

КУРЕЙЧИК К.П.

УЧЕБНОЕ ПОСОБИЕ
ПО ДИСЦИПЛИНЕ "АПЭВМ"

Минск 2006

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	4
ОБЩАЯ АРХИТЕКТУРА	5
16-БИТНЫЕ ПРОЦЕССОРЫ	9
ОБЩИЕ СВЕДЕНИЯ.....	9
РЕГИСТРЫ ОБЩЕГО НАЗНАЧЕНИЯ.....	9
ИНДЕКСНЫЕ РЕГИСТРЫ.....	10
СЕГМЕНТНЫЕ РЕГИСТРЫ	10
СЧЕТЧИК КОМАНД.....	10
РЕГИСТР ФЛАГОВ.....	11
32-БИТНЫЕ ПРОЦЕССОРЫ	12
ОБЩИЕ СВЕДЕНИЯ.....	12
ФИЗИЧЕСКАЯ СТРУКТУРА	13
РЕГИСТРЫ.....	13
РЕАЛЬНЫЙ РЕЖИМ	17
ОБЩИЕ СВЕДЕНИЯ.....	17
РАБОТА С АДРЕСАМИ.....	17
ОБРАБОТКА ПРЕРЫВАНИЙ	18
ЗАЩИЩЕННЫЙ РЕЖИМ	19
ОБЩИЕ СВЕДЕНИЯ.....	19
ОБРАБОТКА ПРЕРЫВАНИЙ	19
РАБОТА С АДРЕСАМИ.....	20
ПОРТЫ	26
ОСНОВНЫЕ ПОНЯТИЯ И ТЕРМИНЫ	26
ПОРТЫ АСИНХРОННОГО АДАПТЕРА.....	29
ПРОГРАММИРОВАНИЕ АСИНХРОННОГО АДАПТЕРА	33
ПАМЯТЬ	36
ПАМЯТЬ	36
ЛОГИЧЕСКАЯ ОРГАНИЗАЦИЯ ПАМЯТИ.....	36
ФИЗИЧЕСКАЯ ОРГАНИЗАЦИЯ ПАМЯТИ.....	39
КЭШИРОВАНИЕ ПАМЯТИ	39
БАНКИ ПАМЯТИ.....	40
ТЕНЕВАЯ ПАМЯТЬ	40
ФАЙЛОВАЯ СИСТЕМА	41
ОБЩИЕ СВЕДЕНИЯ.....	41
FDD.....	41
HDD	44
ИНТЕРФЕЙС АТА	46
Enhanced IDE	48
КЛАВИАТУРА	51
УСТРОЙСТВО	51
ПОРТЫ КЛАВИАТУРЫ.....	52
ПРЕРЫВАНИЯ КЛАВИАТУРЫ	53
ВИДЕОСИСТЕМА	56

АРХИТЕКТУРА.....	56
ТИПЫ ВИДЕОСИСТЕМ	57
ВИДЕОРЕЖИМЫ.....	59
ЗВУКОВАЯ СИСТЕМА	63
ЗВУКОВЫЕ КАРТЫ.....	63
ПОРТЫ SOUND BLASTER.....	63
ПРЕРЫВАНИЯ SOUND BLASTER.....	64
DSP	64

ВВЕДЕНИЕ

В настоящее время в программу обучения студентов 3-го курса Белорусского Государственного Университета Информатики и Радиоэлектроники специальности ВМСиС входит предмет "Архитектура персональных ЭВМ". Сходные курсы читаются на других специальностях БГУИР и в других ВУЗах. Данный курс является основополагающим в подготовке будущих инженеров-системотехников. Это фундамент, на котором строится вся система знаний, создающая из простого студента высококлассного специалиста. Курс "АПЭВМ" охватывает широкую область знаний о мире персональных компьютеров, изучение которой требует огромной массы времени и сил.

Данный документ направлен на усовершенствование методики изучения предмета и является методическим пособием по курсу "АПЭВМ". Пособие по данному курсу чрезвычайно своевременно, так как в настоящее время для изучения АПЭВМ студентам предоставляется множество технической литературы, которая либо устарела, либо тяжела для восприятия и изучения. В современной литературе очень мало места уделяется различным пособиям и учебникам, где были бы собраны основополагающие принципы вычислительной техники. В единичных экземплярах встречается литература, где приведены примеры рабочих программ. Очень мало электронных пособий и справочников, которые гораздо удобнее и практичнее обычных книг.

Информация, содержащаяся в документе, тщательно структурирована с целью облегчения поиска нужного раздела, так что ориентироваться в данном справочнике не составит труда никому. Информация по каждому из разделов бралась из конспектов лекций (курс "Архитектура ПВЭМ" читается как на дневном отделении 3-го курса нашего университета, так и на вечернем и заочном отделениях), а также из других источников (технической литературы, справочников, пособий, сети Internet).

Данный проект рассчитан прежде всего на студентов заочного и вечернего отделений ВУЗов, которым данный учебник позволяет изучать предмет в домашних условиях. Но этот справочник также будет полезен и студентам дневного отделения как при написании курсовых и дипломных (благодаря большому количеству исходного кода, часто используемого в качестве шаблона), так и при подготовке к экзаменам.

ОБЩАЯ АРХИТЕКТУРА

Архитектура ПЭВМ - совокупность методов, алгоритмов и аппаратных средств, предназначенных для функционирования ЭВМ.

Архитектура современной ПЭВМ может быть представлена схемой на рис. 1. Современные ПЭВМ, в отличие от самых первых, обзавелись большим количеством шин, к которым подключаются внешние устройства. Это связано с тем, что не все устройства могут работать на той скорости и тактовой частоте, на которой работает *центральный процессор*. На схеме вверху - самые быстрые устройства (процессор работает на частоте 300-500 или больше мегагерц), внизу - самые медленные (устройства на *шине ISA* - не более 16-ти мегагерц). Для того, чтобы более медленные устройства не замедляли более быстрые, введены несколько промежуточных шин и контроллеров.

Видеосистема переместилась на собственную шину в связи с возрастающей потребностью в более сложной графике, к тому же современные ПЭВМ должны справляться с потоком видеоданных, получаемых с видеокамер и видеомэгафонов для осуществления кодирования и обработки в реальном времени - раньше персональные ЭВМ не годились для таких целей. *Шина PCI* больше не может обслуживать видеоподсистему, тем более *шина ISA* - самая первая системная шина у ПЭВМ.

Память более не подключается к общей шине, а управляется своим *контроллером*, который, в свою очередь, подключен к *шине процессора*. Между памятью и процессором добавлена *кэш-память*, доступ к которой осуществляется быстрее, т.к. она относится к статической памяти. В кэш-памяти сохраняется порции памяти, использовавшиеся ранее. В случае запроса тех порций, которые уже есть в кэше, доступ к обычной *ОЗУ* на производится. В скором времени общая шина будет заменена новой *шиной USB*, позволяющей подключать устройства без выключения питания. Все необходимые ресурсы (порты, прерывания, адреса памяти) будут назначаться автоматически по технологии Plug&Play.

Все внешние устройства, подключенные ко всем шинам, получают порты. Через порты можно получить доступ ко внутренним регистрам состояния устройств и выдавать команды управления. Обратную связь также организуют через аппаратные прерывания (IRQ).



Рис. 1. Архитектура ПЭВМ.

CENTRAL PROCESSING UNIT

- устройство, отвечающее за обработку данных и управление всеми устройствами и узлами ПЭВМ.

CASHE LEVEL1, LEVEL2

-промежуточная статическая память (*SRAM*), буферизирующая доступ к основной памяти путем сохранения ранее запрошенных порций и выдачи их по новому запросу, не обращаясь к основной памяти. Работает на частоте процессора и размещается на его кристалле, за счет чего быстрее обычной кэш-памяти (смотри Системная память)

ШИНА ПРОЦЕССОРА

- магистраль, используемая *процессором* для обмена данными и управляющими сигналами с остальными устройствами и узлами ЭВМ (непосредственно - с контроллером памяти)

КОНТРОЛЛЕР ПАМЯТИ

(North Bridge) - сложный узел современной ЭВМ, осуществляющий координацию потока данных между *процессором*, *ОЗУ* и локальными шинами *PCI* и *AGP*. В настоящее время - самая ответственная часть подсистемы памяти, влияющая на производительность ПЭВМ.

CASHE

-промежуточная статическая память (*SRAM*), буферизирующая доступ к основной памяти путем сохранения ранее запрошенных порций и выдачи их по новому запросу, не обращаясь к основной памяти. (смотри Системная память)

ОЗУ

Оперативное запоминающее устройство, или память - устройство для кратковременного хранения информации, используемой непосредственно для вычислений. Базируется на динамической памяти (*DRAM*).

VIDEO

- набор контроллеров, регистров и памяти, предназначенный для вывода текста и графики на монитор, телевизионный приемник и другие средства отображения информации. В настоящее время также является устройством ввода и обработки видеоизображений.

PCI

Peripheral Component Interconnect - локальная шина, работающая быстрее *шины ISA* и имеющая множество преимуществ - высокую тактовую частоту, встроенное распределение ресурсов (Plug&Play), захват устройством всей шины и т.д.

MOST

- контроллер, согласующий работу двух шин, управляющий работой дисковых устройств.

ATA

AT Attachment - дисковые устройства хранения информации - жесткий диск со встроенным контроллером (EIDE HDD), устройства обработки оптических дисков (CD-ROM, CD-R, CD-RW), магнитная лента и т.п..

USB

Universal Serial Bus - новая шина для подключения внешних устройств с возможностью самонастройки.

ISA

Industrial Standard Architecture - самая первая шина, использованная в ПЭВМ. В настоящее время все меньше устройств подключается к этой шине в связи с малой пропускной способностью и тактовой частотой (не более 16 Мегабайт/с при 16 Меггерцах).

ROM BIOS

Read - Only Memory Base Input-Output System - микросхема ПЗУ, хранящая программы настройки и самодиагностики ПЭВМ, а также часть системы ввода-вывода для обмена с дисковой и другими подсистемами.

SRAM

Static Random Access Memory - статическая память, не требующая циклов регенерации для предотвращения потери (в отличие от *DRAM*). Доступ к ней осуществляется быстрее, особенно если она находится на одном кристалле с *процессором* и работает на его тактовой частоте.

DRAM

Dynamic Random Access Memory - динамическая память, требующая циклов регенерации для предотвращения потери (в отличие от *SRAM*). Доступ к ней осуществляется за 5-7 тактов, а в случае последовательного доступа - за 2 или (в случае синхронной памяти *SDRAM*) за 1 такт задержки между циклами выдачи информации.

16-БИТНЫЕ ПРОЦЕССОРЫ

ОБЩИЕ СВЕДЕНИЯ

16-битные процессоры - первые процессоры, появившиеся в персональных ЭВМ. Процессор, стоявший в компьютере IBM PC, был изготовлен фирмой Intel, назывался i8086 и работал на тактовой частоте 4,77 МГц. Позднее был выпущен процессор i8088, отличавшийся тем, что его внутренняя *шина данных* была 8-ми разрядной. В дальнейшем были выпущены аналогичные ему процессоры V20, V30 фирмы NEC. Процессоры следующего поколения, 80186, 80188, 80286, тоже были 16-ти разрядными, хотя имели более высокую тактовую частоту и возможность работы с памятью выше 1 Мбайта в *защищенном* режиме.

Коротко 16-ти битные процессоры можно описать:

Разрядность ядра - 16 бит

Число регистров - 14

Разрядность шины данных : внутренняя - 16 или 8 бит, внешняя - 16 бит

Адресная шина - 20 бит или 24 бит (80286)

Внутренняя кэш-память - отсутствует

Внешняя шина для подключения устройств ввода/вывода - ISA (Industry Standard Architecture), 16 бит, 8 МГц

Внутренняя логическая структура:

РЕГИСТРЫ ОБЩЕГО НАЗНАЧЕНИЯ

AX:

Accumulator - этот регистр используется во многих арифметических операциях как один из операндов, явно и без явного указания (MUL, SUB, DIV). Многие арифметические операции оптимизированы под использования именно этого регистра. Старшая его часть (AH) используется для указания номера вызываемой функции в обработчиках прерываний (ISR), а также для возврата кода ошибки.

BX:

Base - этот регистр используется как второй операнд в арифметических операциях (MUL, DIV). Его можно использовать как индексный регистр для указания смещения относительно базы (как и DI,SI,BP)

CX:

Counter - регистр используется при организации циклов (loop) и цепочечных операций (rep movs, stos, lods и т.д.)

DX:

Data - регистр общего назначения.

ИНДЕКСНЫЕ РЕГИСТРЫ

SI:

Source Index - применяется для указания смещения относительно регистра DS в операциях с цепочками байтов (stos, lods, movs..). Также - на общих основаниях.

DI:

Destination Index - применяется для указания смещения относительно регистра ES в операциях с цепочками байтов (stos, lods, movs..). Также - на общих основаниях.

BP:

Base Pointer - используется для указания смещения относительно базового сегментного регистра (обычно SS) для операций передачи параметров в подпрограммы через стек. Также - на общих основаниях.

SP:

Stack Pointer - смещение в стеке до первой свободной ячейки относительно SS. Модифицируется аппаратно по командам push, pop, call, ret и др. Для арифметических операций не пригоден.

СЕГМЕНТНЫЕ РЕГИСТРЫ

CS:

Code Segment - указатель сегмента кода(см. Реальный режим, Виртуальный, Защищенный режим). Доступен - только для для чтения.

DS:

Data Segment - указатель сегмента данных(см. Реальный режим, Виртуальный, Защищенный режим). Доступен для чтения и записи из другого регистра. По умолчанию данные адресуются через этот регистр.

ES:

Extra Segment - дополнительный сегментный указатель. Для различных нужд адресации к данным.

SS:

Stack Segment - указатель сегмента стека. Доступен для чтения и записи через регистр.

СЧЕТЧИК КОМАНД

IP:

Instruction Pointer - используется для указания адреса следующей команды относительно сегмента, записанного в CS. Программно не доступен.

РЕГИСТР ФЛАГОВ



Рис.2 Регистр флагов.

OF:

Overflow Flag - флаг переполнения (выхода из старшего разряда в знак) результата на выходе АЛУ при выполнении арифметических операций. Соответствует переходу единицы в старший разряд из предыдущего.

DF:

Direction Flag - флаг направления цепочечных команд (stos, movs, lods и т.п.). При единице регистры DI и SI уменьшаются на единицу, иначе - увеличиваются, что дает возможность получать предыдущий или последующий байт, соответственно.

IF:

Interrupt Flag - флаг запрета прерываний. Если флаг = 1, то процессор не реагирует на поступающие аппаратные прерывания (IRQ), кроме немаскируемого (NMI), и они блокируются.

TF:

Trace Flag - флаг трассировки. В этом режиме процессор вызывает прерывание 04 после выполнения каждой инструкции.

SF:

Sign Flag - флаг отрицательного результата на выходе АЛУ при выполнении арифметических операций и операций сравнения (как признак “меньше”). Соответствует единице в старшем разряде.

ZF:

Zero Flag - флаг нулевого результата на выходе АЛУ при выполнении арифметических операций и операций сравнения (как признак равенства).

AF:

Auxiliary Flag - флаг переноса из старшей тетрады при выполнении арифметических операций с двоично-десятичными упакованными числами (BCD). Эта тема в данном справочнике не рассматривается.

