

Архитектурные особенности модели микропроцессорной системы

Лекция 4

Упрощенная архитектура типовой микро-ЭВМ.

- Архитектура – это совокупность свойств и основных характеристик, раскрывающих возможности ЭВМ и ВС (функциональные средства, принципы обработки данных, организация вычислительного процесса, логическая организация совместной работы различных устройств). Первая архитектура ЭВМ была разработана до её появления, и ей были свойственны следующие характерные черты: Единственная и последовательно адресуемая память; Хранение программ и данных в одной памяти; Последовательное выполнение команд программы до появления специальных указаний (команд перехода). ЭВМ состоит из трех структурных элементов: процессора, памяти и устройств ввода-вывода. Управление системой целиком возложено на процессор, и для пересылки данных между устройством ввода-вывода и памятью требуется прямое управление со стороны процессора.

Архитектура фон Неймана

ПАМЯТЬ

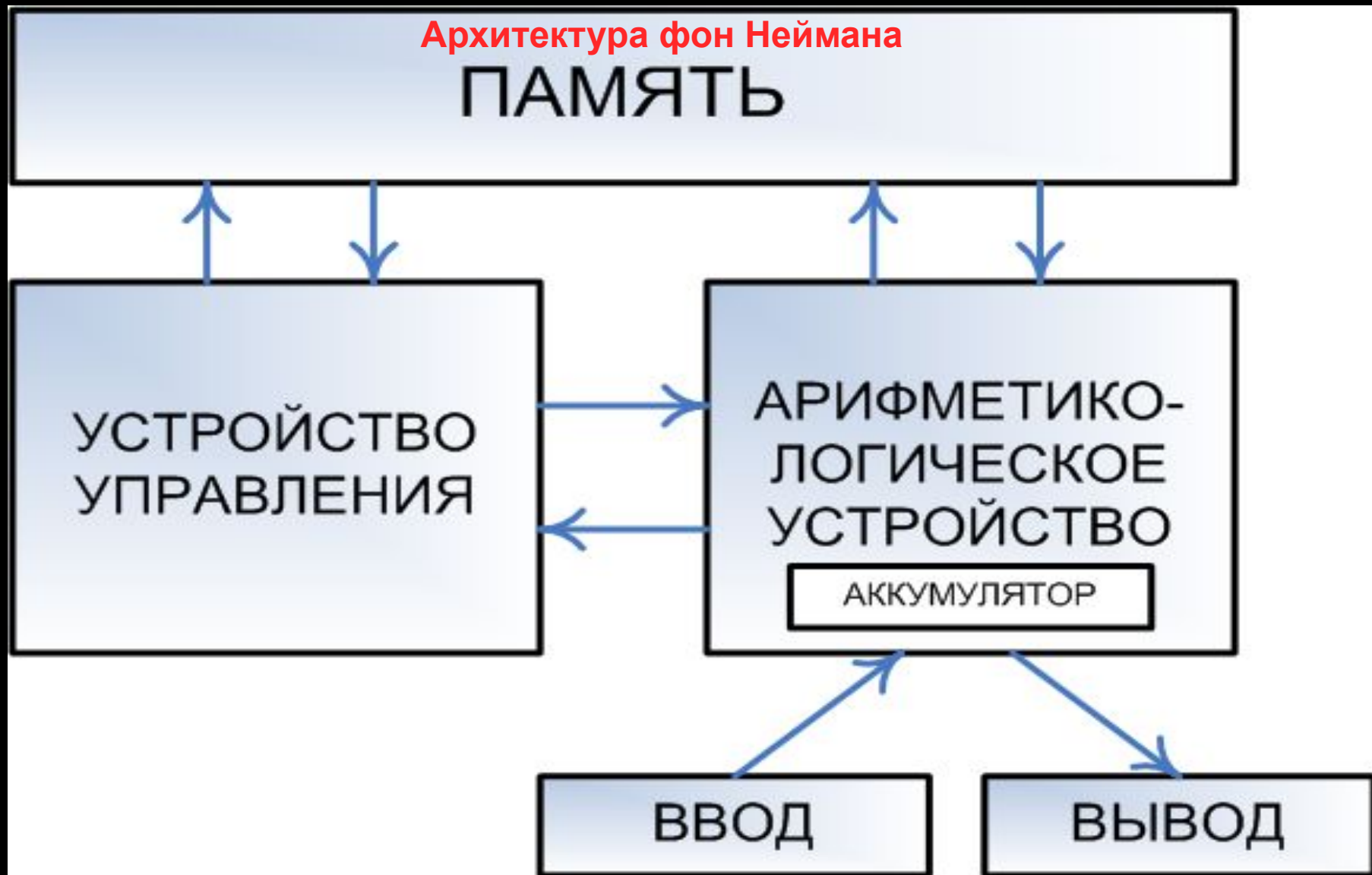
УСТРОЙСТВО
УПРАВЛЕНИЯ

АРИФМЕТИКО-
ЛОГИЧЕСКОЕ
УСТРОЙСТВО

АККУМУЛЯТОР

ВВОД

ВЫВОД



Упрощенная архитектура типовой микро-ЭВМ.



Упрощенная архитектура типовой микро-ЭВМ.

Материнская плата микрокомпьютера (англ. motherboard), или системная плата, основное устройство, определяющее архитектуру и производительность компьютера. На материнской плате прежде всего размещаются:

- Центральный процессор (Central Processor Unit, CPU) – главная микросхема, выполняющая вычислительные и логические действия;
- Оперативная память (оперативное запоминающее устройство, ОЗУ) – набор микросхем для хранения данных во время работы компьютера;
- ПЗУ (постоянное запоминающее устройство) – микросхема для долговременного хранения данных;
- Шины – наборы проводников для обмена сигналами между внутренними компонентами компьютера;
- Набор микросхем, управляющих работой внутренних компонентов компьютера и определяющих функциональные возможности материнской платы;
- Разъемы (слоты) – расширения для подключения дополнительных устройств.

Упрощенная архитектура типовой микро-ЭВМ.

Системная магистраль данных (шина).

- Шина - совокупность проводников, предназначенных для обмена данными между различными устройствами компьютера. Шина - общий канал связи, по которому внутри устройства передаются данные; она используется совместно различными блоками системы. Для рабочих мест на основе микропроцессора Intel-80486 с интенсивным использованием графики (анимация, САПР) где требуется обеспечить высокую пропускную способность ввода – вывода для двух – трёх контроллеров, например, видеоконтроллера и контроллера дисков, целесообразно выбрать локальную видеошину VESA. При этом можно получить компьютер с высокопроизводительными видео- и дисковой системами.
