

Основы компьютерной техники (Computer Organization. Basis)

БГТУ

кафедра ПИ

доцент Самаль Дмитрий Иванович т. 293-23-79, [dmitry\_samal@mail.ru,](mailto:dmitry_samal@mail.ru) a.510б-5

Лекция 4

«Организация памяти-I» 2020



План лекции

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

***Слайд*** 2



## 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

***Слайд*** 3



## 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, экза (kilo, mega, giga, tera, peta, exa) – 103,106,109,1012,1015 и 1018.

Близко, но не равно 210,220,230,240,250 и 260.

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.

***Слайд*** 4

## Память

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

Метод доступа к ЗУ:

**Последовательный доступ.**

*Хранениеинформацииввидепоследовательности блоковданных,называемыхзаписями.Длядоступак нужномуэлементу(словуилибайту)необходимо прочитатьвсепредшествующиеемуданные. (Магнитные ленты)*

**Прямой доступ.** *Каждаязаписьимеетуникальный адрес,отражающийеефизическоеразмещениена носителеинформации.Обращение-адресныйдоступкначалу*

*записи ипоследующийпоследовательныйдоступкединице*

*информациивнутризаписи.(Жёсткийдиск)*



***Слайд*** 5



## Память

#### **Произвольный доступ.** *Каждаяячейкапамятиимеет* уникальныйфизическийадрес.Обращениеклюбой ячейкезанимаетодноитожевремяиможет проводитьсявпроизвольнойочередности.(ОЗУ).

**Ассоциативный доступ.** *Поискячеек,содержащих информацию,вкоторойзначениеотдельныхбитов совпадаетссостояниемодноименныхбитоввзаданном образце.*

#### *Сравнениеосуществляетсяпараллельнодлявсехячеек* памяти,независимоотееемкости.(КЭШ-память)

***Слайд*** 6

Быстродействие ЗУ:

## Память



**Время доступа** - Для памяти с произвольным доступам 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.

***Слайд*** 7

Быстродействие ЗУ:

## Память



**Длительность цикла памяти или период обращения** (ТЦ). Понятие применяется к памяти с произвольным доступом, для которой оно означает *минимальноевремямеждудвумяпоследовательными обращениямикпамяти.Периодобращениявключаетв себявремядоступаплюснекотороедополнительное 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.

***Слайд*** 8

Быстродействие ЗУ:

## Память

**Скорость передачи.** Это скорость, с которой данные могут передаваться в память или из нее. Для памяти с произвольным доступом она равна 1/ТЦ. Для других видов памяти скорость передачи

определяется соотношением:

TN = ТА + N/R ,

где TN - среднее время считывания или записи N битов; ТА – 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.



***Слайд*** 9

## Память. Основная проблема

Память отстаёт по быстродействию от ЦП



**Processor-Memory Performance Gap**

µProc

55%/year

**10000**

(2X/1.5yr)

**1000**

“Moore’s Law”

**100**

**10**

**1**

Processor-Memory Performance Gap (grows 50%/year)

DRAM

7%/year (2X/10yrs)

**1980 1984 1988**

**1992 1996**

**Year**

**2000 2004**

**CSE431 L18 Memory Hierarchy.3**

**Irwin, PSU, 2005**

**Performance**

***Слайд*** 10

## Память. Основная проблема

Память отстаёт по быстродействию от ЦП



**The “Memory Wall”**

* Logic vs DRAM speed gap continues to grow

**1000**

**100**

**10**

**1**

**0.1**

**0.01**

**VAX/1980**

**PPro/1996**

**2010+**

**CSE431 L18 Memory Hierarchy.4**

**Irwin, PSU, 2005**

**Core**

**Memory**

Clocks per instruction

Clocks per DRAM access

***Слайд*** 11



## Иерархия запоминающих устройств

Чем меньше время доступа, тем выше стоимость хранения бита.

