**20.11.2020 ИС 2к Основы архитектуры**

 **Тема: 1.3.** **Классическая архитектура вычислительной машины**

 **1.4.** **Технические и эксплуатационные характеристики**

 **вычислительных машин**

**1.5.** **Процессоры**

 ***(Учебник «Основы архитектуры, устройство и функционирование вычислительных систем» Степина ВВ)***

 ***стр. 17-36*** [**https://yadi.sk/i/1PZf\_-ZO43S0fA**](https://yadi.sk/i/1PZf_-ZO43S0fA)

**Ответить на вопросы:**

1. Опишите принцип работы фон-неймановской вычислительной машины.
2. Перечислите принципы фон Неймана, положенные в основу построения современных вычислительных машин.
3. В чем заключается отличие микроархитектуры процессора от макроархитектуры?
4. Перечислите основные характеристиками процессора.
5. Каково назначение устройства управления микропроцессора?
6. Каково назначение регистров микропроцессора?

**Ответы присылать** kuzn117@yandex.ru

**можно в ВК -**  **id480169637**

* 1. **Классическая архитектура вычислительной машины**

Разнообразие современных вычислительных машин очень велико, но все они представляют собой реализацию так называемой фон-ней- мановской (принстонской) архитектуры, представленной Джорджем фон Нейманом еще в 1945 г. Рассмотрим классическую архитектуру вычислительной машины на примере архитектуры фон Неймана.

Принцип действия вычислительной машины состоит в выполнении программ последовательностей арифметических, логических и других операций, описывающих решение определенной задачи.

**Программа** (для ЭВМ) — это упорядоченная последовательность команд, подлежащая обработке (стандарт ISO 2382/1-84).

**Команда** — это описание операции, которую должна выполнить вычислительная машина.

Результат команды вырабатывается по точно определенным для данной команды правилам, заложенным в конструкцию компьютера. Электронные схемы каждого компьютера могут распознавать и выполнять ограниченный набор простых команд.

На рис. 1.3 представлена схема фон-неймановской вычислительной машины, на основе которой уже более полувека создаются современные вычислительные машины.

**Рис. 1.3.** Схема фон-неймановской вычислительной машины

*17*

Фон-неймановская архитектура состоит из следующих основных устройств:

* памяти, которая включала 4096 машинных слов разрядностью 40 бит. Машинное слово содержало или команды (две команды по 20 бит), или целое число со знаком на 40 бит (8 бит указывали на тип команды, а остальные 12 бит определяли одно из 212 = 4096 слов);
* арифметико-логического устройства (АЛУ), внутри которого находится особый внутренний регистр разрядностью 40 бит, так называемый аккумулятор;
* устройства управления (УУ), выполняющего функции управления устройствами;
* устройства ввода информации;
* устройства вывода информации.

Эти устройства соединены каналами связи, по которым передается информация.

В современных вычислительных машинах арифметико-логическое устройство и устройство управления сочетаются в одной микросхеме, которая называется *центральным процессором* (ЦП).

Машинная команда считывает слово из памяти в аккумулятор или сохраняет содержимое аккумулятора в памяти. Машина фон Неймана выполняла арифметические операции только с фиксированной точкой, поскольку фон Нейман был отличным математиком и считал, что плавающую точку можно держать в голове.

В основу построения современных вычислительных машин были положены следующие принципы фон Неймана.

1. *Принцип однородности памяти.* Команды и данные хранятся в одной и той же памяти и внешне в памяти неразличимы. Распознать их можно только по способу использования; т.е. одно и то же значение в ячейке памяти может использоваться и как данные, и как команда, и как адрес в зависимости лишь от способа обращения к нему. Это позволяет производить над командами те же операции, что и над числами, и, соответственно, открывает ряд возможностей. Так, циклически изменяя адресную часть команды, можно обеспечить обращение к последовательным элементам массива данных. Такой прием носит название модификации команд и с позиций современного программирования не приветствуется. Более полезным является другое следствие принципа однородности, когда команды одной программы могут быть получены как результат исполнения другой программы. Эта возможность лежит в основе трансляции — перевода текста программы с языка высокого уровня на язык конкретной вычислительной машины.