- Шина VESA (или локальная шина, VL – шина) разработана ассоциацией VESA (Video Electronics Standards Association). Эта шина обеспечивает более дешевое и более эффективное подключение высокоскоростных внешних устройств, поддерживая непосредственный доступ центрального процессора к соответствующим контроллерам (видеоконтроллерам, контроллерам жестких дисков, адаптерам локальной сети). Для использования остальных устройств на компьютер устанавливается другая шина EISA. Благодаря разработанным ассоциацией VESA правилам «шинного арбитража» эти шины могут сосуществовать в одном компьютере не мешая друг другу. Компьютеры с шинами VESA и EISA часто называют «VESA/ EISA».

Структура оперативной памяти.

- Оперативная память (RAM, ОЗУ) обеспечивает работу с программным обеспечением. Из неё процессор и сопроцессор берут программы и исходные данные для обработки, в нее же записываются полученные результаты. Английское название RAM – Random Access Memory – переводится как «память с произвольным доступом». Произвольность доступа подразумевает возможность операций записи и чтения с любой ячейкой ОЗУ в произвольном порядке. После прекращения подачи питания вся информация, содержащаяся в оперативной памяти, уничтожается. Поэтому сделанную работу необходимо сохранять в виде файлов на жестком диске ВС, либо других накопителях. Характеристика оперативной памяти – объём, измеряемый в мегабайтах (Мб). Выбирают размер оперативной памяти с учетом задач, которые будут решаться. САПР P-CAD 8.5 требует не менее 4 Мб, а фактически 8 – 12 Мб оперативной памяти.
- Постоянное запоминающее устройство (ПЗУ) служит для хранения программ начальной загрузки компьютера и режимом его работы является считывание данных. ПЗУ – это энергонезависимая память. И при выключении питания содержимое ПЗУ сохраняется. Энергонезависимая память в основном применяется для хранения неизменяемой (или редко изменяемой) информации – системного программного обеспечения (BIOS), таблиц и т.д. Т.к. эта информация обычно является ключевой для функционирования ВС, то энергонезависимая память должна обладать такими важными параметрами как время хранения и устойчивость к электромагнитным воздействиям.

