
универсальная карта, 794 универсальное устройство
LVD, 988 универсальный интерфейс, 56 универсальный
слот, 794 универсальный терминатор, 1004
универсальный хост-контроллер, 892 уникальный 128-
битный идентификатор, 523 уникальный адрес, 897
унифицированный 34-контактный разъем, 454
унифицированный конструктив, 423 управление
заданиями, 999 интерфейсом шины SCSI, 996
параметрами кэширования, 247 потоком, 65, 848,
938, 1020 рабочими станциями, 177 шиной, 782,
904
энергопотреблением, 143, 303, 611, 812, 974
управляющая передача, 877 управляющее
сообщение, 737 управляющий импульс, 386
управляющий регистр, 365, 798 упреждающее
считывание, 468 упреждающее чтение, 289 уровень
активности, 144
защиты
высокий, 976 максимальный, 976
канальный, 810, 904, 922, 945 транзакций,
810, 904 транспортный, 861, 945
физический, 810, 813, 854, 903, 946
усилитель, 688 ускорение построений,
582 ускоренный арбитраж, 993 условный
переход, 259
усовершенствованный периферийный контроллер
прерываний, 113 установка устройств, 541 установление
соединения, 982, 998 устаревшее устройство, 792
устройство
A, 895
ATA, 821,971
ATAPI, 971
B, 896 IDE, 240
PCI, 772 SAS, 1014
SATA, 240, 971
SCSI, 978, 990
USB, 872
логическое, 875
физическое, 874
активное, 772
арифметико-логическое, 20
аутентификации, 55 блочное,
157 ввода, 167
ввода-вывода, 20, 22, 31
ведущее, 779, 783 виртуальной
реальности, 53 внешнее, 425
внутреннее, 425 вывода, 167
активное, 661 векторное, 561
пассивное, 661 растровое, 561
двухролевое, 896 загрузочное,
167 защиты, 416
инициатор транзакций, 768
комбинированное, 874
коммуникационное, 22, 32, 54, 899
конечное, 1015 логическое, 186
мультимедийное, 51 начальной загрузки,
167 периферийное, 22, 826, 872
подключенное
к интерфейсу периферийного уровня, 149 к
системной шине, 148 построчного вывода, 662
поточное, 157
работающее в реальном времени, 830
регистроориентированное, 157 с
выдвигающимся лотком, 502 с
подвижным носителем, 419 со
встроенным контроллером, 927
считывания штрих-кодов, 51 удаленной
загрузки, 169 указатель, 50, 652
устаревшее, 792 физическое, 531, 874
хранения, 22, 50, 517, 519 с
последовательным доступом, 420 с
прямым доступом, 420

устройство (*продолжение*) целевое, 768 экспандер, 1005, 1015 электронный ключ, 54 устройство-функция PCI, 956 утилита Defrag, 556 DrvSpace, 556 ScanDisk, 556 учетверенное слово, 23

Ф

фаза адреса, 775 атрибутов, 776, 780 данных, 775 сообщений, 999 файл, 30 файловая система, 31, 437, 506 файл-сервер, 435 факс-модем, 742 фантомный каталог, 457 ферроэлектрическая память, 400 физическая операция, 152 физическая топология USB, 874 физическая шина, 769 физические параметры, 514, 535 физический адрес, 26, 32, 109, 152, 281, 804 физический банк, 338 физический блок, 525 физический диск, 538 физический интерфейс IEEE 1394, 903 MIDI, 735 USB, 875 физический привод, 531 физический регистр, 266 физический уровень, 363, 810, 813, 854, 903, 946 физическое соединение, 979 физическое устройство, 434, 531, 874 фиксированный диск, 533 фиктивная команда, 972 фильтр поляризационный, 603 фильтрация билинейная, 588 трилинейная, 588 флаг активности, 438 флэш-память, 335, 404 boot block, 405 bulk erase, 405 data polling, 412 flash file, 406 NAND, 407 NOR, 406 toggle bit, 413 асимметричная архитектура, 406 защита BIOS, 174 флэш-память (*продолжение*) использование в BIOS, 173 обновление BIOS, 225 симметричная архитектура, 406 синхронная, 407 фирмы AMD, 412 Intel, 404 форманта, 701 формат PCM, 715 матрицы, 387 потока, 722 сектора, 446 форматирование, 446 в стиле винчестера, 487 дискеты, 487 верхнего уровня, 441, 447 диска, 486, 510 логическое, 554 низкоуровневое, 446, 554 форматированная емкость, 468 фотопринтер, 669 фрагментация, 442 фракция, 490 фрейм-граббер, 572 функциональная группа, 724 функция USB, 872 извлечения носителя, 534 отпирания-запирания, 534 получения параметров устройства, 534 проверки наличия расширения, 534 факта смены носителя, 534 расширенного поиска, 534 расширенного чтения, 534 расширенной верификации, 534 расширенной записи, 534 телетайпного вывода, 635 установки аппаратной конфигурации, 534

