

# Структурная и функциональная организация ЭВМ (Computer Organization and Design)

БГУИР  
кафедра ЭВМ

Лекция 6  
«Организация памяти-I»

2019

## План лекции

1. Характеристики запоминающих устройств
2. Основная проблема памяти
3. Иерархия памяти
4. Триггеры и защёлки
5. Регистры
6. Микросхемы памяти

# Память

Память предназначена для фиксации, хранения и выдачи информации в процессе работы ЭВМ. Процессы чтения и записи информации определяются как процессы обращения к запоминающему устройству (ЗУ).

ЗУ характеризуются:

Местом расположения (на кристалле ЦП, на плате, внешняя память).

Ёмкостью

Единицей пересылки

Методом доступа

Быстродействия

Физическим типом (полупроводники, магнитный носитель, оптика)

Физические особенности (энерго – зависимая/независимая)

Стоимостью

# Память

Емкость ЗУ характеризуют числом битов либо байтов, которое может храниться в запоминающем устройстве.

На практике применяются более крупные единицы – с приставками кило,мега, гига, тера, пета, экса (kilo, mega, giga, tera, peta, exa) –  $10^3, 10^6, 10^9, 10^{12}, 10^{15}$  и  $10^{18}$ .

Близко, но не равно  $2^{10}, 2^{20}, 2^{30}, 2^{40}, 2^{50}$  и  $2^{60}$ .

В последнее время IEEE лоббирует принятие новых обозначений «-binary»: kilobinary, megabinary, gigabinary, terabinary, petabinary, exabinary.

В результате вместо: “килобайт” – «кибибайт», мегабайт – «мебибайт» и т.п.

Сокращённые обозначения – Ki, Mi, Gi, Ti, Pi и Ei.

# Память

Единица пересылки – обычно равна ширине ШД (слову), но не обязательно. Из внешней памяти обычно данные передаются блоками.

Метод доступа к ЗУ:

## **Последовательный доступ.**

*Хранение информации в виде последовательности блоков данных, называемых записями. Для доступа к нужному элементу (слову или байту) необходимо прочитать все предшествующие ему данные. (Магнитные ленты)*

**Прямой доступ.** *Каждая запись имеет уникальный адрес, отражающий ее физическое размещение на носителе информации. Обращение - адресный доступ к началу записи и последующий последовательный доступ к единице информации внутри записи. (Жёсткий диск)*

# Память

**Произвольный доступ.** Каждая ячейка памяти имеет уникальный физический адрес. Обращение к любой ячейке занимает одно и то же время и может проводиться в произвольной очередности. (ОЗУ).

**Ассоциативный доступ.** Поиск ячеек, содержащих информацию, в которой значение отдельных битов совпадает с состоянием одноименных битов в заданном образце.

Сравнение осуществляется параллельно для всех ячеек памяти, независимо от ее емкости. (КЭШ-память)

# Память

## Быстродействие ЗУ:

**Время доступа** - Для памяти с произвольным доступом оно соответствует интервалу времени от момента поступления адреса до момента, когда данные заносятся в память или становятся доступными.

В ЗУ с подвижным носителем информации это время, затрачиваемое на установку головки записи/считывания (или носителя) в нужную позицию.

# Память

## Быстродействие ЗУ:

**Длительность цикла памяти или период обращения (ТЦ).** Понятие применяется к памяти с произвольным доступом, для которой оно означает *минимальное время между двумя последовательными обращениями к памяти. Период обращения включает в себя время доступа плюс некоторое дополнительное время.*

Дополнительное время может требоваться для затухания сигналов на линиях, а в некоторых типах ЗУ, где считывание информации приводит к ее разрушению, - для восстановления считанной информации.



# Память

## Быстродействие ЗУ:

**Скорость передачи.** Это скорость, с которой данные могут передаваться в память или из нее. Для памяти с произвольным доступом она равна  $1/T_{\text{Ц}}$ . Для других видов памяти скорость передачи определяется соотношением:

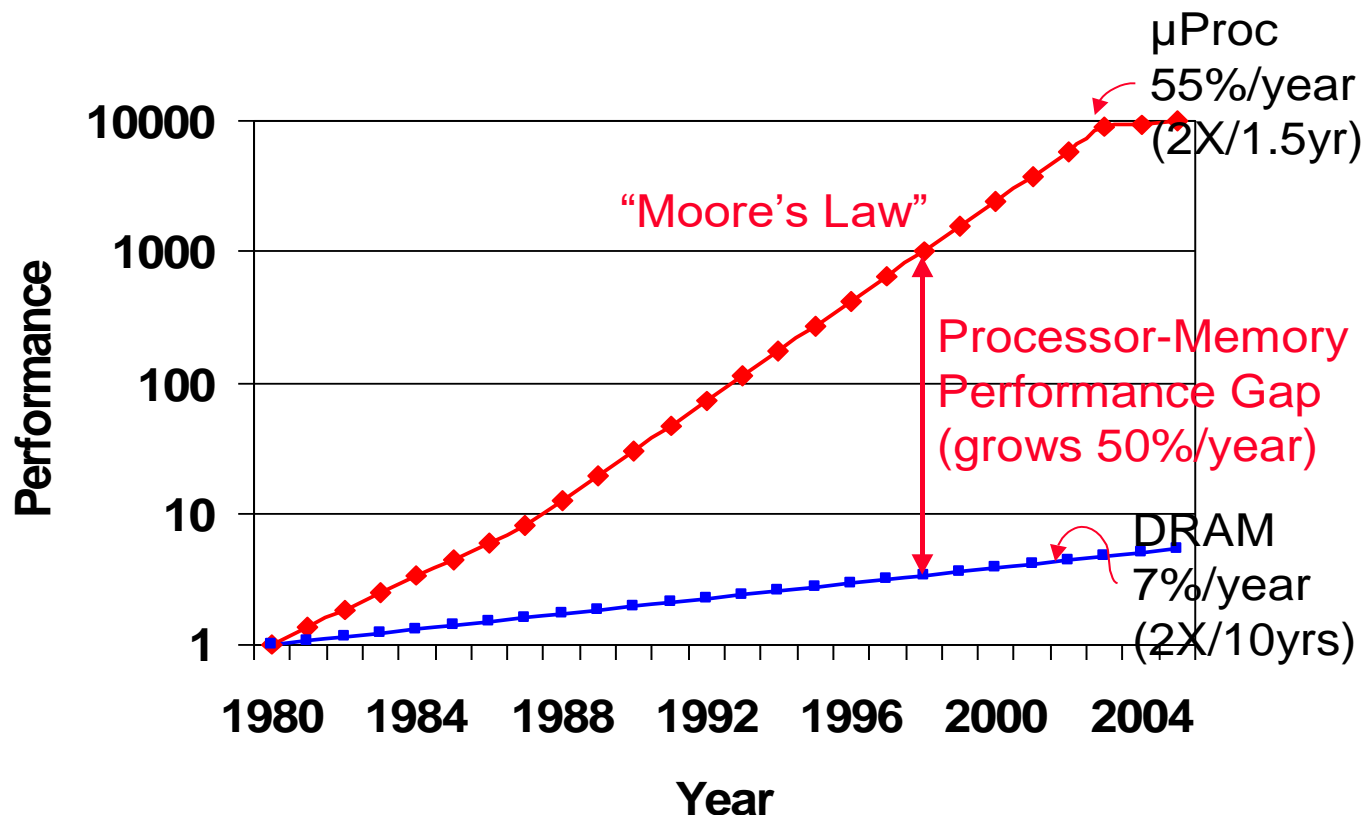
$$T_N = T_A + N/R ,$$

где  $T_N$  - среднее время считывания или записи  $N$  битов;  
 $T_A$  – среднее время доступа;  $R$  - скорость пересылки в битах в секунду.