*18*

1. *Принцип адресности.* Структурно основная память состоит из пронумерованных ячеек, причем процессору в произвольный момент доступна любая ячейка. Двоичные коды команд и данных разделяются на единицы информации, называемые словами, и хранятся в ячейках памяти, а для доступа к ним используются номера соответствующих ячеек — адреса.
2. *Принцип программного управления.* Все вычисления, предусмотренные алгоритмом решения задачи, должны быть представлены в виде программы, состоящей из последовательности управляющих слов — команд. Каждая команда предписывает некоторую операцию из набора операций, реализуемых вычислительной машиной. Команды программы хранятся в последовательных ячейках памяти вычислительной машины и выполняются в естественной последовательности, т.е. в порядке их положения в программе. При необходимости с помощью специальных команд эта последовательность может быть изменена. Решение об изменении порядка выполнения команд программы принимается либо на основании анализа результатов предшествующих вычислений, либо безусловно.
3. *Принцип двоичного кодирования.* Согласно этому принципу вся информация, как данные, так и команды, кодируется двоичными цифрами 0 и 1. Каждый тип информации представляется двоичной последовательностью и имеет свой формат. Последовательность битов в формате, имеющая определенный смысл, называется *полем*. В числовой информации обычно выделяют поле знака и поле значащих разрядов. В формате команды можно выделить два поля: поле кода операции и поле адресов.

Огромным преимуществом фон-неймановской архитектуры является ее простота, поэтому данная концепция легла в основу большинства компьютеров общего назначения. Однако совместное использование шины для памяти программ и памяти данных приводит к так называемому узкому месту архитектуры фон Неймана. Термин «узкое место архитектуры фон Неймана» ввел Джон Бэкус в 1977 г. в своей лекции «Можно ли освободить программирование от стиля фон Неймана?», которую он прочитал при вручении ему Премии Тьюринга.

Фон-неймановская архитектура — не единственный вариант построения ЭВМ, есть и другие, которые не соответствуют указанным принципам (например, потоковые машины). Однако подавляющее большинство современных компьютеров основано именно на указанных принципах, включая и сложные, многопроцессорные комплексы, которые можно рассматривать как объединение фон-ней- мановских машин.

*19*

* 1. **технические и эксплуатационные характеристики вычислительных машин**

**Производительность вычислительной машины.** Этот показатель определяется архитектурой процессора, иерархией внутренней и внешней памяти, пропускной способностью системного интерфейса, системой прерывания, набором периферийных устройств в конкретной конфигурации, совершенством ОС и т.д. Основные единицы оценки производительности:

* абсолютная, определяемая количеством элементарных работ, выполняемых в единицу времени;
* относительная, определяемая для оцениваемой ЭВМ относительно базовой в виде индекса производительности.

Для каждого вида производительности применяются следующие традиционные методы их определения.

**Пиковая производительность** (быстродействие) определяется средним числом команд типа «регистр-регистр», выполняемых в одну секунду без учета их статистического веса в выбранном классе задач.

**Номинальная производительность** (быстродействие) определяется средним числом команд, выполняемых подсистемой «процессорпамять» с учетом их статистического веса в выбранном классе задач. Она рассчитывается, как правило, по формулам и специальным методикам, предложенным для процессоров определенных архитектур, и измеряется с помощью разработанных для них измерительных программ, реализующих соответствующую эталонную нагрузку.

Для данных типов производительностей используются следующие единицы измерения:

* MIPS (Mega Instruction Per Second) — миллион команд в секунду;
* MFLOPS (Mega Floating Operations Per Second) — миллион операций над числами с плавающей запятой в секунду;
* GFLOPS (Giga Floating Operations Per Second) — миллиард операций над числами с плавающей запятой в секунду и т.д.

**Системная производительность** измеряется с помощью синтезированных типовых (тестовых) оценочных программ, реализованных на унифицированных языках высокого уровня. Унифицированные тестовые программы используют типичные алгоритмические действия, характерные для реальных применений, и штатные компиляторы ЭВМ. Они рассчитаны на использование базовых технических средств и позволяют измерять производительность для расширенных конфигураций технических средств. Результаты оценки системной производительности ЭВМ конкретной архитектуры приводятся от

*20*

носительно базового образца, в качестве которого используются ЭВМ, являющиеся промышленными стандартами систем ЭВМ различной архитектуры. Результаты оформляются в виде сравнительных таблиц, двумерных графиков и трехмерных изображений.

**Эксплуатационная производительность** оценивается на основании использования данных о реальной рабочей нагрузке и функционировании ЭВМ при выполнении типовых производственных нагрузок в основных областях применения. Расчеты делаются главным образом на уровне типовых пакетов прикладных программ текстообработки, систем управления базами данных, пакетов автоматизации проектирования, графических пакетов и т.д.

* 1. **Процессоры**

*Центральное процессорное устройство* (ЦПУ; англ. central processing unit, CPU; дословно — «центральное обрабатывающее устройство») — это электронный блок либо интегральная схема (микропроцессор), исполняющие машинные инструкции (код программ), главная часть аппаратного обеспечения компьютера или программируемого логического контроллера. Иногда называют микропроцессором (МП) или просто процессором.

***Характеристики процессора***

Современные процессоры представляют собой сложные, высокотехнологичные устройства и описываются целым рядом показателей. Важнейшими характеристиками процессора являются:

1. **тактовая частота** (clock rate) — это число элементарных операций (тактов), производимых за 1 секунду. Измеряется в герцах (Hz, Гц и их производных по системе СИ — килогерцах, kHz, кГц, мегагерцах, MHz, МГц, гигагерцах, GHz, ГГц).