PF:

Parity Flag - флаг четного числа единиц на выходе АЛУ (обычно - в AX) при выполнении арифметических операций.

CF:

Carry Flag - флаг переноса из старшего разряда (обычно - AX) при выполнении арифметических операций, а также - флаг ошибки обработчика прерываний.

32-БИТНЫЕ ПРОЦЕССОРЫ

ОБЩИЕ СВЕДЕНИЯ

Родоначальником семейства 32-разрядных процессоров для ПЭВМ стал процессор Intel 80386. Этот процессор был значительным шагом вперед по сравнению с предыдущими процессорами этой фирмы. Сохраняя полную совместимость с 16-битными процессорами на уровне команд, он имеет следующие новшества:

- 32-разрядные регистры, шины и команды
- реализованный на аппаратном уровне механизм работы с виртуальной памятью
- механизм защиты памяти, принадлежащей разным задачам - по привилегиям;
- расширенные возможности отладки
- режим виртуальной машины Virtual86
- повышенная производительность, связанная с улучшенным технологическим процессом производства СН MOS3 и конвейерной обработкой данных

Следующий процессор, Intel 80486, стал вдвое более производительным и воплотил в себе следующие нововведения:

- встроенный математический сопроцессор
- кэш-память размером 8 Кбайт, размещенная на кристалле процессора
- Процессор Pentium, пришедший на смену i80486, был еще более насыщен новыми технологическими новинками, которые включали в себя:
- суперскалярную архитектуру (несколько команд за один такт)
- раздельное кэширование данных и команд
- 64-разрядная шину данных
- поддержка многопроцессорности (впрочем, ограниченная)
- средства задания размера страницы памяти
- возможности самодиагностики и управления производительностью

Процессоры семейства Pentium II разделились на несколько подсемейств, ориентированных на различные задачи, но общим у них является:

- новый набор команд MMX для работы с 64-разрядными данными
- кэш-память второго уровня (первый уровень - 64 Кбайт - размещен непосредственно в ядре) размером до 512 Кбайт, работающая на половинной частоте процессора
- динамическое исполнение - возможность программирования некоторых микроопераций процессора, перезаписывая особое ПЗУ микрокода

Так как аналоги процессоров Pentium не имеют каких-либо значимых отличий с точки зрения архитектуры, то рассмотрены отдельно не будут.

ФИЗИЧЕСКАЯ СТРУКТУРА

Физическая структура процессора рассмотрена на примере процессора Pentium. Схема приведена ниже.

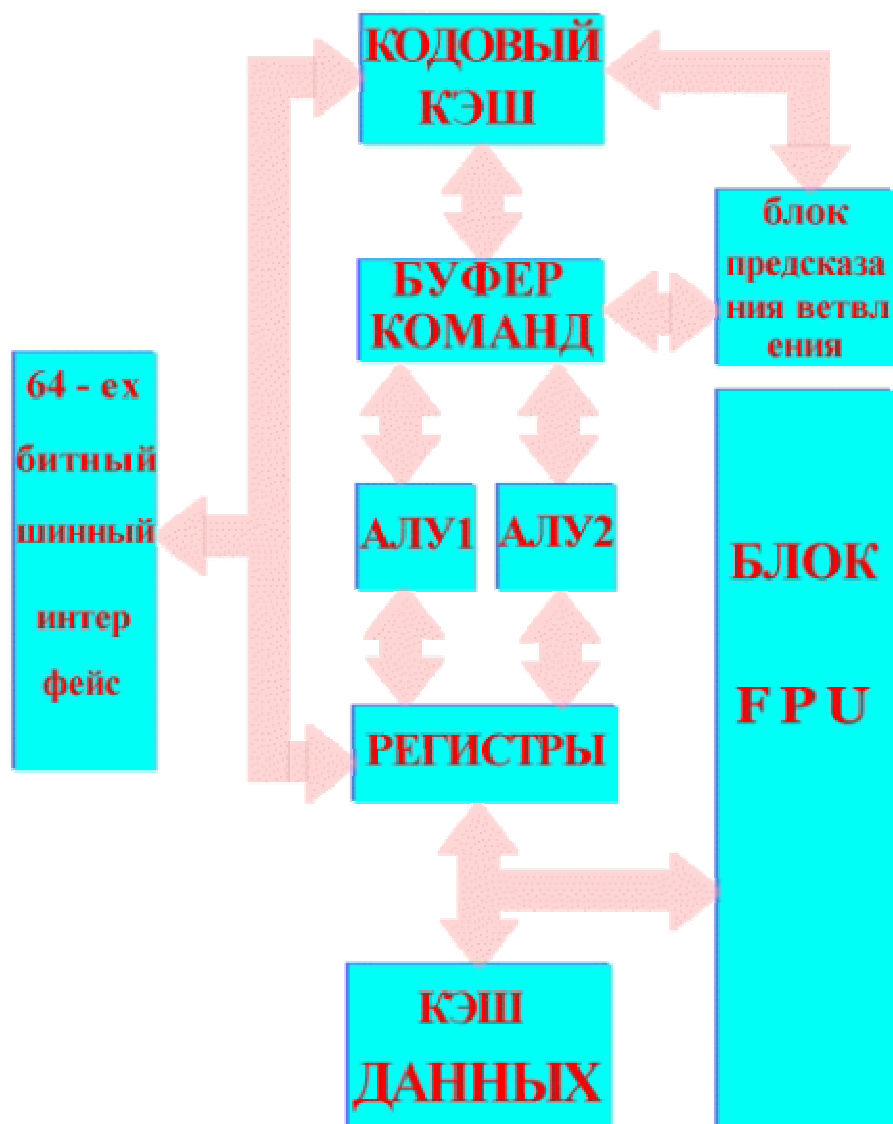


Рис. 3. Физическая структура процессора Pentium.

РЕГИСТРЫ

Процессоры всех семейств имеют общий набор регистров.

РЕГИСТРЫ ОБЩЕГО НАЗНАЧЕНИЯ

EAX

AX

AH

AL

EBX

BX

BH

BL

ECX
CX
CH
CL
EDX
DX
DH
DL
ESI
SI
EDI
DI
EBP
BP
ESP
SP

СЕГМЕНТНЫЕ РЕГИСТРЫ

CS
DS
ES
SS
FS
GS

РЕГИСТР ФЛАГОВ

AC:

флаг проверки выравнивания. При обращении к невыровненному адресу (например, к нечетному при установленном выравнивании на границе слова) генерируется исключение. Только в 486 и выше.

VM:

virtual mode - флаг виртуального режима процессора. Установка флага в 1 переводит процессор в этот режим.

RF:

возобновление. При установленном флаге исключительная ситуация не генерирует отладочного исключения, что позволяет выполнить следующую команду. Используется в отладчиках для возможности обработки исключительной ситуации, не позволяя отладчику вызывать самого себя при возникновении исключения.

NT:

флаг вложенности задачи. Используется для вызовов вложенных задач и возврату к прерванной задаче (обычно при обработке прерываний).

IOPL:

содержит уровень привилегий ввода-вывода

OF

DF
IF
TF
SF
ZF
AF

РЕГИСТРЫ УПРАВЛЕНИЯ ПАМЯТЬЮ

GDTR:

global descriptor table register - регистр глобальной таблицы дескрипторов. Содержит базовый адрес и границу сегмента. Загрузка регистра возможна указанием селектора сегмента, остальная информация берется из таблицы дескрипторов (селектор - индекс в таблице). Большинство защищенных систем не позволяют обращаться к этому регистру.

LDTR:

local descriptor table register - регистр локальной таблицы дескрипторов. Содержит базовый адрес, границу сегмента и селектор сегмента (индекс в таблице глобальных дескрипторов - GDT). Загрузка регистра возможна указанием селектора сегмента, остальная информация берется из таблицы дескрипторов (селектор - индекс в таблице). Большинство защищенных систем не позволяют обращаться к этому регистру.

IDTR

TR:

task register - регистр задачи. Содержит базовый адрес, границу сегмента. Когда происходит прерывание, вектор прерывания используется как индекс в таблице дескрипторов прерываний для получения адреса шлюза (см. Защищенный Режим). Большинство защищенных систем не позволяют обращаться к этому регистру.

РЕГИСТРЫ УПРАВЛЕНИЯ

CR0:

регистр состояния процессора

CR1:

зарезервированный регистр

CR2:

регистр сбоя подкачки страниц. Если при загрузке или выгрузке страниц возникает ошибка, этот регистр содержит линейный адрес команды, вызвавшей ошибку

CR3:

базовый регистр страничного каталога. Используется при реализации страничного механизма работы с памятью

ОТЛАДОЧНЫЕ РЕГИСТРЫ

Используются при отладке программ, особенно зашитых в ПЗУ. Все имеют разрядность 32 бита.

DR0 -- DR3:

линейные адреса контрольных точек останова

DR6:

состояние отладки

DR7:

регистр управления отладкой

DR4, DR5:

зарезервированы

ТЕСТОВЫЕ РЕГИСТРЫ

TR3 -- TR7:

физически зависят от конкретной реализации процессора. Используются для проверки работы кэша и блока ассоциативной трансляции

РЕАЛЬНЫЙ РЕЖИМ

ОБЩИЕ СВЕДЕНИЯ

Реальным называется режим работы процессора после начальной загрузки. Процессоры первого поколения (i8088 и i8086) могли работать только в таком режиме. Процессор i80286 и 32-битные процессоры могут работать и в других режимах (защищенном и виртуальном), но после перезагрузки системы все равно находятся в реальном режиме. Этот режим является базовым режимом для операционной системы DOS, другие системы также используют его перед переключением в другие режимы.

РАБОТА С АДРЕСАМИ

В реальном режиме линейный (физический) адрес формируется при помощи сегмента и смещения. Сегмент обычно хранится в специальном регистре - сегментном или указывается явно. При формировании 20-разрядного адреса (именно такова разрядность адресной шины первых процессоров) сегмент содержит первые 16 разрядов. Смещение содержит младшие 16 разрядов, 12 из которых добавляются к сегменту. Такая схема получения адреса связана с тем, что в первых процессорах разрядности регистров не хватало для формирования полного адреса.



Рис. 4. Схема получения линейного адреса.

Логично предположить, что при сложении сегмента и части смещения может возникнуть переполнения. В этом случае процессоры 8088 и 8086 просто отбросят старший, 21-ый бит и адресация будет производиться от начала памяти. Старшие процессоры поступят также в случае, если их адресная линия A20 (т.е. 21 бит на адресной шине) заблокирована. В противном случае произойдет обращение к памяти за пределами 1-го мегабайта, что используется при работе по интерфейсу НМА.

В отличие от других режимов, в реальном режиме вся память является доступной для чтения и записи (кроме областей ROM BIOS) всем программам, что не позволяет организовать безопасную многозадачную работу.

ОБРАБОТКА ПРЕРЫВАНИЙ

При выполнении инструкции INT или генерации аппаратного (асинхронного) прерывания вызывается системная процедура, адрес которой содержится в таблице векторов прерываний. Для получения нужного индекса в таблице номер прерывания умножается на 4. Из таблицы выбирается дальний указатель (4 байта - сегмент и смещение) на точку входа, сохраняются в стеке CS, IP и флаги, блокируются прерывания (флаг IF), трассировка (флаг TF) и выполняется переход в точку входа. По команде IRET управление возвращается в точку вызова или прерванного управления (в случае асинхронного прерывания). Адрес таблицы прерываний для процессоров 808x всегда один - начало памяти (0000:0000), для старших процессоров он определяется содержимым регистра IDTR.

ЗАЩИЩЕННЫЙ РЕЖИМ

ОБЩИЕ СВЕДЕНИЯ

Перевод процессора в защищенный режим позволяет полностью реализовать все возможности, заложенные в его архитектуру и недоступные в реальном режиме. Сюда можно отнести:

Увеличение адресуемого пространства до 4 Гбайт.

Возможность работать в виртуальном адресном пространстве.

Организация многозадачного режима.

Страничная организация памяти.

ОБРАБОТКА ПРЕРЫВАНИЙ

В реальном режиме для обработки прерываний используется таблица векторов прерываний, расположенная в первом килобайте адресного пространства. Эта таблица состоит из 256 элементов размером 4 байта, которые содержат полный адрес обработчиков прерывания в формате <сегмент:смещение>.

Как вы знаете, существуют аппаратные и программные прерывания. Аппаратные прерывания вырабатываются периферийными устройствами, как правило, при завершении ими операции ввода/вывода. Эти прерывания являются асинхронными по отношению к запущенным программам. Программные прерывания вызываются командой INT. Программные прерывания являются синхронными, так как они инициируются самой программой.

В ответ на прерывание любого типа в реальном режиме в регистры CS:IP процессора загружается адрес, взятый из соответствующей ячейки таблицы векторов прерываний, после чего управление передается по этому адресу. Обработчик прерываний, выполнив все необходимые действия, возвращает управление прерванной программе, выполняя команду IRET.

Программы MS-DOS широко используют программные прерывания для получения обслуживания от MS-DOS и BIOS.

Механизм обработки прерываний в защищенном режиме намного сложнее. Для определения адресов обработчиков прерываний в защищенном режиме используется дескрипторная таблица прерываний IDT (Interrupt Descriptor Table), расположение которой определяется содержимым специального системного регистра. Эта таблица содержит дескрипторы специальных типов - вентили прерываний, вентили исключений и вентили задач.

Вентиль прерываний содержат не только логический адрес обработчика прерывания, но и поле доступа. Программа может вызвать прерывание только в том случае, если она имеет для этого достаточный уровень доступа. Таким образом, операционная система, работающая в защищенном режиме, может запретить прикладным программам вызывать некоторые или все программные прерывания.

Обычные приложения Windows никогда не должны вызывать программные прерывания, так как для взаимодействия с операционной системой используется другой механизм, основанный на вызове функций из библиотек динамической загрузки. Тем не менее, некоторые прерывания (например, INT 21h) все же можно использовать. Для таких прерываний Windows выполняет трансляцию адресов из формата защищенного режима в формат реального режима.

Приложение Windows не должно пытаться изменить дескрипторную таблицу прерываний. Не следует также думать, что эта таблица расположена по адресу 0000h:0000h, селектор 0000h вообще не используется для адресации памяти.

Если в этом нет особой необходимости, приложение Windows не должно вызывать программные прерывания. Для работы с файлами, принтером, для вывода на экран следует вызывать функции программного интерфейса операционной системы Windows.

РАБОТА С АДРЕСАМИ

В защищенном режиме, как и в реальном, логический адрес состоит из двух компонент. Однако эти компоненты называются не сегмент и смещение, а селектор и смещение. Для вычисления физического адреса в процессоре 80286 используются также две таблицы дескрипторов - глобальная таблица дескрипторов GDT (Global Descriptor Table) и локальная таблица дескрипторов LDT (Local Descriptor Table). Селектор используется для адресации ячейки одной из таблиц дескрипторов, содержащей помимо прочей информации базовый 24-разрядный адрес сегментов. Для получения физического адреса базовый адрес складывается со смещением, расширенным до 24 разрядов (рис. 5).

