**КЛАССИФИКАЦИЯ И ТИПОВАЯ СТРУКТУРА МИКРОПРОЦЕССОРА**

1. **УСТРОЙСТВО И ПРИНЦИП ДЕЙСТВИЯ ЭВМ**

На разных этапах развития техники и технологии компьютеры назывались по-разному: арифметическо-логическое устройство (АЛУ), программируемое электронно-вычислительное устройство (ПЭВМ или ЭВМ), вычислительная машина, компьютер.

Основные принципы построения логической схемы и структура вычислительной машины, изложенные выдающемся математиком Джоном фон Нейманом, реализованы в первых двух поколениях ЭВМ. Классическая архитектура ЭВМ, построенная по принципу фон Неймана (фон – неймовская архитектура)и реализованная в вычислительных машинах первого и второго поколений, представлена на рис. 1. Она содержит следующие блоки:

* Арифметическо – логическое устройство (АЛУ), выполняющее арифметические и логические операции;
* Управляющее устройство (УУ), организующее процесс выполнения программ;
* Внешнее запоминающее устройство (ВЗУ), или память, для хранения программ и данных;
* Оперативное запоминающее устройство (ОЗУ);
* Устройство ввода и вывода информации (УВВ).

Внешнее запоминающее устройство (ВЗУ)

Управляющее устройство (УУ)

Арифметическо- логическое устройство (АЛУ)

Процессор

Устройство ввода

Устройство вывода

Оперативное запоминающее устройство (ОЗУ)

Рис. 1. Архитектура ЭВМ, реализующая принципы фон Неймана:

направление потоков информации;

 направление управляющих сигналов от процессора к узлам ЭВМ

Внешняя память отличается от устройств ввода и вывода тем, что данные в нее заносятся в виде, удобном компьютеру, но недоступном для непосредственного восприятия человеком. Например, накопитель на магнитных дисках относится к внешней памяти; устройством ввода является клавиатура, а монитор и принтер – устройства вывода. Причем если монитор можно отнести к устройствам отображения информации, то принтер – типичное печатающее устройство.

Взаимодействие основных устройств компьютера реализуется в определенной последовательности. В память компьютера вводится программа с помощью какого – либо внешнего устройства. Память компьютера состоит из некоторого числа пронумерованных ячеек. В каждой ячейке могут находиться или обрабатываемые данные, или инструкции программ. Номер (адрес) очередной ячейки памяти, из которой будет извлечена следующая команда программы, указывается специальным устройством – счетчиком команд в УУ.

**2. Классификация по составу и сложности команд**

Современная технология программирования ориентирована на языки высокого уровня (ЯВУ), главная цель которых — облегчить процесс программирования. Переход к ЯВУ, однако, породил серьезную проблему: сложные операторы, характерные для ЯВУ, существенно отличаются от простых машинных операций, реализуемых в большинстве вычислительных машин. Проблема получила название *семантического разрыва,* а ее следствием становится недостаточно эффективное выполнение программ на ВМ. Пытаясь преодолеть семантический разрыв, разработчики вычислительных машин в настоящее время выбирают один из трех подходов и, соответственно, один из трех типов АСК:

* архитектуру с полным набором команд: **CISC** (ComplexInstructionSetComputer);
* архитектуруссокращеннымнаборомкоманд: **RISC** (Reduced Instruction Set Computer);
* архитектуру с командными словами сверхбольшой длины: **VLIW** (VeryLongInstructionWord).

В вычислительных машинах типа CISC проблема семантического разрыва решается за счет расширения системы команд, дополнения ее сложными командами, семантически аналогичными операторам ЯВУ. Основоположником CISC-архитектуры считается компания IBM, которая начала применять данный подход с семейства машин IBM 360 и продолжает его в своих мощных современных универсальных ВМ, таких как IBM ES/9000. Аналогичный подход характерен и для компании Intel в ее микропроцессорах серии 8086 и Pentium.

Для CISC-архитектуры типичны:

* наличие в процессоре сравнительно небольшого числа регистров общего назначения;
* большое количество машинных команд, некоторые из них аппаратно реализуют сложные операторы ЯВУ;
* разнообразие способов адресации операндов;
* множество форматов команд различной разрядности;
* наличие команд, где обработка совмещается с обращением к памяти.