Тактовая частота центрального микропроцессора определяет производительность как самого процессора, так и в целом всей компьютерной системы. Тем не менее с развитием современных технологий, таких как, например, суперскалярность, и созданием многоядерных процессоров нельзя установить прямо пропорциональной зависимости между тактовой частотой процессора и его производительностью;

1. **разрядность** — максимальное число одновременно обрабатываемых двоичных разрядов. Разрядность МП обозначается *m/n/k/* и включает:

• *m* — разрядность внутренних регистров, определяет принадлежность к тому или иному классу процессоров;

*21*

* *n* — разрядность шины данных, определяет скорость передачи

информации;

* *k* — разрядность шины адреса, определяет размер адресного пространства. Например, МП i8088 характеризуется значениями

*m/n/k* = 16/8/20;

1. **количество вычислительных ядер** на одном процессорном кристалле. Многоядерность как способ повышения производительности процессоров используется с относительно недавнего времени, но признана самым перспективным направлением их развития. Для домашних компьютеров уже существуют процессоры с восемью ядрами. Для серверов на рынке есть 12-ядерные предложения (Opteron 6100). Разработаны прототипы процессоров, содержащие около 100 ядер.Эффективность вычислительных ядер разных моделей процессоров отличается. Но в любом случае чем ядер больше, тем процессор производительнее;
2. **количество потоков.**Чем больше потоков — тем лучше. Количество потоков не всегда совпадает с количеством ядер процессора. Так, благодаря технологии Hyper-Threading 4-ядерный процессор Intel Core i7—3820 работает в восемь потоков и во многом опережает 6-ядерных конкурентов;
3. **размер кэш-памяти** 2-го и 3-го уровней, иногда и 4-го уровня. *Кэш* — это очень быстрая внутренняя память процессора, которая используется им как буфер для временного хранения информации, обрабатываемой в конкретный момент времени;
4. **скорость шины процессора** (FSB, HyperTransport или QPI).Через эту шину центральный процессор взаимодействует с материнской платой. Ее скорость (частота) измеряется в мегагерцах, и чем она выше — тем лучше;
5. **технологический процесс.** Чем выше степень интеграции (количество транзисторов на кристалле), тем меньше потребляемая электроэнергия процессора. От техпроцесса во многом зависит еще одна важная характеристика процессора — TDP;
6. **Termal Design Point** (TDP) — показатель, отображающий энергопотребление процессора, а также количество тепла, выделяемого им в процессе работы. Единицы измерения — ватты (Вт). TDP зависит от многих факторов, среди которых главными являются количество ядер, техпроцесс изготовления и частота работы процессора.

Кроме прочих преимуществ, «холодные» процессоры (с TDP до 100 Вт) лучше поддаются разгону, когда пользователь изменяет некоторые настройки системы, вследствие чего увеличивается частота

22

процессора. Разгон позволяет без дополнительных финансовых вложений увеличить производительность процессора на 15—25 %, но это уже отдельная тема. В то же время проблему с высоким TDP всегда можно решить приобретением эффективной системы охлаждения;

1. **наличие и производительность видеоядра.** Последние технические достижения позволили производителям помимо вычислительных ядер включать в состав процессоров еще и ядра графические. Такие процессоры, кроме решения своих основных задач, могут выполнять роль видеокарты;
2. **тип и максимальная скорость поддерживаемой оперативной памяти.** Эти характеристики процессора необходимо учитывать при выборе оперативной памяти, с которой он будет использоваться. Нет смысла переплачивать за быстрые модули ОЗУ, если процессор не сможет реализовать все их преимущества.

***Развитие архитектуры процессора***

Развитие вычислительной техники сопровождается совершенствованием центральных процессоров. При проектировании новых моделей процессоров разработчики основывались на принципах совместимости, т.е. разные модели процессоров, различающихся по производительности, должны были «уметь» выполнять одни и те же команды. Чтобы охарактеризовать этот уровень совместимости, компания IBM ввела термин «архитектура». Новое семейство процессоров должно было иметь единую архитектуру, т.е. новая модель процессора разрабатывается на основе какой-либо существующей архитектуры.

Под термином «архитектура процессора» понимают совокупность и способ объединения компонентов процессора, а также его совместимость с определенным набором команд.

Знание этих двух моментов дает возможность грамотно организовать интерфейс аппаратных и программных средств вычислительной системы. Например, с точки зрения программиста, архитектура процессора — это способность процессора выполнять набор машинных кодов, а с точки зрения проектирования компьютерных составляющих архитектура процессора — это отражение основных принципов внутренней организации определенных типов процессоров. Допустим, архитектура Intel Pentium обозначается Р5, Pentium II и Pentium III — Р6, а не так давно популярных Pentium 4 — NetBurst. Когда компания Intel закрыла Р5 для конкурирующих производителей, компания AMD разработала свою архитектуру К7 для Athlon и Athlon XP, а для Athlon 64 — К8.

