

# Архитектура вычислительных машин и систем

доцент кафедры ВТ ИИТ, ктн Мусихин А.Г.

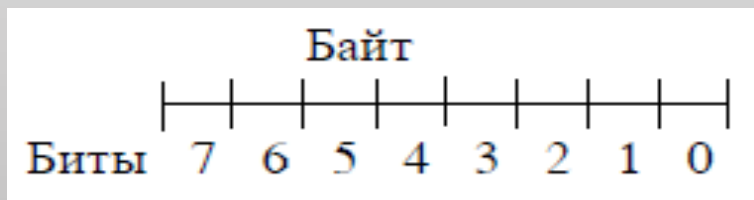
е-mail: [musikhin@mirea.ru](mailto:musikhin@mirea.ru) , [mag.iit.kvt@gmail.com](mailto:mag.iit.kvt@gmail.com)

## Лекция 5

# Основные устройства ВМ. Структура и характеристики памяти

# Структура и характеристики памяти

- **Структура памяти.** Память состоит из огромного числа элементов памяти, каждый из которых может находиться в одном из двух состояний, кодируемых двоичной цифрой 0 или 1 (**биты**, кодирующие состояния элементов). Биты памяти группируются в более крупные единицы хранения информации, минимальной из которых является **байт**.



- Одной из характеристик ЭВМ является **длина машинного слова**. В различных моделях ЭВМ (точнее, процессорах) размер машинного слова различен (8 бит, 16 бит, 32 бита, 64 бита) и зависит от соответствующей характеристики шины данных системной магистрали.
- **Информационная емкость** (или просто емкость) памяти выражается в количестве битов, байтов или слов; так как эта емкость может быть очень велика, обычно используют более крупные единицы, образованные присоединением приставок К (кило -  $2^{10}=1024$ ), М (мега -  $2^{20}=1048676$ ), Г (гига –  $2^{30}=1073741824$ ), Т (тера -  $2^{40}=1\ 099\ 511\ 627\ 776$ ) к перечисленным единицам.

- **Операции с памятью.** В отличие от других функциональных схем, занимающихся переработкой и преобразованием информации, запоминающие устройства выполняют только одну функцию - хранение информации. Поэтому к запоминающим устройствам могут быть применены только две операции - **ЧТЕНИЕ (ВЫБОРКА)** и **ЗАПИСЬ**.
- **Цикл считывания и цикл записи** определяются, соответственно, как время от момента выдачи процессором сигнала на считывание или запись и поступления информации, необходимой для выполнения операции, до того момента, когда заканчиваются все действия, связанные с выполняемой операцией, и память будет готова реализовать следующую операцию. Иногда, при совпадении продолжительности этих циклов, используют обобщенный термин - **цикл памяти**.
- **Способ выборки информации.** Различают два основных типа запоминающих устройств: с **произвольной** выборкой и **последовательной** выборкой. При произвольной выборке время доступа к заданному слову не зависит от местонахождения этого слова в памяти, при последовательной - зависит.

- **Методы размещения и поиска информации.** По этим признакам различают следующие виды основной памяти ЭВМ: **адресная, стековая и ассоциативная**. Доступ к информации в **адресной памяти** производится по **адресу** ячейки, начиная с которой информация в памяти располагается (очевидно, что в общем случае информационный массив может занимать несколько ячеек).

При размещении информации в **стековой памяти** возможен доступ только к ее **верхушке**, хотя на практике это требование может быть смягчено. Поиск информации в **ассоциативной памяти** производится по **содержимому** ее ячеек. **Возможность изменения информации.** Существуют элементы памяти с легко изменяемыми состояниями, способные работать как в режиме чтения, так и в режиме записи. Есть более дешевые элементы памяти, которые один раз установленные в 0 или 1, изменить свое состояние при функционировании ЭВМ не могут. Запоминающие устройства, построенные на таких элементах, используются только в режиме чтения.

- **Сохранение информации при отключении питания ЭВМ.** В некоторых видах запоминающих устройств происходит потеря информации при отключении (и даже кратковременном прерывании) питающего напряжения. Такие запоминающие устройства называют энергозависимыми. В энергонезависимых запоминающих устройствах (например, магнитной памяти) информация сохраняется.