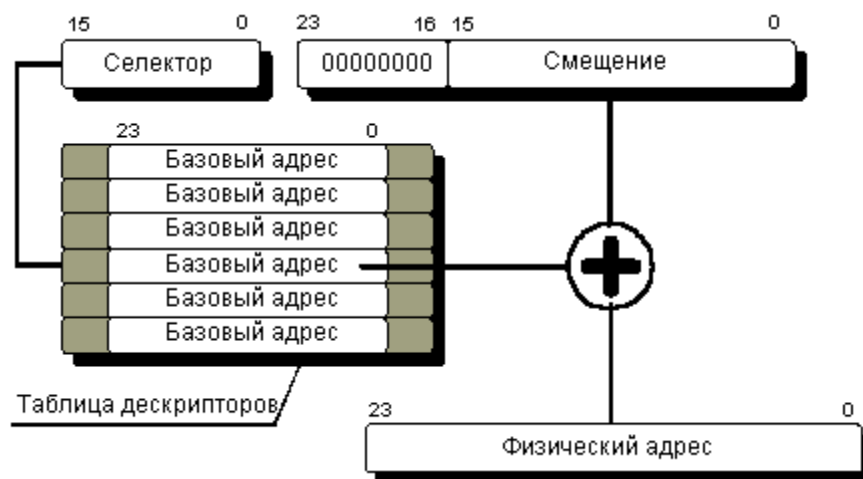


Рис. 5. Получение физического адреса в процессоре 80286.

Согласно этой схеме адресации памяти, селектор содержит номер ячейки таблицы дескрипторов, но не компоненту физического адреса. Программа может задавать не любые значения селекторов, а только те, которые соответствуют существующим ячейкам таблицы дескрипторов. Разумеется,

программа может загрузить в сегментный регистр любое значение, однако при попытке обратиться к сегменту памяти с использованием неправильного селектора работа программы будет прервана.

Таким образом, несмотря на то, что компоненты адреса остались, как и в реальном режиме, 16-разрядными, новая схема адресации защищенного режима процессора 80286 позволяет адресовать до 16 Мбайт памяти, так как в результате преобразования получается 24-разрядный физический адрес.

Кроме индекса, используемого для выбора ячейки дескрипторной таблицы при формировании физического адреса, селектор содержит еще два поля (рис. 6).

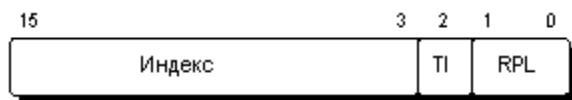


Рис.6. Формат селектора.

Поле TI (Table Indicator) используется для выбора таблицы дескрипторов. Как мы уже говорили, существуют таблицы дескрипторов двух типов. В любой момент времени может использоваться одна глобальная таблица дескрипторов и одна локальная таблица дескрипторов. Если бит TI равен 0, для выборки базового адреса используется глобальная таблица дескрипторов GDT, если 1 - локальная LDT.

Зачем нужно использовать дескрипторные таблицы двух типов?

В мультизадачной операционной системе можно использовать одну глобальную таблицу дескрипторов для описания областей памяти, принадлежащей операционной системе и несколько локальных таблиц дескрипторов для каждой задачи. В этом случае при соответствующей настройке базовых адресов можно изолировать адресные пространства операционной системы и отдельных задач. Если сделать так, что каждая задача будет пользоваться только своей таблицей дескрипторов, любая задача сможет адресоваться только к своим сегментам памяти, описанным в соответствующей таблице, и к сегментам памяти, описанным в глобальной таблице дескрипторов. В системе может существовать только одна глобальная таблица дескрипторов.

Поле RPL (Requested Privilege Level) селектора содержит уровень привилегий, запрошенный программой при обращении к сегменту. Программа может обращаться только к таким сегментам, которые имеют соответствующий уровень привилегий. Поэтому программа не может, например, воспользоваться глобальной таблицей дескрипторов для получения доступа к описанным в ней системным сегментам, если она не обладает достаточным уровнем привилегий. На этом основана защита системных данных от разрушения (преднамеренного или в результате программной ошибки) со стороны прикладных программ.

Таблица дескрипторов содержит, помимо базового адреса сегмента, другую информацию, описывающую сегмент (рис. 7). Точный формат дескриптора, а также других структур данных и системных регистров, имеющих отношение к работе в защищенном режиме, вы можете найти в 6 томе "Библиотеки системного программиста".

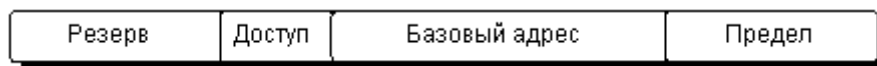


Рис. 7. Формат дескриптора сегмента процессора 80286.

В частности, дескриптор содержит размер сегмента (предел). При вычислении физического адреса процессор следит за тем, чтобы физический адрес не выходил за пределы, указанные в дескрипторе сегмента. Программа не может обратиться к памяти, лежащей вне пределов, указанных в дескрипторе. Если же она попытается это сделать, ее работа будет прервана. Поэтому, создав, например, сегмент для хранения массива размером 100 байт, вы не сможете записать в него 101 байт или 10 Кбайт данных.

Поле доступа содержит уровень привилегий сегмента и информацию о типе сегмента. Существуют сегменты кода, сегменты данных и системные сегменты. Кроме того, для сегмента кода можно запретить операцию чтения, а для сегмента данных - операцию записи. Поэтому операционная система может создать сегменты кода, которые можно выполнять, но нельзя читать, и сегменты данных, защищенные от записи.

Что же касается привилегий, в процессорах 80xxx и Pentium существуют четыре уровня привилегий - от 0 до 3, причем наибольшие привилегии соответствуют уровню 0. Уровни привилегий часто называют также кольцами защиты (рис. 8).

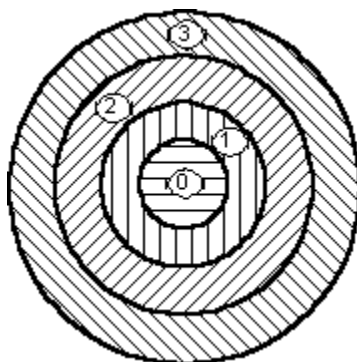


Рис. 8. Кольца защиты.

В кольце 0 обычно работает ядро операционной системы. Кольцо 1 соответствует уровню привилегий драйверов, кольцо 2 - системам, таким как системы управления базами данных. В наименее привилегированном кольце 3 располагаются прикладные программы, запускаемые пользователем.

Описанная выше схема распределения привилегий может изменяться от одной операционной системы к другой. В операционной системе Windows 3.1 в нулевом кольце располагаются только виртуальные драйверы, все остальные модули Windows, а также приложения, работают в кольце 3.

Процессоры 80386, 80486 и Pentium используют более сложную схему адресации памяти, которая, однако, остается прозрачной для программиста.

Преобразование адреса в этих процессорах является многоступенчатым. Программы адресуют память с помощью логического адреса, состоящего из 16-разрядного селектора и 32-разрядного смещения. Так же, как и в процессоре

80286, селектор используется для выборки базового адреса сегмента из глобальной или локальной таблицы дескрипторов. Отличие заключается в том, что во-первых, используются 32-разрядные базовый адрес и смещение, а во-вторых, результат сложения называется линейным адресом и используется не для непосредственной адресации памяти, а для дальнейших преобразований (рис. 9).

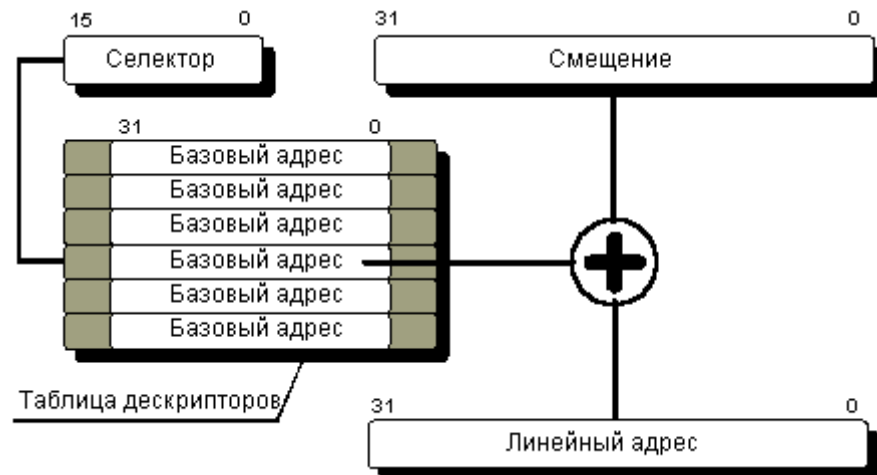


Рис. 9. Преобразование логического адреса в линейный.

Старшие десять бит линейного адреса используются как индекс в каталоге таблиц страниц (рис. 10).



Рис. 10. Преобразование линейного адреса в физический.

В каталоге таблиц страниц хранятся дескрипторы, содержащие помимо другой информации, физические адреса таблиц страниц.

Следующие десять бит линейного адреса используются для выбора из таблицы страниц, адрес которой определяется старшими десятью битами линейного адреса.

Таблица страниц может описывать до 1024 страниц размером 4096 байт.

Младшие двенадцать бит линейного адреса содержат смещение адресуемого байта внутри страницы.

Отметим, что преобразование линейного адреса в физический выполняется процессором с помощью каталога таблиц страниц и таблиц страниц, подготовленных операционной системой. Программист, создающий приложения для Windows, никогда не работает с таблицами страниц или каталогом таблиц страниц. Он пользуется логическим адресом в формате <селектор:смещение>, поэтому схема преобразования логического адреса в физический остается для него прозрачной.

Операционная система Microsoft Windows версии 3.1 может работать, как вы знаете, в стандартном и расширенном режиме. В первом случае используется схема адресации процессора 80286, даже если в компьютере установлен процессор 80386. Если Windows работает на процессоре 80386, 80486 или Pentium, при наличии достаточного объема оперативной памяти (больше 2 Мбайт) по умолчанию используется расширенный режим работы и, соответственно, схема преобразования адресов процессора 80386.

Основное преимущество системы управления памятью расширенного режима работы Windows заключается в использовании виртуальной памяти. Виртуальная память работает на уровне страниц (описанных в каталогах страниц) и совершенно прозрачна для программиста. Операционная система Windows полностью управляет виртуальной памятью. Если программа пытается обратиться к странице, отсутствующей в памяти и выгруженной на диск, происходит прерывание работы программы, страница подгружается с диска, вслед за чем работа программы продолжается. Программа может заказывать для себя блоки памяти огромного размера, адресуясь к ним непосредственно, при этом возникает полная иллюзия работы с оперативной памятью большого размера.

Описанная выше схема адресации в защищенном режиме накладывает ограничения на операции, которое приложение Windows может выполнять над селекторами.

Приложение Windows не должно выполнять над селекторами арифметические операции и операции сравнения.

Программируя для реального режима операционной системы MS-DOS, вы, возможно, при адресации блока памяти большого размера (больше 64 Кбайт) изменяли содержимое сегментных регистров. В защищенном режиме вы не можете делать никаких предположений относительно базового адреса следующего или предыдущего дескриптора в локальной или глобальной таблице дескрипторов.

Сказанное не означает, что приложения Windows не могут работать с блоками памяти, занимающими несколько сегментов. В этом случае для адресации вам нужно использовать специальные методы. Однако, если вы составляете приложение на языке программирования C или C++, при определении указателей на блоки памяти размером больше, чем 64 Кбайт, можно воспользоваться ключевым словом `huge`. Для таких указателей при

необходимости будет автоматически выполняться переключение на нужные селекторы.

Работа приложения Windows не должна зависеть от уровня привилегий, предоставленного приложению операционной системой, так как в новых версиях Windows этот уровень может измениться.

ПОРТЫ

ОСНОВНЫЕ ПОНЯТИЯ И ТЕРМИНЫ

Порт последовательной передачи данных называют еще портом RS-232-C, или асинхронным адаптером RS-232-C. Компьютер IBM PC поддерживает интерфейс RS-232-C не в полной мере, скорее разъем, обозначенный на корпусе компьютера как порт последовательной передачи данных, содержит некоторые из сигналов, входящих в интерфейс RS-232-C и имеющих соответствующие этому стандарту уровни напряжения.

В настоящее время порт последовательной передачи данных используется очень широко. Вот далеко не полный список применений:

подключение мыши;

подключение графопостроителей (плоттеров), сканеров, принтеров, диджитайзеров;

связь двух компьютеров через порты последовательной передачи данных с использованием специального кабеля и таких программ, как FastWire II или Norton Commander;

подключение модемов для передачи данных по телефонным линиям;

подключение к сети персональных компьютеров.

Практически каждый компьютер оборудован хотя бы одним портом для последовательной передачи данных.

Последовательная передача данных означает, что данные передаются с использованием единственной линии. При этом биты байта данных передается по очереди с использованием одного провода. Для синхронизации группе битов обычно предшествует специальный стартовый бит, после группы битов следуют бит проверки на четность и один или два стоповых бита. Иногда бит проверки на четность может отсутствовать.

Сказанное иллюстрируется следующим рисунком:



Рис. 11. Принцип последовательной передачи данных.

Исходное состояние линии последовательной передачи данных - уровень логической 1. Стартовый бит START сигнализирует о начале передачи данных. Далее передаются биты данных, вначале младшие, затем старшие. Если используется бит четности P, то передается и он. Бит четности имеет такое значение, чтобы в пакете битов общее количество единиц (или нулей) было четно или нечетно.

В самом конце передаются один или два стоповых бита STOP, завершающих передачу байта. Затем уровень линии передачи снова устанавливается в 1 до прихода следующего стартового бита.

Использование четности, стартовых и стоповых битов определяют протокол передачи данных. Очевидно, что передатчик и приемник данных должны использовать один и тот же протокол, иначе связь будет невозможной.

Другая важная характеристика - скорость передачи данных. Она также должна быть одинаковой для передатчика и приемника.

Скорость передачи данных обычно измеряется в бодах. Боды - это количество передаваемых битов в секунду. При этом учитываются и старт/стопные биты, а также бит четности.

Иногда используется другой термин - биты в секунду (bps). Здесь имеется в виду эффективная скорость передачи данных, без учета служебных битов.

АППАРАТНАЯ РЕАЛИЗАЦИЯ

Компьютер может быть оснащен одним или двумя портами последовательной передачи данных. Эти порты расположены либо на материнской плате, либо на отдельной плате, вставляемой в слоты расширения материнской платы.

Бывают также платы, содержащие 4 или 8 портов последовательной передачи данных. Их часто используют для подключения нескольких компьютеров или терминалов к одному, центральному, компьютеру.

В основе последовательного порта передачи данных лежит микросхема Intel 8250. Это универсальный асинхронный приемо-передатчик (UART - Universal Asynchronous Receiver Transmitter). Микросхема содержит несколько внутренних регистров, доступных через команды ввода/вывода.

Микросхема 8250 содержит регистры передатчика и приемника данных. При передаче байта он записывается в буферный регистр передатчика, откуда затем переписывается в сдвиговый регистр передатчика. Байт "выдвигается" из сдвигового регистра по битам.

Аналогично имеются сдвиговый и буферный регистры приемника.

Программа имеет доступ только к буферным регистрам, копирование информации в сдвиговые регистры и процесс сдвига выполняется микросхемой 8250 автоматически.

Внешние устройства подключаются к порту ввода/вывода через разъем DB25P (имеющий 25 выводов) или DB9P (имеющий 9 выводов). Приведем разводку разъема последовательной передачи данных DB25P:

Таблица 1. Разводка разъема DB25P.