*23*

Но даже процессоры с одинаковой архитектурой могут существенно отличаться друг от друга. Эти различия обусловлены разнообразием процессорных ядер, которые обладают определенным набором характеристик. Наиболее частыми отличиями являются различные частоты системной шины, а также размеры кэша второго уровня и технологические характеристики, по которым изготовлены процессоры. Очень часто смена ядра в процессорах из одного и того же семейства требует также замены процессорного разъема, а это влечет за собой проблемы с совместимостью материнских плат. Но производители постоянно совершенствуют ядра и вносят постоянные, но незначительные изменения в ядре. Такие нововведения называют ревизией ядер и, как правило, обозначают цифробуквенными комбинациями.

Выделяют понятия микроархитектуры и макроархитектуры.

*Микроархитектура* микропроцессора — это аппаратная организация и логическая структура микропроцессора, регистры, управляющие схемы, арифметико-логические устройства, запоминающие устройства и связывающие их информационные магистрали.

*Макроархитектура* — это система команд, типы обрабатываемых данных, режимы адресации и принципы работы микропроцессора.

Архитектура современных микропроцессоров представляет собой продукт нескольких независимых групп разработчиков, которые развивают эту архитектуру более 15 лет, добавляя новые возможности к первоначальному набору команд.

В настоящее время доминирующее положение на рынке центральных процессоров занимает семейство процессоров фирмы «Intel» с архитектурой X86.

Это семейство открывают 16-разрядные процессоры 8086 и 8088 с 16/8-битной шиной данных и 20-битной шиной адреса. Второе поколение процессоров представлено процессором 80286, в котором шина адреса была расширена до 24 бит, что позволяло в особом режиме (protected mode — защищенный режим) адресовать до 16 Мбайт физической памяти. Начиная с третьего поколения (Intel 386) архитектура процессоров этого семейства стала 32-битной, а основным режимом работы — защищенный. В новых моделях усовершенствована работа с кешем (Intel 486), появились параллельные конвейеры (Pentium), новые архитектурные блоки (Pentium MMX), появился встроенный кеш второго уровня (P6). Эти изменения сопровождались также добавлением новых возможностей при работе в защищенном режиме: VME (Virtual Mode Extension) у Pentium, PAE (Physical Address Extension) у P6 и др.

*24*

В семействе Intel с архитектурой X86 декларируется программная совместимость моделей процессоров сверху вниз. Это значит, что код, написанный для 8086, должен работать и на 80386, и на Pentium 4. С другой стороны, программы, разработанные для более поздних процессоров, могут не работать на более ранних, если в них используются какие-либо специфические особенности новой модели.

***Немного истории***

Появлением первых микропроцессоров мы обязаны компании Intel Corporation of Santa Clara, которая объявила процессор как «компьютер-на-чипе», это был 4-битный Intel 4004 (ноябрь 1971). В одном кристалле размещалась большая часть компонентов процессора. Фактически процессор был реализован на четырех микросхемах. На базе 4-битного Intel 4004-процессора по заказу японской компании «Busicom» был сделан первый микрокалькулятор.

Процессор был однопрограммным — следующая программы исполнялась только после завершения предыдущей — и неуниверсальным — был предназначен исключительно для вычислительных работ и не мог применяться, например, для обработки текстов.

Почти в то же время появился специализированный микропроцессор 8008, который предназначался для использования в терминале вычислительной машины. Развитием микропроцессоров стало появление универсальных многопрограммных 8-битных Intel 8080 (апрель 1974) и MOS Technology 6502 (сентябрь 1975). Они оба использовались в производстве настольных компьютеров и игровых консолей: первый — Altair 8800, второй — Nintendo NES, Atari 2600, Apple I, Apple II, Commodore 64, Агат и др. В отличие от 4004 эти микропроцессоры использовали раздельные шины адреса и данных, а инструкции и данные хранились в одних и тех же областях памяти. Он имел довольно высокую производительность, позволял адресовать значительный объем памяти, да и спектр его возможностей был существенно шире, чем у предшественников. Это были первые CPU, работающие на основе архитектуры фон Неймана и выполняющие функции арифметико-логического устройства и устройства управления.

Intel 8080 быстро стал общепризнанным стандартом, и многие фирмы начали выпускать его по лицензии. Стали появляться улучшенные версии, например Z80 фирмы «Zilog» или V10 фирмы «NEC», но структура оставалась прежней. Кстати, и сама фирма «Intel» в 1976 г. тоже выпустила модернизированный вариант кри

*25*

сталла 8080 — микропроцессор 8085, который был дополнен несколькими командами и улучшен аппаратно — встроены тактовый генератор и контроллер шины, добавлен простой последовательный порт, увеличена тактовая частота.

