

3. Архитектура микропроцессора

3.1. Понятие архитектуры микропроцессора

С точки зрения пользователя (разработчика автоматических систем) при выборе МП для решения конкретной задачи целесообразно располагать некоторыми обобщенными или комплексными характеристиками возможностей МП, т. е. воспринимать его как нечто цельное, имеющее вполне определенные потребительские качества (свойства и характеристики). В конечном итоге разработчик нуждается в уяснении и понимании лишь тех компонентов МП и МПС, которые явно отражаются в программах и (или) должны быть учтены при разработке и выполнении программ: число и имена программно-доступных регистров; разрядность машинного слова; система команд; доступный размер и адреса ОЗУ; быстродействие МП; схему обработки прерываний; способы адресации ОЗУ и внешних устройств. Совокупность таких сведений представляет определенную модель МП (МПС) с точки зрения пользователя (разработчика МПАС).

Указанные выше характеристики и свойства определяются понятием архитектуры МП (МПС, МЭВМ).

Архитектура МП – это его логическая организация, рассматриваемая с точки зрения пользователя; она определяет возможности МП по аппаратной, программной и микропрограммной реализации функций, необходимых для построения МПС и МПАС [2, 3].

Понятие архитектуры МП отражает:

- структуру, т. е. совокупность компонентов, составляющих МП, и связей между ними;
- способы представления и форматы данных;
- способы обращения ко всем доступным для пользователя (программно-доступным) элементам структуры (адресация к регистрам, ячейкам оперативной и постоянной памяти, внешним устройствам);
- набор операций, выполняемых МП, т. е. система команд МП;
- характеристики управляющих слов и сигналов, вырабатываемых микропроцессором и поступающих в МП извне;
- реакцию на внешние сигналы (схема обработки прерываний и т. д.) и другие характеристики. Ниже будут рассмотрены некоторые элементы архитектуры.

3.2. Основные характеристики микропроцессора

Микропроцессор характеризуется:

- 1) тактовой частотой, определяющей максимальное время выполнения переключения элементов в ЭВМ;
- 2) разрядностью, т. е. максимальным числом одновременно обрабатываемых двоичных разрядов;
- 3) архитектурой.

Разрядность МП обозначается $m/n/k$ и включает:

m – разрядность внутренних регистров, определяет принадлежность к тому или иному классу процессоров;

n – разрядность шины данных, определяет скорость передачи информации;

k – разрядность шины адреса, определяет размер адресного пространства.

Например, МП i8088 характеризуется значениями $m/n/k = 16/8/20$;

Понятие архитектуры микропроцессора включает в себя систему команд и способы адресации, возможность совмещения выполнения команд во времени, наличие дополнительных устройств в составе микропроцессора, принципы и режимы его работы. 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 [1, 2, 3].

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

В общем случае под архитектурой ЭВМ понимается абстрактное представление машины в терминах основных функциональных модулей, языка ЭВМ, структуры данных [1, 2, 3].

3.3. Типы архитектур микропроцессоров

Все микропроцессоры можно разделить на следующие группы:

- МП с гарвардской архитектурой;
- МП с фоннеймановской архитектурой;
- МП типа CISC (Complex Instruction Set Computing) с полным набором команд;
- МП типа RISC (Reduced Instruction Set Computing) с сокращенным набором команд;
- МП типа MISC (Minimum Instruction Set Computing) с минимальным набором команд и весьма высоким быстродействием (в настоящее время эти модели находятся в стадии разработки).

3.3.1. Гарвардская и фоннеймановская архитектуры микропроцессора

Архитектуры микропроцессоров различаются по использованию памяти. Наибольшее распространение получили:

- гарвардская архитектура;
- архитектура фон Неймана.

Гарвардская архитектура предполагает раздельное использование памяти программ и данных. Обычно такую архитектуру используют для повышения быстродействия системы за счёт разделения путей доступа к памяти программ и данных. Большинство специализированных микропроцессоров (особенно микроконтроллеры) имеют данную архитектуру.

Антипод гарвардской – архитектура фон Неймана – предполагает хранение программ и данных в общей памяти и наиболее характерна для микропроцессоров, ориентированных на использование в компьютерах. Примером могут служить микропроцессоры семейства x86.

3.3.2. Микропроцессоры типа CISC

Термин *CISC* означает сложную систему команд и является аббревиатурой английского определения Complex Instruction Set Computer.

Большинство современных ПК типа IBM PC (International Business Machine) используют МП типа CISC, характеристики наиболее распространенных из них приведены в табл. 1.

Таблица 1

Характеристики наиболее распространенных CISC МП

Мо- дель МП	Разрядность, бит		Так- товая частота, МГц	Адресное пространство, байт	Число ко- манд	Число элемен- тов	Год выпус- ка
	данных	адреса					
4004	4	4	4,77	4*103	45	2300	1971
8080	8	8	4,77	64*103		10000	1974
8086	16	16	4,77 и 8	106"	134	70000	1982
8088	8, 16	16	4,77 и 8	106	134	70000	1981
80186	16	20	8 и 10	106		140000	1984
80286	16	24	10–33	4*106 (вирту- альное 109)		180000	1985
80386	32	32	25–50	16*106 (вир- туальное 4*109)	240	275000	1987
80486	32	32	33–100	16*106 (вир- туальное 4*109)	240	1,2x106	1989
Pentium	64	32	50–150	4*109	240	3,1*106	1993
Pentium Pro	64	32	66–200	4*109	240	5,5*106	1995

Отметим некоторые характеристики МП:

- начиная с МП 80386 используется конвейерное выполнение команд – одновременное выполнение разных тактов последовательных команд в разных частях МП при непосредственной передаче результатов из одной части МП в другую. Конвейерное выполнение команд увеличивает эффективное быстродействие ПК в 2–3 раза;
- начиная с МП 80286 предусматривается возможность работы в вычислительной сети;
- начиная с МП 80286 имеется возможность многозадачной работы (многопрограммность) и сопутствующая ей защита памяти;
- начиная с МП 80386 обеспечивается поддержка режима системы виртуальных машин, т. е. такого режима многозадачной работы, при котором в одном МП моделируется как бы несколько компьютеров, работающих параллельно и имеющих разные операционные системы;
- начиная с МП 80286 микропроцессоры могут работать в двух режимах: реальном (Real mode) и защищенном (Protected mode). В ре-

альном режиме имитируется (эмулируется) работа МП 8086, естественно, однозадачная. В защищенном режиме возможна многозадачная работа с непосредственным доступом к расширенной памяти (см. подразд. 4.5) и с защитой памяти, отведенной задачам, от посторонних обращений.

Микропроцессоры 80586 (P5) более известны по их товарной марке Pentium, которая запатентована фирмой Intel (МП 80586 других фирм имеют иные обозначения: K5 у фирмы AMD, M 1 у фирмы Cyrix и др.).

Эти микропроцессоры имеют пятиступенную конвейерную структуру, обеспечивающую многократное совмещение тактов выполнения последовательных команд, и КЭШ-буфер для команд условной передачи управления, позволяющий предсказывать направление ветвления программ; по эффективному быстродействию они приближаются к RISC МП, выполняющим каждую команду как бы за один такт. Pentium имеют 32-разрядную адресную шину и 64-разрядную шину данных. Обмен данными с системой может выполняться со скоростью 1 Гб/с.

У всех МП Pentium имеется встроенная КЭШ-память, отдельно для команд, отдельно для данных; имеются специализированные конвейерные аппаратные блоки сложения, умножения и деления, значительно ускоряющие выполнение операций с плавающей запятой.

Микропроцессоры Pentium Pro. В сентябре 1995 г. прошли презентацию и выпущены МП 80686 (P6), торговая марка Pentium Pro. Благодаря новым схемотехническим решениям они обеспечивают для ПК более высокую производительность. Часть этих новшеств может быть объединена понятием динамическое исполнение (dynamic execution), что в первую очередь означает наличие 14-ступенной суперконвейерной структуры (superpipelining), предсказания ветвлений программы при условных передачах управления (branch prediction) и исполнение команд по предполагаемому пути ветвления (speculative execution).

КЭШ-память емкостью 256–512 кб – обязательный атрибут высокопроизводительных систем на процессорах Pentium. Однако у них встроенная КЭШ-память имеет небольшую емкость (16 кб), а основная ее часть находится вне процессора на материнской плате. Поэтому обмен данными с ней происходит не на внутренней частоте МП, а на частоте тактового генератора, которая обычно в 2–3 раза ниже, что снижает общее быстродействие компьютера. В МП Pentium Pro КЭШ-память емкостью 256–512 кб находится в самом микропроцессоре.

Микропроцессоры OverDrive. Интерес представляют также недавно разработанные МП OverDrive, по существу являющиеся своеобраз-

ными сопроцессорами, обеспечивающими для МП 80486 режимы работы и эффективное быстроедействие, характерные для МП Pentium. Появились МП OverDrive, улучшающие характеристики и микропроцессоров Pentium.

3.3.3. Микропроцессоры типа RISC

Термин *RISC* означает сокращённую систему команд и происходит от английского Reduced Instruction Set Computer.

Микропроцессоры типа RISC содержат набор только простых, чаще всего встречающихся в программах команд. При необходимости выполнения более сложных команд в микропроцессоре производится их автоматическая сборка из простых. В этих МП на выполнение каждой простой команды за счет их наложения и параллельного выполнения тратится 1 машинный такт (на выполнение даже самой короткой команды из системы CISC обычно тратится 4 такта).

Некоторые микропроцессоры типа RISC: ARM (на его основе выпускались ПК IBM PC RT) – один из первых 32-разрядных RISC микропроцессоров, имеющий 118 различных команд. Современные RISC микропроцессоры (80860, 80960, 80870, Power PC) являются 64-разрядными при быстроедействии до 150 млн оп./с. Микропроцессоры Power PC (Performance Optimized With Enhanced RISC PC) весьма перспективны и уже сейчас широко применяются в машинах-серверах и в ПК типа Macintosh.

Микропроцессоры типа RISC имеют очень высокое быстроедействие, но программно не совместимы с CISC-процессорами: при выполнении программ, разработанных для ПК типа IBM PC, они могут лишь эмулировать (моделировать, имитировать) МП типа CISC на программном уровне, что приводит к резкому уменьшению их эффективной производительности.

Однако, несмотря на широкую распространённость этих понятий, необходимо признать, что сами названия не отражают главного различия между системами команд CISC и RISC. *Основная идея RISC-архитектуры* – это тщательный подбор таких комбинаций кодов операций, которые можно было бы выполнить за один такт тактового генератора. Основной выигрыш от такого подхода – резкое упрощение аппаратной реализации ЦП и возможность значительно повысить его производительность. Все команды работают с операндами и имеют одинаковый формат. Обращение к памяти выполняется с помощью специальных команд загрузки регистра и записи. Простота структуры и небольшой набор команд позволяет реализовать полностью их аппа-

ратное выполнение и эффективный конвейер при небольшом объеме оборудования. Арифметику RISC-процессоров отличает высокая степень дробления конвейера. Этот прием позволяет увеличить тактовую частоту (значит и производительность) компьютера. Чем более элементарные действия выполняются в каждой фазе работы конвейера, тем выше частота его работы. RISC-процессоры с самого начала ориентированы на реализацию всех возможностей ускорения арифметических операций, поэтому их конвейеры обладают значительно более высоким быстродействием, чем в CISC-процессорах. Поэтому RISC-процессоры в 2–4 раза быстрее имеющих ту же тактовую частоту CISC-процессоров с обычной системой команд и высокопроизводительней, несмотря на больший объем программ, на (30 %). Дейв Паттерсон и Карло Секуин сформулировали 4 основных принципа RISC:

1. Любая операция должна выполняться за один такт, вне зависимости от ее типа.

2. Система команд должна содержать минимальное количество наиболее часто используемых простейших инструкций одинаковой длины.

3. Операции обработки данных реализуются только в формате “регистр–регистр” (операнды выбираются из оперативных регистров процессора, и результат операции записывается также в регистр; а обмен между оперативными регистрами и памятью выполняется только с помощью команд чтения/записи).

4. Состав системы команд должен быть “удобен” для компиляции операторов языков высокого уровня.

Микропроцессоры с архитектурой CISC (Complex Instruction Set Computers) – архитектура вычислений с полной системой команд. Реализующие на уровне машинного языка комплексные наборы команд различной сложности (от простых, характерных для микропроцессора первого поколения, до команд значительной сложности, характерных для современных 32-разрядных микропроцессоров типа 80486, 68040 и др.).

3.4. Архитектурно-функциональные принципы построения ЭВМ

Основополагающие принципы цифровой ЭВМ (Джон фон Нейман, 1946 г.):

1. Принцип использования двоичной системы исчисления для представления информации в ЭВМ.
2. Принцип программного управления ЭВМ.
3. Принцип условного перехода.
4. Принцип хранимой программы.
5. Принцип иерархичности запоминающих устройств ЭВМ.