К типу CISC можно отнести практически все ВМ, выпускавшиеся до середины 1980-х годов, и значительную часть производящихся в настоящее время. Рассмотренный способ решения проблемы семантического разрыва вместе с тем ведет к усложнению аппаратуры ВМ, главным образом устройства управления, что, в свою очередь, негативно сказывается на производительности ВМ в целом. Это заставило более внимательно проанализировать программы, получаемые после компиляции с ЯВУ. Был предпринят комплекс исследований, в результате которых обнаружилось, что доля дополнительных команд, эквивалентных операторам ЯВУ, в общем объеме программ не превышает 10-20%, а для некоторых наиболее сложных команд даже 0,2%. В то же время объем аппаратных средств, требуемых для реализации дополнительных команд, возрастает весьма существенно. Так, емкость микропрограммной памяти при поддержании сложных команд может увеличиваться на 60%. Детальный анализ результатов упомянутых исследований привел к серьезному пересмотру традиционных решений, следствием чего стало появление *RISC-архитектуры.* Термин RISC впервые был использован Д. Паттерсоном и Д. Дитцелем в 1980 году. Идея заключается в ограничении списка команд ВМ наиболее часто используемыми простейшими командами, оперирующими данными, размещенными только в регистрах процессорах. Обращение к памяти допускается лишь с помощью специальных команд чтения и записи. Резко уменьшено количество форматов команд и способов указания адресов операндов. Сокращение числа форматов команд и их простота, использование ограниченного количества способов адресации, отделение операций обработки данных от операций обращения к памяти позволяет существенно упростить аппаратные средства ВМ и повысить их быстродействие. RISC-архитектура разрабатывалась таким образом, чтобы уменьшить TВИЧ за счет сокращения CPI и I/. Как следствие, реализация сложных команд за счет последовательности из простых, но быстрых RISC-команд оказывается не менее эффективной, чем аппаратный вариант сложных команд в CISC-архитектуре. Отметим, что в последних микропроцессорах фирмы Intel и AMD широко используются идеи, свойственные RISC-архитектуре, так что многие различия между CISC и RISC постепенно стираются.

Помимо CISC- и RISC-архитектур в общей классификации был упомянут еще один тип АСК — архитектура с командными словами сверхбольшой длины (VLIW). Концепция VLIW базируется на RISC-архитектуре, где несколько простых RISC-команд объединяются в одну сверхдлинную команду и выполняются параллельно. В плане АСК архитектура VLIW сравнительно мало отличается от RISC. Появился лишь дополнительный уровень параллелизма вычислений, в силу чего архитектуру VLIW логичнее адресовать не к вычислительным машинам, а к вычислительным системам. Таблица 2.1 позволяет оценить наиболее существенные различия в архитектурах типа CISC, RISC и VLIW.

Таблица 2.1. Сравнительная оценка CISC-, RISC- и VLIW-архитектур

|  |  |  |  |
| --- | --- | --- | --- |
| **Характеристика** | **CISC** | **RISC** | **VLIW** |
| Длина команды | Варьируется  | Единая  | Единая  |
| Расположение нолей и команде | Варьируется  | Неизменное  | Неизменное  |
| Количество регистров | Несколько (часто специализированных) | Много регистров общего назначения | Много регистров общего назначения |
| Доступ к памяти | Может выполняться как часть команд различных типов | Выполняется только специальными командами | Выполняется только специальными командами |

**3. СТРУКТУРА МИКРОПРОЦЕССОРА**

**Устройство управления**

*Устройство управления* является функционально наиболее сложным устройством ПК. Оно вырабатывает управляющие сигналы, поступающие по кодовым шинам инструкций во все блоки машины.

Упрощенная функциональная схема УУ показана на [рис2.](http://www.stu.ru/inform/glaves2/glava4/gl_4_3.html#ris_4_5) Здесь представлены:

*Регистр команд -* запоминающий регистр, в котором хранится код команды: код выполняемой операции и адреса операндов, участвующих в операции. Регистр команд расположен в интерфейсной части МП, в блоке регистров команд.

*Дешифратор операций -* логический блок, выбирающий в соответствии с поступающим из регистра команд кодом операции (КОП) один из множества имеющихся у него выходов.

*Постоянное запоминающее устройство микропрограмм -* хранит в своих ячейках управляющие сигналы (импульсы), необходимые для выполнения в блоках ПК операций обработки информации. Импульс по выбранному дешифратором операций в соответствии с кодом операции считывает из ПЗУ микропрограмм необходимую последовательность управляющих сигналов.

*Узел формирования адреса* (находится в интерфейсной части МП) - устройство, вычисляющее полный адрес ячейки памяти (регистра) по реквизитам, поступающим из регистра команд и регистров МПП.



**Рис. 2.** Укрупненная функциональная схема устройства управления

*Кодовые шины данных, адреса и инструкций* - часть внутренней интерфейсной шины микропроцессора. В общем случае УУ формирует управляющие сигналы для выполнения следующих основных процедур:

* выборки из регистра-счетчика адреса команды МПП адреса ячейки ОЗУ, где хранится очередная команда программы;
* выборки из ячеек ОЗУ кода очередной команды и приема считанной команды в регистр команд;
* расшифровки кода операции и признаков выбранной команды;
* считывания из соответствующих расшифрованному коду операции ячеек ПЗУ микропрограмм управляющих сигналов (импульсов), определяющих во всех блоках машины процедуры выполнения заданной операции, и пересылки управляющих сигналов в эти блоки;
* считывания из регистра команд и регистров МПП отдельных составляющих адресов операндов (чисел), участвующих в вычислениях, и формирования полных адресов операндов;
* выборки операндов (по сформированным адресам) и выполнения заданной операции обработки этих операндов;
* записи результатов операции в память;
* формирования адреса следующей команды программы.