# Классы запоминающих устройств

**Общая классификация.** По своему **функциональному назначению** запоминающие устройства можно разделить на крупные классы:

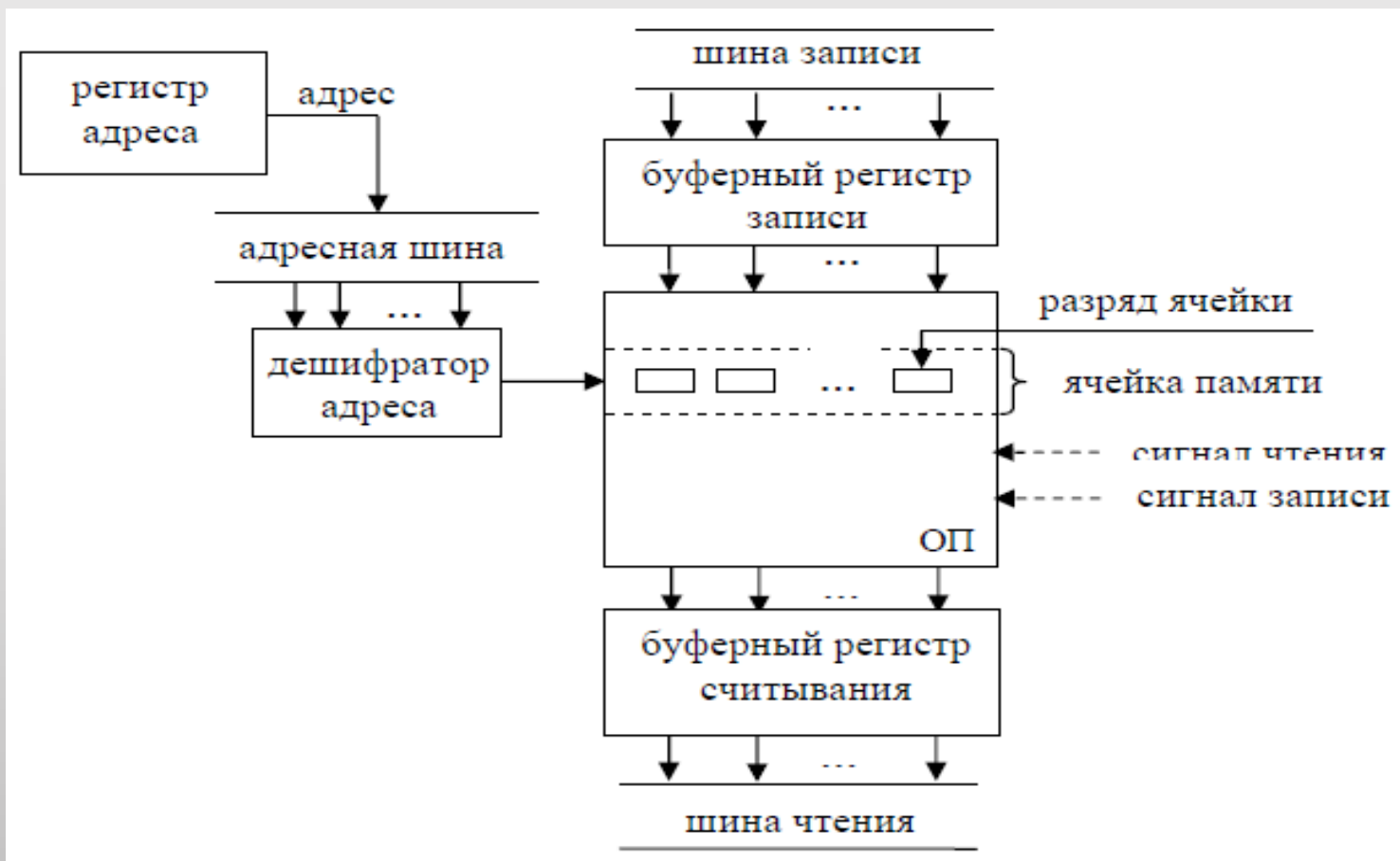
- регистровые внутренние запоминающие устройства;
- КЭШ-память;
- основная память;
- внешние запоминающие устройства (ВЗУ).

### **Состав основной памяти:**

- 1) Оперативное запоминающее устройство (ОЗУ), другое название - запоминающее устройство с произвольной выборкой (ЗУПВ), или в английской терминологии RAM (Random Access Memory). Оперативные запоминающие устройства всех классов ЭВМ представляют собой энергозависимые ЗУПВ.
- 2) Постоянное запоминающее устройство (ПЗУ). В английской терминологии такого рода память называют ROM (Read Only Memory) - память, работающая только на считывание. Информация, находящаяся в такой памяти, заранее закладывается при ее изготовлении ("зашивается") и при отключении питания не разрушается.