**Стоимость** – отношение общей стоимости ЗУ к его ёмкости в битах -> стоимость хранения одного бита информации.

# Память. Основная проблема

## Processor-Memory Performance Gap



CSE431 L18 Memory Hierarchy.3

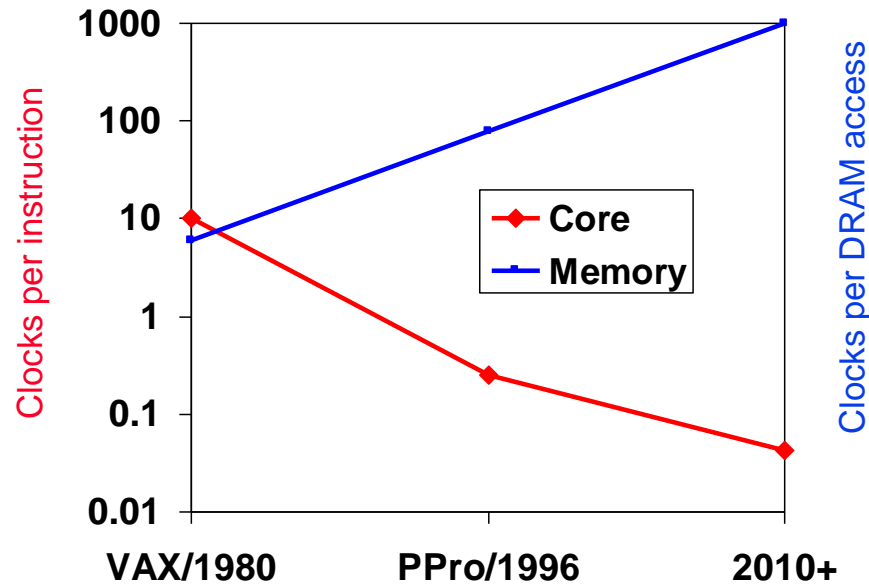
Irwin, PSU, 2005

Память отстаёт по быстродействию от ЦП

# Память. Основная проблема

## The “Memory Wall”

- Logic vs DRAM speed gap continues to grow



Память отстаёт по быстродействию от ЦП

# Иерархия запоминающих устройств

Чем меньше время доступа, тем выше стоимость хранения бита.

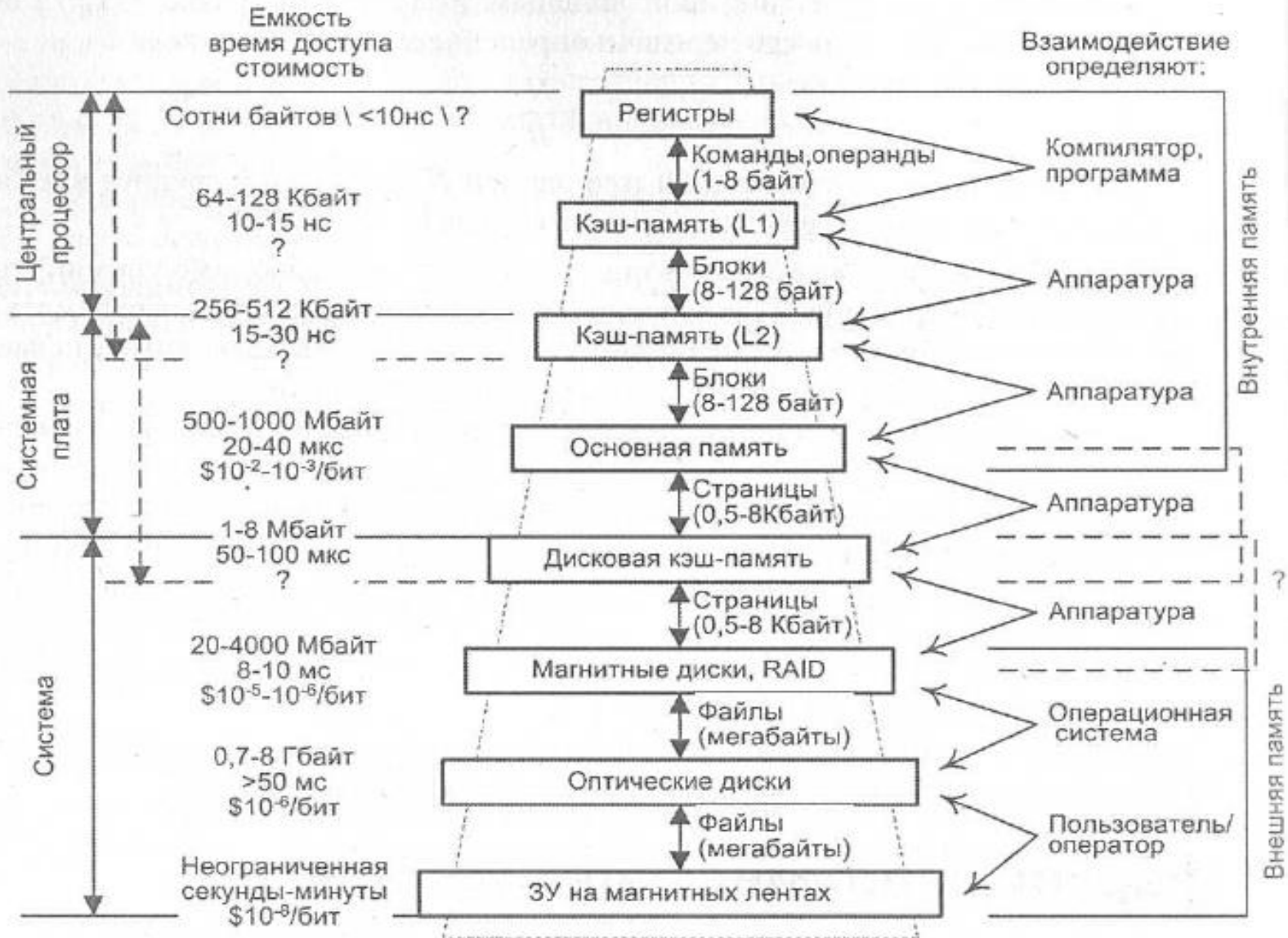
Чем больше ёмкость, тем ниже стоимость хранения, но больше время доступа.

При создании системы памяти постоянно приходится решать задачу обеспечения требуемой емкости и высокого быстродействия за приемлемую цену.

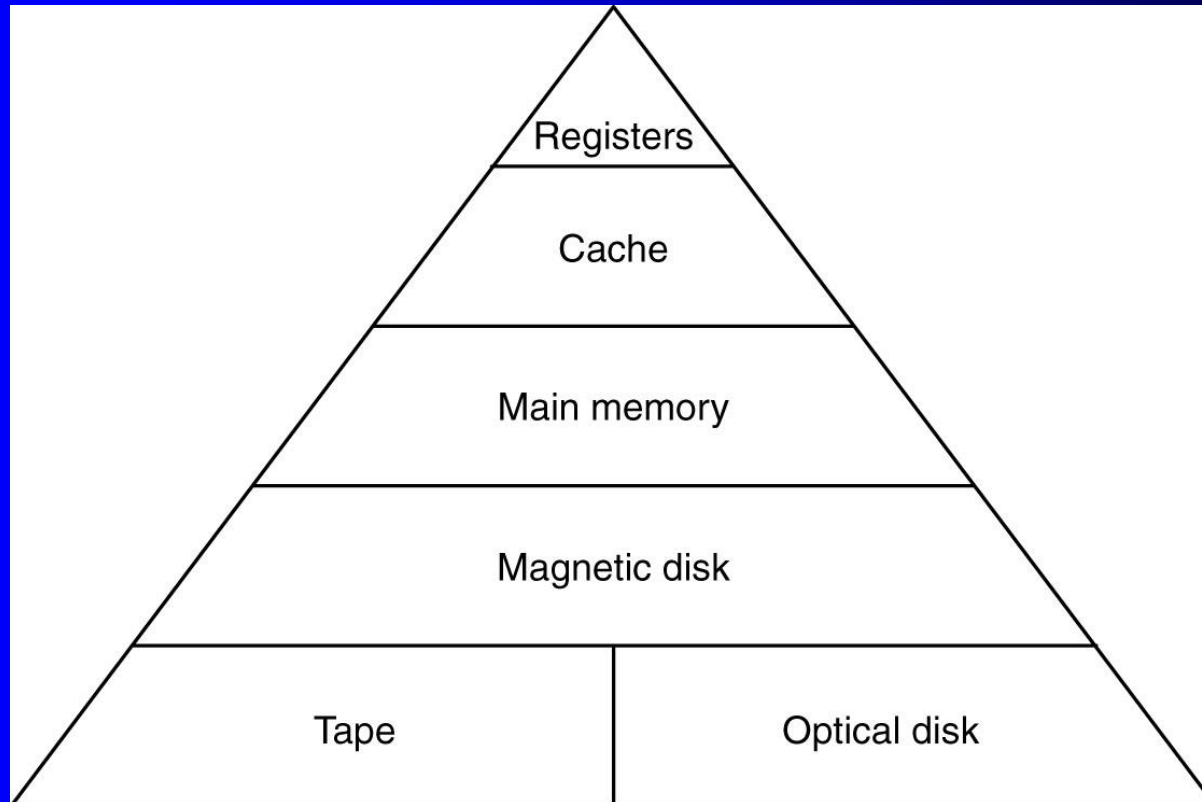
Наиболее эффективным решением является *создание иерархической памяти.*

*Уровни иерархии тесно связаны – все данные на одном уровне продублированы так же на более низком уровне, все данные с более низкого уровня – на следующем нижележащем уровне.*

# Иерархия запоминающих устройств



# Memory Hierarchies



A five-level memory hierarchy.