Следующим этапом в эволюции центральных процессоров стал выпуск 16-битных Intel 8086/88 (июнь 1978), которые положили начало архитектуре Х86 и массовому распространению персональных компьютеров. Процессор обладал весьма высокой по тем временам производительностью и серьезными возможностями, в том числе полной десятичной арифметикой, что позволяло применять его в самых различных областях. В 1980 г. был представлен первый процессор с RISC-архитектурой — IBM 801. По сравнению с CISC-процессорами того времени он имел меньшие размеры и число инструкций, был проще и дешевле в изготовлении. В 1984 г. начали изготовляться первые процессоры VLIW-архитектуры, однако они не получили большого распространения.

В 1978 г. появилось третье поколение микропроцессоров, и опять фирма «Intel» оказалась лидером — кристалл 8086 стал первым микропроцессором, оперирующим 16-разрядными словами данных. Он обладал весьма высокой по тем временам производительностью и серьезными возможностями, в том числе полной десятичной арифметикой, что позволяло применять его в самых различных областях.

В 1979 г. фирма «Intel» упростила процессор 8086, создав микросхему 8088 с 8-разрядной внешней шиной данных.

В 1983 г. «Intel» разработала еще два микропроцессора, представлявших собой усовершенствованные варианты 8086 и 8088, — 80186 и 80188, однако получить широкого распространения они не успели, так как в том же году появился процессор 80286, ставший серьезным шагом вперед. Всего через год на его базе был создан персональный компьютер IBM PC/AT, предоставивший в распоряжение пользователя вычислительные мощности средней ЭВМ. С появлением виртуального режима стало возможным создавать на базе 80286 системы с разделением ресурсов, что раньше было прерогативой больших машин. В микропроцессоре было также реализовано управление памятью.

Дальнейшее развитие процессоров привело к переходу на 32-разрядные модели, позволявшие эффективнее работать с большими числами и адресовать ранее недоступные объемы памяти. В октябре

1. г. вышел первый 32-битный Х86-процессор, Intel 80386, а в
2. г. появились три новые 32-битные RISC-архитектуры —MIPS, SPARC и PA-RISC, представленные компаниями «MIPS Technolo-

*26*

gies», «Sun» и «HP» соответственно. Следующим шагом было появление 64-битных процессоров MIPS R4000 (февраль 1991) и DEC Alpha 21064 (ноябрь 1992). Alpha был также первым CPU, поддерживающим суперскалярность, т.е. возможность исполнять более одной инструкции за такт. Первыми суперскалярными процессорами других архитектур стали Intel Pentium (март 1993), MIPS R8000 (июнь 1994), PA-RISC 7100 (июнь 1994) и 64-битный UltraSPARC (сентябрь 1994).

Следующей вехой в истории центральных процессоров стало динамическое исполнение команд. Заключалось оно в том, что процессор исполнял команды не в том порядке, в котором он их считывал из памяти, а в том, который был более эффективен по времени выполнения, и при этом, конечно же, не нарушал семантики программы. Эта технология была реализована во всех процессорах соответствующих архитектур, начиная с MIPS R10000 (октябрь 1994), PA-RISC 8000 (март 1995), Intel Pentium Pro (ноябрь 1995), Alpha 21264 (декабрь 1998).

После этого появились первые популярные и коммерчески успешные процессоры с архитектурой VLIW —Intel Itanium (октябрь 1999) и Transmeta Crusoe (январь 2000). Затем с некоторым опозданием от других платформ вышли 64-битные расширения для Х86, реализованные в процессорах AMD Opteron (технология AMD64, апрель 2003) и Pentium 4 (EM64T, август 2004).

В начале нового тысячелетия развитие центральных процессоров пошло в сторону увеличения количества ядер в одном процессорном корпусе. Практически одновременно вышли двухъядерные CPU всех популярных архитектур: PA-RISC 8800 (февраль 2004), UltraSPARC-IV (февраль 2004), IBM PowerPC G4 (август 2004), MIPS BCM1255 (октябрь 2004), AMD Athlon X2 (апрель 2005), Pentium D (май 2005), Itanium 2 (октябрь 2005), Intel Core 2 Duo (июль 2006). В ноябре 2006 г. появился уже 7-ядерный Cell в составе Sony PlayStation 3. А в серверном процессоре Intel® Xeon® Processor E7-8890 v4 (дата выпуска — июнь 2016) количество ядер составило уже 24.

***Классификация микропроцессоров***

1. *По числу БИС в микропроцессорном комплекте:*
* однокристальные МП;
* многокристальные МП;
* многокристальные секционные МП.