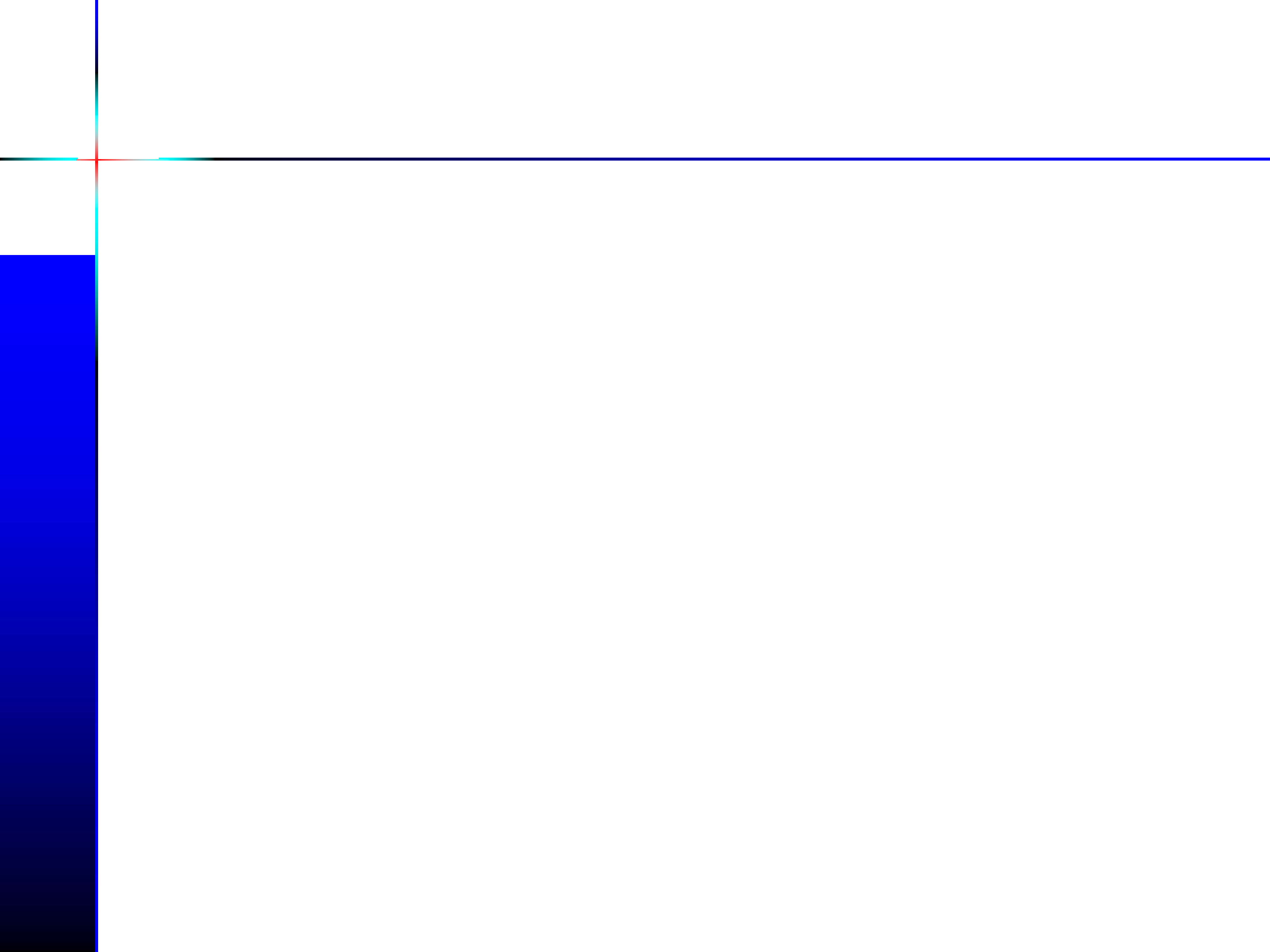
Чем больше ёмкость, тем ниже стоимость хранения, но больше время доступа.