Чем ниже по пирамиде: меньше «стоимость/бит»,  
больше ёмкость, больше время доступа, меньше  
частота обращения ЦП к памяти.

# Иерархия запоминающих устройств

Принцип локальности по обращению. Адрес очередной команды – либо следует либо расположен рядом с текущей (*локальность по адресу*). Данные как правило так же структурируются (*лок-сть по данным*).

Кроме того, программы содержат множество небольших циклов и подпрограмм. Это означает, что небольшие наборы команд могут многократно повторяться в течение некоторого интервала времени, то есть имеет место временная локальность. Все три вида локальности объединяет понятие локальность по обращению. Принцип локальности часто облачают в численную форму и представляют в виде так называемого правила "90/10": *90% времени работы программы связано с доступом к 10% адресного пространства этой программы.*

# Иерархия запоминающих устройств

Из свойства локальности -> программу разумно представить в виде последовательно обрабатываемых фрагментов. Помещая такие фрагменты в более быструю память, можно существенно снизить общие задержки на обращение, *поскольку команды и данные, будучи один раз переданы из медленного ЗУ в быстрое, затем могут использоваться многократно и среднее время доступа к ним в этом случае определяется уже более быстрым ЗУ.*

Между каждыми двумя уровнями иерархии блоками данных своего размера – для пересылки.

При доступе к данным – сперва поиск на более высоком уровне: попадание (**hit**) или промах (**miss**). При очередном промахе – ещё на уровень ниже. При нахождении – блок информации пересылается наверх (пересылка только между 2-мя соседними уровнями!)



# Иерархия запоминающих устройств

При оценке эффективности подобной организации памяти обычно используют следующие характеристики:

- коэффициент попаданий (hit rate) - *отношение числа обращений к памяти, при которых произошло попадание, к общему числу обращений к ЗУ данного уровня;*
- коэффициент промахов (miss rate) - *отношение числа обращений к памяти, при которых имел место промах, к общему числу обращений к ЗУ данного уровня;*
- время обращения при попадании (hit time) - *время, необходимое для поиска нужной информации в памяти верхнего уровня, плюс время на фактическое считывание данных;*
- потери на промах (miss penalty) - *время, требуемое для замены блока в памяти более высокого уровня на блок с нужными данными, расположенный в ЗУ следующего (более низкого) уровня.*

# Иерархия запоминающих устройств

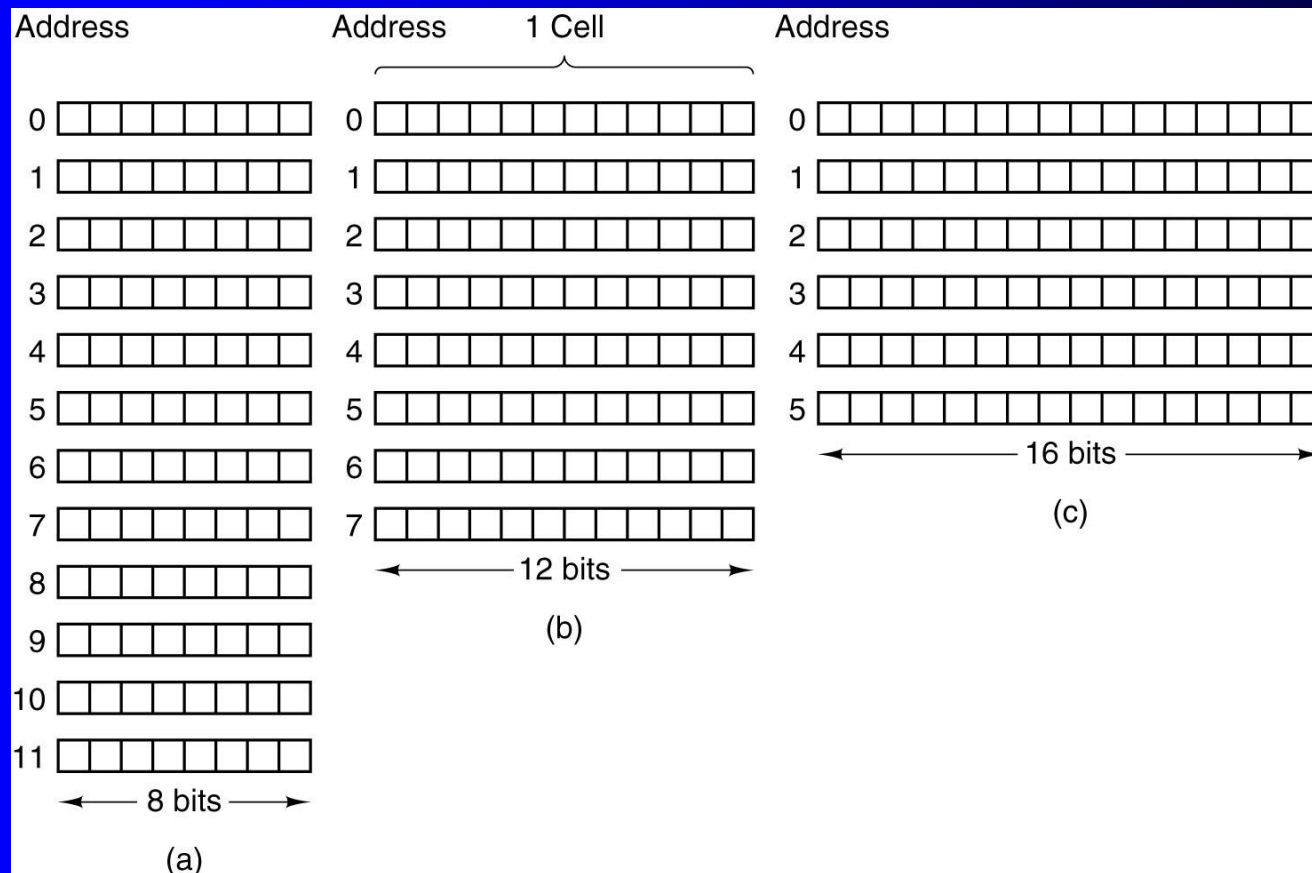
Описание некоторого уровня иерархии ЗУ предполагает конкретизацию четырех моментов:

- размещения блока - допустимого места расположения блока на примыкающем сверху уровне иерархии;
- идентификации блока - способа нахождения блока;
- замещения блока - выбора блока, заменяемого при промахе с целью освобождения места для нового блока;
- согласования копий (когерентность данных) - обеспечения согласованности копий одних и тех же блоков, расположенных на разных уровнях.

В некоторых ВМ фирмы IBM есть расширенная память (**expanded storage**) – меньшее быстродействие и стоимость по отношению к основной. В иерархию не входит – к ней запрещено обращение от внешних устройств.

# Primary Memory

## Memory Addresses (1)



Three ways of organizing a 96-bit memory.  
Каждая ячейка содержит фиксированное число  
запоминающих элементов.