В первую очередь на такое деление повлияли возможности БИС: ограниченное число элементов, выводов корпуса, в то время как МП — довольно сложное устройство, имеющее много логических элементов и требующее большое количество выводов корпуса БИС.

*27*

*Однокристальный МП* получен при реализации всех аппаратных средств МП в виде одной БИС или СБИС. Основные характеристики таких МП зависят от технологии изготовления БИС.

*Многокристальные МП* получены при разбиении его логической структуры на функционально законченные части и реализации их в виде БИС (рис. 1.4).

**Рис. 1.4.** Логические структуры многокристального МП:

ОП — операционный процессор, служит для обработки данных;

УП — управляющий процессор, выполняет функции выборки, декодирова-
ния и вычисления адресов операндов, а также генерирует последователь-
ность команд, формирует очередь команд; ИП — интерфейсный процессор,
позволяет подключить память к МП

УП, ОП, ИП могут работать автономно (параллельно) и тем самым организовывать конвейер операций.

*Многокристальные секционные МП* получаются, когда в виде БИС реализуются логические структуры МП при функциональном разбиении ее вертикальными плоскостями (рис. 1.5).

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  | УП |  |  |  |
|  |  |  |  |
|  |  | ОП |  |  |  |
|  |  |  |
|  | ИП |  |  |  |
|  |  |  |

**Рис. 1.5.** Логические структуры многокристального секционного МП

**Пример.** Если невозможно реализовать ОП 16 разрядов в одной БИС, его делят на части, реализуемые каждая в своей БИС. Они образуют микропроцессорные секции (например, 4-разрядная секция).

*28*

1. *По назначению*:
* универсальные;
* специальные.

*Универсальные МП* могут быть применены для решения широкого круга задач. При этом их эффективная производительность мало зависит от проблемной специфики задачи. Как правило, это определяется достаточно широкой универсальной системой команд.

*Специальные МП* — проблемно ориентированные МП, которые нацелены на ускоренное выполнение определенных функций, что увеличивает эффективную производительность при решении только определенной задачи:

* математические процессоры;
* микроконтроллеры;
* параллельная обработка данных;
* цифровая обработка сигнала — цифровые фильтры и т.д. **Пример.** Сравнение входного сигнала одновременно с несколькими эталонами для выделения нужного сигнала.
1. *По виду обрабатываемых входных сигналов:*
* цифровые;
* аналоговые.

Требования к аналоговым МП:

* большая разрядность;
* высокая скорость арифметических операций.
1. *По характеру временной организации работы:*
* одномагистральные — все устройства имеют одинаковый интерфейс и подключаются к единой информационной магистрали, по которой передаются коды данных, адресов и управляющих сигналов;
* многомагистральные — устройства группами подключаются к своей информационной магистрали, это позволяет осуществить одновременную передачу информационных сигналов по нескольким магистралям. Производительность увеличивается.
1. *По количеству выполняемых программ:*
* однопрограммные;
* мультипрограммные.

Мультипрограммные либо могут одновременно выполнять несколько программ, либо имеют средства для поддержки виртуальной мультипрограммности.

**Структура типового микропроцессора.** Архитектура типичной небольшой вычислительной системы на основе микроЭВМ показана на рис. 1.6. Такая микроЭВМ содержит все пять основных блоков цифровой машины: устройство ввода информации, управляющее устройство (УУ), арифметико-логическое устройство (АЛУ) (входя-

*29*

щее в состав микропроцессора), запоминающие устройства (ЗУ) и устройство вывода информации.

Шина данных (8 линий)

**Рис. 1.6.** Архитектура типового микропроцессора

Микропроцессор координирует работу всех устройств цифровой системы с помощью шины управления (ШУ). Помимо ШУ, имеется 16-разрядная адресная шина (ША), которая служит для выбора определенной ячейки памяти, порта ввода или порта вывода. По 8-разрядной информационной шине или шине данных (ТТТД) осуществляется двунаправленная пересылка данных к микропроцессору и от микропроцессора. Важно отметить, что МП может посылать информацию в память микроЭВМ или к одному из портов вывода, а также получать информацию из памяти или от одного из портов ввода.

Постоянное запоминающее устройство (ПЗУ) в микроЭВМ содержит некоторую программу (на практике — программу инициализации ЭВМ). Программы могут быть загружены в запоминающее устройство с произвольной выборкой (ЗУПВ) и из внешнего запоминающего устройства (ВЗУ). Это программы пользователя.

В качестве примера, иллюстрирующего работу микроЭВМ, рассмотрим процедуру, для реализации которой нужно выполнить следующую последовательность элементарных операций.

1. Нажать клавишу с буквой «А» на клавиатуре.
2. Поместить букву «А» в память микроЭВМ.
3. Вывести букву «А» на экран дисплея.