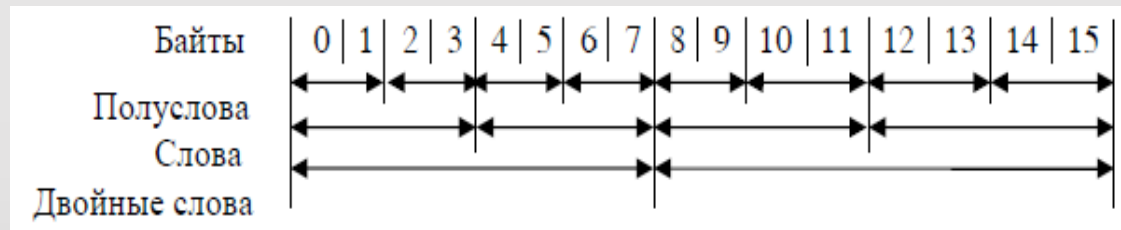
# Структура ОЗУ



# Логическая структура оперативной памяти

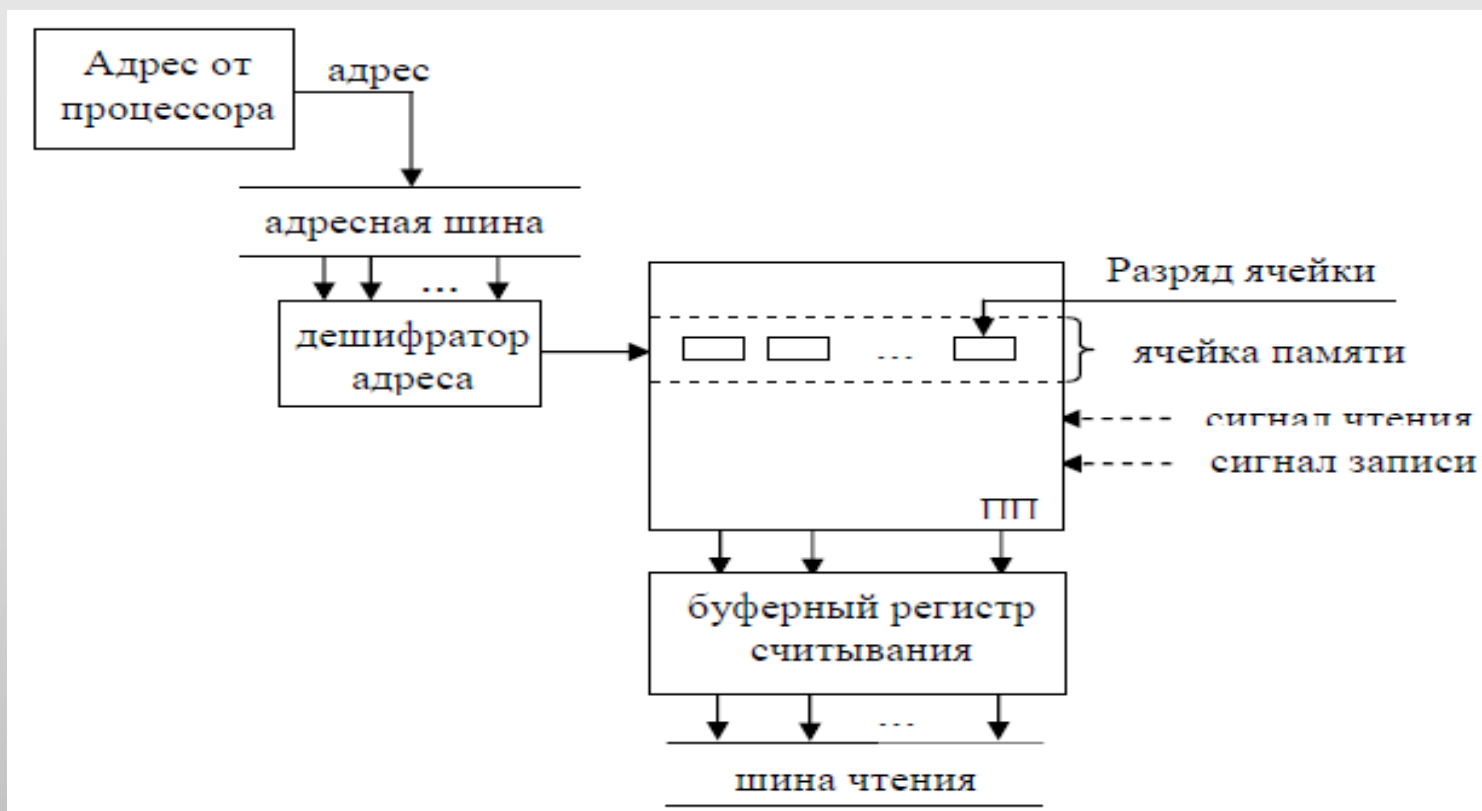
Адрес байта памяти	Байты памяти
0000	10011101
0001	01110010
0002	10101011
...	...