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

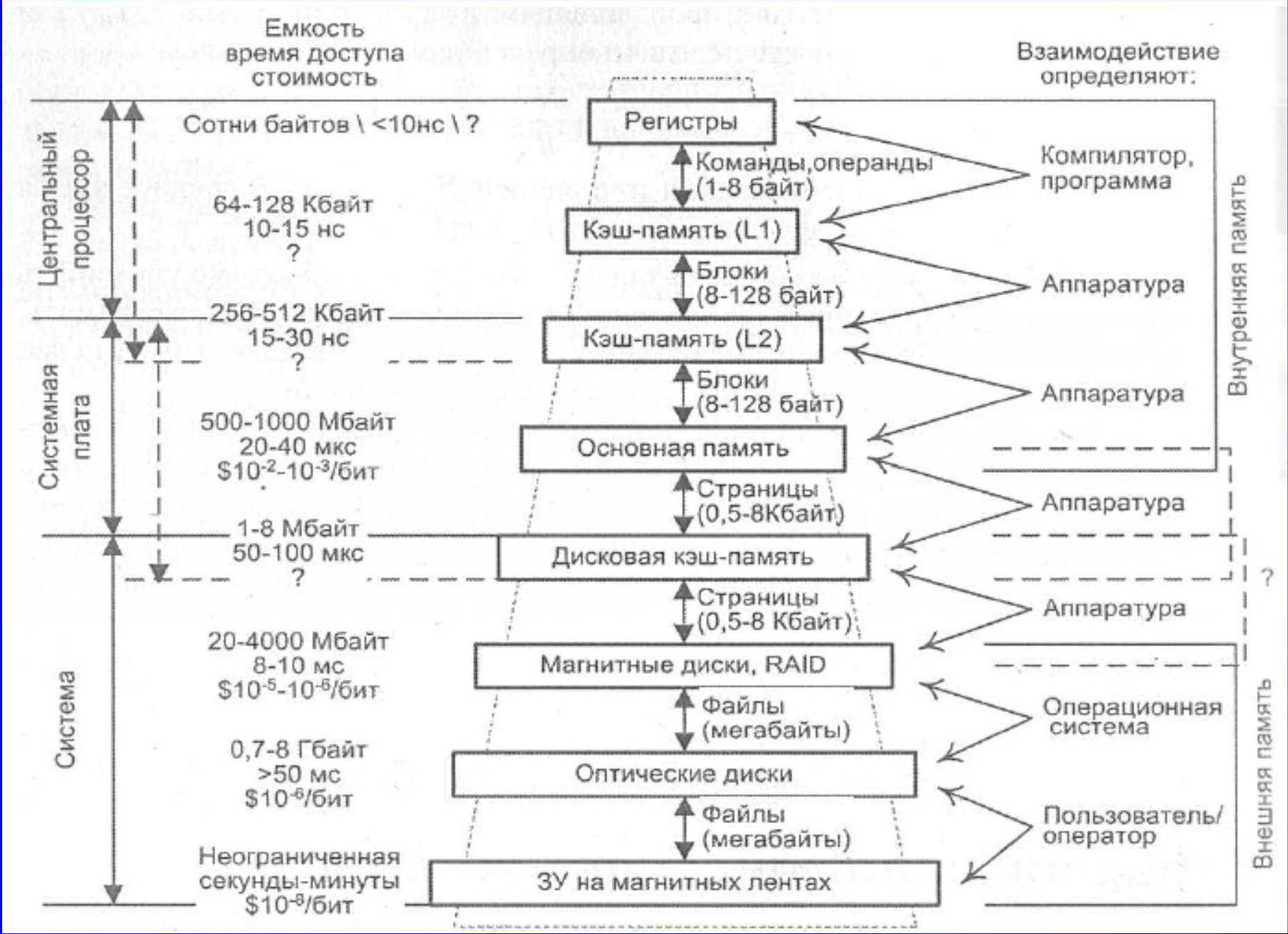
Наиболее эффективным решением является *создание иерархическойпамяти.*

#### *Уровнииерархиитесносвязаны–вседанныенаодном* уровнепродублированытакженаболее низкомуровне, вседанныесболеенизкогоуровня–наследующем нижележащемуровне.