Структура оперативной памяти.

Кэш-память или сверхоперативная память. Скорость обработки информации центральным процессором уже так высока, что современные устройства ОЗУ не справляются с функцией посредника между ЦП и внешней памятью. Поэтому было добавлено еще одно устройство кэш-память – служащее посредником между ОЗУ и ЦП. Современные микропроцессоры имеют встроенную кэш-память.

Кэш память устроена так, что при попытке прочитать данные из ОЗУ сначала аппаратным образом проверяется, нет ли нужных данных в кэше. Если эти данные в кэше, они быстро извлекаются и используются процессором. Однако в противном случае эти данные считываются из ОЗУ, и в момент передачи процессору также помещаются в кэш (на случай, если они понадобятся позже). С точки зрения процессора всё это происходит прозрачно, так как единственное, что отличается между обращениями к данным в кэше или обращением к данным в ОЗУ — это время, необходимое для получения данных.

Система команд и методы адресации.

- Микропроцессор , как и всякий алгоритмический исполнитель , имеет определенную систему команд (СК). Количество , назначение , формат команд пользователь (программист) изменить не может , но он может применять эти команды в любом порядке , определяемом логикой решения задачи .Значительная часть команд из СК предназначена для обработки данных. Сюда относятся: сложение , вычитание , умножение , деление , пересылка данных , сдвиги двоичных кодов и многое другое. Данные , подлежащие обработке в той или иной конкретной команде , принято называть операндами. Операнды могут располагаться в различных местах , в частности , в одном из РОН , в сегменте данных оперативной памяти , в стеке и т. д. Обращение к ним может быть произведено весьма разнообразными способами. Это разнообразие и составляет суть методов адресации .Система команд и методы адресации - два тесно взаимосвязанных между собой понятия. С одной стороны , ни одной конкретной команды нельзя сформировать без использования того или иного метода адресации , с другой стороны , эти методы "не работают" вне конкретных команд.

Система команд и методы адресации.

Перечислим название методов адресации и дадим их краткую характеристику:

- Регистровая адресация. Операнд или операнды располагаются в одном из РОН , либо (если речь идет о сегментной части адреса) в сегментном регистре. Это обеспечивает очень быстрый доступ к данным. Соответствующие команды , как правило , имеют короткий двоичный код. Однако количество РОН в процессоре невелико , их может не хватить для размещения всех переменных при решении сложной задачи .
Непосредственная адресация. Операнд(ы) располагаются в самой команде , в ее последних байтах. Эти байты , как правило , находятся во внутренней очереди команд процессора , поэтому доступ к ним осуществляется также достаточно быстро.
- Прямая адресация. В составе команды находится не сам операнд , а его адрес. Это простейший способ обратиться к данным , находящимся в ОЗУ.

Система команд и методы адресации.

- Косвенно - регистровая адресация. В этом случае адрес операнда размещается в одном из регистров. Как правило, для этого используются регистры SI, DI, BX, BP. Содержимое регистра можно изменять (например, в цикле), при этом одна и та же команда будет оперировать различными ячейками памяти.
- Базовая адресация. В этом случае адрес операнда (исполнительный адрес) получается как сумма содержимого регистров BX или BP и числовой константы, называемой смещением. Если использован регистр BX, то будет происходить обращение к сегменту данных в ОЗУ, а если регистр BP - то к сегменту стека. Такой вид адресации можно, например, использовать для доступа к элементу некоего массива, номер которого заранее известен: регистр BX указывает на начало массива, а смещение представляет собой номер элемента.
- Индексная адресация. Манипулирует содержимым сегмента данных и во всех микропроцессорах фирмы Intel по существу аналогична базовой. Адрес операнда вычисляется как сумма содержимого регистров SI или DI и смещения.
- Базово - индексная адресация (а также - базово - индексная со смещением). Адрес операнда здесь образуется из суммы содержимого регистров BX (или BP), регистров SI (или DI) и необязательного смещения.
- Стековая адресация. Является разновидностью неявной. Операнд находится в стеке, на вершину которого указывает регистр SP. Неявная (или подразумеваемая) адресация используется, например, при обращении к отдельным флагам или регистру флагов в целом, а также в командах обработки строк (цепочек данных) типа MOVS, SCAS и т. п.
- Относительная адресация. В микропроцессорах фирмы Intel не применяется к командам обработки данных, а используется лишь в командах переходов, вызовов подпрограмм и управления циклами. Адрес перехода образуется как смещение относительно текущего содержимого счетчика команд.