# Primary Memory

## Memory Addresses (2)

Number of bits per cell for some historically interesting commercial computers

| Computer         | Bits/cell |
|------------------|-----------|
| Burroughs B1700  | 1         |
| IBM PC           | 8         |
| DEC PDP-8        | 12        |
| IBM 1130         | 16        |
| DEC PDP-15       | 18        |
| XDS 940          | 24        |
| Electrologica X8 | 27        |
| XDS Sigma 9      | 32        |
| Honeywell 6180   | 36        |
| CDC 3600         | 48        |
| CDC Cyber        | 60        |

# Основная память

Основная память (ОП) -> единственный вид памяти, к которой ЦП может обращаться непосредственно. Основная память - ЗУ с произвольным доступом.

Основная память может включать в себя два типа устройств:

- оперативные запоминающие устройства (ОЗУ) и
- постоянные запоминающие устройства (ПЗУ).

ОЗУ – RAM (Random Access Memory). Для большинства типов полупроводниковых ОЗУ характерна энергозависимость: *даже при кратковременном прерывании питания хранимая информация теряется. Микросхема ОЗУ должна быть постоянно подключена к источнику питания и поэтому может использоваться только как временная память.*

# Основная память

Вторую группу полупроводниковых ЗУ основной памяти образуют

энергонезависимые микросхемы ПЗУ (ROM - Read-Only Memory). ПЗУ обеспечивает считывание информации, но не допускает ее изменения (в ряде случаев информация в ПЗУ может быть изменена, но этот процесс сильно отличается от считывания и требует значительно большего времени).

Энергозависимые ОЗУ можно подразделить на две основные подгруппы:

динамическую память (DRAM - Dynamic Random Access Memory) и статическую память (SRAM - Static Random Access Memory).

# Основная память

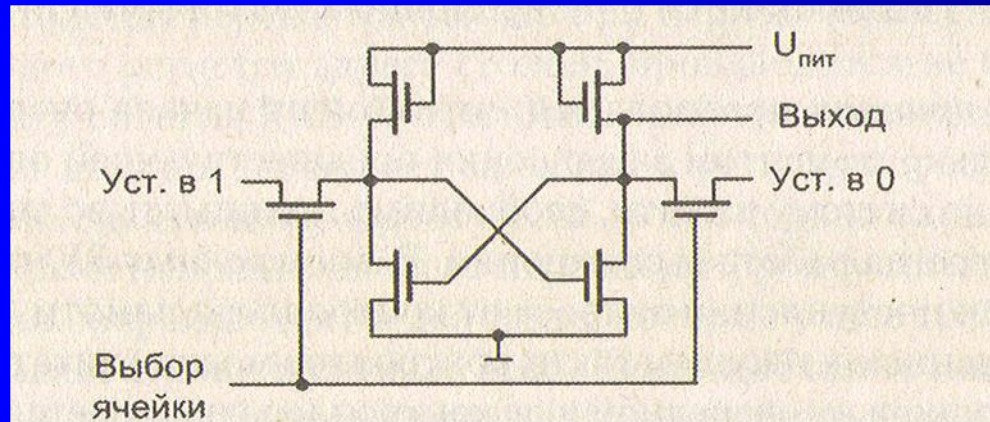
В **статических ОЗУ** запоминающий элемент может хранить записанную информацию неограниченно долго (при наличии питающего напряжения).

Запоминающий элемент **динамического ОЗУ** способен хранить информацию только в течение достаточно короткого промежутка времени, после которого информацию нужно восстанавливать заново, иначе она будет потеряна. Динамические ЗУ, как и статические, энергозависимы.

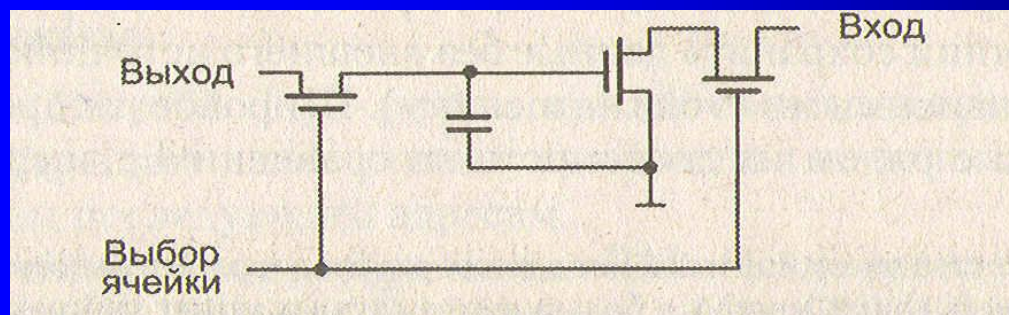
Роль запоминающего элемента в статическом ОЗУ выполняет триггер (схема с двумя устойчивыми состояниями, обычно состоящую из четырех или шести транзисторов).



# Статическое и динамическое ОЗУ



SRAM, на 6 транзисторах – быстрее, менее чувствительны к внешним излучениям

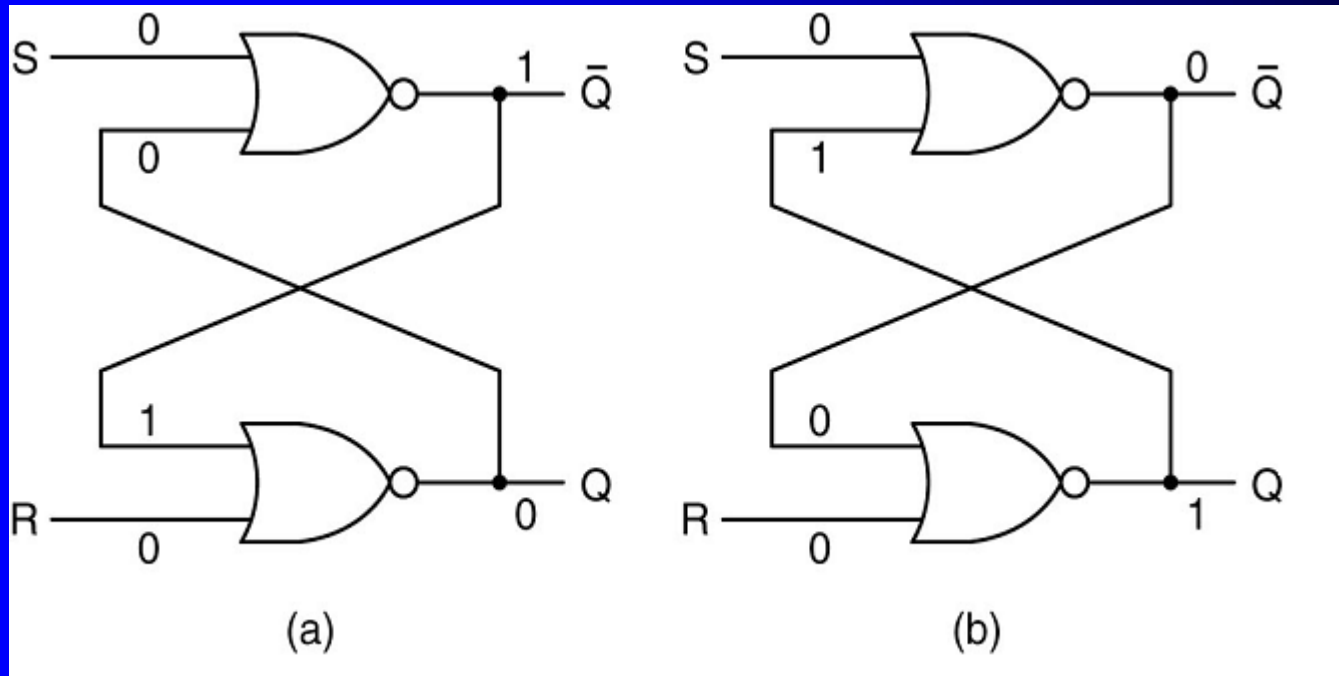


DRAM

Простота схемы DRAM позволяет достичь высокой плотности размещения, в итоге, *снизить стоимость. Главный недостаток - что накапливаемый на конденсаторе заряд со временем теряется. Среднее время утечки заряда DRAM составляет сотни(десятки) миллисекунд, поэтому, заряд следует регенерировать.*