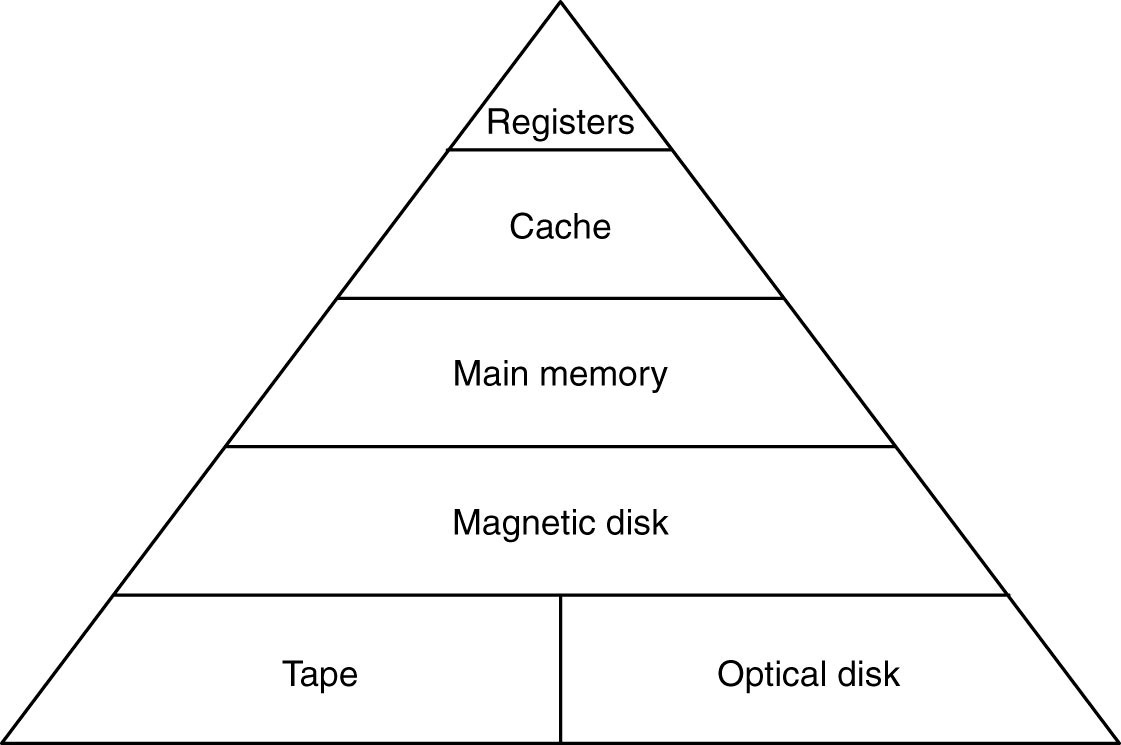
***Слайд*** 12



## Иерархия запоминающих устройств

***Слайд*** 13

# 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.

***Слайд*** 14

## Иерархия запоминающих устройств

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

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 облекают в численную форму и представляют в виде так называемого правила "90/10": *90%времениработы программысвязаносдоступомк10%адресного*

#### *пространстваэтойпрограммы.*



***Слайд*** 15



## 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, *посколькукомандыиданные,будучиодин разпереданыизмедленногоЗУвбыстрое,затеммогут использоватьсямногократноисреднеевремядоступак нимвэтомслучаеопределяетсяужеболеебыстрымЗУ.*

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!)