Номер контакта	Назначение контакта	Вход или выход
1	Защитное заземление	-
2	Передаваемые данные (Transmitted Data)	Выход
3	Принимаемые данные (Received Data)	Вход
4	Запрос для передачи (Request to send, RTS)	Выход

5	Сброс для передачи (Clear to Send, CTS)	Вход
6	Готовность данных (Data Set Ready, DSR)	Вход
7	Сигнальное заземление	-
8	Детектор принимаемого с линии сигнала (Data Carrier Detect, DCD)	Вход
9-19	Не используются	
20	Готовность выходных данных (Data Terminal Ready, DTR)	Выход
21	Не используется	
22	Индикатор вызова (Ring Indicator, RI)	Вход
23-25	Не используется	

Наряду с 25-контактным разъемом часто используется 9-контактный разъем:

Таблица 1. Разводка разъема DB9.

Номер контакта	Назначение контакта	Вход или выход
1	Детектор принимаемого с линии сигнала (Data Carrier Detect, DCD)	Вход
2	Принимаемые данные (Received Data)	Вход
3	Передаваемые данные (Transmitted Data)	Выход
4	Готовность выходных данных (Data Terminal Ready, DTR)	Выход
5	Сигнальное заземление	-
6	Готовность данных (Data Set Ready, DSR)	Вход
7	Запрос для передачи (Request to send, RTS)	Выход
8	Сброс для передачи (Clear to Send, CTS)	Вход
9	Индикатор вызова (Ring Indicator, RI)	Вход

Уровни напряжения на линиях разъема составляют для логического нуля -15 вольт, для логической единицы - +15 вольт.

Доступ к отдельным линиям возможен через порты ввода/вывода асинхронного адаптера, которые мы рассмотрим в следующем разделе. Там же будет описано назначение отдельных линий разъема.

ПОРТЫ АСИНХРОННОГО АДАПТЕРА

На этапе инициализации системы модуль POST BIOS тестирует имеющиеся асинхронные адаптеры и инициализирует первые два. Их базовые адреса располагаются в области данных BIOS начиная с адреса 0000:0400h.

Первый адаптер COM1 имеет базовый адрес 3F8h и занимает диапазон адресов от 3F8h до 3FFh. Второй адаптер COM2 имеет базовый адрес 2F8h и занимает адреса 2F8h...2FFh.

Асинхронные адаптеры могут вырабатывать прерывания:

- COM1 - IRQ4 (соответствует INT 0Ch)
- COM2 - IRQ3 (соответствует INT 0Bh)

Рассмотрим назначение отдельных битов этих портов.

ПОРТ 3F8h

Этот порт соответствует регистру передаваемых данных. Для передачи в порт 3F8h необходимо записать передаваемый байт данных. После приема данных от внешнего устройства они могут быть прочитаны из этого порта.

В зависимости от состояния старшего бита управляющего слова, выводимого в управляющий регистр с адресом 3FBh, назначение порта 3F8h может изменяться. Если этот бит равен 0, порт используется для записи передаваемых данных. Если же бит равен 1, порт используется для вывода значения младшего байта делителя частоты тактового генератора. Изменяя содержимое делителя, можно изменять скорость передачи данных. Старший байт делителя записывается в порт 3F9h.

Зависимость скорости передачи данных от значения делителя частоты представлено в следующей таблице:

Таблица 3. Зависимость скорости передачи от значения делителя частоты.

Делитель	Скорость передачи в бодах
1040	110
768	150
384	300
192	600
96	1200

48	2400
24	4800
12	9600
6	19200
3	38400
2	57600
1	115200

ПОРТ 3F9h

Порт используется либо как регистр управления прерываниями от асинхронного адаптера либо (после вывода в порт 3F9h байта с установленным в 1 старшим битом) для вывода значения старшего байта делителя частоты тактового генератора.

В режиме регистра управления прерываниями порт имеет следующий формат:

Таблица 4. Описание регистра управления.

Номер бита	Значение
0	1 - разрешение прерывания при готовности принимаемых данных
1	1 - разрешение прерывания после передачи байта (когда выходной буфер передачи пуст)
2	1 - разрешение прерывания по обнаружению состояния "BREAK" или по ошибке
3	1 - разрешение прерывания по изменению состояния входных линий на разъеме RS-232-C (CTS, DSR, RI, DCD)
4 - 7	Не используются, должны быть равны 0.

ПОРТ 3FAh

Регистр идентификации прерывания. Считывая его содержимое, программа может определить причину прерывания.

Формат регистра:

Таблица 5. Формат регистра идентификации прерывания.

Номер бита	Значение
0	1 - Нет прерываний, ожидающих обслуживания
1 - 2	00 - Прерывание по линии состояния приемника, возникает при переполнении приемника, ошибках четности или формата данных или при состоянии "BREAK". Сбрасывается после чтения состояния линии из порта 3FDh. 01 - Данные приняты и доступны для чтения. Сбрасывается после чтения данных из порта 3F8h. 10 - Буфер передатчика пуст. Сбрасывается при записи новых данных в регистр данных передатчика, порт 3F8h. 11 - Состояние модема. Устанавливается при изменении состояния входных линий CTS, RI, DCD, DSR. Сбрасывается после чтения состояния модема из порта 3FEh.
3 - 7	Должно быть равно 0

ПОРТ 3FBh

Управляющий регистр, доступен по записи и чтению.

Таблица 6. Формат управляющего регистра.

Номер бита	Значение
0 - 1	Длина слова в битах: 00 - 5 бит; 01 - 6 бит; 10 - 7 бит; 11 - 8 бит.
2	Количество стоповых бит: 0 - 1 бит; 1 - 2 бита.
3 - 4	Четность: X0 - контроль на четность не используется; 01 - контроль на нечетность; 11 - контроль на четность.
5	Фиксация четности. При установке этого бита бит четности всегда принимает значение 0 (если биты 3-4 равны 11) или 1 (если биты 3-4 равны 01).
6	Установка перерыва. Вызывает вывод строки нулей в качестве сигнала "BREAK" для подключенного устройства.

7	1 - порты 3F8h и 3F9h используются для для загрузки делителя частоты тактового генератора; 0 - порты используются как обычно.
---	--

ПОРТ 3FCh

Регистр управления модемом. Управляет состоянием выходных линий DTR, RTS , линий, специфических для модемов OUT1 и OUT2, для запуска диагностики соединенных вместе замкнутых входе и выходе асинхронного адаптера.

Формат порта:

Таблица 7. Формат регистра управления модемом.

Номер бита	Значение
0	1 - линия DTR;
1	1 - линия RTS;
2	1 - линия OUT1 (запасная);
3	1 - линия OUT2 (запасная);
4	1 - запуск диагностики при входеасинхронного адаптера, замкнутом на его выход;
5 - 7	1 - должно быть равно 0.

ПОРТ 3FDh

Регистр состояния линии.

Таблица 8. Формат регистра состояния линии.

Номер бита	Значение
0	Данные получены и готовы для чтения, сбрасывается при чтении данных.
1	Ошибка переполнения. Был принят новый байт данных, а предыдущий еще не был считан программой. Предыдущий байт потерян.
2	Ошибка четности, сбрасывается после чтения состояния линии.
3	Ошибка синхронизации.

4	Обнаружен запрос на прерывание передачи "BREAK" - длинная строка нулей.
5	Регистр хранения передатчика пуст, в него можно записывать новый байт для передачи.
6	Регистр сдвига передатчика пуст. Этот регистр получает данные из регистра хранения и преобразует их в последовательный вид для передачи.
7	Таймаут (устройство не связано с компьютером).

ПОРТ 3FEh

Регистр состояния модема.

Таблица 9. Формат регистра состояния модема.

Номер бита	Значение
0	линия CTS изменила состояние
1	линия DSR изменила состояние
2	линия RI изменила состояние
3	линия DCD изменила состояние
4	состояние линии CTS
5	состояние линии DSR
6	состояние линии RI
7	состояние линии DCD

ПРОГРАММИРОВАНИЕ АСИНХРОННОГО АДАПТЕРА

К сожалению, MS-DOS не содержит сколько-нибудь серьезной поддержки асинхронного адаптера. Две функции прерывания INT 21h с номерами 3 и 4 предназначены для чтения и записи байтов через асинхронный адаптер. Обе эти функции имеют дело с адаптером COM1 или AUX. Функция 3 получает в регистре AL символ, принятый из адаптера, функция 4 посылает в адаптер символ, записанный в регистр DL.

Основной недостаток функций MS-DOS, предназначенных для работы с адаптером, заключается в отсутствии их функциональной полноты. Используя только функции MS-DOS, вы не сможете проанализировать ошибочные

ситуации и изменить режим работы асинхронного адаптера - нет соответствующих средств.

Функции BIOS, обслуживающие адаптер, более разнообразны. Однако и им присущи недостатки. Например, вы не сможете установить скорость передачи более 9600 бод или использовать режим фиксации четности. Нет возможности узнать текущий режим асинхронного адаптера, отсутствует поддержка модема. Поэтому для программирования асинхронного адаптера мы рекомендуем использовать порты ввода/вывода микросхемы 8250.

ИНИЦИАЛИЗАЦИЯ АСИНХРОННОГО АДАПТЕРА

Первое, что должна сделать программа, работающая с асинхронным адаптером - установить протокол обмена и скорость передачи данных. После загрузки операционной системы для асинхронных адаптеров устанавливается скорость 2400 бод, не выполняется проверка на четность, используются один стоповый бит и восьмибитовая длина передаваемого символа. Вы можете изменить этот режим командой MS-DOS MODE.

Выполнив ввод из порта 3FBh, программа может получить текущий режим адаптера. Для установки нового режима измените нужные вам поля и запишите новый байт режима по адресу 3FBh.

Если вам надо задать новое значение скорости обмена данными, перед записью байта режима установите старший бит этого байта в 1. Затем последовательно двумя командами вывода загрузите делитель частоты тактового генератора. Младший байт запишите в порт 3F8h, старший - в порт 3F9h.

Перед началом работы необходимо также проинициализировать регистр управления прерываниями (порт 3F9h), даже если в вашей программе не используются прерывания от асинхронного адаптера. Если прерывания вам не нужны, запишите в этот порт значение 0.

На этом инициализацию можно считать законченной.

ПЕРЕДАЧА ДАННЫХ

Перед записью байта данных в регистр передатчика необходимо убедиться в том, что регистр хранения передатчика свободен, то есть убедиться в том, что передача предыдущего символа завершена.

Признаком того, что регистр передатчика свободен, является установленный в 1 бит 5 регистра состояния линии с адресом 3FDh.

ПРИЕМ ДАННЫХ

Аналогично передаче данных, перед вводом символа из порта приемника 3F8h необходимо убедиться в том, что бит 0 порта 3FDh установлен в 1. Это означает, что символ принят из линии и находится в буферном регистре приемника.

ИСПОЛЬЗОВАНИЕ ПРЕРЫВАНИЙ

Так как процесс последовательной передачи данных протекает достаточно медленно, имеет смысл выполнять его в фоновом режиме, используя прерывания по окончании передачи или приема символа. Напомним, что порту COM1 соответствует аппаратное прерывание INT 0Ch, а COM2 - INT 0Bh.

Для разрешения прерываний необходимо установить в 1 биты порта управления прерываниями 3F9h, соответствующие тем прерываниям, которые мы желаем обрабатывать.

Когда произошло прерывание, программа-обработчик прерывания должна проанализировать причину прерывания, прочитав содержимое порта идентификации прерывания с адресом 3FAh.

Не забудьте, что в конце обработчика аппаратного прерывания должна находиться последовательность команд:

```
mov al, 20h  
out 20h, al  
iret
```

Может случиться так, что одновременно произойдет несколько прерываний. В этом случае бит 0 регистра идентификации прерывания будет установлен в 1. Если такая ситуация имеет место, перед завершением обработки прерывания вам надо снова прочитать регистр идентификации прерывания и обработать следующее прерывание. Так следует поступать до тех пор, пока бит 0 регистра идентификации прерывания не станет равным нулю.

ПАМЯТЬ

Оперативная память ПЭВМ (RAM) физически представляет собой набор модулей динамической памяти (DRAM), подключаемый к общей шине (в старых системах) или к специальному контроллеру памяти. Логически память является одномерной матрицей элементов (байтов), в которой хранятся данные и исполняемый код программы.

ЛОГИЧЕСКАЯ ОРГАНИЗАЦИЯ ПАМЯТИ

Правила работы с памятью в защищенном режиме достаточно сложны и рассматриваются отдельно. Для реального режима работы процессора доступ к памяти осуществляется с помощью двух адресов - сегмента, содержащего старшие 16 разрядов 20-ти разрядного физического адреса (обусловлено разрядностью шины 16-битного процессора) и смещения, 12 старших бит которого добавляется к сегменту, а 4 бита составляют младшие разряды физического адреса. Если при добавлении старших 12-ти бит смещения сегмент переполняется, то сегмент должен обнуляться, что, однако, не происходит в процессорах 80286 и старше (см. НМА).

Так как первой системой для ПЭВМ была MS-DOS фирмы Microsoft, то и разделение памяти на области в реальном режиме связано со структурой этой системы.

Области памяти при работе DOS показаны на рисунке:

640 Кб	384 Кб	16 Кб	остальное
conventional MEM	upper MEM	НМА	extended MEM
0	640	1024	1088

Рис. 12. Организация памяти в DOS

Первые 640 Кбайт памяти при загрузке DOS распределяются следующим образом:

1024 б	256 б	512 б	638 Кб
таблица векторов	BDA	DDA	драйвера DOS
0	0040h	0050h	0070h

Рис. 13. Структура первых 640 Кбайт в DOS.

UPPER MEMORY

Верхняя память первоначально предназначалась для загрузки системы DOS (первой версии). Впоследствии в этот диапазон адресов подключались ПЗУ и память внешних устройств и ROM BIOS. Схема распределения этого диапазона:

64 Kb	32 Kb	32 Kb	64 Kb	128 Kb	64 Kb
graphics vram	mono vram	color vram	video bios	UMB	bios setup

Рис. 14. Структура верхней памяти в DOS.

HMA

Область памяти в диапазоне адресов 10000h:0010h..10000h:FFFFh. Она доступна в реальном режиме процессора 80286 и старше ввиду того, что при переполнении сегментного адреса (при вычислении физического адреса старшие 12 бит смещения добавляются к сегменту) не происходит его обнуления и обращение идет к адресам за пределами пространства в 1024Кбайт. Эта область становится доступной при загрузке специального драйвера, деблокирующего адресную линию A20 (т.е. 21-ый разряд адресной шины - у 8086/8088 только 20 адресных линий). В MS-DOS это драйвер HIMEM.SYS. После этого можно загружать программы в эту область, используя прерывание Int 2Fh (мультиплексное), функции 4Axxh (недокументированны). Впрочем, DOS чаще использует эту область для загрузки туда своей части (драйверы, буферы, стек, обработчики функций и т.д.).

EXTENDED MEMORY

К расширенной памяти относят всю остальную память за пределами первого мегабайта. Доступ к ней в реальном режиме невозможен без специальных средств (впрочем, есть исключение). Одним из них является интерфейс XMS - eXtended Memory Specification. Он позволяет выделять, копировать, удалять блоки расширенной памяти с помощью функций прерывания Int 2Fh 43xxh. Альтернативой XMS являются интерфейсы DPMI (DOS Protected Mode Interface) для защищенного режима и VCPI для виртуального.