В цифровых вычислительных устройствах все операции (математические и логические) производятся над числами (величинами), представленными в кодовом выражении. Причём в данном случае все операции производятся в одном и том же узле. Большинство задач цифровой обработки может быть решено при помощи устройств, математическая структура которых изображена на рис. 2.

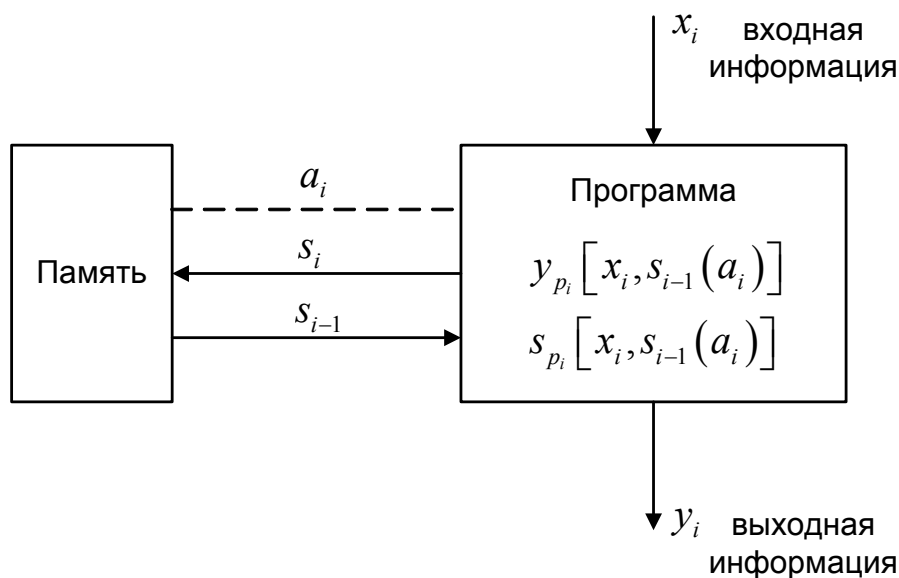


Рис. 2. Схема конечного автомата фон Неймана

На вход данного устройства (рис. 2) подаётся массив входной информации

$$x_i = \{a_i, p_i, x_i^{(0)}, x_i^{(1)}, \dots, x_i^{(n)}\}, \quad (1)$$

где a_i – адресная часть массива, указывающая ячейку (ячейки) памяти, содержание которой $s_{i-1}(a_i)$ необходимо для обработки информации; p_i – номер программы (адрес программы), по которой должна быть произведена обработка; $x_i^{(0)}, \dots, x_i^{(n)}$ – множество входных величин, наиболее полно и точно отражающее входную информацию величины x .

В ходе обработки получается выходной массив

$$y_i = y_{p_i} [x_i, s_{i-1}(a_i)], \quad (2)$$

где y_i – выходной массив, который образуется путём обработки x_i программой p_i по адресу $s_{i-1}(a_i)$.

Выходной массив y_i представляется через множество выходных элементов

$$y_i = (y_i^{(0)}, \dots, y_i^{(n)}). \quad (3)$$

В память ЭВМ помещается массив

$$s_i = s_{p_i} [x_i, s_{i-1}(a_i)], \quad (4)$$

т. е. новый массив памяти.

3.5. Структура типовой ЭВМ (персонального компьютера)

3.5.1. Основные блоки ПК и их назначение

Архитектура компьютера обычно определяется совокупностью ее свойств, существенных для пользователя. Основное внимание при этом уделяется структуре и функциональным возможностям машины, которые можно разделить на основные и дополнительные.

Основные функции определяют назначение ЭВМ: обработка и хранение информации, обмен информацией с внешними объектами. *Дополнительные* функции повышают эффективность выполнения основных функций: обеспечивают эффективные режимы ее работы, диалог с пользователем, высокую надежность и др. Названные функции ЭВМ реализуются с помощью ее компонентов: аппаратных и программных средств.

Структура компьютера – это некоторая модель, устанавливающая состав, порядок и принципы взаимодействия входящих в нее компонентов.

Персональный компьютер (ПК) – это настольная или переносная ЭВМ, удовлетворяющая требованиям общедоступности и универсальности применения.

Достоинствами ПК являются:

- малая стоимость, находящаяся в пределах доступности для индивидуального покупателя;
- автономность эксплуатации без специальных требований к условиям окружающей среды;
- гибкость архитектуры, обеспечивающая ее адаптивность к разнообразным применениям в сфере управления, науки, образования, в быту;
- "дружественность" операционной системы и прочего программного обеспечения, обуславливающая возможность работы с ней пользователя без специальной профессиональной подготовки;
- высокая надежность работы (более 5 тыс. ч наработки на отказ).

Структура типовой ЭВМ представлена на рис. 3.

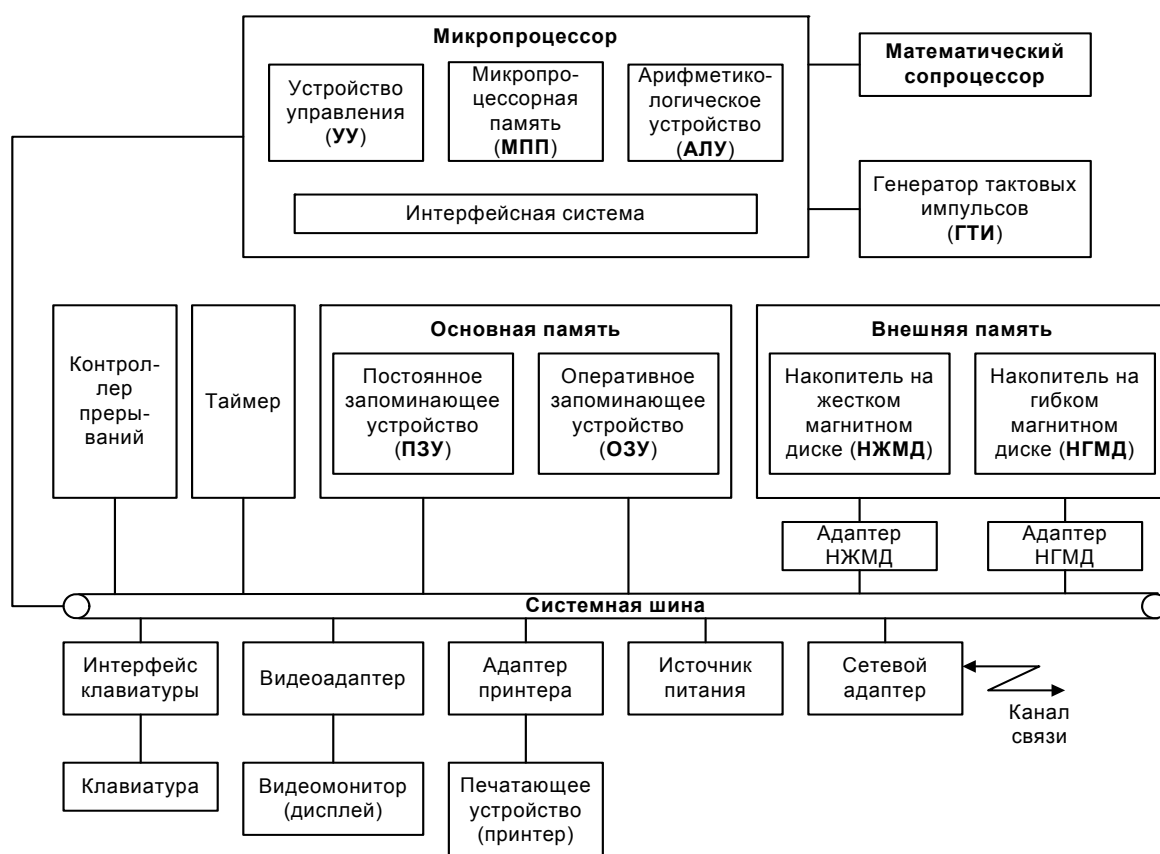


Рис. 3. Структура типовой ЭВМ

Микропроцессор (МП) — это центральный блок ПК, предназначенный для управления работой всех блоков машины и для выполнения арифметических и логических операций над информацией.

В состав микропроцессора входят:

- *устройство управления (УУ)* – формирует и подает во все блоки машины в нужные моменты времени определенные сигналы управления (управляющие импульсы), обусловленные спецификой выполняемой операции и результатами предыдущих операций; формирует адреса ячеек памяти, используемых выполняемой операцией, и передает эти адреса в соответствующие блоки ЭВМ. Опорную последовательность импульсов устройство управления получает от генератора тактовых импульсов;

- *арифметико-логическое устройство (АЛУ)* – предназначено для выполнения всех арифметических и логических операций над числовой и символьной информацией (в некоторых моделях ПК для ускорения выполнения операций к АЛУ подключается дополнительный математический сопроцессор);

- *микропроцессорная память (МПП)* – служит для кратковременного хранения, записи и выдачи информации, непосредственно используемой в вычислениях в ближайшие такты работы машины. МПП строится на регистрах и используется для обеспечения высокого быстродействия машины, т. к. основная память (ОП) не всегда обеспечивает скорость записи, поиска и считывания информации, необходимую для эффективной работы быстродействующего микропроцессора. **Регистры** – быстродействующие ячейки памяти различной длины (в отличие от ячеек ОП, имеющих стандартную длину 1 байт и более низкое быстродействие);

- *интерфейсная система микропроцессора* – реализует сопряжение и связь с другими устройствами ПК; включает в себя внутренний интерфейс МП, буферные запоминающие регистры и схемы управления портами ввода/вывода (ПВВ) и системной шиной. **Интерфейс** (interface) – совокупность средств сопряжения и связи устройств компьютера, обеспечивающая их эффективное взаимодействие. **Порт ввода/вывода** (I/O \approx Input/Output port) – аппаратура сопряжения, позволяющая подключить к микропроцессору другое устройство ПК.

Генератор тактовых импульсов генерирует последовательность электрических импульсов; частота генерируемых импульсов определяет тактовую частоту машины. Промежуток времени между соседними импульсами определяет время одного такта работы машины или просто такт работы машины.

Частота генератора тактовых импульсов является одной из основных характеристик персонального компьютера и во многом определяет скорость его работы, ибо каждая операция в машине выполняется за определенное количество тактов.

Системная шина – это основная интерфейсная система компьютера, обеспечивающая сопряжение и связь всех его устройств между собой.

Системная шина включает в себя:

- *кодovou шину данных (КШД)*, содержащую провода и схемы сопряжения для параллельной передачи всех разрядов числового кода (машинного слова) операнда;
- *кодovou шину адреса (КША)*, включающую провода и схемы сопряжения для параллельной передачи всех разрядов кода адреса ячейки основной памяти или порта ввода/вывода внешнего устройства;
- *кодovou шину инструкций (КШИ)*, содержащую провода и схемы сопряжения для передачи инструкций (управляющих сигналов, импульсов) во все блоки машины;
- *шину питания*, имеющую провода и схемы сопряжения для подключения блоков ПК к системе энергоснабжения.

Системная шина обеспечивает три направления передачи информации:

- 1) между микропроцессором и основной памятью;
- 2) между микропроцессором и портами ввода/вывода внешних устройств;
- 3) между основной памятью и портами ввода/вывода внешних устройств (в режиме прямого доступа к памяти).

Все блоки, а точнее их порты ввода/вывода, через соответствующие унифицированные разъемы (стыки) подключаются к шине единообразно: непосредственно или через *контроллеры (адаптеры)*. Управление системной шиной осуществляется микропроцессором либо непосредственно, либо, что чаще, через дополнительную микросхему – *контроллер шины*, формирующий основные сигналы управления.

Основная память предназначена для хранения и оперативного обмена информацией с прочими блоками машины. ОП содержит два вида запоминающих устройств: постоянное запоминающее устройство (ПЗУ) и оперативное запоминающее устройство (ОЗУ).

ПЗУ служит для хранения неизменяемой (постоянной) программной и справочной информации, позволяет оперативно только считывать хранящуюся в нем информацию (изменить информацию в ПЗУ нельзя).

ОЗУ предназначено для оперативной записи, хранения и считывания информации (программ и данных), непосредственно участвующей в информационно-вычислительном процессе, выполняемом ПК в текущий период времени. Главными достоинствами оперативной памяти являются ее высокое быстродействие и возможность обращения к каж-

дой ячейке памяти отдельно (прямой адресный доступ к ячейке), В качестве недостатка ОЗУ следует отметить невозможность сохранения информации в ней после выключения питания машины (энергозависимость).

Внешняя память относится к внешним устройствам ПК и используется для долговременного хранения любой информации, которая может когда-либо потребоваться для решения задач. В частности, во внешней памяти хранится все программное обеспечение компьютера. Внешняя память содержит разнообразные виды запоминающих устройств, но наиболее распространенными, имеющимися практически на любом компьютере, являются накопители на жестких (НЖМД) и гибких (НГМД) магнитных дисках.