# Latches - Защёлки



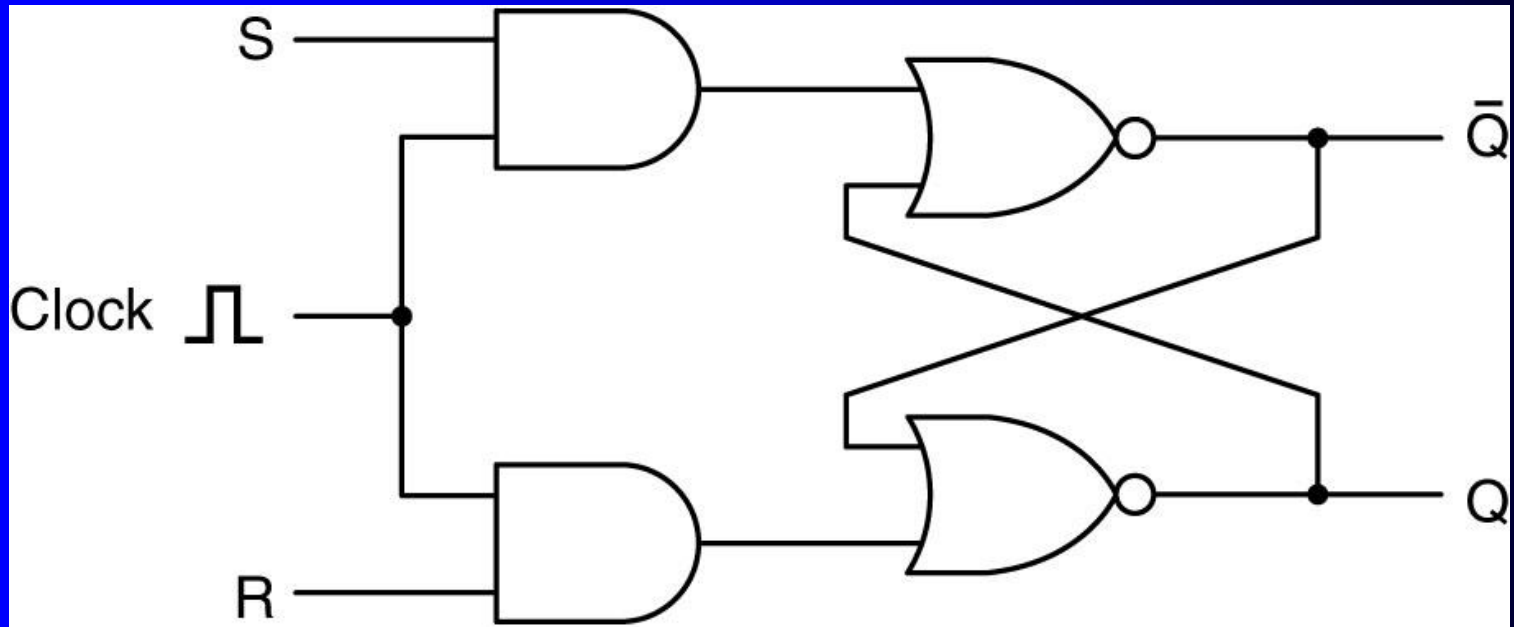
При  $R=S=0$  два стабильных состояния:

(a) Защёлка ИЛИ-НЕ (NOR) в состоянии 0.

(b) Защёлка ИЛИ-НЕ (NOR) в состоянии 1.

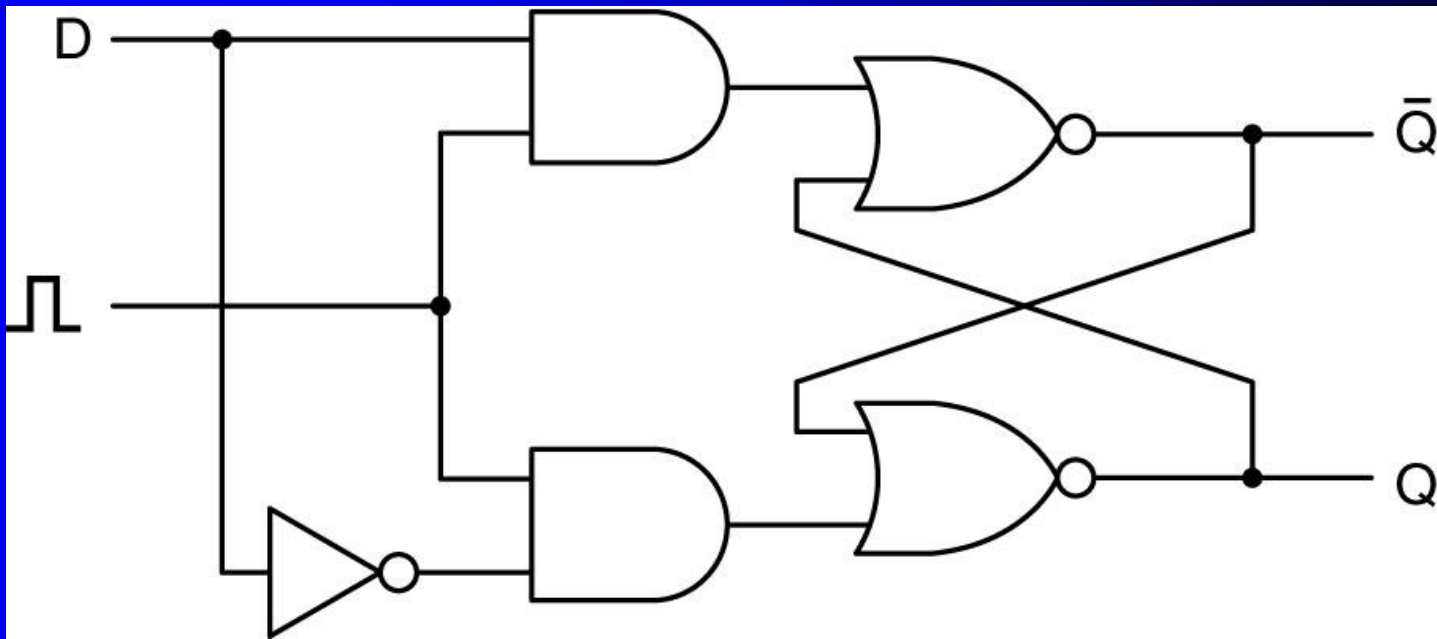
$R=S=1$  – запрещено в следствие неопределённости при возврате к  $R=S=0$ .

# Latches - Защёлки



Синхронные RS-защёлки.  
A clocked SR latch.

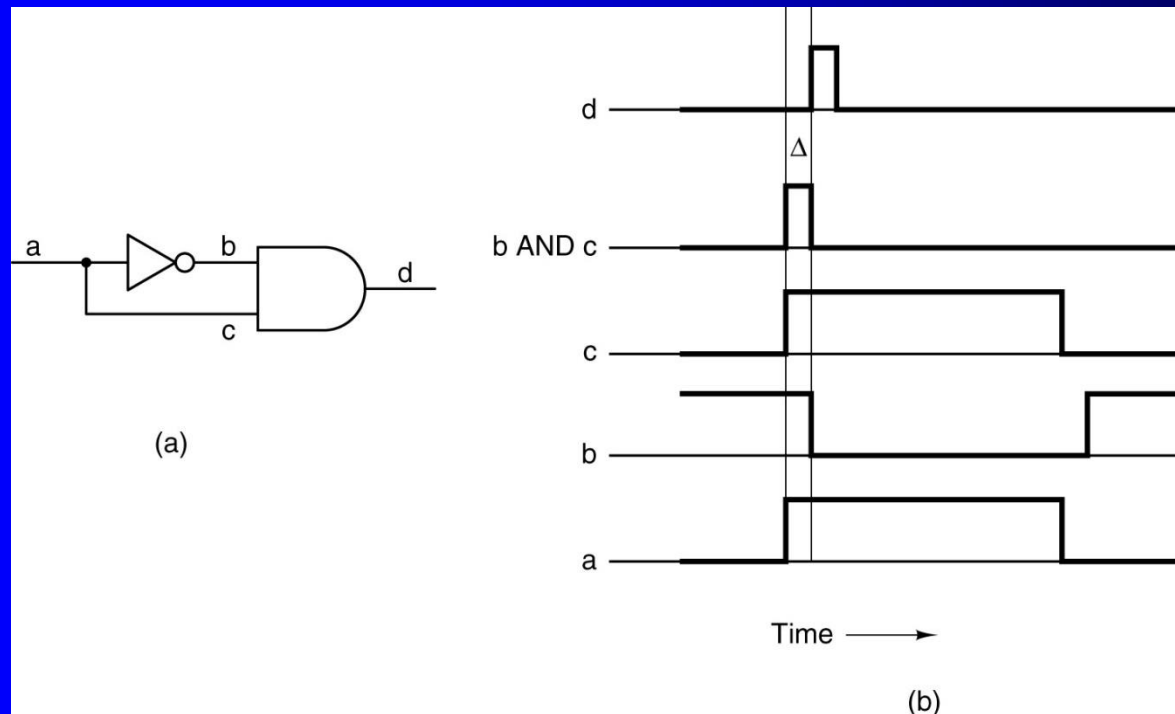
# Latches - Защёлки



Синхронная D защёлка - A clocked D latch.  
(Используется 11 транзисторов).