***Слайд*** 16

## Иерархия запоминающих устройств

При оценке эффективности подобной организации памяти обычно используют следующие характеристики:

* коэффициент попаданий (hit rate) - *отношениечисла обращенийкпамяти,прикоторыхпроизошлопопадание, кобщемучислуобращенийкЗУданногоуровня;*
* коэффициент промахов (miss rate) - *отношениечисла обращенийкпамяти,прикоторыхимелместопромах,к общемучислуобращенийкЗУданногоуровня;*

#### время обращения при попадании (hit time) - *время,* необходимоедляпоисканужнойинформациивпамяти верхнегоуровня,плюсвремянафактическоесчитывание данных;

* потери на промах (miss penalty) *-время,требуемое длязаменыблокавпамятиболеевысокогоуровняна*

#### *блокснужнымиданными,расположенныйвЗУ* следующего(болеенизкого)уровня.



***Слайд*** 17



## Иерархия запоминающих устройств

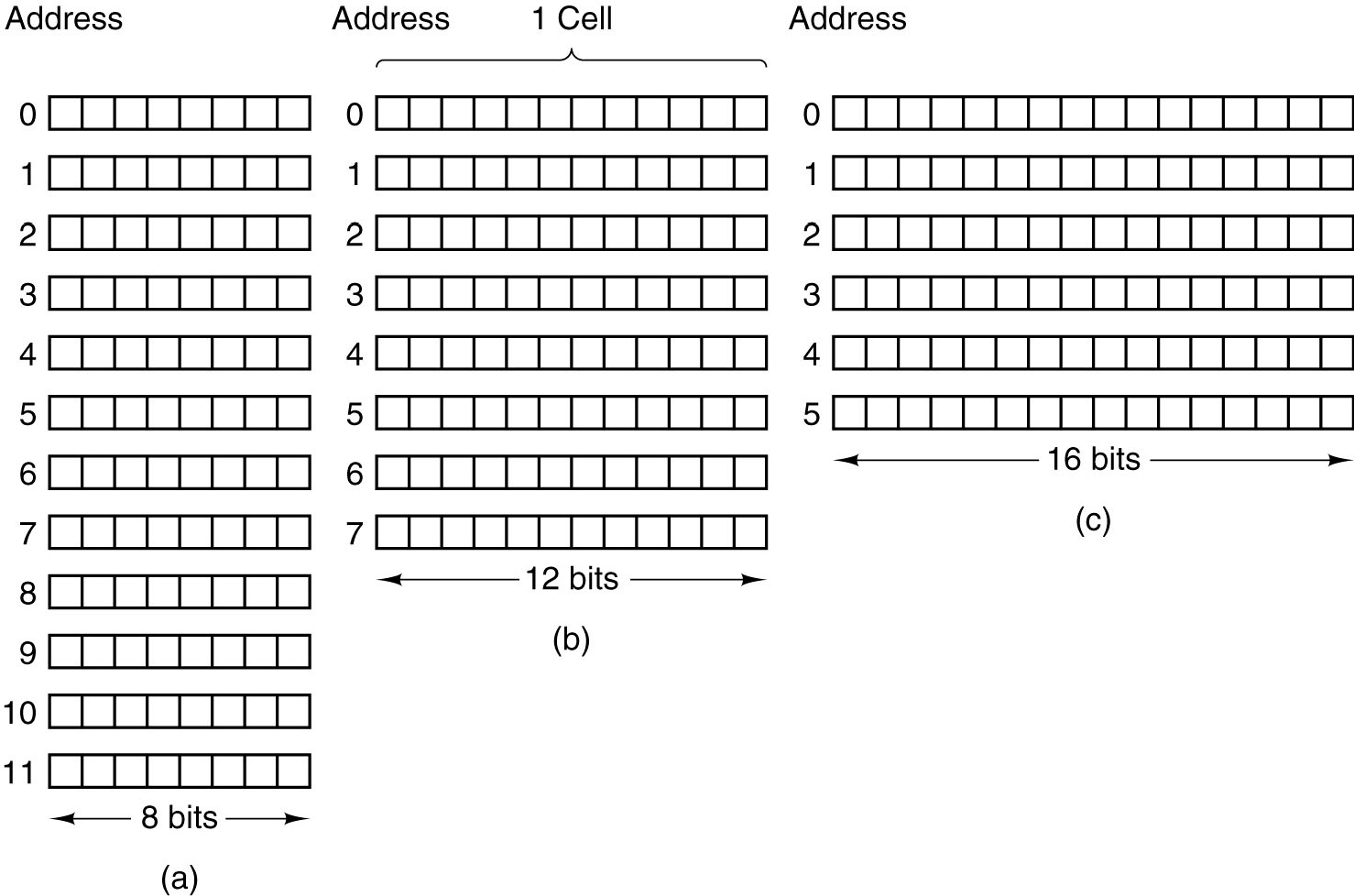
Описание некоторого уровня иерархии ЗУ предполагает конкретизацию четырех моментов:

* 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 устр***Сл***-***ай***в***д***.18

# Primary Memory

Memory Addresses (1)



Three ways of organizing a 96-bit memory.



Каждая ячейка содержит фиксированное число запоминающих элементов.

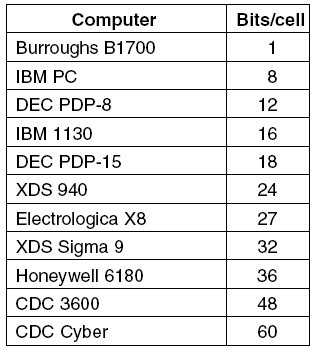
***Слайд*** 19

# Primary Memory



Memory Addresses (2)

Number of bits per cell for some historically interesting commercial computers



***Слайд*** 20

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: *дажеприкратковременном прерываниипитанияхранимаяинформациятеряется.*

#### *МикросхемаОЗУдолжнабытьпостоянноподключенак* источникупитанияипоэтомуможетиспользоваться

*толькокаквременнаяпамять.*



***Слайд*** 21



## Основная память

Вторую группу полупроводниковых ЗУ основной памяти образуют

энергонезависимые микросхемы ПЗУ (ROM - Read-Only Меmоrу). 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 Rаndоm Access Меmory) и статическую память (SRAM - Static Rаndоm

Access Меmory).