Назначение этих накопителей – хранение больших объемов информации, запись и выдача хранимой информации по запросу в оперативное запоминающее устройство. Различаются НЖМД и НГМД лишь конструктивно, объемами хранимой информации и временем поиска, записи и считывания информации.

В качестве устройств внешней памяти используются также запоминающие устройства на кассетной магнитной ленте (стримеры), накопители на оптических дисках (CD-ROM – Compact Disk Read Only Memory – компакт-диск с памятью, только читаемой) и др.

Источник питания – это блок, содержащий системы автономного и сетевого энергопитания ПК.

Таймер – это внутримашинные электронные часы, обеспечивающие при необходимости автоматический съем текущего момента времени (год, месяц, часы, минуты, секунды и доли секунд). Таймер подключается к автономному источнику питания – аккумулятору – и при отключении машины от сети продолжает работать.

Внешние устройства – это важнейшая составная часть любого вычислительного комплекса. Достаточно сказать, что по стоимости ВУ иногда составляют 50–80 % стоимости всего ПК. От состава и характеристик ВУ во многом зависят возможность и эффективность применения ПК в системах управления и в народном хозяйстве в целом.

Внешние устройства ПК обеспечивают взаимодействие машины с окружающей средой; пользователями, объектами управления и другими ЭВМ. ВУ весьма разнообразны и могут быть классифицированы по ряду признаков. Так, по назначению можно выделить следующие виды ВУ:

- внешние запоминающие устройства (ВЗУ) или внешняя память ПК;
- диалоговые средства пользователя;

- устройства ввода информации;
- устройства вывода информации;
- средства связи и телекоммуникации.

Диалоговые средства пользователя включают в свой состав видеомониторы (дисплеи), реже – пультовые пишущие машинки (принтеры с клавиатурой) и устройства речевого ввода/вывода информации.

Видеомонитор (дисплей) – устройство для отображения вводимой и выводимой из ПК информации.

Устройства речевого ввода/вывода относятся к быстроразвивающимся средствам мультимедиа. Устройства речевого ввода – это различные микрофонные акустические системы, "звуковые мыши", например, со сложным программным обеспечением, позволяющим распознавать произносимые человеком буквы и слова, идентифицировать их и кодировать.

Устройства речевого вывода – это различные синтезаторы звука, выполняющие преобразование цифровых кодов в буквы и слова, воспроизводимые через громкоговорители (динамики) или звуковые колонки, подсоединенные к компьютеру.

К устройствам ввода информации относятся:

- *клавиатура* – устройство для ручного ввода числовой, текстовой и управляющей информации в ПК;
- *графические планшеты (диджитайзеры)* – для ручного ввода графической информации, изображений путем перемещения по планшету специального указателя (пера); при перемещении пера автоматически выполняются считывание координат его местоположения и ввод этих координат в ПК;
- *сканеры* – для автоматического считывания с бумажных носителей и ввода в ПК машинописных текстов, графиков, рисунков, чертежей; в устройстве кодирования сканера в текстовом режиме считанные символы после сравнения с эталонными контурами специальными программами преобразуются в коды ASCII, а в графическом режиме считанные графики и чертежи преобразуются в последовательности двумерных координат;
- *манипуляторы (устройства указания)*: джойстик – рычаг, мышь, трекбол – шар в оправе, световое перо и др. – для ввода графической информации на экран дисплея путем управления движением курсора по экрану с последующим кодированием координат курсора и вводом их в ПК;
- *сенсорные экраны* – для ввода отдельных элементов изображения, программ или команд с полиэкрана дисплея в ПК.

К устройствам вывода информации относятся:

- *принтеры* – печатающие устройства для регистрации информации на бумажный носитель (см. подразд. 4.5);
- *графопостроители (плоттеры)* – для вывода графической информации (графиков, чертежей, рисунков) из ПК на бумажный носитель; плоттеры бывают векторные с вычерчиванием изображения с помощью пера и растровые: термографические, электростатические, струйные и лазерные. По конструкции плоттеры подразделяются на планшетные и барабанные. Основные характеристики всех плоттеров примерно одинаковые: скорость вычерчивания – 100–1000 мм/с, у лучших моделей возможно цветное изображение и передача полутонов; наибольшая разрешающая способность и четкость изображения – у лазерных плоттеров, но они самые дорогие.

Устройства связи и телекоммуникации используются для связи с приборами и другими средствами автоматизации (согласователи интерфейсов, адаптеры, цифроаналоговые и аналого-цифровые преобразователи и т. п.) и для подключения ПК к каналам связи, к другим ЭВМ и вычислительным сетям (сетевые интерфейсные платы, "стыки", мультиплексоры передачи данных, модемы).

В частности, показанный на рис. 3 сетевой адаптер является внешним интерфейсом ПК и служит для подключения его к каналу связи для обмена информацией с другими ЭВМ, для работы в составе вычислительной сети. В глобальных сетях функции сетевого адаптера выполняет модулятор-демодулятор.

Многие из названных выше устройств относятся к условно выделенной группе – средствам мультимедиа.

Средства мультимедиа (multimedia – многосредовость) – это комплекс аппаратных и программных средств, позволяющих человеку общаться с компьютером, используя самые разные естественные для себя среды: звук, видео, графику, тексты, анимацию и др.

К средствам мультимедиа относятся устройства речевого ввода и вывода информации; широко распространенные уже сейчас сканеры (поскольку они позволяют автоматически вводить в компьютер печатные тексты и рисунки); высококачественные видео- (video-) и звуковые (sound-) платы, платы видеозахвата (videograbber), снимающие изображение с видеомagneтофона или видеокамеры и вводящие его в ПК; высококачественные акустические и видеовоспроизводящие системы с усилителями, звуковыми колонками, большими видеоэкранами. Но, пожалуй, еще с большим основанием к средствам мультимедиа относят внешние запоминающие устройства большой емкости на оптических дисках, часто используемые для записи звуковой и видеоинформации.

Стоимость компактных дисков (CD) при их массовом тиражировании невысокая, а учитывая их большую емкость (650 Мб, а новых типов – 1 Гб и выше), высокую надежность и долговечность, стоимость хранения информации на CD для пользователя оказывается несравнимо меньшей, нежели на магнитных дисках. Это уже привело к тому, что большинство программных средств самого разного назначения поставляется на CD. На компакт-дисках за рубежом организуются обширные базы данных, целые библиотеки; на CD представлены словари, справочники, энциклопедии; обучающие и развивающие программы по общеобразовательным и специальным предметам.

CD широко используются, например, при изучении иностранных языков, правил дорожного движения, бухгалтерского учета, законодательства вообще и налогового законодательства в частности. И все это сопровождается текстами и рисунками, речевой информацией и мультипликацией, музыкой и видео. В чисто бытовом аспекте CD можно использовать для хранения аудио- и видеозаписей, т. е. использовать вместо плеерных аудиокассет и видеокассет. Следует упомянуть, конечно, и о большом количестве программ, компьютерных игр, хранимых на CD.

Таким образом, CD-ROM открывает доступ к огромным объемам разнообразной и по функциональному назначению, и по среде воспроизведения информации, записанной на компакт-дисках.

Дополнительные схемы. К системной шине и к микропроцессору ПК, наряду с типовыми внешними устройствами, могут быть подключены и некоторые дополнительные платы с интегральными микросхемами, расширяющие и улучшающие функциональные возможности микропроцессора: математический сопроцессор, контроллер прямого доступа к памяти, сопроцессор ввода/вывода, контроллер прерываний и др.

Математический сопроцессор широко используется для ускоренного выполнения операций над двоичными числами с плавающей запятой, над двоично-кодированными десятичными числами, для вычисления некоторых трансцендентных, в том числе тригонометрических, функций. Математический сопроцессор имеет свою систему команд и работает параллельно (совмещенно во времени) с основным МП, но под управлением последнего. Ускорение операций происходит в десятки раз. Последние модели МП, начиная с МП 80486 DX, включают сопроцессор в свою структуру.

Контроллер прямого доступа к памяти освобождает МП от прямого управления накопителями на магнитных дисках, что существенно повышает эффективное быстродействие ПК. Без этого контрол-

лера обмен данными между ВЗУ и ОЗУ осуществляется через регистр МП, а при его наличии данные непосредственно передаются между ВЗУ и ОЗУ, минуя МП.

Сопроцессор ввода/вывода за счет параллельной работы с МП значительно ускоряет выполнение процедур ввода/вывода при обслуживании нескольких внешних устройств (дисплей, принтер, НЖМД, НГМД и др.); освобождает МП от обработки процедур ввода/вывода, в том числе реализует и режим прямого доступа к памяти.

Важнейшую роль играет в ПК контроллер прерываний.

Прерывание – временный останов выполнения одной программы в целях оперативного выполнения другой, в данный момент более важной (приоритетной) программы.

Прерывания возникают при работе компьютера постоянно [4]. Достаточно сказать, что все процедуры ввода/вывода информации выполняются по прерываниям. Например, прерывания от таймера возникают и обслуживаются контроллером прерываний 18 раз в секунду (естественно, пользователь их не замечает).

Контроллер прерываний обслуживает процедуры прерывания, принимает запрос на прерывание от внешних устройств, определяет уровень приоритета этого запроса и выдает сигнал прерывания в МП. МП, получив этот сигнал, приостанавливает выполнение текущей программы и переходит к выполнению специальной программы обслуживания того прерывания, которое запросило внешнее устройство. После завершения программы обслуживания восстанавливается выполнение прерванной программы. Контроллер прерываний является программируемым.

Элементы конструкции ПК

Конструктивно ПК выполнены в виде центрального системного блока, к которому через разъемы подключаются внешние устройства: дополнительные устройства памяти, клавиатура, дисплей, принтер и др.

Системный блок обычно включает в себя *системную плату*, блок питания, накопители на дисках, разъемы для дополнительных устройств и *платы расширения* с контроллерами – адаптерами внешних устройств.

На системной плате (часто ее называют материнской платой Mother Board), как правило, размещаются :

- микропроцессор;
- математический сопроцессор;
- генератор тактовых импульсов;

- блоки (микросхемы) ОЗУ и ПЗУ;
- адаптеры клавиатуры, НЖМД и НГМД;
- контроллер прерываний;
- таймер и др.

3.5.2. Внутримашинный системный интерфейс

Внутримашинный системный интерфейс – система связи и сопряжения узлов и блоков ЭВМ между собой – представляет собой совокупность электрических линий связи (проводов), схем сопряжения с компонентами компьютера, протоколов (алгоритмов) передачи и преобразования сигналов.

Существуют два варианта организации внутримашинного интерфейса.

1. *Многосвязный интерфейс*: каждый блок ПК связан с прочими блоками своими локальными проводами; многосвязный интерфейс применяется, как правило, только в простейших бытовых ПК.

2. *Односвязный интерфейс*: все блоки ПК связаны друг с другом через общую или системную шину.

В подавляющем большинстве современных ПК в качестве системного интерфейса используется системная шина. Структура и состав системной шины были рассмотрены ранее. Важнейшими функциональными характеристиками системной шины являются: количество обслуживаемых ею устройств и ее пропускная способность, т. е. максимально возможная скорость передачи информации. Пропускная способность шины зависит от ее разрядности (есть шины 8-, 16-, 32- и 64-разрядные) и тактовой частоты, на которой шина работает.

В качестве системной шины в разных ПК использовались и могут использоваться:

- *шины расширений* – шины общего назначения, позволяющие подключать большое число самых разнообразных устройств,
- *локальные шины*, специализирующиеся на обслуживании небольшого количества устройств определенного класса.

Сравнительные технические характеристики некоторых шин приведены в табл. 2.

Шины расширений

• Шина *Multibus1* имеет две модификации: PC/XT bus (Personal Computer eXtended Technology – ПК с расширенной технологией) и PC/AT bus (PC Advanced Technology – ПК с усовершенствованной технологией).

- Шина *PC/XT bus* – 8-разрядная шина данных и 20-разрядная шина адреса, рассчитанная на тактовую частоту 4,77 МГц; имеет 4 линии для аппаратных прерываний и 4 канала для прямого доступа в память (каналы DMA – Direct Memory Access). Шина адреса ограничивала адресное пространство микропроцессора величиной 1 Мб. Используется с МП 8086, 8088.

- Шина *PC/AT bus* – 16-разрядная шина данных и 24-разрядная шина адреса, рабочая тактовая частота до 8 МГц, но может использоваться и МП с тактовой частотой 16 МГц, так как контроллер шины может делить частоту пополам; имеет 7 линий для аппаратных прерываний и 4 канала DMA. Используется с МП 80286.