Форматы команд и данных.

Алгоритм, написанный пользователем программы, в конечном счете реализуется в виде машинных команд. Под **командой** понимают совокупность сведений, представленных в виде двоичных кодов, необходимых процессору для выполнения очередного шага. В ходе команды для сведений о типе операции, адресной информации о нахождении обрабатываемых данных, а также для информации о месте хранения результатов выделяются определенные разряды (поля).

Форматом команды называется заранее обговоренная структура полей в её кодах, позволяющая ЭВМ распознавать составные части кода.

Главным элементом кода команды является **код операции** (КОП), что определяет, какие действия будут выполнены по данной команде. Под него выделяется N старших разрядов формата. В остальных разрядах размещаются A_1 и A_2 - адреса операндов. A_3 - адрес результата.

Распределение полей в формате команды может изменяться при смене способа адресации. Длина команды зависит от числа адресных полей. По числу адресов команды делятся на:

- Безадресные;
- Одноадресные;
- Двухадресные;
- Трехадресные.

Форматы команд и данных.

Длина кода команды измеряется в машинных словах. Чтобы получить возможность работать с минимальным числом адресных полей, результат, к примеру, можно размещать по месту хранения одного из операндов. Либо предварительно размещают один или несколько операндов в специально выделенных регистрах процессора. Множество реализуемых машинных действий образует её **систему команд**. Система команд часто определяет области и эффективность применения ЭВМ. Состав и число команд должны быть ориентированы на стандартный набор операций, используемых пользователем для решения своих задач.

По функциональному назначению в системе команд ЭВМ различают следующие группы:

- Команды передачи данных (обмен входами между регистрами процессора, процессора и оперативной памятью, процессора и периферийными установками).
- Команды обработки данных (команды сложения, умножения, сдвига, сравнения-)
- Команды передачи управления (команды безусловного и условного перехода)
- Команды дополнительные (типа RESET, TEST,-).

Форматы команд и данных.

Группа команд передачи управления обеспечивает принудительное изменение порядка выполнения команд в программе. Оттранслированные команды записываются в соседние ячейки памяти в порядке их следования в программе. При естественном порядке выполнения команд в программе, адрес каждой следующей команды определяется по содержимому специального счетчика команд, который входит в состав процессора. Содержимое этого счетчика автоматически наращивается на 1 при выполнении очередной команды. При организации ветвления цикла или для перехода на подпрограмму в счетчик в счетчик команд принудительно записывается адрес перехода, указанный в ходе команды. Большинство алгоритмов может быть реализовано небольшим базовым набором команд. Вместе с тем система команд должна быть полной, т.е. содержать все команды, которые необходимы для интерпретации алгоритма в машинных кодах. ЭВМ общего назначения имеет универсальный набор команд и применяется в основном для решения тривиальных (стандартных) задач. Существуют 2 различных принципа поисков операндов в памяти: ассоциативный и адресный.

- **Ассоциативный** поиск (поиск по содержанию запоминающей ячейки) предполагает просмотр содержимого всех ячеек памяти для выявления кода, содержащего заданный командой ассоциативный признак.
- **Адресный** поиск предполагает, что операнд находится по адресу, указанному в адресном поле команд.

Форматы команд и данных.

Исполнительным адресом операнда называется двоичный код номера ячейки памяти, по которому будет записан или считан оператором.

Адресным кодом команды называется двоичный код в адресном поле команды, с помощью которого необходимо сформировать исполнительный адрес операнда. В ЭВМ адресный код и исполнительный адрес не совпадают, поэтому способ адресации можно определить, как способ формирования исполнительного адреса по адресному коду команды. Способы адресации классифицируют:

- по наличию адресной информации в команде (явная и неявная адресация).
- по кратности обращения в оперативную память.
- по способу формирования адресов ячеек памяти.