EXPANDED MEMORY

Даже на заре эры ПЭВМ объема памяти в 640 Кбайт не всегда хватало, поэтому возник интерфейс EMS - Expanded Memory Specification. Этот интерфейс предполагал наличие отдельного устройства - платы с памятью, которая подключалась к доступным адресам верхней памяти - обычно начиная с адресов D000h - E800h. В этом диапазоне организовывалась область размером

64 Кбайта, разделяемая на 4 окна (фрейма) по 16 Кбайт. В эти окна, используя вызовы драйвера EMS, программист мог “видеть” любые страницы памяти, размещенной на плате. Однако с появлением 32-разрядных процессоров использование дополнительных контроллеров стало нецелесообразным, т.к. это значительно замедляло систему (работа памяти по общей шине ISA) и повышало ее стоимость. Поэтому от внешней памяти быстро отказались, но принципы, заложенные в основу интерфейса EMS, оказались настолько удачны, что драйвера EMS используются и по сей день для организации доступа к памяти за пределами первого мегабайта. Обычно эмуляцию EMS осуществляют на базе XMS - интерфейса, разделяя общую память на две части по мере поступления запросов на блоки памяти.

UPPER MEMORY BLOCKS

Свободная память в пределах 640-1024 Кбайт используют не только для организации окон EMS, но и для логической переадресации туда страниц памяти. Приблизительно это выглядит так:



Рис. 15. Организация окон EMS.

В организованные области, наполненные памятью, загружаются резидентные программы и драйверы DOS. Для работы этого механизма нужен драйвер XMS - EMM386 или подобный ему, иначе области памяти окажутся “пустыми”, т.е. без реальной памяти, соответствующей адресам.

ФИЗИЧЕСКАЯ ОРГАНИЗАЦИЯ ПАМЯТИ

Память ПЭВМ основана на модулях DRAM, т.е. динамической памяти. Схема работы такого модуля представлена на рисунке:

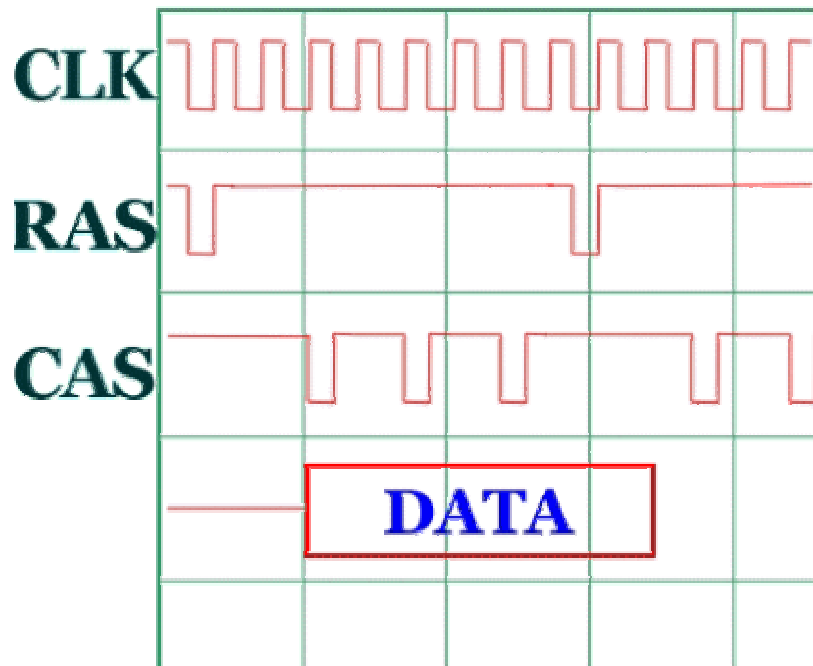


Рис. 16. Временная диаграмма работы модуля DRAM.

Так как сигналы RAS (номер строки памяти) и CAS (номер столбца) подаются на те же входы микросхемы, между ними должна быть некоторая задержка (2-3 такта). В пакетном режиме (как на рисунке) сигнал RAS подавать все время не нужно, т.к. считывание происходит с той же строки матрицы памяти. Если память синхронизирована с шиной процессора (память SDRAM), то данные будут поступать без задержек - по одному блоку (64 бита) за такт или по два блока (память DDR SDRAM), иначе между выдачей данных на шину будет задержка в 1-2 такта (память EDO или FPM).

Динамическая память работает достаточно медленно за счет необходимости задержек на регенерацию. Статическая память не требует этих задержек, но является более дорогой, громоздкой и потребляет больше энергии. Выход - использование кэширования.

КЭШИРОВАНИЕ ПАМЯТИ

В основе механизма кэширования лежит принцип использования дополнительных модулей статической памяти для хранения промежуточных данных при чтении и записи их в память. Механизм кэширования достаточно сложен, но упрощенно он выглядит так. При запросе процессора на чтение определенного блока данных по определенному адресу контроллер памяти проверяет, не был ли этот блок прочитан ранее. Все читаемые данные попадают сначала в кэш, поэтому запрос на чтение можно выполнить без обращения к основной динамической памяти, взяв их из кэша (статической памяти). Скорость работы кэша почти в 2 раза превышает скорость работы обычной

памяти на той же тактовой частоте (впрочем, память SDRAM в пакетном режиме дает не такую большую разницу). После обработки данные попадают тоже в кэш, откуда могут быть взяты при следующих запросах на чтение.

Кэш-память может находиться как в составе ядра процессора (см. 32-битный процессор), так и на плате совместно с обычной DRAM.

Механизм кэширования широко используется начиная с процессора 80386. С развитием ПЭВМ ему отводится все большая роль в повышении производительности подсистемы “Процессор - память”.

БАНКИ ПАМЯТИ

Еще одним способом обхода задержек на регенерацию динамической памяти является разбиение памяти на банки. Для этого всю память разделяют на несколько (2 или 4) аппаратно независимых частей (обычно - разные блоки памяти или микросхемы). При этом если четные слова памяти размещаются в одном банке, то нечетные - в другом. При последовательном чтении / записи слов банк регенерируется в то время, когда доступ осуществляется ко второму банку. После чего банки меняются местами. При этом чтение идет непрерывно, т.к. банк, к которому происходит доступ, уже готов, будучи регенерирован до этого во время работы с другим банком. В настоящее время разбиение на банки применяется повсеместно и практически все модули памяти работают в этом режиме.

ТЕНЕВАЯ ПАМЯТЬ

Теневая память (shadow RAM) - еще одно средство повышения производительности ПЭВМ. Известно, что микросхема ПЗУ, хранящая BIOS, Setup, POST и другие важные системные программы, подключается к общей системе через общую шину ISA (см. Общая архитектура ПЭВМ). Поэтому доступ к ней намного медленнее, чем к обычной памяти. Чтобы преодолеть этот недостаток, содержимое ПЗУ копируется в обычную память на этапе начальной загрузки ПЭВМ и остается там все время работы системы. Обращение по адресам 0F000h - 0FFFFh производится не к ПЗУ, а к обычной динамической памяти, и является столь же быстрым. Возможно также и кэширование этих адресов, однако в данном случае оно не играет большой роли, т.к. в ПЗУ хранятся в основном обработчики прерываний BIOS.

ФАЙЛОВАЯ СИСТЕМА

ОБЩИЕ СВЕДЕНИЯ

Дисковая подсистема современного персонального компьютера является одной из самых значимых для дальнейшего увеличения быстродействия и удобства работы. Рост производительности жестких дисков является основной задачей фирм-производителей систем массовой памяти; появляются новые устройства, позволяющие использовать другие, альтернативные технологии для хранения больших массивов данных. С каждым годом все больше информации может обрабатывать персональный компьютер, и во многом это обеспечивается непрерывным развитием системы дисковой памяти.

Хранение больших объемов информации осуществляется на двух типах устройств - сменных накопителях (Removable Media) и фиксированных дисках (Fixed Media). К первым относят гибкие диски, магнитооптические диски, лазерные компакт-диски (не записываемые (CD-ROM), однократно записываемые (CD-R) и перезаписываемые (CD-RW)) и кассеты с лентой; эти устройства не отличаются большой плотностью записи, надежностью и скоростью, впрочем, это не касается магнитооптики и компакт-дисков. Ко вторым относятся жесткие диски. Они не предназначены для переноса данных от компьютера к компьютеру, так как устанавливаются внутри корпуса. Жесткий диск объединен с устройством чтения-записи, сменные накопители могут вставляться в любое устройство чтения-записи, способное их обрабатывать.

FDD

Гибкие диски были первой системой дисковой памяти на персональном компьютере. Дисковая подсистема первого IBM PC состояла только из них. Накопитель на гибких дисках состоит из контроллера, соединительного шлейфа (36 линий связи), устройства чтения-записи (дисковод) и гибких дисков

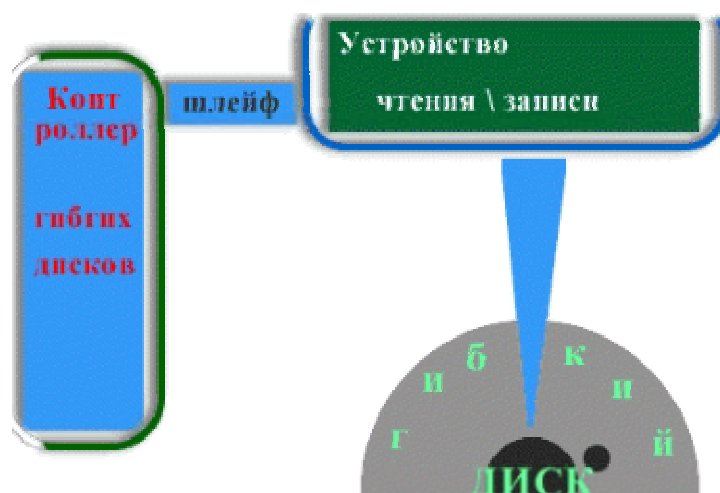


Рис. 17. Дисковая подсистема ПЭВМ.

Гибкий диск представляет собой пластину из гибкого материала (полимера) и нанесенное на нее магнитное покрытие. Покрытие наносится с двух сторон диска (раньше было с одной стороны). Диск находится в защитном конверте. В центре диска есть отверстие со специальным кольцом (металлическим у дисков 3.5 дюйма), за которое привод мотора дисководов захватывает диск.

Как выполняется чтение

Две головки чтения-записи (с обеих сторон диска) прижимаются к поверхности при поступлении диска в дисковод. Для выполнения операций записи и чтения шаговый двигатель перемещает головки вдоль радиуса диска. Это связано с тем, что вся поверхность диска разбивается на концентрические кольца - дорожки (цилиндры, треки). Позиционирование головок выполняется на тот трек, данные с которого требуется прочитать. Каждый трек делится на несколько секторов. Позиционирование головок на сектор, конечно, не производится. Для чтения сектора контроллер ищет начало дорожки (запись определенного формата, содержит биты со смещенной синхронизацией), после чего просматривается каждый найденный сектор, который содержит информацию о своем номере и позиции на диске. После выполнения считывания сектора данные (512 байт) проверяются на корректность методом расчета контрольной суммы (CRC) и сверки полученной суммы с контрольной в самом секторе. Если правильность не подтверждается, проводится повторное чтение несколько раз. В случае успеха данные поступают в оперативную память, используя метод DMA.

Типы дисков

В современных персональных компьютерах используется только один тип гибких дисков - диск размером 3.5 дюйма с плотностью 1.44 Мбайт (135 дорожек на дюйм). Однако раньше использовались следующие типы дисков:

Таблица 10. Типы НГМД.

Размер	Емкость	Число дорожек	Число секторов	Плотность, ТРІ
5.25"	360К	40	9	48
5.25"	720К	80	9	96
5.25"	1.2М	80	15	96
3.5"	720К	80	9	135
3.5"	1.44М	80	18	135
3.5"	2.88М	80	36	135

Кроме того, раньше использовались односторонние диски, диски размером 8 дюймов, а также диски малой плотности (160, 320 Кбайт).

Уже существует новый стандарт гибких дисков - LS-120, а также стандарт A:DRIVE, но эти стандарты не являются распространенными и общепринятыми, хотя получили поддержку производителей компьютерных систем.

Для выполнения работы с гибкими дисками требуется проверить наличие и тип дисководов.

Параметры дискового устройства

Существует специальная таблица - Disk Parameter Table. Она содержит специфические низкоуровневые параметры дисководов, формируется BIOS при инициализации системы. На эту таблицу указывает вектор прерывания 1Eh (адрес находится по адресу 0000h:0078h). Таблица имеет размер 11 байт.

Таблица 11. Таблица параметров диска.

Смещение	Описание
0	Биты 0...3 - SRT (Step Rate Time) - задержка для переключения головок, лежит в пределах 1-16 мс и задается с интервалом 1 мс (0Fh - 1мс, 0Eh - 2 мс, 0Dh - 3 мс, ...); биты 4...7 - задержка разгрузки головки, лежит в пределах 16-240 мс и задается с интервалом 16 мс (1 - 16 мс, 2 - 32 мс, ..., 0Fh - 240 мс).
1	Бит 0 - значение этого бита, равное 1, говорит о том, что используется прямой доступ к памяти (DMA); биты 2...7 - время загрузки головок HLT - интервал между сигналом загрузки головок и началом операции чтение/запись, лежит в пределах 2-254 мс и задается с интервалом 2 мс (1 - 2 мс, 2 - 4 мс, ..., 0FFh - 254 мс).
2	Задержка перед выключением двигателя.
3	Код размера сектора в байтах (0 - 128 байтов, 1 - 256, 2 - 512, 3 - 1024).
4	Номер последнего сектора на дорожке
5	Длина межсекторного промежутка для чтения/записи.
6	Максимальная длина передаваемых данных, используется когда не задана длина сектора.
7	Длина межсекторного промежутка для операции форматирования.
8	Байт-заполнитель для форматирования (обычно используется

	F6h).
9	Время установки головки в миллисекундах.
10	Время запуска двигателя в 1/8 долях секунды.

Использовать эту таблицу можно при выполнении операций чтения-записи через порты контроллера дисководов.

HDD

В современных компьютерах устройства для записи данных на магнитные диски, устанавливаемые стационарно в корпусе и жестко объединенные с самими дисками-носителями информации являются основным средством долговременного хранения информации. В настоящее время работа жестких дисков осуществляется с использованием интерфейсов АТА (старое название - IDE) и SCSI. Другие интерфейсы остались далеко в прошлом. Единственный недостаток жестких дисков - отсутствие возможности смены диска - компенсируется большой плотностью записи и производительностью.

История

Устройство для хранения информации на основе вращающегося магнитного диска было разработано фирмой IBM в самом начале 70-х годов. Этот громоздкий 14-дюймовый диск позволял записать 30 дорожек по 30 секторов в каждой из них; обозначение диска "30/30" напоминало название широко распространенной модели ружья фирмы "Winchester", в результате чего в английском языке для обозначения дисковых устройств с несъемными дисками стали широко применять слово "winchester". В 1973 году фирма IBM создала первый накопитель с несколькими дисками емкостью 140 Мб, который продавался по цене \$8600.

При разработке первой модели компьютера IBM PC в нем не был предусмотрен встроенный винчестер, однако уже в следующей модели он был установлен (20 Мб). В компьютере PC AT винчестер уже являлся основным средством повышения эффективности компьютера - пользователям не требовалось больше загружать операционную систему с дискет и искать каждый раз новую дискету при начале работы с какой-либо программой. Возможность хранения на встроенном диске больших объемов данных и программ значительно расширила диапазон использования персональных компьютеров.