Память из байтов



Логические хранилища информации в ОП

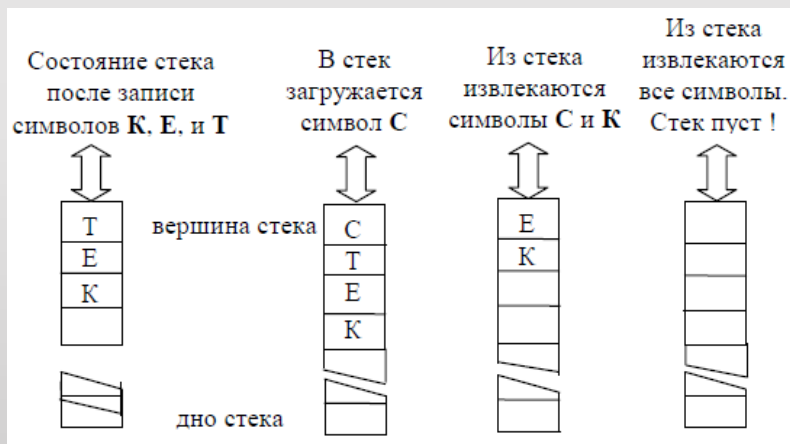
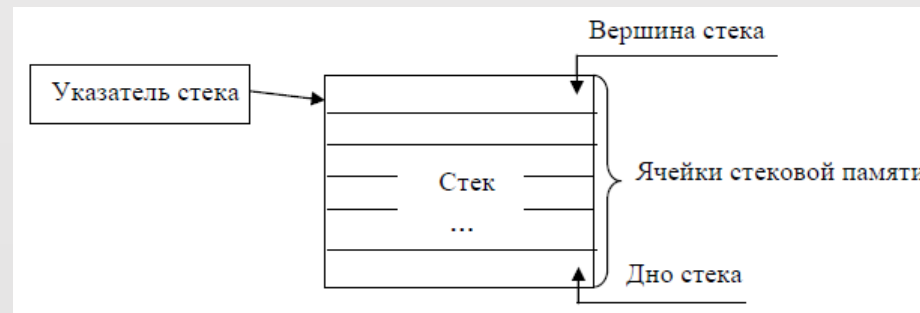
# Постоянное запоминающее устройство



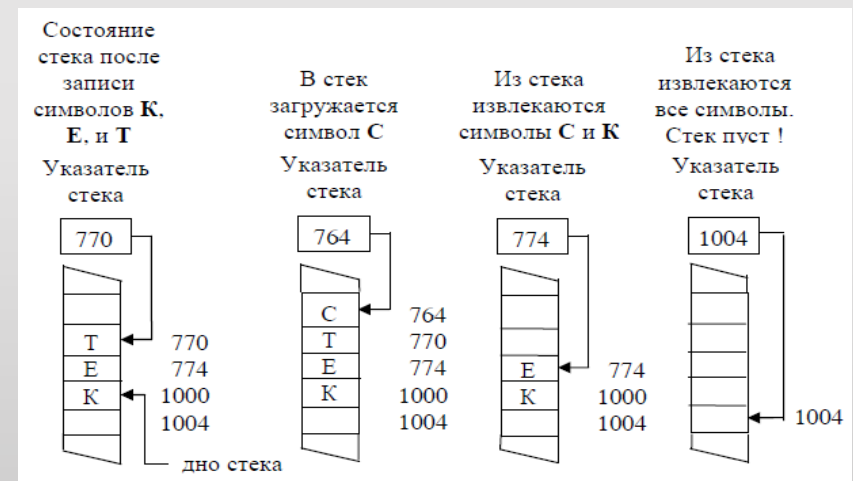
# Стековая память

- Стековая память состоит из ячеек, причем обмен информацией между остальными устройствами ЭВМ и стеком всегда выполняется только через верхнюю ячейку – вершину стека. При записи нового слова (команды, числа, символа и т.п.) все ранее записанные слова сдвигаются на одну ячейку вниз, а новое слово помещается в вершину стека. Считывание возможно только с вершины стека и производится с удалением или без удаления считываемого слова. Такую память часто называют памятью типа LIFO (Last – In First – Out – последним вошел, первым вышел).