# Flip-Flops - Триггеры

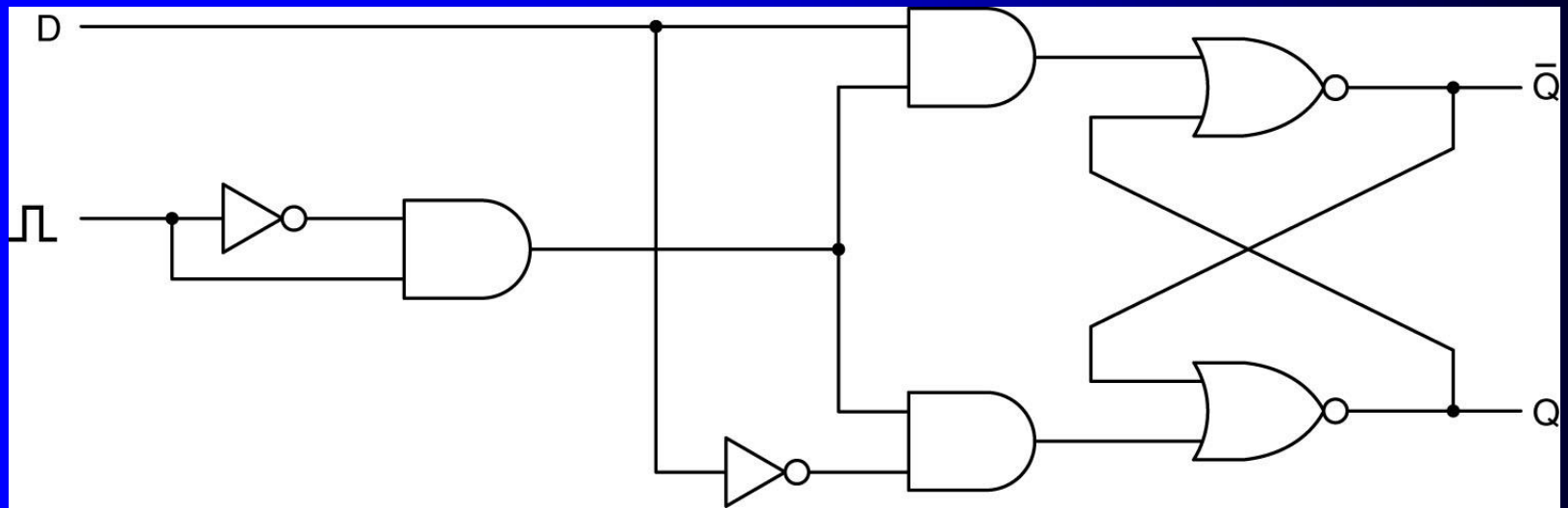
*Защёлки запускаются уровнем сигнала, триггеры – фронтом!*



(a) A pulse generator.

(b) Timing at four points in the circuit.

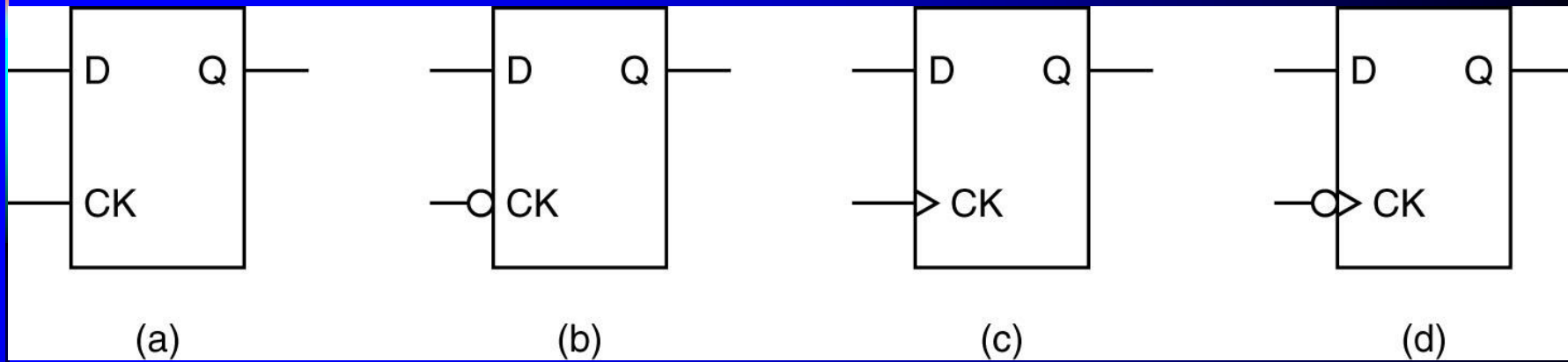
# Flip-Flops



A D flip-flop.

D- триггер (на практике используются более сложные схемы).

# Flip-Flops

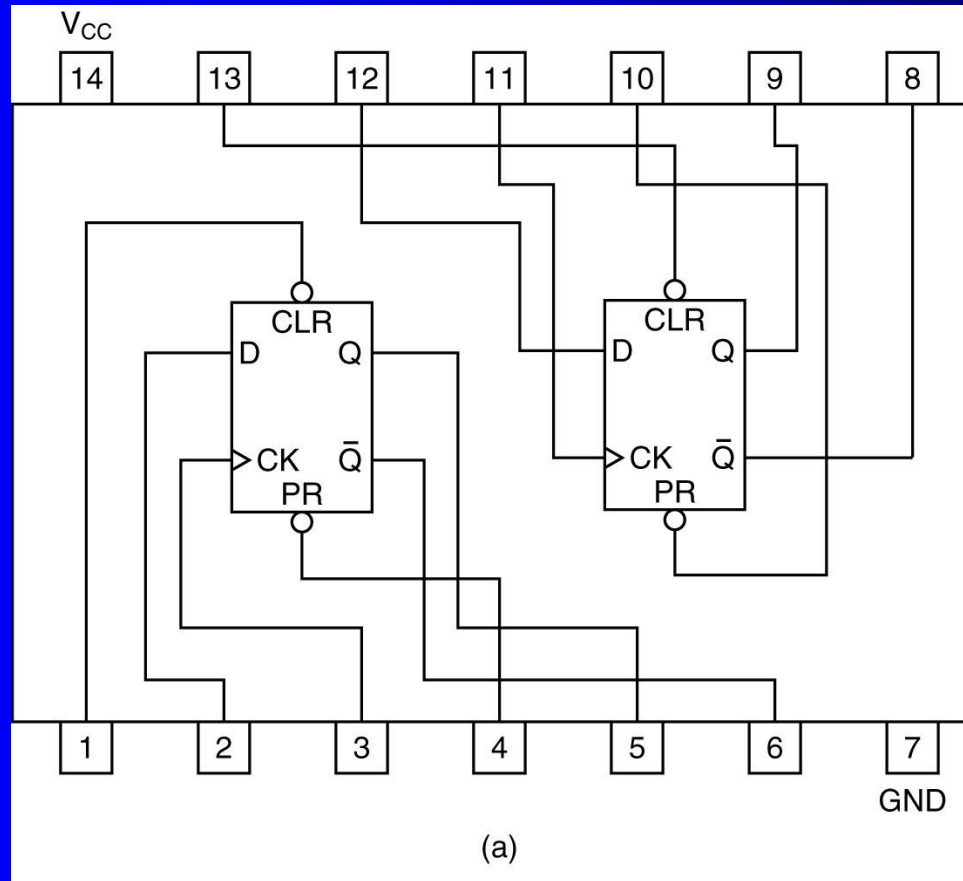


D latches and flip-flops.