- Шине *ISA* (Industry Standard Architecture – архитектура промышленного стандарта) – 16-разрядная шина данных и 24-разрядная шина адреса, рабочая тактовая частота 8 МГц, но может использоваться и МП с тактовой частотой выше 8 МГц (коэффициент деления увеличен); по сравнению с шинами PC/XT и PC/AT увеличено количество линий аппаратных прерываний с 7 до 15 и каналов прямого доступа к памяти DMA с 7 до 11. Благодаря 24-разрядной шине адреса, адресное пространство увеличилось с 1 до 16 Мб. Теоретическая пропускная способность шины данных равна 16 Мб/с, но реально она ниже (около 4-5 Мб/с), ввиду ряда особенностей ее использования. С появлением 32-разрядных высокоскоростных МП шина ISA стала существенным препятствием увеличения быстродействия ПК.

- Шина *EISA* (Extended ISA) – 32-разрядная шина данных и 32-разрядная шина адреса, создана в 1989 г. Адресное пространство шины 4 Гб, пропускная способность 33 Мб/с. Причем скорость обмена по каналу МП–КЭШ–ОП определяется параметрами микросхем памяти, увеличено число разъемов расширений (теоретически может подключаться до 15 устройств, практически – до 10). Улучшена система прерываний, шина EISA обеспечивает автоматическое конфигурирование системы и управление DMA; полностью совместима с шиной ISA (есть разъем для подключения ISA), шина поддерживает многопроцессорную архитектуру вычислительных систем. Шина EISA весьма дорогая и применяется в скоростных ПК, сетевых серверах и рабочих станциях.

- Шина *MCA* (Micro Channel Architecture) – 32-разрядная шина, созданная фирмой IBM в 1987 г. Для машин PS/2, пропускная способность 76 Мб/с, рабочая частота 10-20 МГц. По своим прочим характеристикам близка к шине EISA, но не совместима ни с ISA, ни с EISA. Поскольку ЭВМ PS/2 не получили широкого распространения, в первую очередь ввиду отсутствия наработанного обилия прикладных программ, шина MCA также используется не очень широко.

Локальные шины

Современные вычислительные системы характеризуются:

- стремительным ростом быстродействия микропроцессоров (например, МП Pentium может выдавать данные со скоростью 528 Мб/с по 64-разрядной шине данных) и некоторых внешних устройств. Так, для отображения цифрового полноэкранного видео с высоким качеством необходима пропускная способность 22 Мб/с;
- появлением программ, требующих выполнения большого количества интерфейсных операций (например, программы обработки графики в Windows, работа в среде Multimedia).

В этих условиях пропускной способности шин расширения, обслуживающих одновременно несколько устройств, оказалось недостаточно для комфортной работы пользователей, ибо компьютеры стали подолгу "задумываться".

Разработчики интерфейсов пошли по пути создания локальных шин, подключаемых непосредственно к шине МП и работающих на тактовой частоте МП (но не на внутренней рабочей его частоте), а также обеспечивающих связь с некоторыми скоростными внешними по отношению к МП устройствами: основной и внешней памятью, видео-системами и др.

Сейчас существуют два основных стандарта универсальных локальных шин: VLB и PCI.

Шина *VLB* (VESA Local Bus – локальная шина VESA) – разработана в 1992 г. Ассоциацией стандартов видеооборудования (VESA – Video Electronics Standards Association), поэтому часто ее называют шиной VESA.

Шина VLB, по существу, является расширением внутренней шины МП для связи с видеоадаптером и реже с винчестером, платами Multimedia, сетевым адаптером. Разрядность шины – 32 бита, на подходе 64-разрядный вариант шины. Реальная скорость передачи данных по VLB – 80 Мб/с (теоретически достижимая – 132 Мб/с).

Недостатки шины:

- рассчитана на работу с МП 80386, 80486, пока не адаптирована для процессоров Pentium, Pentium Pro, Power PC;
- жесткая зависимость от тактовой частоты МП (каждая шина VLB рассчитана только на конкретную частоту);
- малое количество подключаемых устройств – к шине VLB могут подключаться только четыре устройства;
- отсутствует арбитраж шины – могут быть конфликты между подключаемыми устройствами.

Шина *PCI* (Peripheral Component Interconnect – соединение внешних устройств) – разработана в 1993 г. фирмой Intel.

Шина *PCI* является намного более универсальной, чем *VLB*, имеет свой адаптер, позволяющий ей настраиваться на работу с любым МП: 80486, Pentium, Pentium Pro, Celeron, Pentium 2 – Pentium 4, Power PC и др. Позволяет подключать 10 устройств самой разной конфигурации с возможностью автоконфигурирования, имеет свой "арбитраж", средства управления передачей данных.

Разрядность *PCI* – 32 бита с возможностью расширения до 64 бит, теоретическая пропускная способность 132 Мб/с, а в 64-битовом варианте – 263 Мб/с (реальная вдвое ниже).

Шина *PCI* хотя и является локальной, выполняет и многие функции шины расширения, в частности, шины расширения *ISA*, *EISA*, *MCA* (а она совместима с ними) при наличии шины *PCI* подключаются не непосредственно к МП (как это имеет место при использовании шины *VLB*), а к самой шине *PCI* (через интерфейс расширения).

Варианты конфигурации систем с шинами *VLB* и *PCI* показаны, соответственно, на рис. 4 и 5. Следует иметь в виду, что использование в ПК шин *VLB* и *PCI* возможно только при наличии соответствующей *VLB*- или *PCI*-материнской платы.

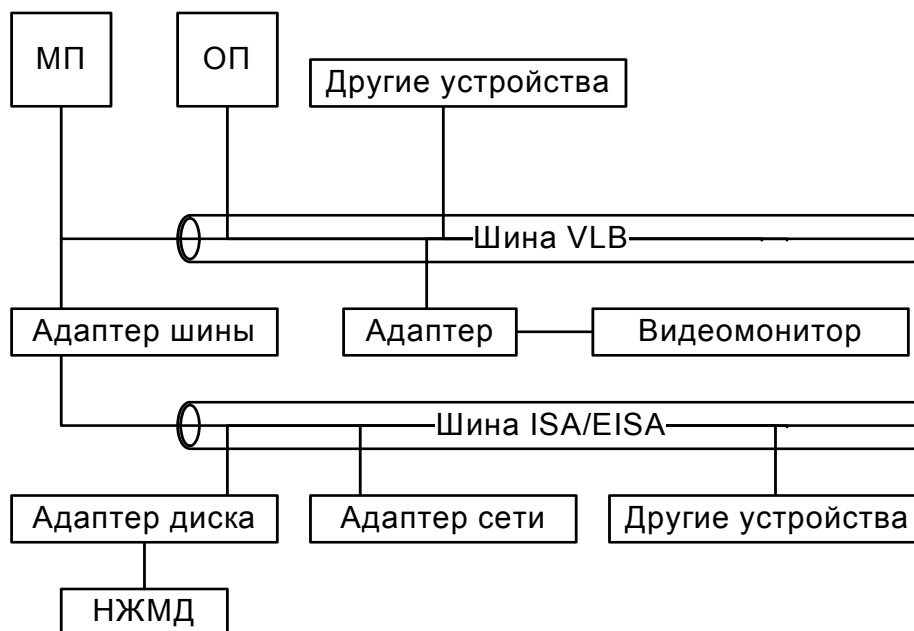


Рис. 4. Конфигурация системы с шиной *VLB*

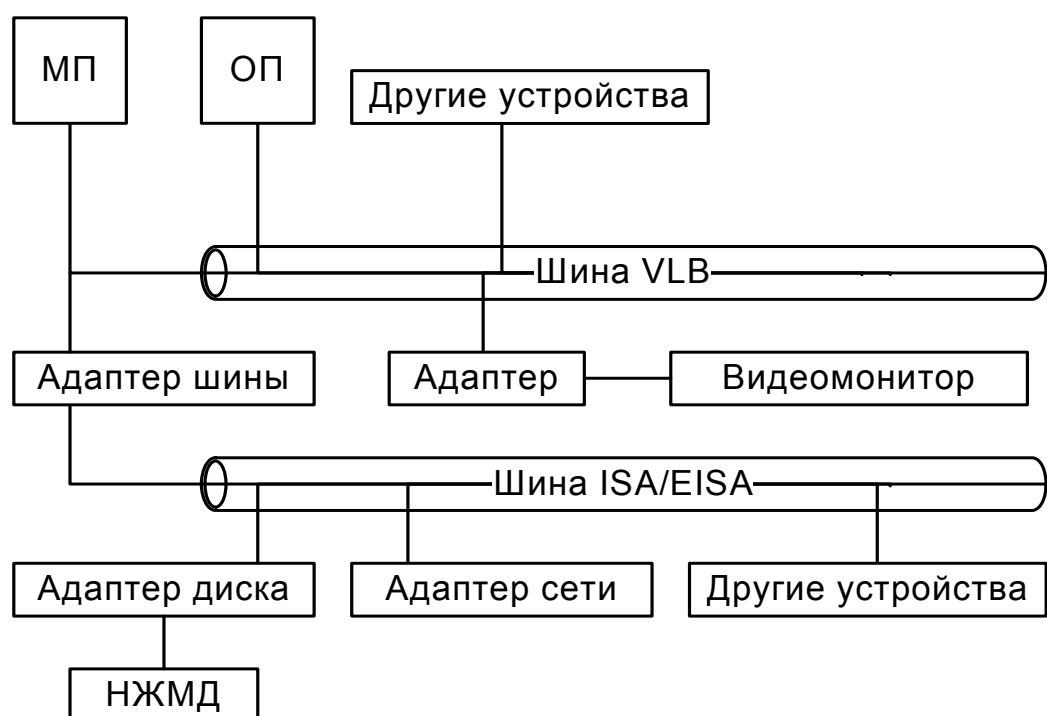


Рис. 5. Конфигурация системы с шиной PCI

Таблица 2

Основные характеристики шин

Параметр	ISA	EISA	MCA	VLB	PCI
Разрядность шины, бит	16	32	32;64	32;64	32; 64
Данных	24	32	32	32	32
Адреса					
Рабочая частота, МГц	8	8–33	10–20	до 33	до 33
Пропускная способность, Мб/с	4	33	76	132	132;264
теоретическая	2	8	20	80	50;100
практическая					
Число подключаемых устройств, шт.	6	15	15	4	10

3.5.3. Функциональные характеристики ПК

Основными характеристиками ПК являются:

1. Быстродействие, производительность, тактовая частота.

Единицами измерения быстродействия служат:

- МИПС (MIPS – Mega Instruction Per Second) – миллион операций над числами с фиксированной запятой (точкой);
- МФЛОПС (MFLOPS – Mega FLoating Operations Per Second) – миллион операций над числами с плавающей запятой (точкой);
- КОПС (KOPS – Kilo Operations Per Second) для низкопроизводительных ЭВМ – тысяча неких усредненных операций над числами;
- ГФЛОПС (GFLOPS – Giga FLoating Operations Per Second) – миллиард операций в секунду над числами с плавающей запятой (точкой).

Оценка производительности ЭВМ всегда приблизительная, ибо при этом ориентируются на некоторые усредненные или, наоборот, на конкретные виды операций. Реально при решении различных задач используются и различные наборы операций. Поэтому для характеристики ПК вместо производительности обычно указывают тактовую частоту, более объективно определяющую быстродействие машины, так как каждая операция требует для своего выполнения вполне определенного количества тактов. Зная тактовую частоту, можно достаточно точно определить время выполнения любой машинной операции.

2. Разрядность машины и кодовых шин интерфейса.

Разрядность – это максимальное количество разрядов двоичного числа, над которым одновременно может выполняться машинная операция, в том числе и операция передачи информации; чем больше разрядность, тем, при прочих равных условиях, будет больше и производительность ПК.

3. Типы системного и локальных интерфейсов.

Разные типы интерфейсов обеспечивают разные скорости передачи информации между узлами машины, позволяют подключать разное количество внешних устройств и различные их виды.

4. Емкость оперативной памяти.

Емкость оперативной памяти измеряется чаще всего в мегабайтах (Мб), реже в килобайтах (кб). Напоминаем: $1 \text{ Мб} = 1024 \text{ кб} = 1048576 \text{ байт}$.

Многие современные прикладные программы при оперативной памяти емкостью меньше 8 Мб просто не работают либо работают, но очень медленно.

Следует иметь в виду, что увеличение емкости основной памяти в 2 раза, помимо всего прочего, дает повышение эффективной производительности ЭВМ при решении сложных задач примерно в 1,7 раза.

5. Емкость накопителя на жестких магнитных дисках (винчестер).

Емкость винчестера измеряется обычно в мегабайтах или гигабайтах ($1 \text{ Гб} = 1024 \text{ Мб}$).

6. Тип и емкость накопителей на гибких магнитных дисках.

Сейчас применяются в основном накопители на гибких магнитных дисках, использующие дискеты диаметром 3,5 и 5,25 дюйма (1 дюйм = 25,4 мм). Первые имеют стандартную емкость 1,44 Мб, вторые – 1,2 Мб.

7. Виды и емкость КЭШ-памяти.