При **явной** адресации операнда в команде есть поле адреса этого операнда.

При **неявной** в адресное поле в команде отсутствует, а адрес операнда подразумевается кодом операции. Например, из команды может быть исключен адрес приемника адресата, при этом подразумевается, что результат записывается на месте второго операнда.

По кратности обращения в оперативную память различают:

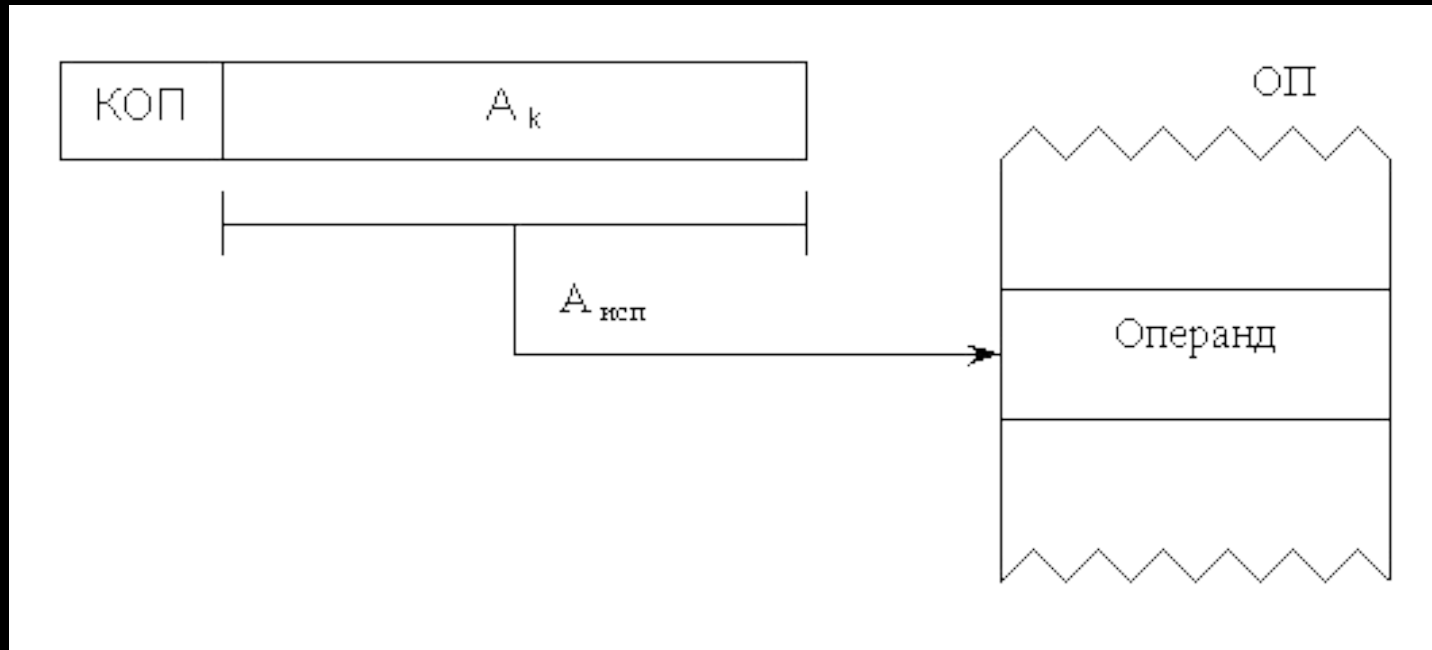
непосредственную адресацию (direct addressing)

прямую адресацию (immediate addressing)

косвенную адресацию (indirect addressing)