D – защёлки (a,b) и триггеры (c,d).

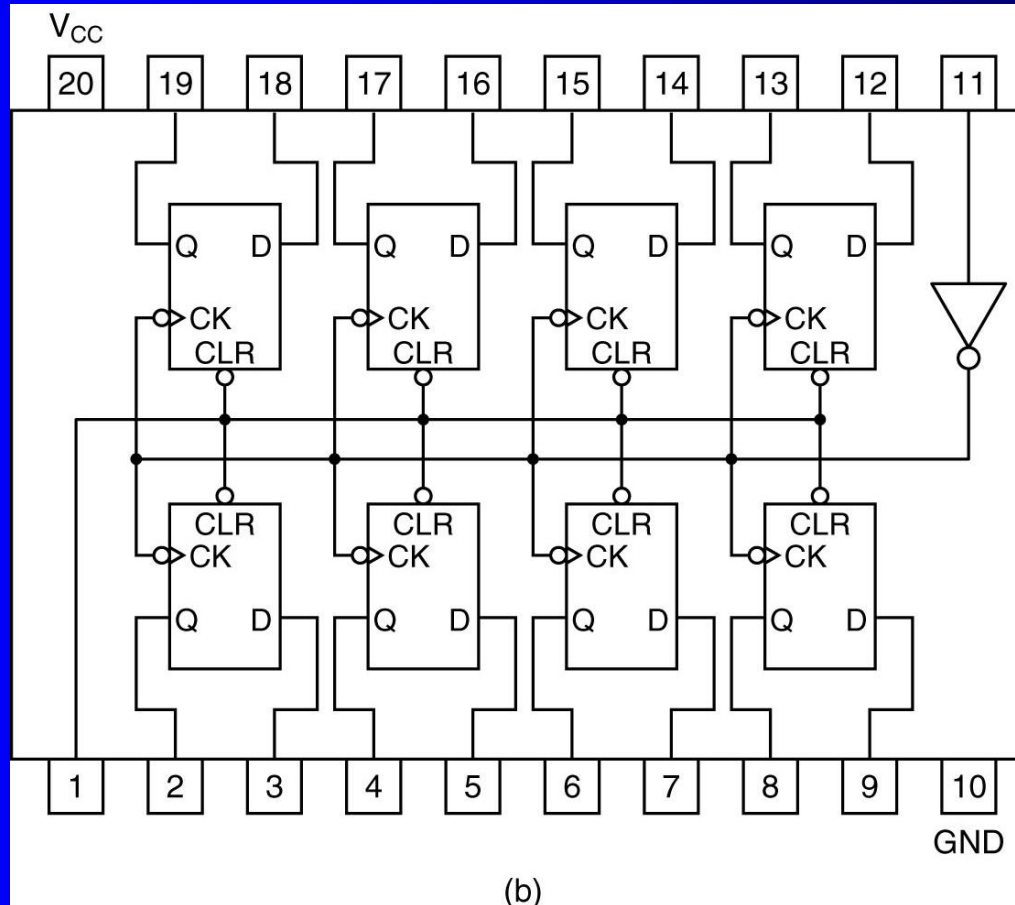
Триггер (c) меняет состояние на возрастающем фронте (с 0 на 1), а (d) – на заднем фронте (с 1 на 0). У многих схем есть  $\#Q$ , *Set (Preset)*  $\rightarrow Q=1$ , *Reset (Clear)*  $\rightarrow Q=0$

# Flip-Flops



Два независимых D-триггера на одной микросхеме (Dual D flip-flop).

# Flip-Flops



Восьмиразрядный триггер (Octal flip-flop) – в качестве восьмиразрядного регистра.