КЭШ-память – это буферная, не доступная для пользователя быст-родействующая память, автоматически используемая компьютером для ускорения операций с информацией, хранящейся в более медленно действующих запоминающих устройствах. Например, для ускорения операций с основной памятью организуется регистровая КЭШ-память внутри микропроцессора (КЭШ-память первого уровня) или вне микропроцессора на материнской плате (КЭШ-память второго уровня); для ускорения операций с дисковой памятью организуется КЭШ-память на ячейках электронной памяти.

Следует иметь в виду, что наличие КЭШ-памяти емкостью 256 кб увеличивает производительность ПК примерно на 20 %.

8. Тип видеомонитора (дисплея) и видеоадаптера.

9. Тип принтера.

10. Наличие математического сопроцессора.

Математический сопроцессор позволяет в десятки раз ускорить выполнение операций над двоичными числами с плавающей запятой и над двоично-кодированными десятичными числами.

11. Имеющееся программное обеспечение и вид операционной системы.

12. Аппаратная и программная совместимость с другими типами ЭВМ.

Аппаратная и программная совместимость с другими типами ЭВМ означает возможность использования на компьютере, соответственно, тех же технических элементов и программного обеспечения, что и на других типах машин.

13. Возможность работы в вычислительной сети.

14. Возможность работы в многозадачном режиме.

Многозадачный режим позволяет выполнять вычисления одновременно по нескольким программам (многопрограммный режим) или для нескольких пользователей (многопользовательский режим). Совмещение во времени работы нескольких устройств машины, возможное в таком режиме, позволяет значительно увеличить эффективное быстродействие ЭВМ.

15. Надежность.

Надежность – это способность системы выполнять полностью и правильно все заданные ей функции. Надежность ПК измеряется обычно средним временем наработки на отказ.

16. Стоимость.

17. Габариты и масса.

3.6. Система команд микропроцессора

Решение задач на ЭВМ реализуется программным способом, т. е. путем выполнения последовательно во времени отдельных операций над информацией, предусмотренных алгоритмом решения задачи.

Алгоритм – это точно определенная последовательность действий, которые необходимо выполнить над исходной информацией, чтобы получить решение задачи.

Алгоритм решения задачи, заданный в виде последовательности команд на языке вычислительной машины (в кодах машины), называется *машинной программой* [2, 6].

Проектирование системы команд оказывает влияние на структуру ЭВМ. Оптимальную систему команд иногда определяют как совокупность команд, которая удовлетворяет требованиям проблемно-ориентированных применений таким образом, что избыточность аппаратных и аппаратно-программных средств на реализацию редко используемых команд оказывается минимальной. В различных программах ЭВМ частота появления команд различна; например, по данным фирмы DEC в программах для ЭВМ семейства PDP-11 наиболее часто встречается команда передачи MOV(B), на ее долю приходится приблизительно 32 % всех команд в типичных программах. Систему команд следует выбирать таким образом, чтобы затраты на редко используемые команды были минимальными.

При наличии статистических данных можно разработать (выбрать) ЭВМ с эффективной системой команд. Одним из подходов к достижению данной цели является разработка команд длиной в одно слово и кодирование их таким образом, чтобы разряды таких коротких команд использовать оптимально, что позволит сократить время реализации программы и ее длину.

Другим подходом к оптимизации системы команд является использование микроинструкций. В этом случае отдельные биты или группы бит команды используются для кодирования нескольких элементарных операций, которые выполняются в одном командном цикле.

Эти элементарные операции не требуют обращения к памяти, а последовательность их реализации определяется аппаратной логикой.

Сокращение времени выполнения программ и емкости памяти достигается за счет увеличения сложности логики управления.

Важной характеристикой команды является ее формат, определяющий структурные элементы команды, каждый из которых интерпретируется определенным образом при ее выполнении. Среди таких элементов (полей) команды выделяют следующие: *код операции*, определяющий выполняемое действие; *адрес* ячейки памяти, регистра процессора, внешнего устройства; режим адресации; операнд при использовании непосредственной адресации; код анализируемых признаков для команд условного перехода [2, 6].

Классификация команд по основным признакам представлена на рис. 6.

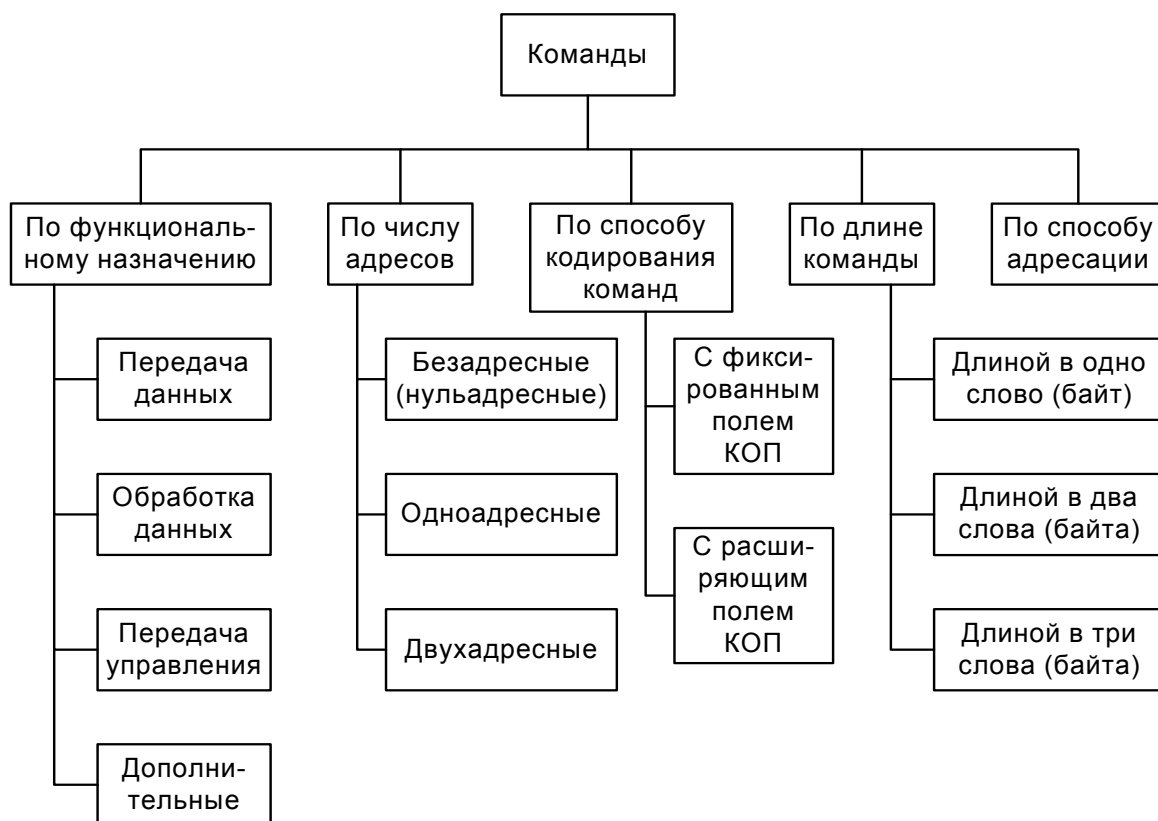


Рис. 6. Классификация команд

Важнейшим структурным элементом формата любой команды является *код операции* (КОП), определяющей действие, которое должно быть выполнено. Большое число КОП в процессоре очень важно, так как аппаратная реализация команд экономит память и время. Но при

выборе ЭВМ необходимо концентрировать внимание на полноте операций с конкретными типами данных, а не только на числе команд, на доступных режимах адресации. Число бит, отводимое под КОП, является функцией полного набора реализуемых команд [3, 6].

При использовании фиксированного числа бит под КОП для кодирования всех команд необходимо в поле КОП выделить m двоичных разрядов. Однако, учитывая ограниченную длину слова мини- и микроЭВМ, различное функциональное назначение команд, источники и приемники результатов операций, а также то, что не все команды содержат адресную часть для обращения к памяти и периферийным устройствам, в малых ЭВМ для кодирования команд широко используется принцип кодирования с переменным числом бит под поле КОП для различных групп команд.

В некоторых командах необходим только один операнд и они называются *однооперандными* (или *одноадресными*) командами в отличие от *двухоперандных* (или *двухадресных*), в которых требуются два операнда. При наличии двух операндов командой обычно изменяется только один из них. Так как информация берется только из одной ячейки, эту ячейку называются *источником*; ячейка, содержимое которой изменяется, называется *приемником*.

3.6.1. Структура и виды команд

Команда машинной программы (иначе, машинная команда) – это элементарная инструкция машине, выполняемая ею автоматически без каких-либо дополнительных указаний и пояснений.

Машинная команда состоит из двух частей: операционной и адресной.

Операционная часть команды – это группа разрядов в команде, предназначенная для представления кода операции машины.

Адресная часть команды – это группа разрядов в команде, в которой записываются коды адреса (адресов) ячеек памяти машины, предназначенных для оперативного хранения информации или иных объектов, задействованных при выполнении команды. Часто эти адреса называются адресами операндов, т. е. чисел, участвующих в операции.

По количеству адресов, записываемых в команде, команды делятся на безадресные, одно-, двух- и трехадресные.

Типовая структура трехадресной команды [2, 3, 6]:

КОП a_1, a_2, a_3 ,

где a_1 и a_2 – адреса ячеек (регистров), где расположены, соответственно, первое и второе числа, участвующие в операции;

a3 – адрес ячейки (регистра), куда следует поместить число, полученное в результате выполнения операции.

Типовая структура двухадресной команды:

КОП a1, a2,

где a1 – это обычно адрес ячейки (регистра), где хранится первое из чисел, участвующих в операции, и куда после завершения операции должен быть записан результат операции;

a2 – обычно адрес ячейки (регистра), где хранится второе участвующее в операции число.

Типовая структура одноадресной команды:

КОП a1,

где a1 в зависимости от модификации команды может обозначать либо адрес ячейки (регистра), где хранится одно из чисел, участвующих в операции, либо адрес ячейки (регистра), куда следует поместить число результат операции.

Безадресная команда содержит только код операции, а информация для нее должна быть заранее помещена в определенные регистры машины (безадресные команды могут использоваться только совместно с командами другой адресности).

Пример. Поступила представленная на языке символического кодирования команда:

СЛ 0103, 5102.

Такую команду следует расшифровать так: "сложить число, записанное в ячейке 0103 памяти, с числом, записанным в ячейке 5102, а затем результат (т. е. сумму) поместить в ячейку 0103".

3.6.2. Особенности программного и микропрограммного управления

В микропроцессорах используют два метода выработки совокупности функциональных управляющих сигналов: программный и микропрограммный [2, 3, 6].

Выполнение операций в машине сводится к элементарным преобразованиям информации (передача информации между узлами в блоках, сдвиг информации в узлах, логические поразрядные операции, проверка условий и т. д.) в логических элементах, узлах и блоках под воздействием функциональных управляющих сигналов блоков (устройств) управления. Элементарные преобразования, неразложимые на более простые, выполняются в течение одного такта сигналов синхронизации и называются микрооперациями.

В аппаратных (схемных) устройствах управления каждой операции соответствует свой набор логических схем, вырабатывающих определенные функциональные сигналы для выполнения микроопераций в определенные моменты времени. При этом способе построения устройства управления реализация микроопераций достигается за счет однажды соединенных между собой логических схем, поэтому ЭВМ с аппаратным устройством управления называют ЭВМ с жесткой логикой управления. Это понятие относится к фиксации системы команд в структуре связей ЭВМ и означает практическую невозможность каких-либо изменений в системе команд ЭВМ после ее изготовления.

При микропрограммной реализации устройства управления в состав последнего вводится ЗУ, каждый разряд выходного кода которого определяет появление определенного функционального сигнала управления. Поэтому каждой микрооперации ставится в соответствие свой информационный код – микрокоманда. Набор микрокоманд и последовательность их реализации обеспечивают выполнение любой сложной операции. Набор микроопераций называют микропрограммами. Способ управления операциями путем последовательного считывания и интерпретации микрокоманд из ЗУ (наиболее часто в виде микропрограммного ЗУ используют быстродействующие программируемые логические матрицы), а также использования кодов микрокоманд для генерации функциональных управляющих сигналов называют микропрограммным, а микроЭВМ с таким способом управления – микропрограммными или с хранимой (гибкой) логикой управления.

К микропрограммам предъявляют требования функциональной полноты и минимальности. Первое требование необходимо для обеспечения возможности разработки микропрограмм любых машинных операций, а второе связано с желанием уменьшить объем используемого оборудования. Учет фактора быстродействия ведет к расширению микропрограмм, поскольку усложнение последних позволяет сократить время выполнения команд программы.

Преобразование информации выполняется в универсальном арифметико-логическом блоке микропроцессора. Он обычно строится на основе комбинационных логических схем.

Для ускорения выполнения определенных операций вводятся дополнительно специальные операционные узлы (например, узел циклического сдвига). Кроме того, в состав микропроцессорного комплекта (МПК) БИС вводятся специализированные оперативные блоки арифметических расширителей.