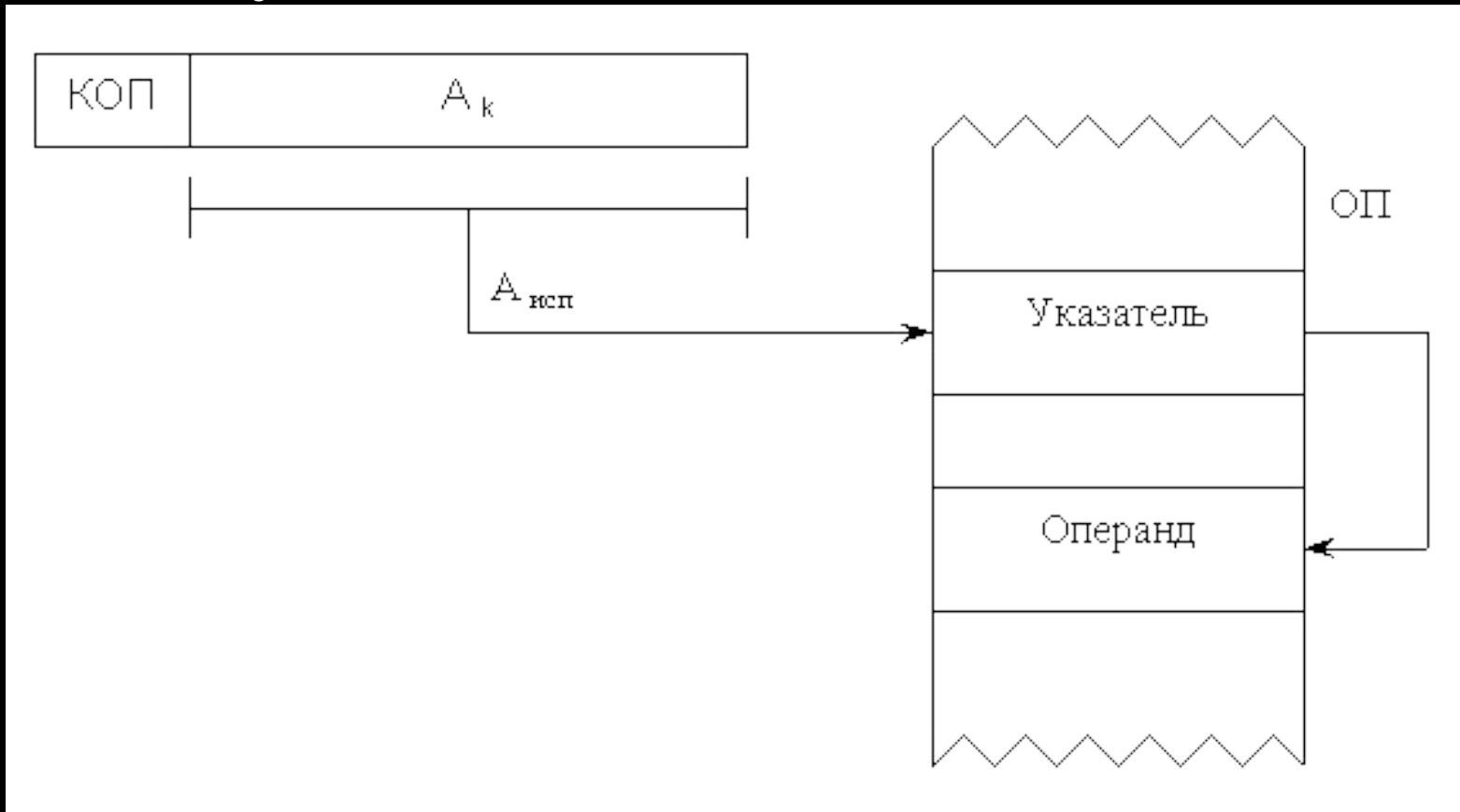
Форматы команд и данных.

Прямая адресация-При прямой адресации обращение за операндом производится по адресу коду в поле команды. При этом исполнительный адрес совпадает с адресом кода команды.