## Структура стека



Аппаратная реализация стека



Модель стека, реализованная в оперативной памяти

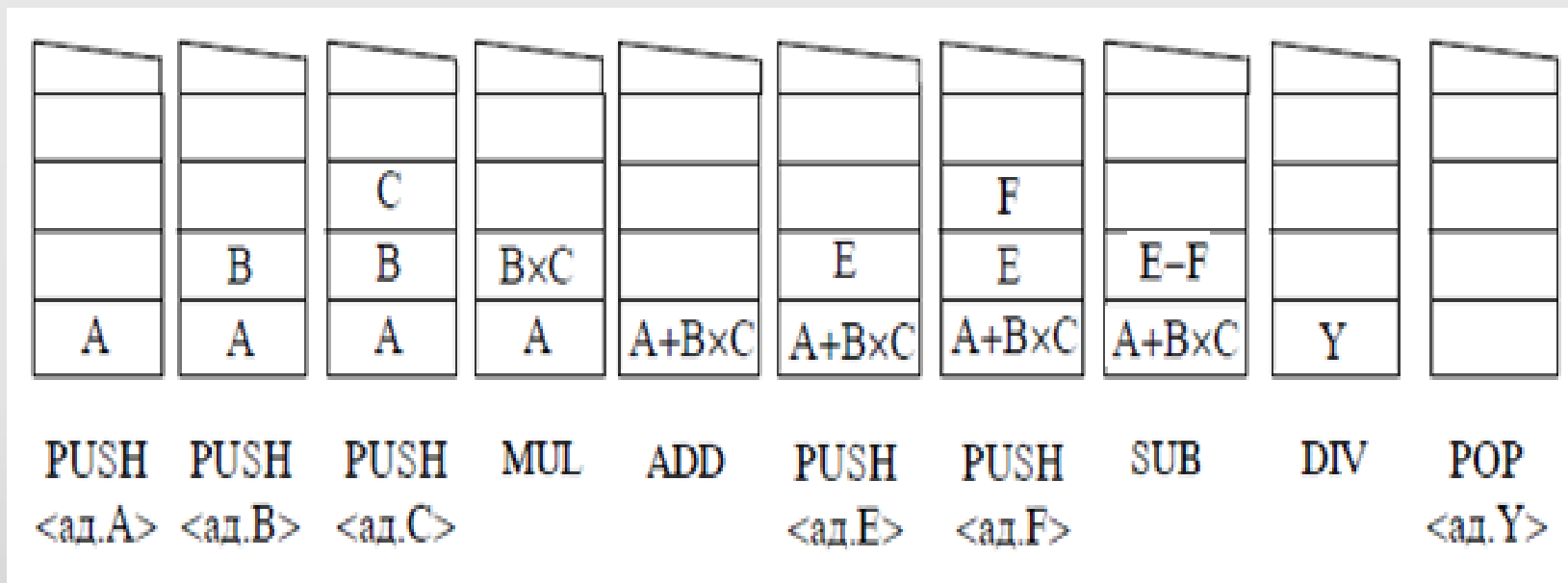
Для вычисления арифметических выражений с использованием стека используют Польскую Инверсную (бесскобочную) Запись (ПОЛИЗ). В ПОЛИЗе операция записывается не между операндами ( $X+Y$ ), а после них ( $XY+$ ).

Подобные выражения можно вычислять по следующему алгоритму:

- 1) проанализировать каждый символ бесскобочной записи формулы, начиная с крайнего левого символа, до тех пор, пока не встретится знак операции;
- 2) взять ближайших два операнда (если операция двуместная) или один операнд (при унарной операции), расположенные слева от обнаруженного знака операции, выполнить операцию и результат поместить в формулу на место выделенных операндов и знака операции;
- 3) если после выполнения пункта 2) формула состоит из одного значения, это значение и есть результат, то есть алгоритм завершен, в противном случае перейти к пункту 1).

- Формула  $Y = (A+B*C)/(E-D)$  преобразуется в ПОЛИЗе к виду  
 $Y=ABC*+ED- /$
- При значениях  $A=8$ ,  $B=2$ ,  $C=5$ ,  $E=6$  и  $D=4$  процесс вычисления представлен в таблице:

Шаг	Формула, подлежащая расчету	Левый знак операции	Операнды	Результат	Новая формула после выполнения операции
1	8 2 5 * + 6 4 - /	-	2 и 5	10	8 10 + 6 4 - /
2	8 10 + 6 4 - /	+	8 и 10	18	18 6 4 - /
3	18 6 4 - /	-	6 и 4	2	18 2 /
4	18 2 /	/	18 и 2	9	9