Операционные возможности микропроцессора можно расширить за счет увеличения числа регистров. Если в регистровом буфере закре-

пление функций регистров отсутствует, то их можно использовать как для хранения данных, так и для хранения адресов. Подобные регистры микропроцессора называются регистрами общего назначения (РОН). По мере развития технологии реально осуществлено изготовление в микропроцессоре 16, 32 и более регистров.

В целом же, принцип микропрограммного управления (ПМУ) включает следующие позиции:

1) любая операция, реализуемая устройством, является последовательностью элементарных действий – микроопераций;

2) для управления порядком следования микроопераций используются логические условия;

3) процесс выполнения операций в устройстве описывается в форме алгоритма, представляемого в терминах микроопераций и логических условий, называемого микропрограммой;

4) микропрограмма используется как форма представления функции устройства, на основе которой определяются структура и порядок функционирования устройства во времени.

ПМУ обеспечивает гибкость микропроцессорной системы и позволяет осуществлять проблемную ориентацию микро- и миниЭВМ.

3.7. Структура микропроцессора

3.7.1. Типовая структура микропроцессора

Типовая структура микропроцессора [1, 3, 7] приведена на рис. 7. Микропроцессор состоит из трех основных блоков: арифметико-логическое устройство (АЛУ), блок внутренних регистров (микропроцессорная память) и устройство управления. Для передачи данных между этими блоками используется внутренняя шина данных.

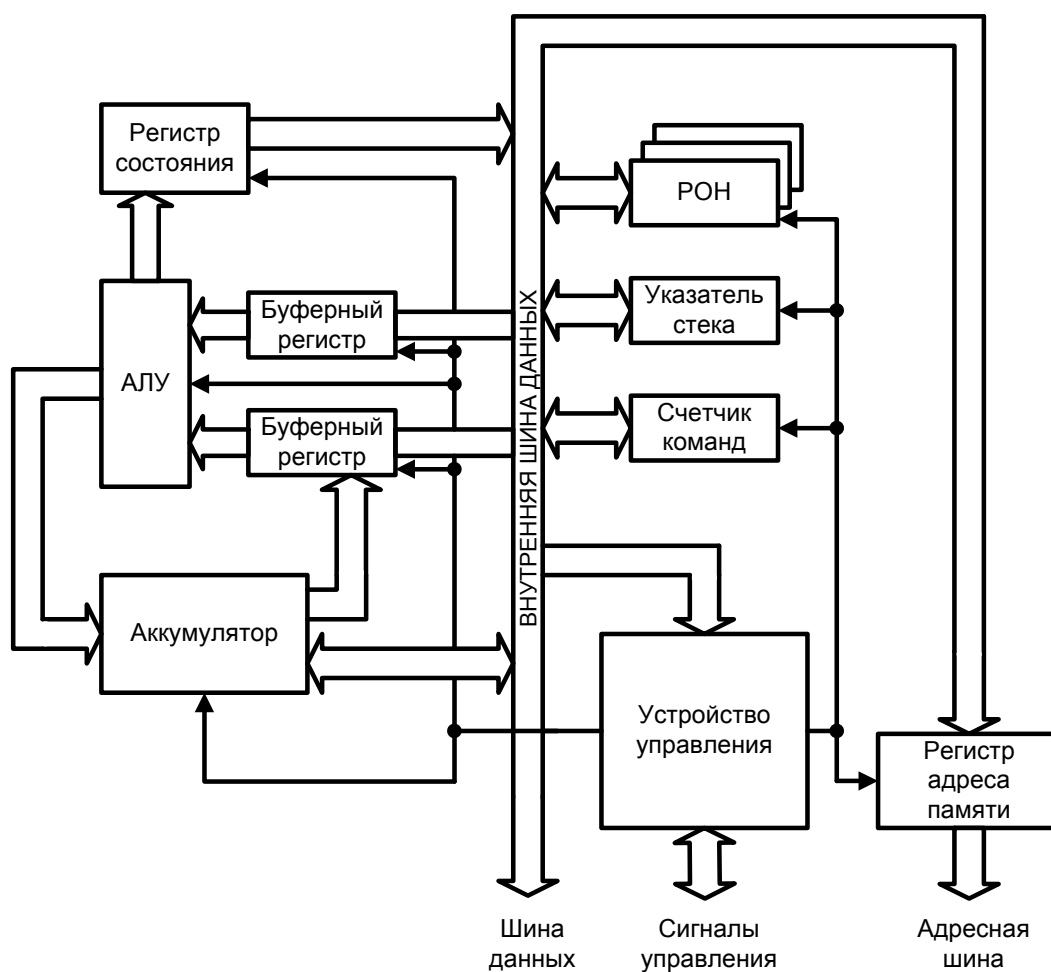


Рис. 7. Типовая структурная схема микропроцессора

Арифметико-логическое устройство выполняет одну из главных функций микропроцессора – обработку данных [7]. Перечень функций АЛУ зависит от типа микропроцессора. Некоторые АЛУ способны вы-

полнять множество различных операций, у других набор операций ограничен. Функции АЛУ определяют архитектуру микропроцессора в целом. Операции, выполняемые АЛУ, большинства микропроцессоров следующие: сложение, вычитание, И, ИЛИ, исключающее ИЛИ, инверсия, сдвиг вправо, сдвиг влево, приращения положительное и отрицательное.

Важная составная часть микропроцессора – регистры. Каждый регистр микропроцессора можно использовать для временного хранения одного слова данных. Некоторые регистры имеют *специальное назначение*, другие – *многоцелевое*. Последние называются регистрами *общего назначения* (РОН) и могут использоваться программистом по его усмотрению. Количество и назначение регистров в микропроцессоре зависят от его архитектуры.

Рассмотрим назначение основных регистров, имеющих почти во всех микропроцессорах.

Аккумулятор – это главный регистр микропроцессора при различных манипуляциях с данными [7]. Большинство арифметических и логических операций осуществляется путем использования АЛУ и аккумулятора. Любая из таких операций над двумя словами данных (операндами) предполагает размещение одного из них в аккумуляторе, а другого в памяти или каком-либо регистре. Так, при сложении двух слов, называемых условно А и В и расположенных в аккумуляторе и памяти соответственно, результирующая сумма С загружается в аккумулятор, замещая слово А. Результат выполнения операции АЛУ тоже обычно размещается в аккумуляторе, содержимое которого при этом теряется.

Операцией другого типа, использующей аккумулятор, является программируемая передача данных из одной части микропроцессора в другую. Например, пересылка данных между портом ввода/вывода и памятью, между двумя областями памяти и т. д. Выполнение операции «программируемая передача данных» осуществляется в два этапа: сначала выполняется пересылка данных из источника в аккумулятор, затем – из аккумулятора в пункт назначения.

Микропроцессор может выполнять некоторые действия над данными непосредственно в аккумуляторе. Например, аккумулятор можно очистить путем записи двоичных нулей во все его разряды, установить в единичное состояние путем записи во все его разряды двоичных единиц. Содержимое аккумулятора можно сдвигать влево или вправо, получать его инвертированное значение, а также выполнять другие операции. Аккумулятор является наиболее универсальным регистром микропроцессора: для выполнения любой операции над данными пре-

жде всего необходимо поместить их в аккумулятор. Данные поступают в него с внутренней шины данных микропроцессора. В свою очередь, аккумулятор может посылать данные на эту шину.

Количество разрядов аккумулятора соответствует длине слова микропроцессора, однако некоторые микропроцессоры имеют аккумуляторы двойной длины. В дополнительные разряды аккумулятора записываются при этом биты, появляющиеся при выполнении некоторых арифметических операций. Например, при умножении двух 8-битовых слов результат (16-битовое число) размещается в аккумуляторе двойной длины.

Счетчик команд – это один из наиболее важных регистров микропроцессора [7]. Как известно, *программа* – это последовательность команд (инструкций), хранимых в памяти микроЭВМ и предназначенных для того, чтобы инструктировать машину, как решать поставленную задачу. Для корректного ее выполнения команды должны поступать в строго определенном порядке. Счетчик команд обеспечивает формирование адреса очередной команды, записанной в памяти.

Когда микропроцессор начинает работать, то по команде начальной установки в счетчик команд загружаются данные из области памяти, заданной проектировщиком микропроцессора. Когда программа начинает выполняться, первым значением содержимого счетчика команд является этот, заранее определенный адрес.

В отличие от аккумулятора счетчик команд не может выполнять операции различного типа. Набор команд, его использующих, крайне ограничен по сравнению с подобным набором для аккумулятора.

Перед выполнением программы счетчик команд необходимо загрузить адресом, указывающим на первую команду программы. Адрес первой команды программы посылается по адресной шине к схемам управления памятью, в результате чего считывается ее содержимое по указанному адресу. Далее эта команда передается в специальный регистр микропроцессора, называемый *регистром команд* [7].

После извлечения команды из памяти микропроцессор автоматически дает приращение содержимому счетчика команд. Это приращение счетчик команд получает в тот момент, когда микропроцессор начинает выполнять команду, только что извлеченную из памяти. Следовательно, с этого момента счетчик команд содержит адрес следующей команды.

Счетчик команд можно загрузить иным содержимым при выполнении особой группы команд. Может возникнуть необходимость выполнить часть программы, которая «выпадает» из последовательности команд основной (главной) программы. Например, такую часть про-

граммы, которая повторяется в процессе выполнения всей программы. Вместо того чтобы писать эту часть программы каждый раз, когда в ней возникает необходимость, ее записывают один раз и возвращаются к ее повторному выполнению, отступая от указанной последовательности. Часть программы, выполняемая путем отступления от последовательности команд главной программы, называется подпрограммой. В данном случае в счетчик команд непосредственно записывается требуемый адрес.

Часто счетчик команд имеет намного больше разрядов, чем длина слова данных микропроцессора. Так, в большинстве 8-разрядных микропроцессоров число разрядов счетчика команд равно 16.

Регистр команд содержит команду в процессе ее дешифрования и выполнения. Входные данные поступают в регистр из памяти по мере последовательной выборки команд. Обычно существует возможность записи данных в регистр команд при помощи набора переключателей и кнопок на пульте управления ЭВМ. Как правило, этой возможностью пользуются для передачи управления в начало программы.

Регистр адреса памяти при каждом обращении к памяти микроЭВМ указывает адрес области памяти, подлежащей использованию микропроцессором. Регистр адреса памяти содержит двоичное число – адрес области памяти. Выход этого регистра называется **адресной шиной** и используется для выбора области памяти или порта ввода/вывода [7].

В течение выборки команды из памяти регистры адреса памяти и счетчика команд имеют одинаковое содержимое, т. е. регистр адреса памяти указывает местоположение команды, извлекаемой из памяти.

После декодирования команды счетчик команд получает приращение в отличие от регистра адреса памяти.

В процессе выполнения команды содержимое регистра адреса памяти зависит от выполняемой команды. Если в соответствии с командой микропроцессор должен произвести еще одно обращение к памяти, то регистр адреса памяти подлежит вторичному использованию в процессе обработки этой команды. Для некоторых команд, например команды очистки аккумулятора, адресация к памяти не требуется. При обработке таких команд регистр адреса памяти используется лишь один раз – в течение выборки команды из памяти.

В большинстве микропроцессоров регистры адреса памяти и счетчика команд имеют одинаковое количество разрядов. Как и счетчик команд, регистр адреса памяти должен располагать количеством разрядов, достаточным для адресации любой области памяти микроЭВМ. У

большинства 8-разрядных микропроцессоров количество разрядов регистра адреса памяти равно 16.

Поскольку регистр адреса памяти подключен к внутренней шине данных микропроцессора, он может загружаться от различных источников. Большинство микропроцессоров располагают командами, позволяющими загружать этот регистр содержимым счетчика команд, регистра общего назначения или какой-либо области памяти. Некоторые команды предоставляют возможность изменять содержимое регистра адреса памяти путем выполнения вычислений: новое значение содержимого этого регистра получается путем сложения или вычитания содержимого счетчика команд с числом, указанным в самой команде. Адресация такого типа называется адресацией с использованием смещения.

Буферный регистр предназначен для временного хранения (буферирования) данных [7].

Регистр состояния предназначен для хранения результатов некоторых проверок, осуществляемых в процессе выполнения программы. Разряды регистра состояний принимают то или иное значение при выполнении операций, использующих АЛУ и некоторые регистры. Запоминание результатов упомянутых проверок позволяет использовать программы, содержащие переходы (нарушения естественной последовательности выполнения команд) [7].

При наличии в программе перехода по заданному признаку выполнение команд начинается с некоторой новой области памяти, т. е. счетчик команд загружается новым числом. В случае условного перехода такое действие имеет место, если результаты определенных проверок совпадают с ожидаемыми значениями. Указанные результаты находятся в регистре состояния. Регистр состояния предоставляет программисту возможность организовать работу микропроцессора так, чтобы при определенных условиях менялся порядок выполнения команд.

Рассмотрим некоторые наиболее часто используемые разряды регистра состояния.