В 1988 был создан Комитет по стандартизации методов доступа (Common Access Method Committee) для разработки программного интерфейса устройств SCSI. Одним из результатов деятельности этого комитета явилась разработка спецификации интерфейса АТА (AT Attachment - присоединение к шине АТ), известного также под именем IDE (Integrated Drive Electronics). Новый стандарт произвел настоящий бум в индустрии дисковых накопителей, когда производители PC получили возможность использования этих недорогих устройств. С конца 80-х годов интерфейс АТА (такое название более

правильно, чем IDE) стал самым распространенным среди недорогих настольных систем.

Устройство жесткого диска

В самом первом магнитном накопителе, разработанном фирмой IBM, диски и головки вместе с несущей конструкцией размещались в отдельном закрытом корпусе (его называли модулем данных), устанавливаемом для работы на приводное устройство. При установке модуля данных на привод автоматически подключалась система подачи в модуль данных очищенного воздуха. Головки, благодаря малой массе, прижимались к поверхности диска с усилием всего 0.1Н, а при вращении диска между головкой и поверхностью образовывался воздушный зазор толщиной около 0.5мкм. Прослойка воздуха между головкой и диском

создавалась за счет потоков, образующихся при вращении диска, и обеспечивала возможность многократной записи и считывания данных без повреждения магнитной поверхности диска.

В современных устройствах модуль данных и привод составляют единое целое и система подачи очищенного воздуха уже не используется. Каждый современный накопитель содержит пакет магнитных дисков, установленных на одной оси. В первых устройствах использовалась скорость вращения 3600 об/мин, однако по мере роста требований к скорости записи/считывания частота вращения блока дисков была повышена во многих устройствах до 7200 об/мин. Повышение скорости вращения обеспечивает возможность ускорения работы всего устройства, однако рост скорости ограничен механической прочностью дисков.

Диски представляют собой пластины из алюминия, стекла или керамики с нанесенным на них слоем высококачественного ферромагнетика. Состав магнитного покрытия достаточно сложен - оно, как правило, наносится путем напыления или вакуумного осаждения. В первых дисках использовалось покрытие из оксида железа, сегодня в качестве материалов для магнитного покрытия используются как материалы на основе железа и его окислов, так и пленки других магнитных металлов. Покрытия на основе окислов железа и бариевых ферритов являются достаточно мягкими, поэтому их использование в новых разработках почти прекратилось. Металлические пленочные покрытия обеспечивают более высокую плотность записи и прочность поверхности диска. Прочность покрытия особенно важна при использовании дисков в переносных компьютерах, где велика вероятность ударов.

После нанесения покрытия диски подвергаются специальной обработке для обеспечения высококачественной поверхности. Обработанные диски собирают в один пакет (обычно в пакете содержится от 2 до 12 дисков) и закрепляют на оси, устанавливаемой в привод. Каждый диск имеет две рабочих поверхности, однако в некоторых устройствах внешние поверхности крайних дисков пакета не используются из конструктивных соображений.

Для надежной и качественной работы винчестера важно обеспечить отсутствие пыли в корпусе блока дисков и головок, для чего широко используются барометрические фильтры, выравнивающие давление внутри и снаружи блока

дисков. Если вы хотите, чтобы ваш винчестер работал долго и обеспечивал высокую надежность хранения данных, никогда не открывайте корпус блока дисков и не срываете с него защитных наклеек.

Головки чтения-записи относятся к числу важнейших элементов дискового накопителя. Принцип действия головок винчестера похож на принцип работы головок обычного магнитофона, однако требования к ним предъявляются значительно более жесткие по сравнению с магнитофонными головками. Отличаются головки дисковых накопителей и своими малыми размерами. Головка всегда находится на некотором расстоянии от поверхности диска (около 0.13мкм), обеспечиваемом за счет потока воздуха при быстром вращении диска.

Устройство привода магнитных головок (head positioner) является одной из важнейших частей винчестера. От типа используемого привода непосредственно зависит скорость работы устройства в целом - привод обеспечивает важнейший параметр винчестера: время позиционирования головок (seek time). Для перемещения головок обычно используются шаговые двигатели, обеспечивающие высокую точность позиционирования. Существуют два различных варианта приводов: линейные и поворотные. При поворотном приводе головки перемещаются по дуге окружности как в обычном электропроигрывателе, линейный привод обеспечивает перемещение головок по радиусу диска. Преимущество линейного привода заключается в том, что зазор магнитной головки всегда перпендикулярен дорожке и расстояние между дорожками сохраняется постоянным, поворотные приводы обеспечивают меньшую инерционность и, как следствие, более быстрое позиционирование головок. Кроме того, поворотные приводы более устойчивы к ударам и вибрации, поскольку допускают точную балансировку.

На каждом винчестере кроме блока дисков и привода установлена печатная плата (как правило она крепится снизу), обеспечивающая управление приводами головок и дисков, а также усиление сигналов записи/считывания. Кроме того, на этой плате установлен дешифратор команд управления головками, схемы стабилизации и др. На современных винчестерах, изготавливаемых в рамках программы Energy Star, имеется также устройство, обеспечивающее отключение привода дисков при отсутствии запросов к устройству и другие функции энергосбережения.

Современные дисковые устройства выпускаются четырех типоразмеров по ширине (диаметру дисков) и трех - по высоте. Диаметр дисков в большинстве случаев равен 1.8, 2.5, 3.5 или 5.25 дюйма, высота - 3.25 (устройство полной высоты), 1.63 (устройство половинной высоты) или менее 1 дюйма (низкопрофильное устройство).

ИНТЕРФЕЙС АТА

Предложенный в конце 80-х годов интерфейс АТА (AT Attachment), известный также под названием IDE (Integrated Drive Electronics) очень быстро завоевал

широкую популярность среди производителей и пользователей персональных компьютеров. Основными причинами быстрого и широкого распространения нового интерфейса послужили невысокая цена устройств, простота их установки и эксплуатации, а также высокий уровень совместимости устройств IDE. Спецификация ATA уступает по скорости обмена с дисками и ряду других параметров интерфейсу SCSI, однако для большинства пользователей производительность играет меньшую роль, по сравнению с совместимостью, простотой и стоимостью устройств.

Ограниченность ATA

Первоначальная версия стандарта IDE обеспечивала возможность подключения к компьютеру четырех винчестеров и позволяла обмениваться данными с диском на скорости до 10Мбайт/сек, однако реальная скорость ограничивалась прежде всего возможностями самого винчестера. Подключение четырех устройств, предусмотренное спецификацией IDE, в компьютерах семейства IBM PC AT, в спецификации ATA/IDE реализовано не было. Кроме того, совместное использование стандарта ATA и программного интерфейса Int 13 BIOS ограничивало размер дисковых устройств 528Мб. Природу этого ограничения, связанную с различиями форматами чисел, используемых для адресации головок, секторов и цилиндров винчестера (CHS-адрес, т.е. Cylinder-Head-Sector) в стандарте IDE и BIOS, можно понять из приведенной ниже таблицы.

Таблица 12. Ограничительные параметры дисков.

	BIOS	IDE	Результат
Максимальное число секторов на дорожке	63	255	63
Максимальное число головок	255	16	16
Максимальное число цилиндров	1024	65535	1024
Максимальный размер диска	8.4Г	136.9Г	528М

Другим существенным ограничением стандарта IDE/ATA является невозможность подключения к контроллеру IDE каких-либо устройств, за исключением винчестеров.

Присущие интерфейсу IDE/ATA ограничения тормозили развитие недорогих компьютерных систем, рассчитанных на массового пользователя, в связи с чем целый ряд фирм предпринял попытки расширения возможностей классического IDE. Наибольших успехов на этом пути добилась компания Western Digital, разработавшая спецификацию Enhanced IDE (EIDE), позволяющую использовать диски, размер которых превышает 528 Мб, и обеспечившую

реальную возможность подключения к компьютеру до четырех устройств IDE (не только винчестеров, но и приводов CD-ROM или стриммеров).

Другим интересным вариантом развития интерфейса IDE является технология Fast ATA, разработанная компанией Seagate Technology при поддержке фирмы Quantum. Эта технология направлена в первую очередь на повышение скорости обмена данными с дисковыми устройствами и не поддерживает возможность подключения CD-ROM или стриммеров, но обеспечивает более высокий уровень совместимости за счет соответствия широко распространенным промышленным стандартам.

Enhanced IDE

Технология Enhanced IDE компании Western Digital была разработана в результате анализа потребностей современного рынка ПК. В 1984 году компания Western Digital создала контроллеры дисководов (WD1002) и винчестеров (ST506), которые были использованы фирмой IBM при разработке компьютера IBM PC AT. Успех архитектуры AT привел к значительному расширению рынка IBM-совместимых ПК и сделал контроллеры Western Digital стандартом de facto.

В процессе становления рынка персональных компьютеров фирма Western Digital пришла к выводу о необходимости интеграции электроники контроллера AT и дискового устройства. В результате сотрудничества с Compaq Computer Corporation был разработан интерфейс IDE (Integrated Drive Electronics), называемый также ATA (AT Bus attachment). Первые промышленные устройства на базе IDE/ATA были выпущены в 1986 году.

Продолжая лидировать в сфере IDE-устройств, компания Western Digital предложила расширение интерфейса IDE. Новая спецификация - Enhanced IDE - повышает скорость обмена с диском, допускает применение более скоростных дисков и обеспечивает возможность установки в компьютере до четырех устройств IDE. Кроме того, Enhanced IDE позволяет подключать к контроллеру не только винчестеры, но и другие устройства (приводы CD-ROM, стриммеры), поддерживающие спецификацию ATAPI (AT Attachment Packet Interface). Эта спецификация определяет интерфейс с приводами CD-ROM и другими недисковыми устройствами, позволяющий использовать стандартные контроллеры и кабели. Стандарт ATAPI получил широкую поддержку среди производителей CD-ROM-устройств и разработчиков операционных систем, что еще более расширило сферу распространения интерфейса IDE/ATA.

Спецификация EIDE позволяет избавиться от целого ряда ограничений, присущих интерфейсу IDE/ATA. Такое расширение обеспечивает существенный рост возможностей устройств долговременного хранения информации без роста цен, усложнения использования и потери совместимости. Ограничения, присущие IDE, по сравнению с другими интерфейсами (такими, как SCSI) не угрожают доминированию IDE на современном рынке недорогих систем. Однако повышение производительности процессоров, разработка новых ОС и высокие запросы современных

приложений к дисковому пространству привели к тому, что стандарт IDE уже не может удовлетворять всем современным требованиям. Существенно и то, что стандартный интерфейс IDE менее гибок и более ограничен по сравнению со SCSI. Технология Enhanced IDE позволяет избавиться от основных ограничений стандарта IDE/ATA:

- предельный размер диска 528Мб
- малая скорость обмена с диском
- поддержка только двух устройств

невозможность подключения приводов CD-ROM и стриммеров

Технология Enhanced IDE включает в себя четыре основных компоненты, реализуемые совместно в периферийных устройствах, контроллерах EIDE, BIOS и операционной системе:

- использование винчестеров размером более 528Мб

Эта возможность обусловлена внесенными в BIOS изменениями, позволяющими адресовать большие диски. Режим, называемый LBA (Logical Block Access), позволяет адресоваться по схеме CHS для дисков с числом цилиндров более допустимой (1024 - при адресации через Int 13h используется только 10 бит для номера цилиндра) путем трансляции вызовов, т.е. создается виртуальная конфигурация диска с большим числом головок (до 255) и меньшим числом цилиндров. Для работы с дисками более 528Мб требуется поддержка со стороны винчестера, BIOS и операционной системы. Винчестер должен поддерживать спецификацию ATA (этому требованию удовлетворяет большинство современных винчестеров). BIOS также должна поддерживать спецификацию ATA; большинство производителей системных плат и разработчиков BIOS давно поддерживают этот режим.

Ускоренный обмен данными

Основан на использовании скоростных режимов программируемого ввода-вывода (PIO mode 3) и прямого доступа к памяти (DMA mode 1). Эти режимы обеспечивают скорость обмена 11.1 и 13.3 Мб/сек, соответственно. Для реализации скоростного обмена требуется поддержка со стороны винчестера и BIOS. Скоростные режимы обмена позволяют полнее реализовать возможности дисковых контроллеров, использующих широкополосную локальную шину VESA или PCI.

Поддержка двух портов ATA

Для обеспечения этой возможности используются зарезервированные для второго контроллера адреса портов ввода-вывода и прерывание IRQ15. Для реализации этой возможности требуется поддержка со стороны контроллера (хост-адаптера), BIOS и операционной системы.

Возможность подключения приводов CD-ROM и стриммеров

Этот режим является недорогой альтернативой SCSI. Для обеспечения такой возможности требуется поддержка нескольких новых команд. Спецификация ATAPI (ATA Packet Interface), находящаяся в стадии разработки, содержит описание новых команд.

Для обеспечения возможности подключения к хост-адаптеру недисковых устройств требуется поддержка спецификации АТАPI со стороны BIOS, хост-адаптера, операционной системы и самих периферийных устройств. В настоящее время большинство устройств CD-ROM уже использует интерфейс IDE/АТАPI для подключения ко второму порту современных контроллеров.

КЛАВИАТУРА

УСТРОЙСТВО

Клавиатура выполнена, как правило, в виде отдельного устройства, подключаемого к компьютеру тонким кабелем. Малогабаритные компьютеры Notebook используют встроенную клавиатуру.

Что же находится внутри клавиатуры? Оказывается, там есть компьютер! Только этот компьютер состоит из одной микросхемы и выполняет специализированные функции. Он отслеживает нажатия на клавиши и посылает номер нажатой клавиши в центральный компьютер.

Если рассмотреть сильно упрощенную принципиальную схему клавиатуры, можно заметить, что все клавиши находятся в узлах матрицы. Все горизонтальные линии матрицы подключены через резисторы к источнику питания +5 В. Клавиатурный компьютер имеет два порта - выходной и входной. Входной порт подключен к горизонтальным линиям матрицы, а выходной - к вертикальным. Устанавливая по очереди на каждой из вертикальных линий уровень напряжения, соответствующий логическому 0, клавиатурный компьютер опрашивает состояние горизонтальных линий. Если ни одна клавиша не нажата, уровень напряжения на всех горизонтальных линиях соответствует логической 1 (т.к. все эти линии подключены к источнику питания +5 В через резисторы). Если оператор нажмет на какую-либо клавишу, то соответствующая вертикальная и горизонтальная линии окажутся замкнутыми. Когда на этой вертикальной линии процессор установит значение логического 0, то уровень напряжения на горизонтальной линии также будет соответствовать логическому 0. Как только на одной из горизонтальных линий появится уровень логического 0, клавиатурный процессор фиксирует нажатие на клавишу. Он посылает в центральный компьютер запрос на прерывание и номер клавиши в матрице. Аналогичные действия выполняются и тогда, когда оператор отпускает нажатую ранее клавишу.

Номер клавиши, посылаемый клавиатурным процессором, однозначно связан с распайкой клавиатурной матрицы и не зависит напрямую от обозначений, нанесенных на поверхность клавиш. Этот номер называется скан-кодом (Scan Code). Слово scan ("сканирование") подчеркивает тот факт, что клавиатурный компьютер сканирует клавиатуру для поиска нажатой клавиши.