*30*

Это типичная процедура ввода-запоминания-вывода, рассмотрение которой дает возможность пояснить принципы использования некоторых устройств, входящих в микроЭВМ.

На рис. 1.7 приведена подробная диаграмма выполнения процедуры ввода-запоминания-вывода. Обратите внимание, что команды уже загружены в первые шесть ячеек памяти. Хранимая программа содержит следующую цепочку команд.

1. Ввести данные из порта ввода 1.
2. Запомнить данные в ячейке памяти 200.
3. Переслать данные в порт вывода 10.

Код символа *«А»*

**Рис. 1.7.** Диаграмма выполнения процедуры ввода-запоминания-вывода

*31*

В данной программе всего три команды, хотя на рис. 1.7 может показаться, что в памяти программ записано шесть команд. Это связано с тем, что команда обычно разбивается на части. Первая часть команды 1 в приведенной выше программе — команда ввода данных. Во второй части команды 1 указывается, откуда нужно ввести данные (из порта 1). Первая часть команды, предписывающая конкретное действие, называется *кодом операции* (КОП), а вторая часть — *операндом.* Код операции и операнд размещаются в отдельных ячейках памяти программ. На рис. 1.7 КОП хранится в ячейке 100, а код операнда — в ячейке 101 (порт 1); последний указывает, откуда нужно взять информацию.

В МП на рис. 1.7 выделены еще два новых блока — регистры: аккумулятор и регистр команд.

Рассмотрим прохождение команд и данных внутри микроЭВМ с помощью занумерованных кружков на диаграмме. Напомним, что микропроцессор — это центральный узел, управляющий перемещением всех данных и выполнением операций.

Итак, при выполнении типичной процедуры ввода-запоминания- вывода в микроЭВМ происходит следующая последовательность действий.

1. МП выдает адрес 100 на шину адреса. По шине управления поступает сигнал, устанавливающий память программ (конкретную микросхему) в режим считывания.
2. ЗУ программ пересылает первую команду («Ввести данные») по шине данных, и МП получает это закодированное сообщение. Команда помещается в регистр команд. МП декодирует (интерпретирует) полученную команду и определяет, что для команды нужен операнд.
3. МП выдает адрес 101 на ША; ШУ используется для перевода памяти программ в режим считывания.
4. Из памяти программ на ШД пересылается операнд «Из порта 1». Этот операнд находится в программной памяти в ячейке 101. Код операнда (содержащий адрес порта 1) передается по ШД к МП и направляется в регистр команд. МП теперь декодирует полную команду («Ввести данные из порта 1»).
5. МП, используя ША и ШУ, связывающие его с устройством ввода, открывает порт 1. Цифровой код буквы «А» передается в аккумулятор внутри МП и запоминается. Важно отметить, что при обработке каждой программной команды МП действует согласно микропроцедуре выборки-декодирования-исполнения.
6. МП обращается к ячейке 102 по ША. ШУ используется для перевода памяти программ в режим считывания.

*32*

1. Код команды «Запомнить данные» подается на ШД и пересылается в МП, где помещается в регистр команд.
2. МП дешифрирует эту команду и определяет, что для нее нужен операнд. МП обращается к ячейке памяти 103 и приводит в активное состояние вход считывания микросхем памяти программ.
3. Из памяти программ на ШД пересылается код сообщения «В ячейке памяти 200». МП воспринимает этот операнд и помещает его в регистр команд. Полная команда «Запомнить данные в ячейке памяти 200» выбрана из памяти программ и декодирована.
4. Теперь начинается процесс выполнения команды. МП пересылает адрес 200 на ША и активизирует вход записи, относящийся к памяти данных.
5. МП направляет хранящуюся в аккумуляторе информацию в память данных. Код буквы «А» передается по ШД и записывается в ячейку 200 этой памяти. Выполнена вторая команда. Процесс запоминания не разрушает содержимого аккумулятора. В нем по- прежнему находится код буквы «А».
6. МП обращается к ячейке памяти 104 для выбора очередной команды и переводит память программ в режим считывания.
7. Код команды вывода данных пересылается по ШД к МП, который помещает ее в регистр команд, дешифрирует и определяет, что нужен операнд.
8. МП выдает адрес 105 на ША и устанавливает память программ в режим считывания.
9. Из памяти программ по ШД к МП поступает код операнда «В порт 10», который далее помещается в регистр команд.
10. МП дешифрирует полную команду «Вывести данные в порт 10». С помощью ША и ШУ, связывающих его с устройством вывода, МП открывает порт 10, пересылает код буквы «А» (все еще находящийся в аккумуляторе) по ШД. Буква «А» выводится через порт 10 на экран дисплея.

В большинстве микропроцессорных систем (МПС) передача информации осуществляется способом, аналогичным рассмотренному выше. Наиболее существенные различия возможны в блоках ввода и вывода информации.