**Арифметико-логическое устройство**

*Арифметико-логическое устройство* предназначено для выполнения арифметических и логических операций преобразования информации.

Функционально АЛУ ([рис.](http://www.stu.ru/inform/glaves2/glava4/gl_4_3.html#ris_4_6) 3) состоит обычно из двух регистров, сумматора и схем управления (местного устройства управления).

****

**Рис. 3**. Функциональная схема АЛУ

***Сумматор*** - вычислительная схема, выполняющая процедуру сложения поступающих на ее вход двоичных кодов; сумматор имеет разрядность двойного машинного слова.

***Регистры*** - быстродействующие ячейки памяти различной длины: регистр 1 (Рг1) имеет разрядность двойного слова, а регистр 2 (Рг2) - разрядность слова.

При выполнении операций в Рг1 помещается первое число, участвующее в операции, а по завершении операции - результат; в Рг2 - второе число, участвующее в операции (по завершении операции информация в нем не изменяется). Регистр 1 может и принимать информацию с кодовых шин данных, н выдавать информацию на них, регистр 2 только получает информацию с этих шин.

***Схемы управления*** принимают по кодовым шинам инструкций управляющие сигналы от устройства управления и преобразуют их в сигналы для управления работой регистров и сумматора АЛУ.

АЛУ выполняет арифметические операции (+, -, \*,:) только над двоичной информацией с запятой, фиксированной после последнего разряда, т.е. только над целыми двоичными числами.

Выполнение операций над двоичными числами с плавающей запятой и над двоично-кодированными десятичными числами осуществляется или с привлечением математического сопроцессора, или по специально составленным программам.

**Микропроцессорная память**

***Микропроцессорная память*** - память небольшой емкости, но чрезвычайно высокого быстродействия (время обращения к МПП, т.е. время, необходимое на поиск, запись или считывание информации из этой памяти, измеряется наносекундами - тысячными долями микросекунды).

Она предназначена для кратковременного хранения, записи и выдачи информации, непосредственно в ближайшие такты работы машины участвующей в вычислениях, МПП используется для обеспечения высокого быстродействия машины, ибо основная память не всегда обеспечивает скорость записи, поиска и считывания информации, необходимую для эффективной работы быстродействующего микропроцессора.

Микропроцессорная память состоит из быстродействующих *регистров* с разрядностью не менее машинного слова. Количество и разрядность регистров в разных микропроцессорах различны: от 14 двухбайтных регистров у МП 8086 до нескольких десятков регистров разной длины у МП Pentium.

***Регистры микропроцессора*** делятся на регистры общего назначения и специальные.

*Специальные регистры* применяются для хранения различных адресов (адреса команды, например), признаков результатов выполнения операций и режимов работы ПК (регистр флагов, например) и др.

*Регистры общего назначения* являются универсальными и могут использоваться для хранения любой информации, но некоторые из них тоже должны быть обязательно задействованы при выполнении ряда процедур.

**Интерфейсная часть микропроцессора**

Интерфейсная часть МП предназначена для связи и согласования МП с системной шиной ПК, а также для приема, предварительного анализа команд выполняемой программы и формирования полных адресов операндов и команд.

Интерфейсная часть включает в свой состав адресные регистры МПП, узел формирования адреса, блок регистров команд, являющийся буфером команд в МП, внутреннюю интерфейсную шину МП и схемы управления шиной и портами ввода-вывода.

***Порты ввода-вывода*** - это пункты системного интерфейса ПК, через которые МП обменивается информацией с другими устройствами. Всего портов у МП может быть 65536. Каждый порт имеет адрес - номер порта, соответствующий адресу ячейки памяти, являющейся частью устройства ввода-вывода, использующего этот порт, а не частью основной памяти компьютера.

Порт устройства содержит аппаратуру сопряжения и два регистра памяти - для обмена данными и обмена управляющей информацией. Некоторые внешние устройства используют и основную память для хранения больших объемов информации, подлежащей обмену. Многие стандартные устройства (НЖМД, НГМД, клавиатура, принтер, сопроцессор и др.) имеют постоянно закрепленные за ними порты ввода-вывода.

***Схема управления шиной и портами*** выполняет следующие функции:

* формирование адреса порта и управляющей информации для него (переключение порта на прием или передачу и др.);
* прием управляющей информации от порта, информации о готовности порта и его состоянии;
* организацию сквозного канала в системном интерфейсе для передачи данных между портом устройства ввода-вывода и МП.

Схема управления шиной и портами использует для связи с портами кодовые шины инструкций, адреса и данных системной шины: при доступе к порту МП посылает сигнал по КШИ, который оповещает все устройства ввода-вывода, что адрес на КША является адресом порта, а затем посылает и сам адрес порта. То устройство, адрес порта которого совпадает, дает ответ о готовности, после чего по КШД осуществляется обмен данными.