1. Перенос/заем. Данный разряд указывает, что последняя выполненная операция сопровождалась переносом или займом (отрицательным переносом). Значение разряда переноса устанавливается равным 1, если в результате сложения двух чисел имеет место перенос из старшего разряда АЛУ. Отрицательный перенос (заем) фиксируется в регистре состояния при вычитании большего числа из меньшего.

2. Нулевой результат. Принимает единичное значение, если после окончания операции во всех разрядах регистра результата обнаружены

двоичные нули. Установка этого разряда в 1 происходит не только при отрицательном приращении содержимого регистра, но и при любой другой операции, результатом которой – число из двоичных нулей.

3. Знаковый. Принимает единичное значение, когда старший значащий бит содержимого регистра, предназначенного для записи результата операции, становится равным 1. При выполнении арифметических операций с числами в дополнительном коде единичное значение старшего значащего бита показывает, что в регистре находится отрицательное число. Многие микропроцессоры располагают дополнительными разрядами состояний. В некоторых предусмотрены специальные команды для сброса или очистки всех разрядов состояния.

Регистры общего назначения (РОН) [7]. Большинство МП имеют в своем составе набор регистров, используемых в качестве запоминающих устройств. Так как АЛУ может совершать операции с содержимым РОН без выхода на внешнюю магистраль адресов и данных, то они происходят много быстрее, чем операции с внешней памятью. Поэтому иногда РОН называют сверхоперативной памятью. Количество РОН и возможность программного доступа к ним у равных микропроцессоров различны.

Указатель стека. *Стек* – это набор регистров микропроцессора или ячеек оперативной памяти, откуда данные или адреса выбираются «сверху» по принципу: первым – поступивший последним. При записи в стек очередного слова все ранее записанные слова смещаются на один регистр вниз. При выборке слова из стека оставшиеся слова перемещаются на один регистр вверх.

Указанные процедуры иллюстрирует рис. 8 [7]. Здесь стек состоит из семи регистров. Если в стек загружается какое-либо слово, например А5, то оно записывается в верхнем регистре, а каждое из слов А1 ... А4 перемещается в соседние нижние регистры. Если же А5 извлекается из стека, то каждое из слов А1 ... А4 перемещается в соседние верхние регистры. Нельзя извлечь А4 раньше А5, т. е. автоматически реализуется отмеченный выше принцип. Стек обычно используется в микропроцессорах для хранения адресов возврата при обращении к подпрограммам, а также для запоминания состояния внутренних регистров при обработке прерываний. При организации стека в памяти время на обращение к нему будет равно циклу обращения к памяти. Эта операция выполняется значительно быстрее, если стек в виде набора регистров входит в состав микропроцессора. Важным параметром в таком случае является число регистров стека. При попытке записать в стек большее количество слов, чем число его регистров, первое слово будет утеряно. В некоторых микропроцессорах при переполнении ре-

гистров стека соответствующие слова записываются в стек памяти. Таким образом, процесс его функционирования напоминает работу с пачкой документов, когда каждый новый документ кладется сверху пачки.

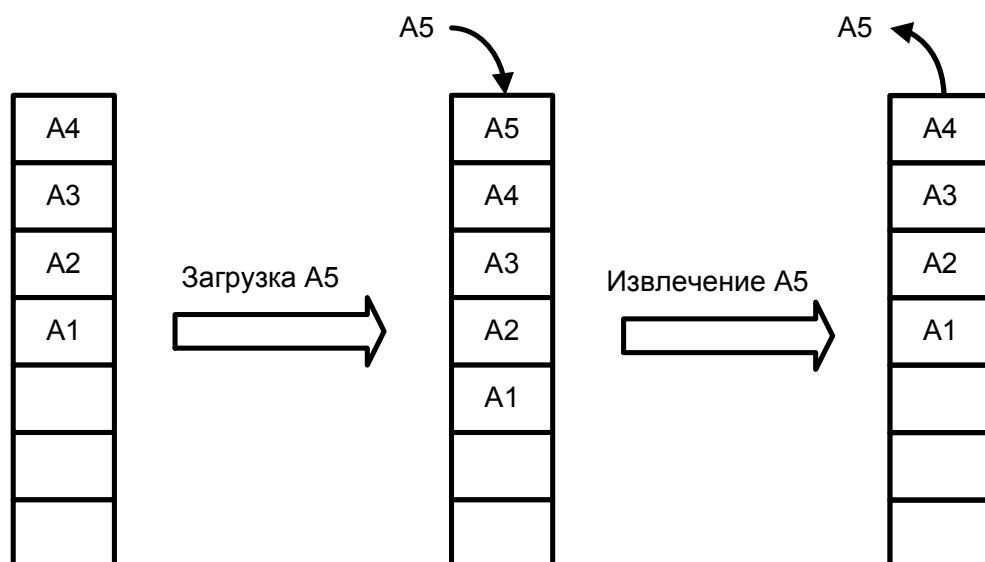


Рис. 8. Принципы работы стека

В приведенном примере при записи в стек новых данных все его содержимое перемещается на один регистр. В реальных МП такой способ не применяется, так как он сопряжен с большим количеством вспомогательных операций по переносу соседних регистров стека.

В реальных МП данные между ячейками не переносятся, а для хранения адреса последнего по времени поступления элемента стека используется специальный регистр – указатель стека (УС).

На рис. 9 указатель стека представляет собой трехразрядный регистр с двоичным представлением информации [7].

Первоначально указатель стека содержит число 0112. Это означает, что последний элемент – «верхушка стека» – находится в регистре с адресом 0112 (или 310). При операции загрузки в регистр 410 записывается число A5, а содержимое указателя стека изменяется так, что он указывает на регистр 410. При операции извлечения из стека производятся обратные действия.

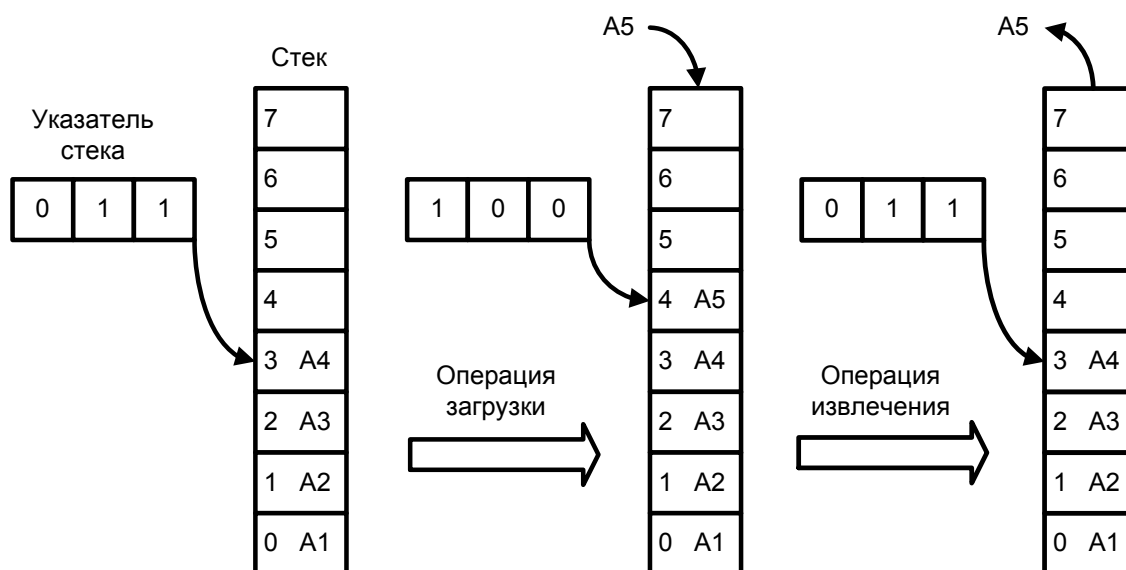


Рис. 9. Адресация элемента стека с использованием указателя стека

Схемы управления. Роль схем управления в микропроцессоре заключается в поддержании требуемой последовательности функционирования всех остальных его звеньев. По сигналам схем управления очередная команда извлекается из регистра команд. При этом определяется, что необходимо делать с данными, а затем обеспечивается последовательность действий для выполнения поставленной задачи.

Одна из главных функций схем управления – декодирование команды, находящейся в регистре команд, посредством дешифратора команд, который в результате выдает сигналы, необходимые для ее выполнения.

Помимо указанных выше действий, схемы управления выполняют некоторые специальные функции: управление последовательностью включения питания и процессами прерываний. *Прерывание* – это своего рода запрос, поступающий на схемы управления от других устройств (памяти, ввода/вывода). Прерывание связано с использованием внутренней шины данных микропроцессора. Схемы управления принимают решение, когда и в какой последовательности другие устройства могут пользоваться внутренней шиной данных.

Система шин. На характеристики микропроцессора большое влияние оказывает способ организации его связи с внешней средой – устройствами ввода/вывода (УВВ) и запоминающими устройствами (ЗУ). По способу организации связей с внешней средой различают микропроцессоры с мультиплексированной шиной адреса и данных (рис. 10,а) и с отдельными шинами адреса и данных (рис. 10,б). Мик-

ропроцессор с отдельными шинами адресов и данных изображен на рис. 7.

В микропроцессорах с мультиплексированной шиной адрес хранится на шине только короткий промежуток времени, поэтому устройствам, подключенным к шине, требуются регистры адреса (РгА). Для организации обмена информацией в таких микропроцессорах необходимо использовать управляющий сигнал *адрес–данные*. При отдельных шинах адреса и данных такой управляющий сигнал не нужен. Кроме того, у устройств, подключенных к шинам, отпадает необходимость в регистре адреса, так как он может быть размещен непосредственно на кристалле микропроцессора. Разрядность адресной шины в таких микропроцессорах не связана с разрядностью шины данных.

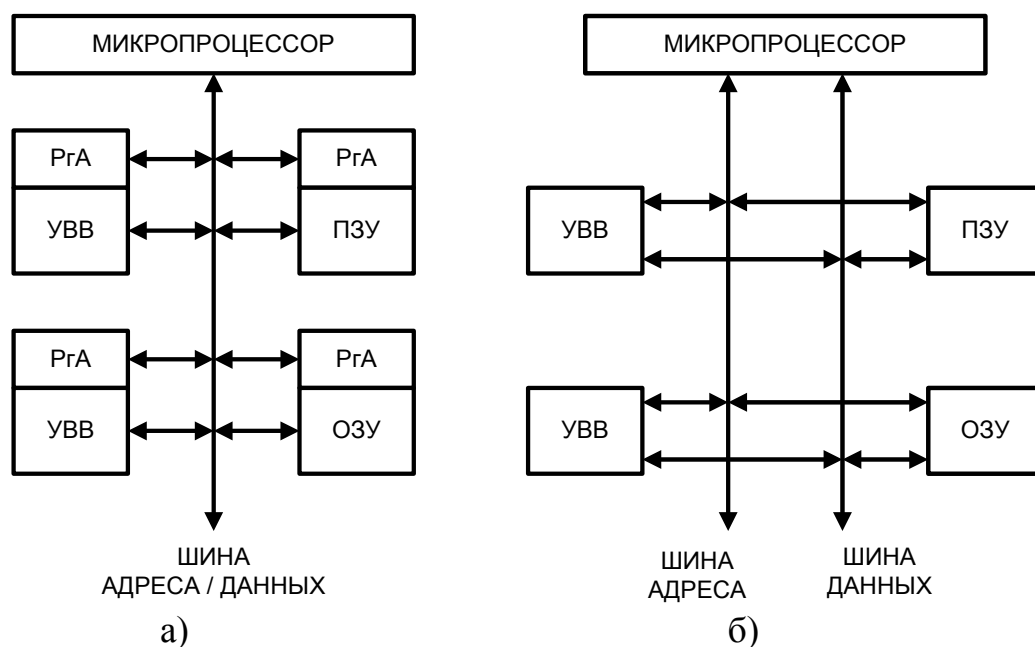


Рис. 10. Система шин микроЭВМ

3.7.2. Устройство управления

Основные функции:

1. Формирование адреса инструкции.
2. Считывание инструкции из ОЗУ (ПЗУ) и её хранение во время выполнения.
3. Дешифрация кода операции.
4. Формирование управляющих сигналов.
5. Считывание из регистра команд и регистров микропроцессорной памяти отдельных составляющих адресов операндов (чисел), участвующих в вычислениях, и формирование полных адресов операндов.
6. Выборка операндов (по сформированным адресам) и выполнение заданной операции обработки этих операндов.
7. Запись результатов операции в память.
8. Формирования адреса следующей команды программы.

Упрощенная структурная схема устройства управления (УУ) показана на рис. 11 [3, 4, 7].