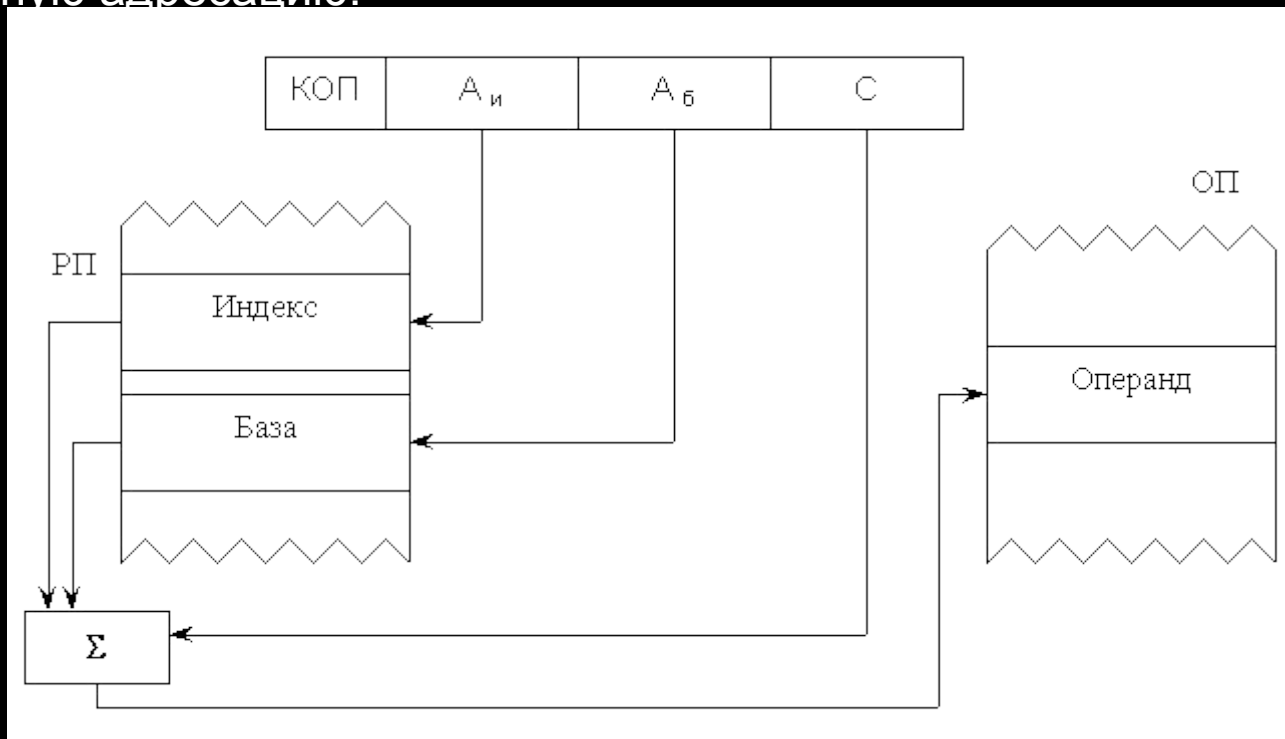
Форматы команд и данных.

- **Косвенная адресация**-При косвенной адресации код команды указывает адрес ячейки памяти, в которой находится не сам операнд, а его адрес, называемый *указателем*.



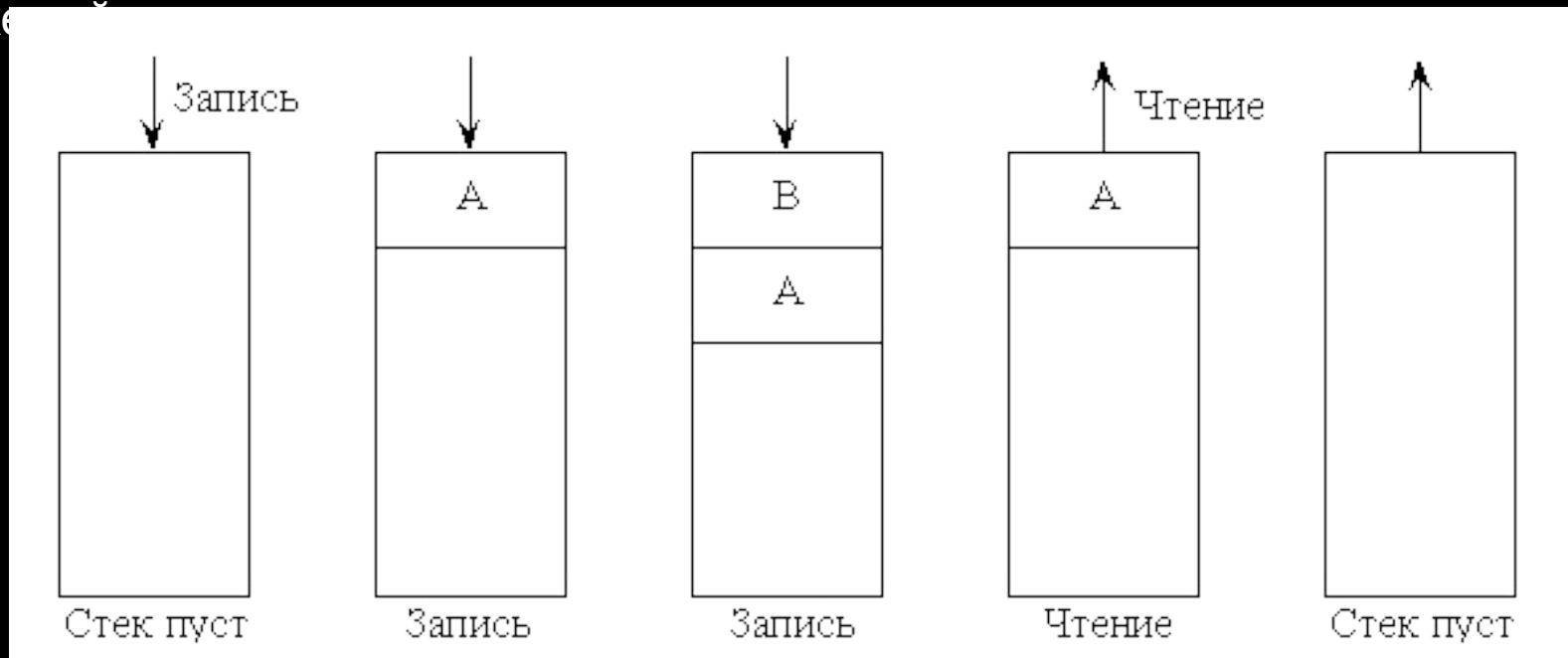
Форматы команд и данных.