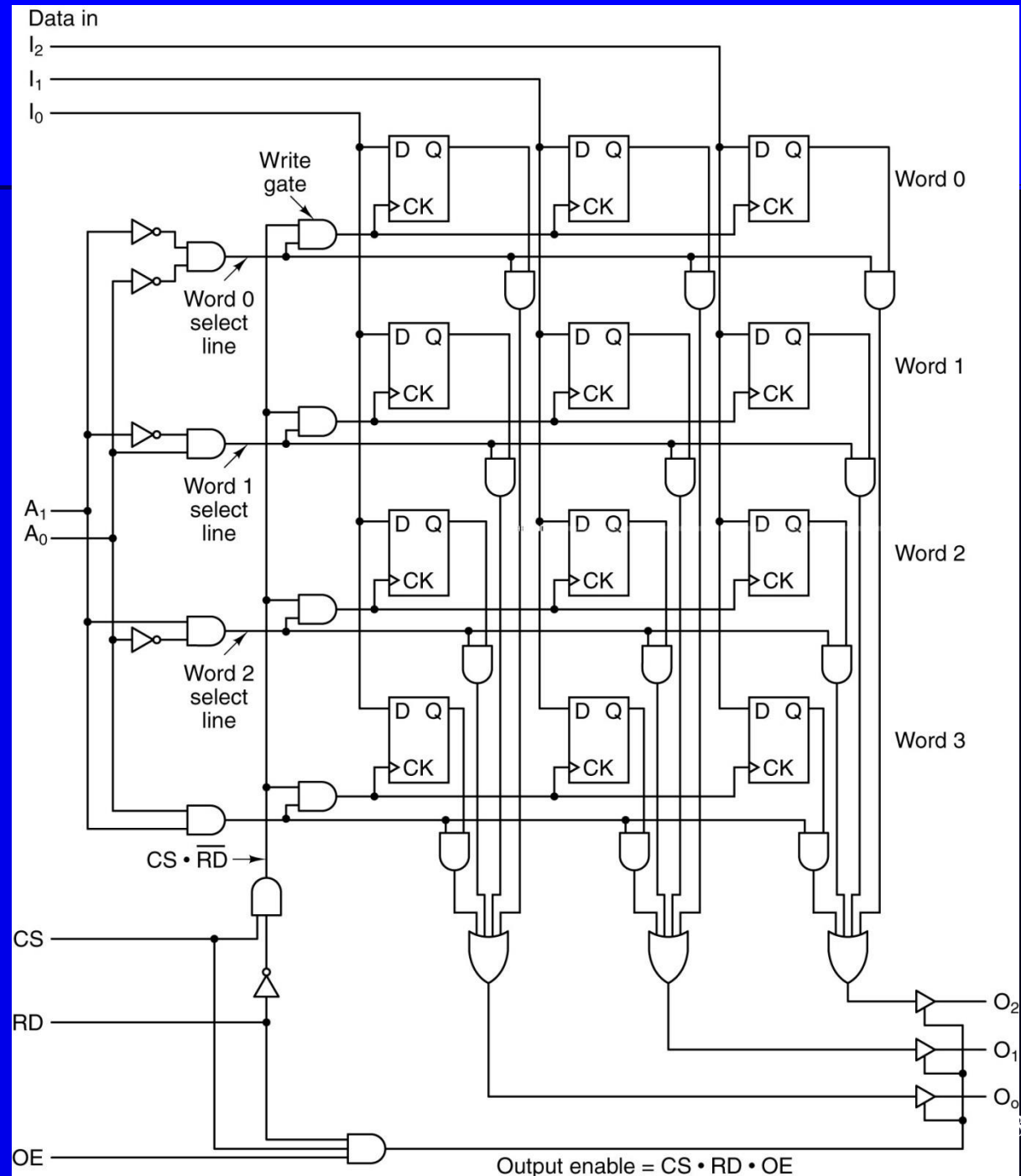


# Memory Organization

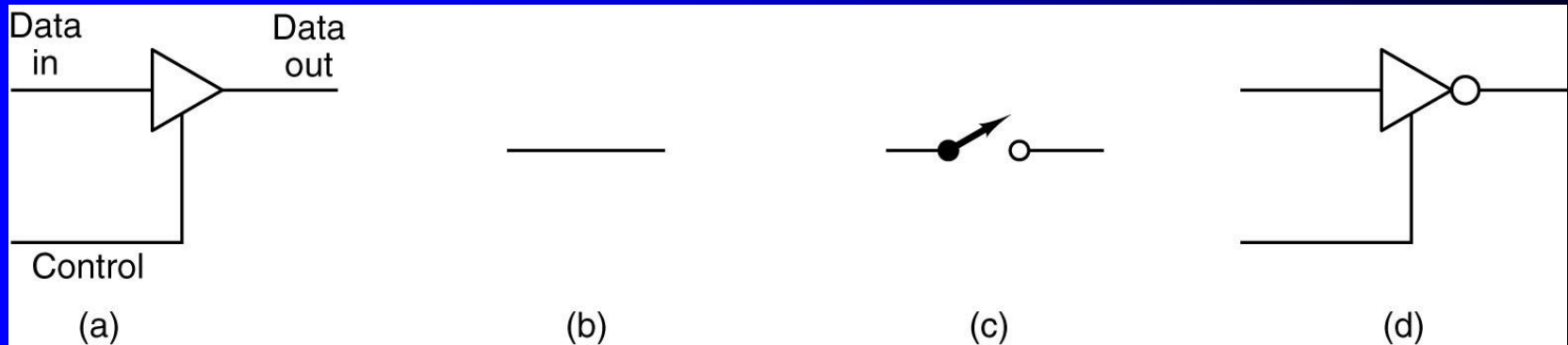
Logic diagram for a 4 x 3 memory.

Each row is one of the four 3-bit words.

CS – chip select  
RD – read  
OE – output enable



# Memory Organization (2)



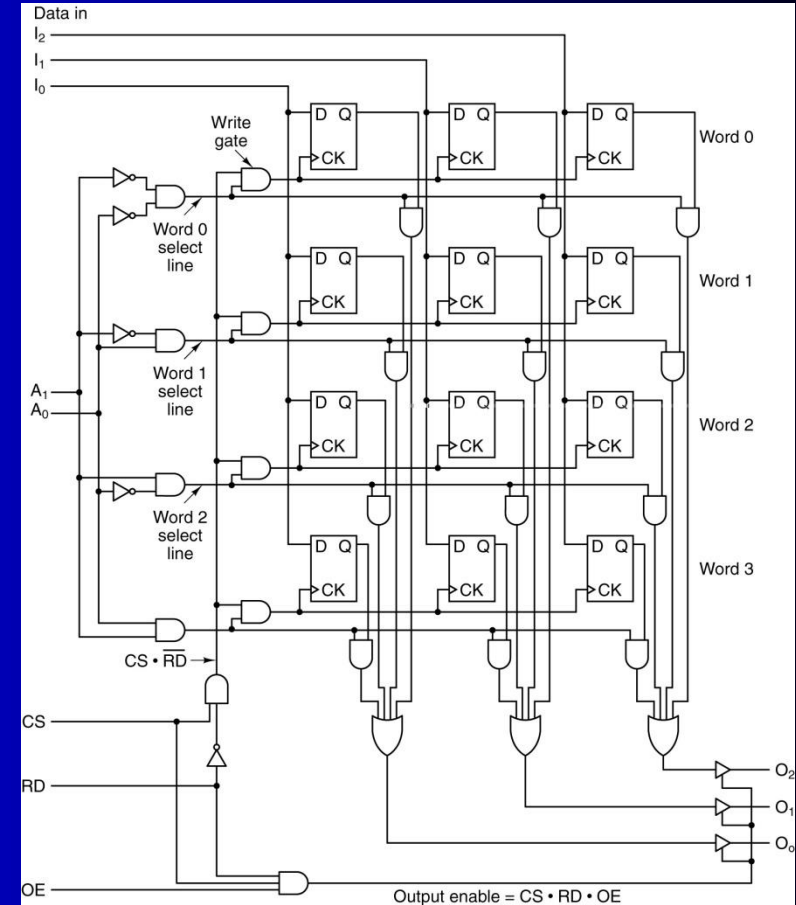
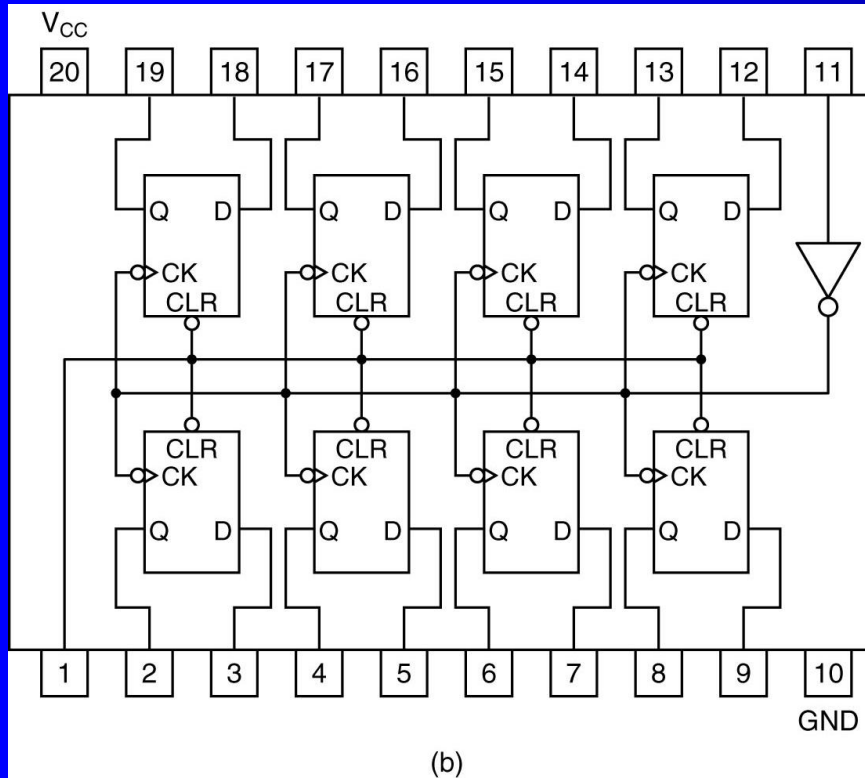
(a) A noninverting buffer – буферный элемент без инверсии.

(b) Effect of (a) when control is high.

(c) Effect of (a) when control is low.

(d) An inverting buffer – буферный элемент с инверсией.

# Memory Organization



8-бит – 20 выводов,

12 бит – 14 выводов.

# Computer of the Day

- Early 1940's: ENIAC – Eckert & Mauchley at U. Penn
  - general purpose; conditional jumps;
  - programmed via plug cables
  - 80 feet long, 18,000 vacuum tubes, 1900 10-digit adds/sec
- Univac I – first commercial computer ('51) ...
  - Only 8 tons (ENIAC was 20 tons). Clock speed 2.25 MHz.
  - 48 machines built – priced \$1M to \$1.5M
  - In 50's, "Univac" was synonymous with "computer"
- ... and first fights over intellectual property
  - E&M applied for patent in '47
    - U. of P. dean said university should get patent
    - E&M were fired or quit
  - Lawsuit in 60's, Honeywell v. Sperry Rand over patents
    - 164 cubic feet of evidence
    - Decision: Atanasoff (Iowa State) invented computer in 30's

**Decision: No timely patent on computers; everyone can build them.**

# Computer of the Day

- The IBM 1620 (1959)
  - 2-е поколение ЭВМ: транзисторы & и память на магнитных сердечниках  
(Первое поколение – на вакуумных лампах и память на задержке сигнала)
  - Пример оригинальной архитектуры
  - Всего было выпущено ~ 2000 шт. Относительно недорогие ( аренда < \$1620/месяц)
- Десятичный компьютер – 6 бит на цифру или символ
  - 4 бита, флаг (+/- или «конец слова»), бит чётности
  - Данные переменной длины – поля заканчивались флагом
- Арифметические действия – по таблице!
- Кодовое имя CADET
  - “Can’t Add, Doesn’t Even Try”