Подчеркнем еще раз, что именно микропроцессор является ядром системы и осуществляет управление всеми операциями. Его работа представляет последовательную реализацию микропроцедур вы- борки-дешифрации-исполнения. Однако фактическая последовательность операций в МПС определяется командами, записанными в памяти программ.

*33*

Таким образом, в МПС микропроцессор выполняет следующие функции:

* выборку команд программы из основной памяти;
* дешифрацию команд;
* выполнение арифметических, логических и других операций, закодированных в командах;
* управление пересылкой информации между регистрами и основной памятью, между устройствами ввода/вывода;
* отработку сигналов от устройств ввода/вывода, в том числе реализацию прерываний с этих устройств;
* управление и координацию работы основных узлов МП.

***Логическая структура микропроцессора***

Логическая структура микропроцессора, т.е. конфигурация составляющих микропроцессор логических схем и связей между ними, определяется функциональным назначением. Именно структура задает состав логических блоков микропроцессора и то, как эти блоки должны быть связаны между собой, чтобы полностью отвечать архитектурным требованиям. Срабатывание электронных блоков микропроцессора в определенной последовательности приводит к выполнению заданных архитектурой микропроцессора функций, т.е. к реализации вычислительных алгоритмов. Одни и те же функции можно выполнить в микропроцессорах со структурой, отличающейся набором, количеством и порядком срабатывания логических блоков. Различные структуры микропроцессоров, как правило, обеспечивают их различные возможности, в том числе и различную скорость обработки данных. Логические блоки микропроцессора с развитой архитектурой показаны на рис. 1.8.

При проектировании логической структуры микропроцессоров необходимо рассмотреть:

1. номенклатуру электронных блоков, необходимую и достаточную для реализации архитектурных требований;
2. способы и средства реализации связей между электронными блоками;
3. методы отбора если не оптимальных, то наиболее рациональных вариантов логических структур из возможного числа структур с отличающимся составом блоков и конфигурацией связей между ними.

При проектировании микропроцессора приводятся в соответствие внутренняя сложность кристалла и количество выводов корпуса. Относительный рост числа элементов по мере развития микроэлектронной технологии во много раз превышает относительное

*34*

увеличение числа выводов корпуса, поэтому проектирование БИС в виде конечного автомата, а не в виде набора схем, реализующих некоторый набор логических переключательных функций и схем памяти, дает возможность получить функционально законченные блоки и устройства ЭВМ.

**Рис. 1.8.** Общая логическая структура микропроцессора:

I — управляющая часть; II — операционная часть; БУПК — блок управ-
ления последовательностью команд; БУВОп — блок управления выполне-
нием операций; БУФКА — блок управления формированием кодов адресов;

БУВП — блок управления виртуальной памятью; БЗП — блок защиты
памяти; БУПРПр — блок управления прерыванием работы процессора;
БУВВ — блок управления вводом/выводом; РгСОЗУ — регистровое сверх-
оперативное запоминающее устройство; АЛБ — арифметико-логический
блок; БДА — блок дополнительной арифметики; БС — блок синхронизации

Использование микропроцессорных комплектов БИС позволяет создать микроЭВМ для широких областей применения вследствие программной адаптации микропроцессора к конкретной области применения: изменяя программу работы микропроцессора, изменяют функции информационно-управляющей системы. Поэтому за счет составления программы работы микропроцессоров в конкретных условиях работы определенной системы можно получить оптимальные характеристики последней.

Если уровень только программной «настройки» микропроцессоров не позволит получить эффективную систему, доступен следующий уровень проектирования — микропрограммный. За счет изменения содержимого ПЗУ или программируемой логической мат

15

рицы (ПЛМ) можно «настроиться» на более специфичные черты системы обработки информации. В этом случае частично за счет изменения микропрограмм затрагивается аппаратный уровень системы. Технико-экономические последствия здесь связаны лишь с ограниченным вмешательством в технологию изготовления управляющих блоков микроЭВМ.

Изменение аппаратного уровня информационно-управляющей микропроцессорной системы, включающего в себя функциональные БИС комплекта, одновременно с конкретизацией микропрограммного и программного уровней позволяет наилучшим образом удовлетворить требованиям, предъявляемым к системе.

Решение задач управления в конкретной системе чисто аппаратными средствами (аппаратная логика) дает выигрыш в быстродействии, однако приводит к сложностям при модификации системы. Микропроцессорное решение (программная логика) является более медленным, но более гибким решением, позволяющим развивать и модифицировать систему. Изменение технических требований к информационно-управляющей микропроцессорной системе ведет лишь к необходимости перепрограммирования работы микропроцессора. Именно это качество обеспечивает высокую логическую гибкость микропроцессоров, определяет возможность их широкого использования, а значит, и крупносерийного производства.