X

хаб, 188, 750 USB, 873, 891 корневой, 872 северный, 188, 192 южный, 194 Хабовая архитектура, 188 холодильник Пельтье, 86 хоп, 309 хост, 419, 434, 769, 773, 826, 827, 920 хост-адаптер SAS, 1013 SCSI, 978, 986 хост-интерфейс, 741, 948

хост-компьютер, 873 хост-
контроллер, 872, 873, 891, 939
открытый, 893 расширенный, 893
универсальный, 892 хранение данных,
738 хранилище, 435
непосредственно подключенное к серверу, 435
подсоединенное к сети, 436

Ц

ЦАП, 693 цвет, 569, 586
цветной матричный принтер, 663
цветной монитор, 595, 596 цветовая
температура, 599 цветопередача, 637
целевое устройство, 768, 800, 924, 980, 991
целостность данных, 813 цент, 700
центральный процессор, 21 цепочка кластеров,
442 команд, 981 цикл записи, 828 обмена
вложенный, 828 связанный, 828
слежения, 290 транзакций, 912
чтения, 828 шинный, 409
циклический избыточный код, 64
цилиндр, 444, 927 цифровая звуковая
карта, 713 цифровая передача
аудиопотока, 689 изображений, 573
цифровая фотокамера, 51 цифровое
управление, 601 цифровой
аудиоканал, 687

Ч

частично управляемая шина, 915 частное
кольцо, 1027 частота
FSB, 316, 326 вращения, 448 выборки
точек, 622 вывода точек, 562
обслуживания, 885 передачи данных,
336 переключения шины данных, 989
порта AGP, 222 работы RAMDAC,
637 развертки, 598
частота (*продолжение*)
регенерации, 346, 636
синхронизации памяти, 222
системной шины, 221, 299
сканирования, 636
слияния мельканий критическая, 562
тактовая, 326 тактового сигнала, 336 шины
ISA, 222 PCI, 222 памяти, 316 ядра, 221, 316
частотная модуляция, 445 чередование
банков, 337, 349 секторов, 465
чересстрочная развертка, 561 чип, 69
чипсет, 69, 185 асинхронный, 196, 222
инициализация, 198 синхронный, 222
Хабовая архитектура, 188 шинно-
мостовая архитектура, 186 число
восьмеричное, 23 двоичное, 23
десятичное, 23 шестнадцатеричное, 23
чистая болванка, 511 чистая строка, 288
чистота цвета, 600 чтение
адреса, 828
высокоприоритетное, 803 данных, 828,
838 низкоприоритетное, 803 регистров,
963 со сравнением цветов, 624
упреждающее, 289 чувствительность к
перепаду, 118 к уровню, 118, 122

Ш

шаблон
инициализации, 797 сообщения, 787
шаговый двигатель, 449 шаговый привод,
460 шина, 47 CardBus, 821 EISA, 219 I2S,
729 IEEE 1394, 903 ISA, 218, 250

шина (*продолжение*)
MCA, 219 Mini PCI, 819 PC Card, 218, 798, 820 PCI, 218, 250, 769, 772, 774, 795, 810 PCIMCIA, 218 SCSI, 240, 986, 1012 Small PCI, 818 USB, 872 VLB, 219
адреса, 327 внутренняя, 625
вспомогательная, 365 вторичная, 788
главная, 788 данных, 327 неуправляемая, 915 памяти, 337 параллельная, 427, 978
первичная, 788 подчиненная, 788
полностью управляемая, 915
расширения, 767, 768 ввода-вывода, 218
конфигурирование, 219 слоты, 219
системная, 147 физическая, 769
частично управляемая, 915 шинная топология, 902 шинная транзакция, 892
шинный цикл INTA, 116
внешнего интерфейса, 409 широковещательная асинхронная передача, 912 широковещательное сообщение, 770 шлем виртуальной реальности, 609
шлюз, 277
шум квантования, 692
шумовой звук, 699

Щ

щелевая маска, 596

Э

ЭВМ, 19
экономия энергопотребления, 958
экспандер, 1005
коммуникативный, 1006
простой, 1006
электрический интерфейс
ATA, 933
Fibre Channel, 1029 электрический сигнал, 60 электрический субблок, 810
электромагнитное излучение, 604
электронный ключ, 54, 853
электропитание бесперебойное, 93
заземление, 89 ИБП, 93
оборудования локальных сетей, 92
сетевой фильтр, 89, 92 ЭЛТ, 594 ЭЛТ-дисплей, 640 эмуляция
сопроцессора, 271 трехмерной геометрии, 552 энергонезависимая память, 335, 400 энергопотребление, 303, 975 эргономика, 603 эффект перспективы, 586 эффективный адрес, 279 эхо, 696

Ю

южная часть чипсета, 186 южный мост, 194 южный хаб, 194

Я

язык
BIFS, 578 HP-GL, 673 PCL, 672 PostScript, 673 яркость, 569 ячейка памяти, 21, 25