Но программе нужен не порядковый номер нажатой клавиши, а соответствующий обозначению на этой клавише ASCII-код. Этот код не зависит однозначно от скан-кода, т.к. одной и той же клавише могут соответствовать несколько значений ASCII-кода. Это зависит от состояния других клавиш. Например, клавиша с обозначением '1' используется еще и для ввода символа '!' (если она нажата вместе с клавишей SHIFT). Поэтому все преобразования скан-кода в ASCII-код выполняются программным обеспечением. Как правило, эти преобразования выполняют модули BIOS. Для использования символов кириллицы эти модули расширяются клавиатурными драйверами.

Если нажать на клавишу и не отпускать ее, клавиатура перейдет в режим автоповтора. В этом режиме в центральный компьютер автоматически через некоторый период времени, называемый периодом автоповтора, посылаются код нажатой клавиши. Режим автоповтора облегчает ввод с клавиатуры большого количества одинаковых символов. Фактически все современные персональные компьютеры позволяют настроить как скорость автоповтора, так и период удержания клавиши до начала включения этого режима.

Следует отметить, что клавиатура содержит внутренний 16-байтовый буфер, через который она осуществляет обмен данными с компьютером.

В настоящее время существует три различных типа клавиатуры. Это клавиатура для компьютеров IBM PC/XT, 84-клавишная клавиатура для IBM AT и 101-клавишная (расширенная) клавиатура для IBM AT. Используется только последняя разновидность. С выходом т.н. операционной системы Windows95® на клавиатурах появились две новые клавиши, которые удобно использовать для работы в этой среде. Также наметилась тенденция добавлять еще одну клавишу неопределенного назначения (чаще всего эта клавиша позволяет устанавливать скорость автоповтора). Таким образом, можно сказать, что современная клавиатура настольного ПК имеет 104 или 105 клавиш; переносные ПК (Notebook то есть) имеют меньше клавиш.

ПОРТЫ КЛАВИАТУРЫ

Для работы с клавиатурой используют порты 60h и 61h. Порт 60h используется для чтения скэн-кода последней нажатой или отпущенной клавиши.

Порт 61h управляет не только клавиатурой, но и другими устройствами компьютера, например, работой встроенного динамика. Этот порт доступен как для чтения, так и для записи. Для нас важен самый старший бит этого порта. Если в старший бит порта 61h записать значение 1, клавиатура будет заблокирована, если 0 - разблокирована.

Так как порт 61h управляет не только клавиатурой, при изменении содержимого старшего бита необходимо сохранить состояние остальных битов этого порта. Для этого можно сначала выполнить чтение содержимого порта в регистр, изменить состояние старшего бита, затем выполнить запись нового значения в порт.

Есть возможность управлять скоростными характеристиками клавиатуры, а также зажигать или гасить светодиоды на лицевой панели клавиатуры - Scroll Lock, Num Lock, Caps Lock. Для расширенного управления клавиатурой используется порт 60h в режиме записи. Этот порт используется для управления подчиненным процессором Intel 8042 (или схем, выполняющих такую же функцию), ответственным за обмен данными с клавиатурным компьютером.

Процессор 8042 обслуживает не только клавиатуру, но и другие системы компьютера. Через порт 64h, например, выполняется сброс (отключение) процессора 80286 для возврата из защищенного режима работы в реальный.

Для посылки команды процессору 8042 вначале необходимо убедиться в том, что его внутренняя очередь команд пуста. Это можно сделать, прочитав слово состояния 8042 из порта с адресом 64h. Бит с номером 1 должен быть равен нулю. После того, как программа дожидается готовности процессора 8042, она может послать ему команду, записав ее в порт с адресом 60h. Некоторые команды состоят более чем из одного байта. Остальные байты команды необходимо записать в порт 60h, предварительно убедившись в готовности процессора 8042. В большинстве случаев можно также использовать простую временную задержку.

Ниже приводится формат двух команд процессора 8042, имеющих отношение к работе с клавиатурой - команду установки задержки и периода автоповтора и команду управления светодиодами, расположенными на клавиатуре.

Для установки характеристик режима автоповтора в порт 60h необходимо записать код команды 0F3h, затем байт, определяющий характеристики режима.

Таблица 13. Байт управления режимом работы клавиатуры.

Биты	Значение
0-4	Период автоповтора: 0 - 30, 4 - 20, 8 - 15, 0Ah - 10, 10h - 7.5, 14h - 5, 1Fh - 2 символов в сек.
5-6	Задержка до начала автоповтора: 00 - 250, 01 - 500, 10 - 750, 11 - 1000 миллисек.
7	Не используется, должно быть 0

Для управления светодиодами, расположенными на лицевой панели клавиатуры, используйте команду 0EDh. Вслед за этой командой в порт 60h необходимо записать байт, имеющий следующий формат:

Таблица 14. Байт состояния индикации клавиатуры.

Биты	Значение
0	Scroll Lock
1	Num Lock
2	Caps Lock
3-7	Не используются

ПРЕРЫВАНИЯ КЛАВИАТУРЫ

Клавиатура подключена к линии прерывания IRQ1. Этой линии обычно соответствует прерывание INT 09h.

Клавиатурное прерывание обслуживается модулями BIOS. Драйверы клавиатуры и резидентные программы могут организовывать дополнительную обработку прерывания INT 09h.

Стандартный обработчик клавиатуры выполняет следующие действия:

- читает из порта 60h скан-код нажатой клавиши;
- записывает вычисленное по скан-коду значение ASCII-кода нажатой клавиши в специальный буфер клавиатуры, расположенный в области данных BIOS;
- устанавливает в 1 бит 7 порта 61h, разрешая дальнейшую работу клавиатуры;
- возвращает этот бит в исходное состояние;
- записывает в порт 20h значение 20h для правильного завершения обработки аппаратного прерывания.

Обработчик прерывания INT 09h не просто записывает значение ASCII-кода в буфер клавиатуры. Дополнительно отслеживаются нажатия таких комбинаций клавиш, как Ctrl-Alt-Del, обрабатываются специальные клавиши PrtSc и SysReq. При вычислении кода ASCII нажатой клавиши учитывается состояние клавиш Shift и CapsLock.

Буфер клавиатуры имеет длину 32 байта и расположен по адресу 0000h:041Eh для машин IBM PC/XT. В IBM AT и PS/2 расположение клавиатурного буфера задается содержимым двух слов памяти с адресами 0000h:0480h (компонента смещения адреса начала буфера) и 0000h:0482h (смещение конца буфера). Обычно в IBM AT эти ячейки памяти содержат значения, соответственно, 001Eh и 003Eh. Так как смещения заданы относительно сегментного адреса 0040h, то видно, что обычное расположение клавиатурного буфера в IBM AT и PS/2 соответствует его расположению в IBM PC/XT.

Клавиатурный буфер организован циклически. Это означает, что при его переполнении самые старые значения будут потеряны. Две ячейки памяти, находящиеся в области данных BIOS с адресами 0000h:041Ah и 0000h:041Ch содержат, соответственно, указатели на начало и конец буфера. Если значения этих указателей равны друг другу, буфер пуст. (Можно удалить все символы из буфера клавиатуры, установив оба указателя на начало буфера. Однако есть более предпочтительный способ с использованием прерывания BIOS INT 16h).

Указателями на начало и конец клавиатурного буфера обычно управляют обработчики прерываний INT 09h и INT 16h.

Программа извлекает из буфера коды нажатых клавиш, используя различные функции прерывания INT 16h.

Помимо управления содержимым буфера клавиатуры, обработчик прерывания INT 09h отслеживает нажатия на так называемые переключающие клавиши - NumLock, ScrollLock, CapsLock, Ins. Состояние этих клавиш записывается в область данных BIOS в два байта с адресами 0000h:0417h и 0000h:0418h.

Программой обработки прерывания INT 09h отслеживаются некоторые комбинации клавиш, такие, как Ctrl-Alt-Del или Ctrl-Break.

Современные клавиатуры имеют отдельную альтернативную цифровую панель, напоминающую клавиатуру калькулятора. Если одновременно с нажатием на

клавишу Alt набрать число на этой панели (не большее, чем 255 и не равное 0), то это число будет помещено в буфер клавиатуры, как будто бы оно было введено нажатием на одну клавишу. Это число будет также записано в слове по адресу 0000h:0419h в области данных BIOS.

При переполнении внутреннего буфера клавиатуры или буфера, расположенного в области данных BIOS, программа-обработчик прерывания INT 09h генерирует звуковой сигнал.

ВИДЕОСИСТЕМА

АРХИТЕКТУРА

Видеосистема персонального компьютера состоит из следующих составных элементов:

- Графический процессор
- Видеопамять
- Контроллер CRT
- RAMDAC (только в современных видеосистемах)

В современных видеосистемах важную роль играют так называемые «ускорители» - специализированные процессоры для ускорения таких специфических задач, как вывод на экран 2D и 3D графики, декодирования MPEG-видеоданных и оцифровки аналогового видеосигнала.

Как правило, все компоненты, кроме видеопамяти, встраиваются в графический контроллер (чип). Видеопамять подключается к графическому чипу по специальной локальной высокоскоростной шине разрядностью 64 или 128 бит. Необходимость в такой шине появилась в связи с тем, что минимальная частота обновления экрана дисплея, при которой человек не замечает мерцания, составляет 75 Герц, чем она больше, тем лучше. При разрешении 1200 на 1024 точек частота выдачи информации на видеовыход уже превышает 75 МГц. Поэтому минимально допустимой частотой работы *RAMDAC* считается 170 МГц, и такое же требование накладывается на видеопамять. Поэтому она чаще всего дороже и сложнее обычной памяти. Для быстрой «закачки» больших массивов видеоданных из основной памяти в видеопамять видеосистема подключается к наиболее быстродействующей шине - *PCI или AGP*.

Графический процессор - специальное устройство, выполняющее различные операции над поступающими в видеосистему данными и записывающее данные в видеопамять. Как правило, включает в себя контроллер шины (*PCI или AGP*).

Видеопамять - модули оперативной памяти (RAM), расположенные на видеоплате (видеокарте) или часть оперативной памяти, выделенной для нужд видеосистемы (стратегия DiME). Как правило, для видеопамяти применяются как стандартные типы модулей памяти (*EDO DRAM, SDRAM*), так и специализированные (*SGRAM, MDRAM, VRAM*). В видеопамяти хранится содержимое экрана - его образ, который используется при выводе на экран. Текстовая информация хранится в виде «символ + атрибут» (по 8 бит на каждый), графическая информация может храниться как в плоскостном виде (каждому пикселю отводится один бит, в памяти хранится несколько плоскостей), так и в поэлементном с глубиной 8, 15, 16, 24 или 32 бита на каждый пиксель.

Контроллер CRT - устройство, управляющее выдачей информации видеовыход дисплея. Контроллер может оперировать как цифровыми данными (*EGA* и более старые системы), так и аналоговыми (современные системы).

RAMDAC - цифроаналоговый преобразователь с оперативной памятью. Используется для конвертации данных из видеопамяти (8, 15, 16, 24, 32 бита) в

аналоговый вид (подача аналогового напряжения на три канала - R, G и B аналогового дисплея).

Bank Interleaving - такая организация видеопамати, при которой все строки пикселей разбиваются на две или четыре группы. Строки одной группы хранятся вместе в одном банке - области видеопамати - друг за другом. Таким образом, размещение строк в видеопамати не соответствует размещению их на экране. Такая схема применяется в видеосистемах:

1) CGA - банки имеют адреса 0B800:0000 и 0B800:2000, в них хранятся отдельно четные и нечетные строки

2) HGC - банки имеют адреса 0B000:0000,2000,4000,6000, в них хранятся первые, вторые, третьи и четвертые строки.

Битовые плоскости - особая организация видеопамати, при которой каждому пикселю соответствует один бит в блоке данных. Каждый блок (плоскость) отвечает за один бит глубины цвета для всех пикселей. Для получения цвета пикселя выбираются биты из всех плоскостей по соответствующему адресу.

Наиболее распространена модель четырех плоскостей -- R- ,G- ,B- и I- плоскости. Первые три отвечают за компоненты красного, зеленого и синего цвета соответственно, последняя - за повышенную яркость пикселя.

VESA- Video Engineer Standard Association - ассоциация инженеров по стандартизации видеотехнологий. На ее совести несколько общепринятых стандартов:

VESA VBE - VESA Video BIOS Extension - стандарт на дополнительные функции видеоBIOS, как то переключение между видеорежимами, получение и установка параметров видеорежимов и т.д. Все режимы, управляемые с помощью функций VBE, называют VESA- режимами.

VESA DPMS - VESA Display Power Management System - стандарт энергосберегающих режимов (Suspend, Standby, Power-off) видеосистемы (дисплея в основном) и сигналов, позволяющих переключаться между этими режимами.

VESA DDC - стандарт двунаправленного общения с дисплеем для получения информации о его параметрах (производитель, размер экрана, зерна, тип покрытия, тип фосфора, температура цвета и т.д.) для оптимальной настройки средств операционной системы на конкретный дисплей.

VESA VLB - VESA Video Local Bus - уже благополучно покинувший наш мир стандарт локальной видеошины. Не был поддержан корпорацией Intel® и был вытеснен шиной PCI. Этот стандарт, по слухам, использовал некоторые неучтенные возможности процессоров и был потенциально опасен. Впрочем, это только слухи.

ТИПЫ ВИДЕОСИСТЕМ

MDA:

Первые персональные компьютеры IBM PC оснащались видеосистемой MDA - Monochrome Data Adapter. Эта видеосистема работает только в монохромном текстовом видеорежиме 7.

HGC:

Фирма Hercules Computer Technology, Inc. выпустила Hercules Graphics Card - видеосистему, использующую текстовый режим, как у MDA, и реализующую видеорежим 8 - монохромный режим 748 на 380 пикселей.

CGA:

Первая графическая система фирмы IBM - Color Graphics Adapter. Работает в текстовых режимах 0-3 и графических 4-6. Имеет крайне низкое разрешение, даже по сравнению с предыдущими видеосистемами.

EGA:

Следующее поколение видеосистем фирмы IBM - Enhanced Graphics Adapter. Улучшено графическое разрешение в текстовых режимах, добавлены новые графические - 0Dh - 10h с разрешением до 640x350.

VGA:

Последний общепринятый стандарт на видеосистему - Video Graphics Array. Улучшено разрешение в текстовых режимах, добавлены режимы 11h-13h. Применен аналоговый монитор - ограничение на максимальное число цветов снято.

MCGA:

Стандарт, похожий на VGA - Multi-Color Graphics Adapter.

После появления стандарта VGA фирма IBM окончательно потеряла контроль над рынком своих «клонов», и инициатива развития видеосистем перешла в руки отдельных специализированных фирм. Поэтому для того, чтобы как-то стандартизировать новые видеорежимы, ассоциация VESA (Video Engineer Standard Association) приняла стандарт определения и управления расширенными видеорежимами - VBE (Video BIOS Extension). Современные (1999 год) видеосистемы поддерживают VBE версии 3.0. Инициатива фирмы IBM с ее видеосистемой XGA так и не была поддержана, и каждый изготовитель видеосистемы продолжает создавать свою архитектуру, придерживаясь рекомендаций VESA.