Пример использования стека для вычисления выражений



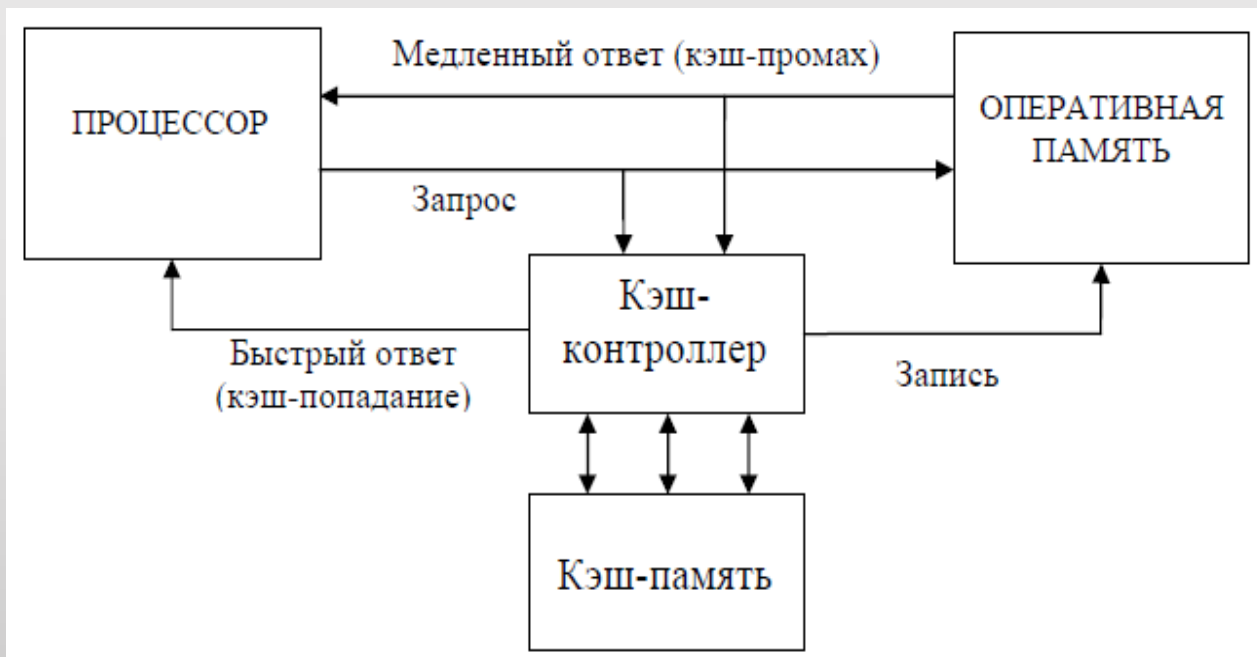
# Ассоциативные запоминающие устройства

- АЗУ обеспечивает возможность поиска информации, хранящейся в ассоциативной памяти (АП), по содержимому ячеек (ассоциативному признаку, контексту), а не по адресам.

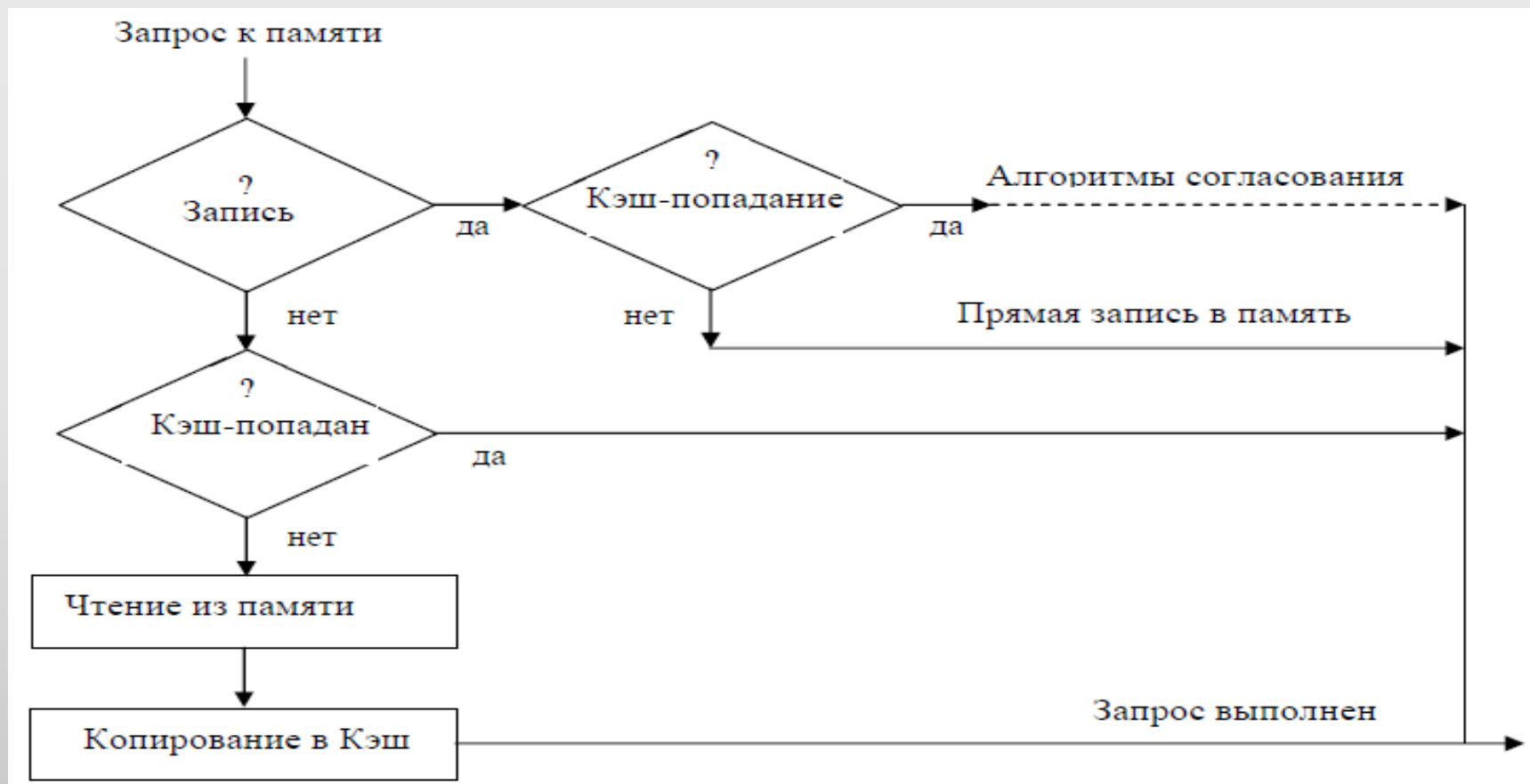
Биты		0	1	2	3	4	5	6	7		
		1	0	1	1	0	1	1	0	– регистр маски	
		0	0	1	0	1	0	1	0	– регистр контекста	
ячейки Ассоциативной памяти	{	1	1	0	0	1	1	1	1	0	– регистр совпадений
		0	1	1	0	0	0	1	1	1	
		1	0	1	0	0	1	0	0	0	
		0	1	1	1	0	1	0	1	0	
		0	0	1	0	1	0	1	0	1	