- **Индексная адресация**-Для работы программ с массивами, требующими однотипных операций над элементами массива, удобно использовать индексную адресацию.



Форматы команд и данных.

- **Стековая адресация**-Стековая память широко используется в современных ЭВМ. Хотя адрес обращения в стек отсутствует в команде, он формируется схемой



Форматы команд и данных.

- **Автоиндексная адресация**-При автоиндексации косвенный адрес, находящийся в регистре РП, автоматически увеличивается (автоинкрементная адресация), или уменьшается (автодекрементная адресация) на постоянную величину до или после выполнения операции
- **Относительная адресация**-При относительной адресации применяется способ вычисления адреса путем суммирования кодов, составляющих адрес.

$$A = B + I + C$$

$$A = B + C$$

$$A = I + C$$

- **Непосредственная адресация**-При непосредственной адресации Операнд располагается непосредственно в адресном поле команды.

Ос как средство управления ресурсами типовой ЭВМ.

- Операционная система, применяющая такие алгоритмы, становится средством управления всеми ресурсами компьютера. Это в конечном счете и определяет общий алгоритм работы вычислительной системы, включая характеристики производительности, область применения и даже пользовательский интерфейс. К числу основных ресурсов современных вычислительных систем могут быть отнесены такие ресурсы, как процессоры, основная память, таймеры, наборы данных, диски, накопители на магнитных лентах, принтеры, сетевые устройства и некоторые другие.
- Ресурсы распределяются между процессами. Процесс (задача) представляет собой базовое понятие большинства современных ОС и часто кратко определяется как программа в стадии выполнения.
- Программа — это статический объект, представляющий собой файл с кодами и данными.
- Процесс — это динамический объект, который возникает в операционной системе после того, как пользователь или сама операционная система решает «запустить программу на выполнение», то есть создать новую единицу вычислительной работы. Например, ОС может создать процесс в ответ на команду пользователя `run prgl. exe`, где `prgl. exe` — это имя файла, в котором хранится код программы.

Ос как средство управления ресурсами типовой ЭВМ.

Управление ресурсами включает решение следующих общих, не зависящих от типа ресурса задач:

- Планирование ресурса — то есть определение, какому процессу, когда и в каком количестве (если ресурс может выделяться частями) следует выделить данный ресурс;
- Удовлетворение запросов на ресурсы;
- Отслеживание состояния и учет использования ресурса — то есть поддержание оперативной информации о том, занят или свободен ресурс и какая доля ресурса уже распределена;
- Разрешение конфликтов между процессами.

Таким образом, управление ресурсами составляет важную часть функций любой операционной системы, в особенности мультипрограммной. В отличие от функций расширенной машины большинство функций управления ресурсами выполняются операционной системой автоматически и прикладному программисту недоступны.