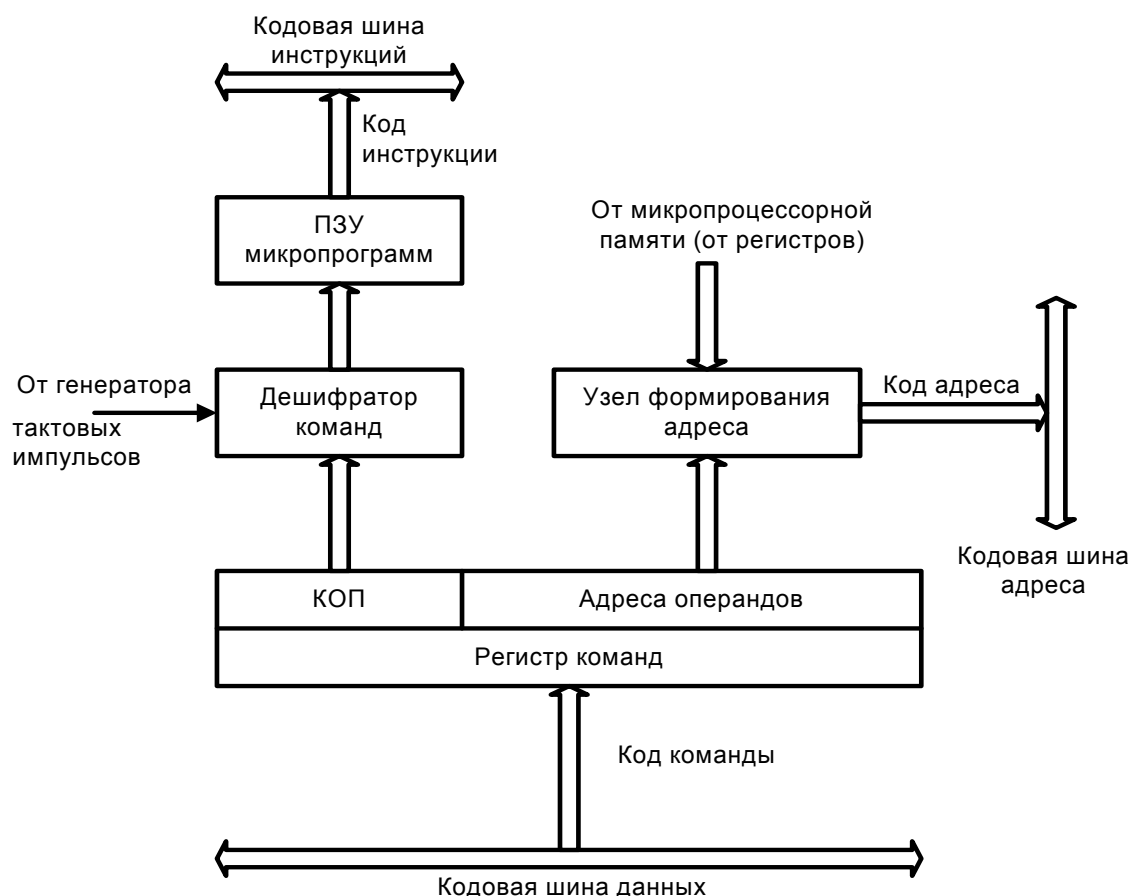


Рис. 11. Упрощённая структурная схема устройства управления

Состав структурной схемы:

Регистр команд – запоминающий регистр, в котором хранится код команды: код выполняемой операции и адреса операндов, участвующих в операции.

Дешифратор операций – логический блок, выбирающий из регистра команд код операции (КОП).

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

Узел формирования адреса – устройство, вычисляющее полный адрес ячейки памяти (регистра) по реквизитам, поступающим из регистра команд и регистров микропроцессорной памяти.

Кодовые шины данных, адреса и инструкций – часть внутренней интерфейсной шины микропроцессора.

3.7.3. Арифметико-логическое устройство

Основные функции АЛУ:

1. Приём операндов из ОЗУ и регистров (микропроцессорной памяти).
2. Выполнение арифметических и логических операций.
3. Передача результатов в регистры и в ОЗУ.
4. Формирование признаков результатов выполнения операций и их запись в регистр слова состояния процессора.

В состав АЛУ входят сумматоры, которые непосредственно выполняют элементарные операции, регистры для хранения операндов и результатов, сдвиговые регистры, логические схемы И, ИЛИ, НЕ, схемы для преобразования прямого кода числа в дополнительную форму и дополнительные схемы аппаратного умножения и деления для целочисленных операндов.

Упрощенная структурная схема АЛУ показана на рис. 12 [3, 4, 7].

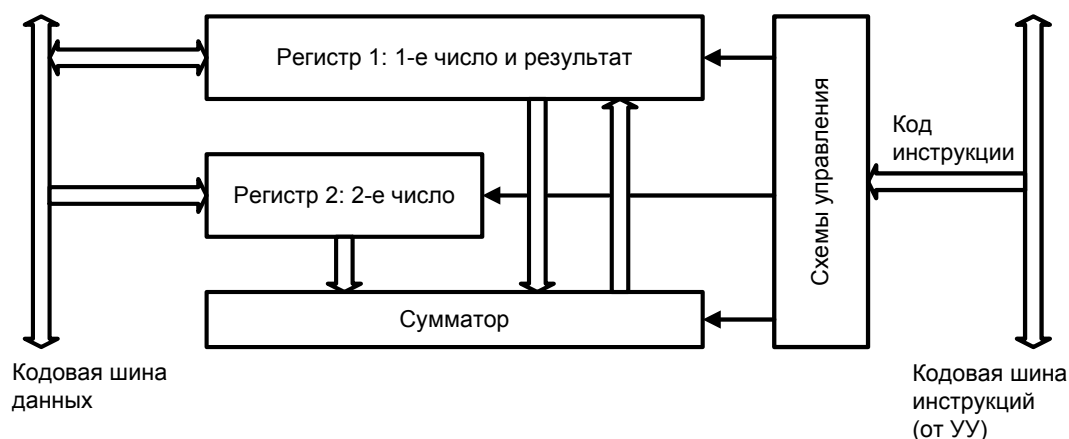


Рис. 12. Упрощённая структурная схема АЛУ

Состав структурной схемы:

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, и 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.

Основные элементы АЛУ. Основой АЛУ является одnorазрядный сумматор, схема которого обеспечивает суммирование цифры одного разряда 2-го числа с учётом бита переноса из соседнего младшего раз-

ряда 1-го числа. Схема одноразрядного сумматора представлена на рис. 13.

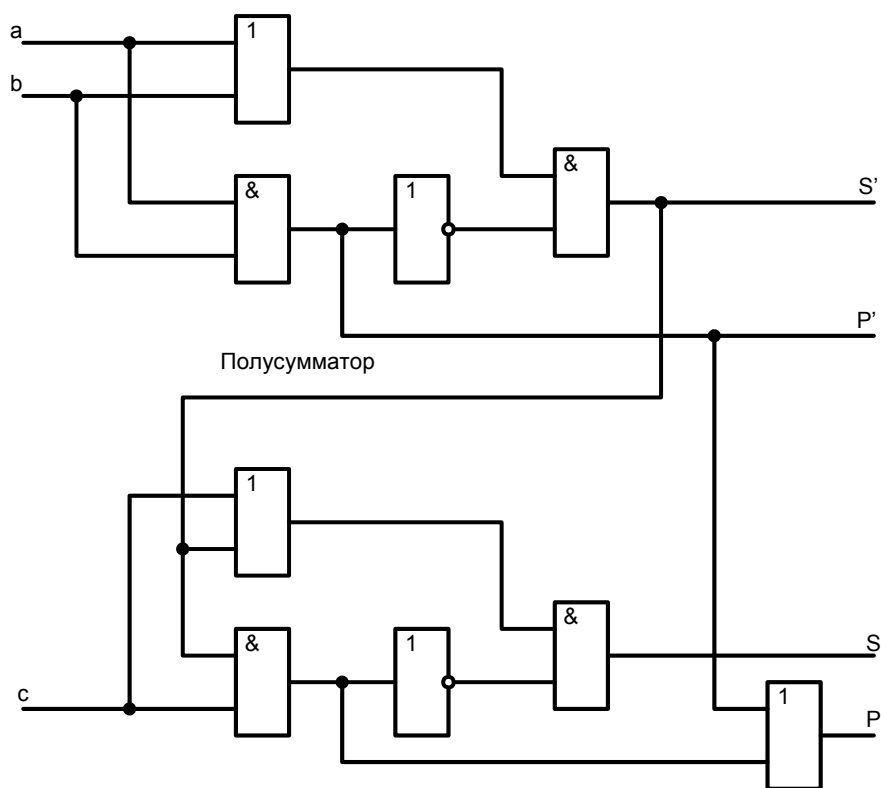


Рис. 13. Одноразрядный сумматор

Рассмотренный сумматор выполняет операцию сложения в соответствии с табл. 3.

Таблица 3

a	b	c	S'	P'	S	P
0	0	0	0	0	0	0
1	0	0	1	0	1	0
0	1	0	1	0	1	0
1	1	0	0	1	0	1
0	0	1	0	0	1	0
1	0	1	1	0	0	1
0	1	1	1	0	0	1
1	1	1	0	1	1	1

Примечание: a – соответствующий разряд 1-го числа;

- b – соответствующий разряд 2-го числа;
- c – бит переноса из соседнего младшего разряда;
- S' – значение цифры суммы в данном разряде полусумматора;
- P' – цифра переноса в следующий (старший) разряд полусумматора;
- S – значение цифры суммы в данном разряде;
- P – цифра переноса в следующий (старший) разряд.

Практически все операции в АЛУ сводятся к сложению и дополнительным операциям сдвига и преобразования кода.

3.7.4. Последовательность работы микропроцессора

После запуска в регистре счетчика команд микропроцессора устанавливается адрес ячейки ОЗУ, в которой находится начало (первая команда) программы.

После этого автоматически начинается выполнение команд программы друг за другом. Каждая команда требует для своего исполнения нескольких тактов работы машины (такты определяются периодом следования импульсов от генератора тактовых импульсов).

В первом такте выполнения любой команды производятся считывание кода самой команды из ОЗУ по адресу, установленному в регистре-счетчике команд, и запись этого кода в блок регистров команд устройства управления. Содержание второго и последующих тактов исполнения определяется результатами анализа команды, записанной в блок регистров команд, т. е. зависит уже от конкретной команды.

Пример. Выполнение команды сложения аккумулятора с прямоадресуемой ячейкой памяти ОЗУ (длина команды два байта) [1, 4, 8]:

add a, addr.

При выполнении данной команды будут выполнены следующие действия:

- Первый такт: считывание КОП из ОЗУ по адресу, установленному в регистре-счетчике команд, запись этого кода в блок регистров команд устройства управления, декодирование КОП, приращение регистра счётчика команд.
- Второй такт: запись содержимого аккумулятора в буферный регистр АЛУ.
- Третий такт: считывание из ячейки ОЗУ по адресу addr второго слагаемого и перемещение его во второй буферный регистр АЛУ, приращение регистра счётчика команд;

- Четвёртый такт: сложение в АЛУ переданных туда чисел и формирование суммы.
- Пятый такт: считывание из АЛУ суммы чисел и запись её в аккумулятор.

Далее МП приступит к выполнению следующей команды по адресу, содержащемуся в регистре-счётчике команд. Команды будут выполняться последовательно одни за другой, пока не завершится вся программа.

При наличии в программе перехода по заданному признаку выполнение команд начинается с некоторой новой области памяти, т. е. счётчик команд загружается новым числом. В случае условного перехода такое действие имеет место, если результаты определенных проверок совпадают с ожидаемыми значениями. Указанные результаты находятся в регистре состояния. Регистр состояния предоставляет программисту возможность организовать работу микропроцессора так, чтобы при определенных условиях менялся порядок выполнения команд.

3.7.5. Режимы адресации

Для взаимодействия с различными модулями в ЭВМ должны быть средства идентификации ячеек внешней памяти, ячеек внутренней памяти, регистров МП и регистров устройств ввода/вывода. Поэтому каждой из запоминающих ячеек присваивается адрес, т. е. однозначная комбинация бит. Количество бит определяет число идентифицируемых ячеек. Обычно ЭВМ имеет различные адресные пространства памяти и регистров МП, а иногда – отдельные адресные пространства регистров устройств ввода/вывода и внутренней памяти. Кроме того, память хранит как данные, так и команды. Поэтому для ЭВМ разработано множество способов обращения к памяти, называемых режимами адресации.

Режим адресации памяти – это процедура или схема преобразования адресной информации об операнде в его исполнительный адрес.

Все способы адресации памяти можно разделить на:

1) прямой, когда исполнительный адрес берется непосредственно из команды или вычисляется с использованием значения, указанного в команде, и содержимого какого-либо регистра (прямая адресация, регистровая, базовая, индексная и т. д.);

2) косвенный, который предполагает, что в команде содержится значение косвенного адреса, т. е. адреса ячейки памяти, в которой находится окончательный исполнительный адрес (косвенная адресация).

В каждой микроЭВМ реализованы только некоторые режимы адресации, использование которых, как правило, определяется архитектурой МП.