- Важнейшим свойством АЗУ является то, что проверка содержимого проводится одновременно для всех ячеек ассоциативной памяти, что существенно ускоряет поиск по сравнению с последовательным адресным обращением к ячейкам ОЗУ.
- Такое ускорение достигается за счет усложнения схемы управления памятью, поскольку для осуществления параллельного (одновременного) просмотра каждая ячейка должна снабжена локальным блоком управления, что резко повышает стоимость такой памяти.
- В современных серийных моделях ЭВМ АЗУ нашли применение в некоторых архитектурах "сверхоперативных" запоминающих устройств, чаще называемых памятью типа кэш.

# Кэш-память



- **Пространственная локальность.** Если произошло обращение по некоторому адресу, то с высокой степенью вероятности в ближайшее время произойдет обращение к соседним адресам. Пространственная локальность позволяет надеяться, что имеет смысл копировать целый блок данных в кэш, так как, скорее всего, в ближайшее время обращение будет к данным из этого блока.
- **Временная локальность.** Учет циклических участков программы. Если произошло обращение по некоторому адресу, то в ближайшее время с большой вероятностью произойдет обращение к этому же адресу.



Алгоритм работы кэш-памяти

- **Эффективность кэш** - отношение числа попаданий к общему количеству обращений процессора. Эффективность, таким образом, — это число от 0 до 1. Нулевая эффективность означает, что кэш нисколько не ускорил работу системы; эффективность, равная единице, означает, что ускорение максимально, и время обращения к памяти определяется скоростью работы кэша, а не скоростью работы оперативной памяти.

Эффективность кэша зависит от факторов:

- **Объём**
- **Алгоритм функционирования**
- **Выполняемая процессором программа**

Адрес данных в ОП	Блоки данных	Управляющая информация
-------------------	--------------	------------------------

### Структура строки кэш – памяти

Пусть имеется ОП емкостью 256К байтов ( $256K = 2^{18}$ ); для адресации такой памяти необходим 18-разрядный адрес.

Пусть ОП разбивается на блоки по 16 байтов в каждом. Очевидно, что ОП удобно рассматривать как линейную последовательность из  $16384 = 2^{14}$  блоков.

При такой организации 18-разрядный адрес можно условно разделить на две части:

младшие 4 разряда определяют адрес байта в пределах блока,  
старшие 14 разрядов определяют номер одного из 16384 блоков.

В дальнейшем старшие 14 разрядов адреса ОП будем называть **адресом блока ОП**.

Пусть имеется кэш-память емкостью  $2K$  байтов ( $2K=2^{11}$ ). Таким образом, для адресации кэш-памяти необходим 11-разрядный адрес.

В строку (ячейку) кэш-памяти отображается блок ОП, следовательно, кэш содержит  $128=2^7$  слов. 11-разрядный адрес слова в кэш-памяти можно представить состоящим из двух частей:

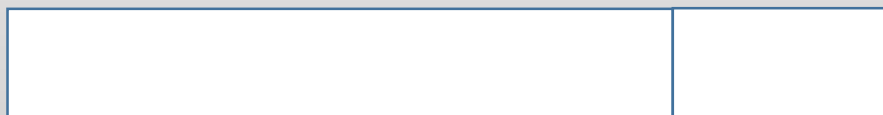
адрес байта в строке (4 младших разряда),

адрес строки кэш-памяти (7 старших разрядов).

Так как расположение байтов в блоке ОП и строке кэш одинаково, доступ к конкретному байту в строке кэш-памяти определяется 4 младшими разрядами адреса ОП.

Остается задача преобразования 14-разрядного адреса блока ОП в 7-разрядный адрес строки кэш-памяти, то есть способа определения взаимного соответствия строки кэш-памяти и области ОП.