ДИСПЛЕИ

Первым дисплеем был Monochrome Data Display, который применялся с видеосистемами MDA и HGC. Это - цифровой дисплей с одним цифровым видеовходом. Для видеосистемы CGA используется другой тип дисплея - Color Display. Это цифровой дисплей с возможностью передачи 16-ти уровней сигнала по трем каналам RGB. Частота горизонтальной развертки (т.е. число полных перерисовок экрана в секунду) составляет 60Гц. Дальнейшим его усовершенствованием стал Enhanced Color Display - для применения с видеосистемой EGA. Может отображать 64 цвета на больших, чем у Color Display, разрешениях. Для видеосистемы VGA был разработан дисплей, основанный на совершенно другом принципе передачи видеoinформации. Для того, чтобы снять ограничения на число отображаемых цветов, в видеосистему был добавлен ЦАП. На выходе видеoinформация представлена уже непрерывным аналоговым напряжением на трех каналах RGB. Поэтому все современные дисплеи используют схему VGA для работы с видеосистемой компьютера.

Первоначально каждый видеорежим имел определенную заранее частотную характеристику. Однако с развитием компьютерной техники потребовались мониторы Multi-Scan, которые автоматически подстраиваются под частоту на выходах синхронизации видеосистемы. Поэтому современные видеосистемы способны обеспечивать работу не только на разных разрешениях, но и на разных частотах вертикальной и горизонтальной развертки, а дисплей способен сам определить и настроиться на нужный режим работы автоматически.

В связи с необходимостью сберечь электроэнергию и ресурсы дисплея (ЭЛТ и электронику), а также в целях конфигурации и настройки цветовой гаммы в современных дисплеях применяются цифровые схемы управления по интерфейсам DDC и DPMS. DPMS (Display Power Management System) позволяет дисплею переключаться в режимы с отключенной разверткой и пониженным напряжением на ЭЛТ, а также отключаться при отсутствии сигналов развертки. DDC (Digital Display Configuration) позволяет организовать обратную связь дисплея с компьютером для более рационального управления видеосистемой средствами операционной системы.

ВИДЕОРЕЖИМЫ

Все видеорежимы можно разделить на две группы - текстовые и графические - по логике работы с содержимым видеопамати.

ТЕКСТОВЫЕ ВИДЕОРЕЖИМЫ

К текстовым относятся режимы с номерами от 0 до 3 и режим 7. В этом режиме видеопамать отображается на диапазон физических адресов 0B8000h-0BFFFFh для режимов 0-3 и на адреса 0B0000h-0B7FFFFh для режима 7. (Об адресации см. системная память). Вся видеопамать разбивается на 8 видеостраниц по 2000 или 4000 байт (для режимов 40x25 и 80x25 текселей). Тексель - знакоместо на экране, символ, состоящий из матрицы точек. Для CGA матрица составляет 8x8, для EGA - 8x14, для MDA и HGC - 9x14, для VGA - 9x16 точек. В видеопамати каждому текселю соответствует два байта - первый - ASCII-код символа, второй - атрибут. Атрибут имеет формат:

I R G B i r g b

I - бит яркости фона или мигания

R - бит присутствия красного

G - бит присутствия зеленого для цвета фона символов

B - бит присутствия синего

R G B - для цвета фона символов

i - бит яркости

r - бит красного

g - бит зеленого для цвета символов

b - бит синего

i r g b - для цвета символов

В видеопамати хранится 8 видеостраниц, на экране отображается только одна. При необходимости их можно легко переключать.

Каждая видеостраница представляет собой простую последовательность 16-битных слов. При этом первое слово соответствует текселю в левом верхнем углу экрана, следующие слова соответствуют текселям в первой строке, второй строке и т.д., т.е. данные хранятся построчно сверху вниз.

Работа в текстовом режиме поддерживается и средствами BIOS, и средствами DOS (немного). Правда, вывод через BIOS чаще всего бывает медленнее, чем работа напрямую с видеопамью.

ГРАФИЧЕСКИЕ ВИДЕОРЕЖИМЫ

При работе видеосистемы в графическом режиме каждому пикселю (логической точке на экране) соответствует определенное количество бит видеопамью, называемое глубиной цвета. (Следует заметить, что нет прямого соответствия между физическим пикселем (зерном) экрана и логическим пикселем - размер логического пикселя может быть как больше, так и меньше физического, и не обязательно кратен ему.) При этом организация данных, отвечающих за цвета пикселей, может быть различной. Существует две основные модели организации видеопамью :

плоскостная

последовательная, упакованная, или RGB

Существует такая особенность организация видеопамью, как банковое чередование (Bank Interleaving). Впрочем, такая схема применяется только в очень древних видеосистемах.

Вторая модель соответствует модели хранения данных в текстовых режимах и является более простой для программирования. Все пиксели содержатся друг за другом, каждому соответствует некоторое количество бит (обычно кратное 8-ми).

Первая модель подразумевает хранение битовых плоскостей друг за другом. При этом, как правило, число этих плоскостей соответствует глубине цвета. Впрочем, при глубине цвета более 4 бит (количество цветов более 16-ти) такая модель не применяется.

Работа в графическом режиме с помощью средств BIOS не рекомендуется в связи с неприлично низкой скоростью вывода на экран. Поэтому единственными операциями, где без функций BIOS обойтись трудно, являются :

переключение видеорежимов

загрузка палитры

переключение банков при работе с VESA-режимами

Все графические видеорежимы подразделяются на две группы - стандартные и VESA- режимы. Стандартные режимы были воплощены в видеосистемах VGA и более старых и являются общим стандартом для всех производителей видеосистем и видеоBIOS . VESA - режимы не являются стандартными для всех видеосистем, другими словами, некоторые видеосистемы могут не поддерживать все режимы (так чаще всего и бывает). Это связано с тем, что стандарт VGA разрабатывался крупной компанией («Голубым Гигантом») и был принят всеми как незыблемая истина. После того, как разработка

видеостандартов перешла в руки более мелких фирм, потребовалось вмешательство VESA для того, чтобы сохранить хотя бы подобие стандартов. Стандартные видеорежимы переключаются функцией 0 прерывания 10h (прерывания videoBIOS), для переключения VESA- режимов требуется использование функций 480xh того же прерывания.

Обобщенная информация по всем видеорежимам представлена в сводной таблице:

Таблица 15. Перечень видеорежимов.

Номер режима	Тип (текст, графика)	Глубина цвета, бит	Разрешение (пиксели, тексели)	Организация (1-упак., 2-плоск)	Поддерживается видеосистемами
0	Текст	-	40x25		CGA и старше
1	Текст	-	80x25		CGA и старше
2	Текст	-	40x25		CGA и старше
3	Текст	-	80x25		CGA и старше
4	Графика	2	320x200	1	CGA и старше
5	Графика	2	320x200	1	CGA и старше
6	Графика	1	640x200	1	CGA и старше
7	Текст	-	80x25		Все
8	Графика	1	739x348	1	HGC
9 - 0Ch	-	-	-	-	Резерв
0Dh	Графика	4	320x200	2	EGA и старше
0Eh	Графика	4	640x200	2	EGA и старше
0Fh	Графика	1,5	640x350	-	EGA и старше
10h	Графика	4	640x350	2	VGA
11h	Графика	1	640x480	-	VGA
12h	Графика	4	640x480	2	VGA
13h	Графика	8	320x200	1	VGA
100h	Графика	8	640x400	1	sVGA(VESA)
101h	Графика	8	640x480	1	sVGA(VESA)
102h	Графика	4	800x600	2	sVGA(VESA)
103h	Графика	8	800x600	1	sVGA(VESA)
104h	Графика	4	1024x768	2	sVGA(VESA)
105h	Графика	8	1024x768	1	sVGA(VESA)
108h	Текст	-	80x60		sVGA(VESA)
109h	Текст	-	132x25		sVGA(VESA)
10Ah	Текст	-	132x43		sVGA(VESA)

110h	Графика	15	640x480	1	sVGA(VESA)
111h	Графика	16	640x480	1	sVGA(VESA)
112h	Графика	24	640x480	1	sVGA(VESA)
113h	Графика	15	800x600	1	sVGA(VESA)
114h	Графика	16	800x600	1	sVGA(VESA)
116h	Графика	15	1024x768	1	sVGA(VESA)
117h	Графика	16	1024x768	1	sVGA(VESA)

ЗВУКОВАЯ СИСТЕМА

ЗВУКОВЫЕ КАРТЫ

Современные звуковые карты являются специальными платами расширения, работающими на шине PCI. Современная звуковая карта должна иметь 16 бит стерео цифро-аналоговый и аналого-цифровой преобразователи, обеспечивать запись и воспроизведение стереозвука "CD качества" (44100 Гц частота преобразования) в режиме полного "дуплекса" (возможность одновременной записи и воспроизведения без каких-либо ограничений) и иметь синтезатор MIDI основанный на Wave-Table принципе. Желательно иметь возможность аппаратного микширования нескольких звуковых потоков, осуществлять аппаратное ускорение при построении геометрической акустической модели (для обеспечения трехмерного позиционирования звука).

ПОРТЫ SOUND BLASTER

Родоначальницей массового проникновения звуковых карт в персональные компьютеры является фирма Creative Labs.

Sound Blaster 2.0 использует так называемый "базовый порт", обычно 220h. Все остальные порты отсчитываются от базового, который может быть изменен программно.

В дальнейшем знак "(w)" обозначает доступность для записи, "(r)" - для чтения.

2x0h (w) порт данных CMS Music Voice 1-6

2x1h (w) регистры CMS Music Voice 1-6

2x2h (w) порт данных CMS Music Voice 7-12

2x3h (w) регистры CMS Music Voice 7-12

2x8h (r) статус FM-синтезатора

2x8h (w) регистр адреса FM-синтезатора

2x9h (w) порт данных FM-синтезатора

FM-синтезатор также доступен через порты 388h и 389h - для совместимости с синтезаторами AdLib.

2x6h (w) сброс DSP

2xAh (r) чтение данных из DSP

2xCh (r) статус порта записи в DSP

2xCh (w) порт данных и команд DSP

2xEh (r) статус доступности DSP

Sound Blaster Pro использует 24 порта, начиная с базового

2x0h (r) статус левого FM-канала

2x0h (w) левый регистр FM-синтезатора

2x1h (w) порт данных левого FM-канала

2x2h (r) статус правого FM-канала

2x2h (w) правый регистр FM-синтезатора

2x3h (w) порт данных правого FM-канала
2x4h (w) регистр микшера
2x5h (r/w) порт данных микшера
2x8h (r) статус FM-синтезатора
2x8h (w) регистр FM-синтезатора
2x9h (w) порт данных FM-синтезатора
2x6h (w) - 2xEh (r) аналогично Sound Blaster 2.0
2x0+10h (r) - 2x0+13h (w) работа с CD-ROM (устарело).

ПРЕРЫВАНИЯ SOUND BLASTER

Sound Blaster использует аппаратное прерывание IRQ2,3,5,7 или 9.

Эти прерывания используются для выполнения команд:

10h Direct mode 8-bit DAC (single-byte data transfer)
14h DMA mode 8-bit DAC
16h DMA mode 2-bit ADPCM DAC
17h DMA mode 2-bit ADPCM DAC with Reference byte
20h Direct mode 8-bit ADC (single-byte data transfer)
24h DMA mode 8-bit ADC
30h MIDI Read (polling mode)
31h MIDI Read (interrupt mode)
34h MIDI UART mode (polling mode)
35h MIDI UART mode (interrupt mode)
38h MIDI Write (polling mode)
40h Set Time Constant
74h DMA mode 4-bit ADPCM DAC
75h DMA mode 4-bit ADPCM DAC with Reference byte
76h DMA mode 2.6-bit ADPCM DAC
77h DMA mode 2.6-bit ADPCM DAC with Reference byte
80h Set Silence Mode Data length
91h High-Speed DMA mode 8-bit DAC
92h High-Speed DMA mode 8-bit ADC
D1h Turn on Speaker
D3h Turn off Speaker
D8h Get Speaker Status
00h = off
FFh = on
D0h Halt DMA in progress
D4h Continue DMA
E1h Get DSP Version
low-byte = Major
High-byte = Minor

DSP

DSP (Digital Signal Processor - обработчик цифровых сигналов) может использоваться в двух режимах - непосредственное программирование и DMA.

Первый способ проще для программирования, но он требует программной установки всех необходимых задержек, а также ограничивает формат проигрываемого звука 8-мибитным PCM (Pulse Code Modulation - импульсная кодовая модуляция) без сжатия. DMA-режим позволяет также работать со сжатым звуком.

Перед началом работы DSP нужно проинициализировать.

1. Запись байта = 1 в порт сброса DSP
2. Задержка не менее 3 микросекунд
3. Запись байта = 0 в порт сброса DSP
4. Ожидание установленного бита 7 в порту статуса доступности DSP (2xEh)
5. Чтение байта из порта данных DSP (2xAh)
6. Если прочитано 0AAh, DSP готов, иначе - пункт 4.

DSP требует около 100 микросекунд для инициализации. Если после 100 повторов не получено значение 0AAh, звуковая карта не работает (или ее нет вообще).

Запись производится:

Чтение статуса буфера записи (порт 2xCh)

Если бит 7 установлен, повтор шага 1

Запись в порт данных (2xCh)

Чтение производится:

Чтение порта доступности данных DSP (2xEh)

Если бит 7 установлен в 0, повтор шага 1

Чтение байта из порта 2xAh

Обработка прерываний

В режиме DMA DSP предоставляет прерывания для выполнения операций цифроаналогового преобразования (вывод звука), аналого-цифрового (для ввода звука) и MIDI. Для выполнения этих операций требуется перехват прерывания. Т.е. чтение из порта DSP (2xAh) выполняется обработчиком прерывания. Вызов прерывания осуществляется путем чтения порта доступности DSP (2xEh) без проверки результата чтения. Обработчик прерывания должен послать сигнал EOI (записать 20h в порт 20h - как и любой другой обработчик аппаратного прерывания).

Работа с DSP в непосредственном режиме

Запись байта 10h в порт команд DSP (2xCh)

Запись одного байта некомпьютеризованного звука (PCM 8bit) в порт записи (2xCh)

Осуществление необходимой задержки

Повторить с шага 1

До начала этой операции нужно проинициализировать звуковую карту.

Работа с DSP в режиме DMA

Установить обработчики прерываний

Запрограммировать частоту дискретизации передаваемых звуковых данных:

отключить колонки (запись байта 0D1h в порт команд (2xCh)

команда "Установить время" (40h) в тот же порт

записать значение дискретизации в тот же порт (оно находится по формуле $256 - (1\ 000\ 000 / \text{частота})$). Для 8 кГц это 131

Запрограммировать контроллер DMA на прием блока данных

Выбор канала 1 (запись байта 5 в регистр маски DMA (порт 0Ah)

Запись 0 в порт 0Ch

Установить DMA для ЦАП (45h) или АЦП (49h) записью в порт 0Bh

Послать адрес блока данных записью младшего, затем старшего байта адреса в порт 02h

Запись номера физической страницы в порт 83h

Запись младшего, затем старшего байта числа (байт в блоке данных - 1) в порт 03h

Включить канал 1 (запись байта 01h в регистр маски - порт 0Ah)

Программирование DSP для приема/выдачи данных

Команда “DMA-режим 8-битного ЦАП” - 14h в порт 2xCh

Запись младшего, затем старшего байта длины блока данных в порт команд (2xCh)

Вызов прерывания

Сохранить регистры (если нужно)

Запросить прерывание (чтение из порта 2xEh), только один раз (!)

Если это был последний блок данных - на шаг f)

Запрограммировать DMA (см. шаг 3) на новый блок данных

Запрограммировать DSP (см. шаг 4)

Послать сигнал EOI

Восстановить регистры

Вызвать IRET

Восстановить обработчик прерывания