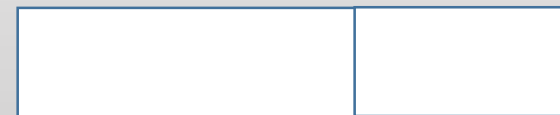
Адрес ячейки памяти – 18 разрядов



Адрес блока памяти – 14 разрядов

Адрес  
смещения  
байта в  
блоке  
памяти – 4  
разряда

Адрес ячейки кэш-памяти – 11 разрядов



Адрес блока кэш-памяти – 7 разрядов

Адрес  
смещения  
байта в  
блоке  
памяти – 4  
разряда



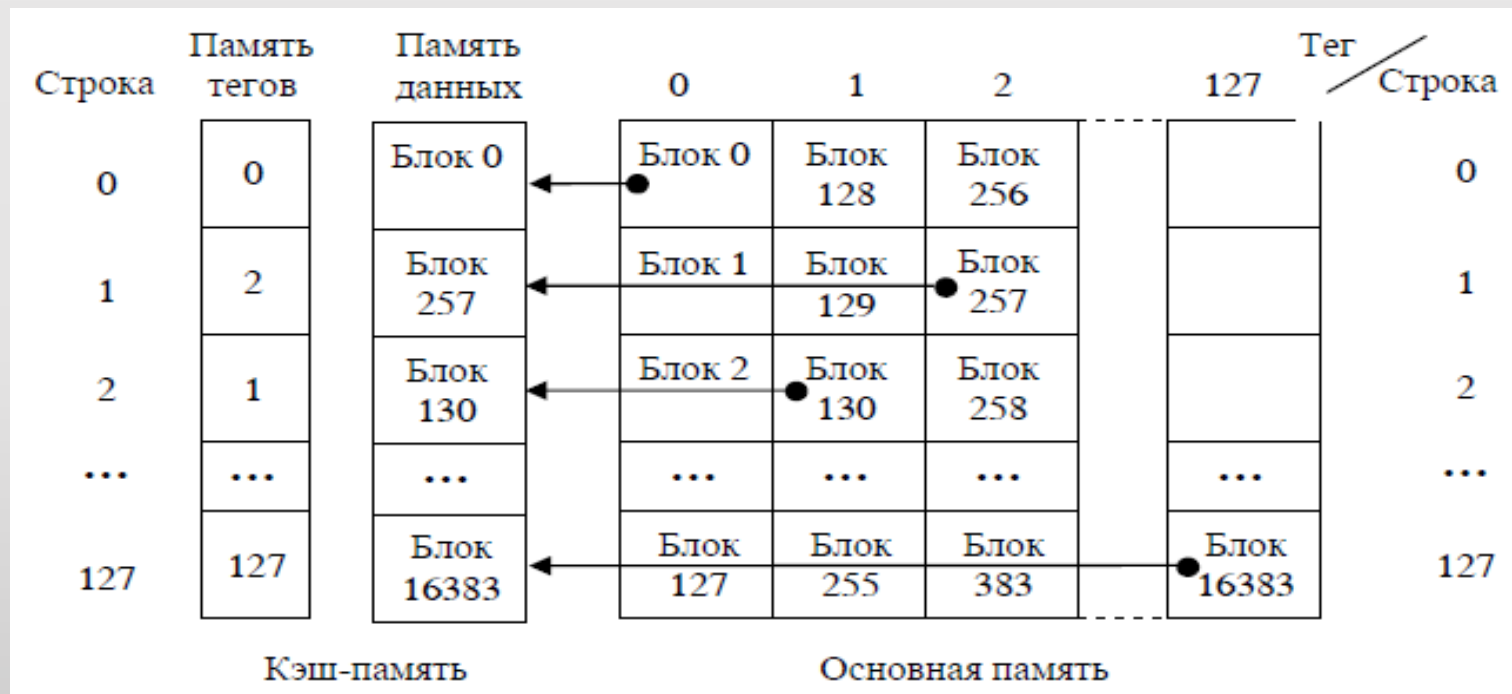
По способу определения взаимного соответствия строки кэш-памяти и области ОП различают три архитектуры кэш-памяти:

- кэш прямого отображения,
- полностью ассоциативный кэш,
- комбинация первых двух подходов - частично-ассоциативный кэш.

**Прямое отображение.** При прямом отображении адрес  $i$ -й строки кэш-памяти, на которую может быть отображен  $j$ -й блок ОП, однозначно определяется выражением  $i = j \bmod m$ , где  $m$  – общее количество строк в кэш-памяти. В примере  $i = j \bmod 128$ , где  $i$  может принимать значения от 0 до 127, а  $j$  – от 0 до 16383.

Таким образом, на строку кэш-памяти с номером  $i$  отображается каждый из 128 блоков ОП, начиная с номера  $i$ .

## Организация кэш-памяти с прямым отображением:



# Спасибо за внимание!