***Слайд*** 22



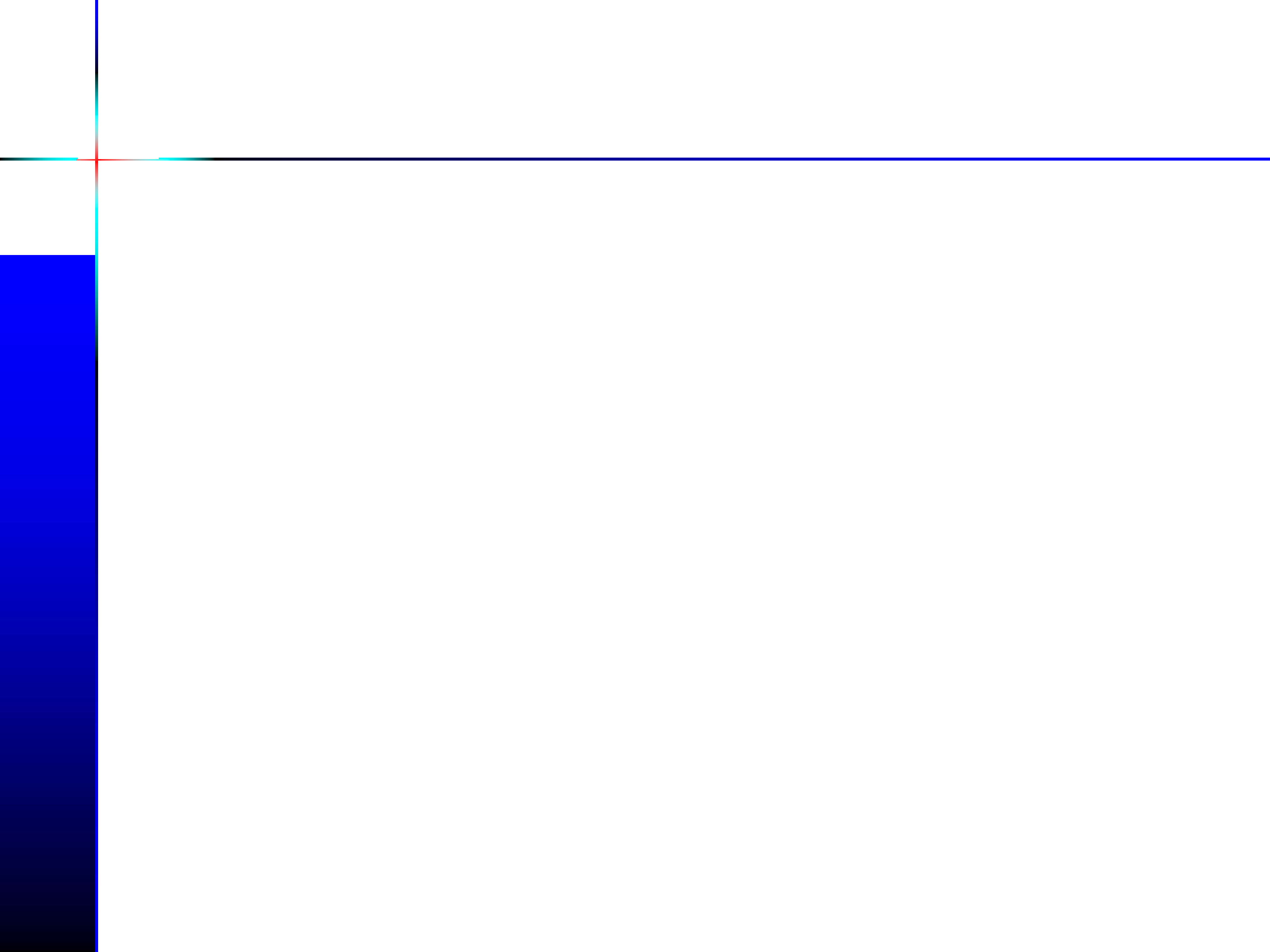
## Основная память

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

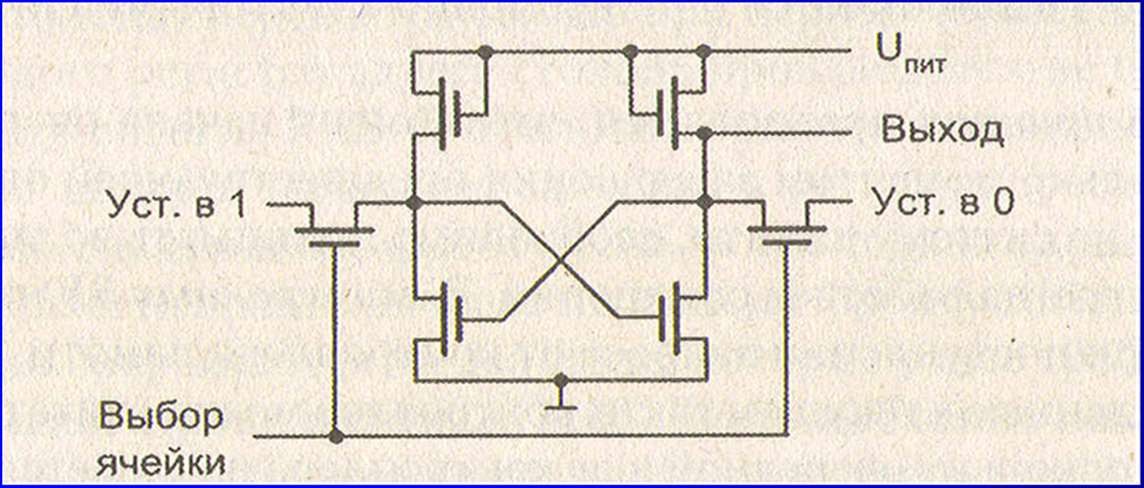
Запоминающий элемент динамического ОЗУ способен хранить информацию только в течение достаточно короткого промежутка времени, после которого информацию нужно восстанавливать заново, иначе она будет потеряна. Динамические ЗУ, как и статические, энергозависимы.

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

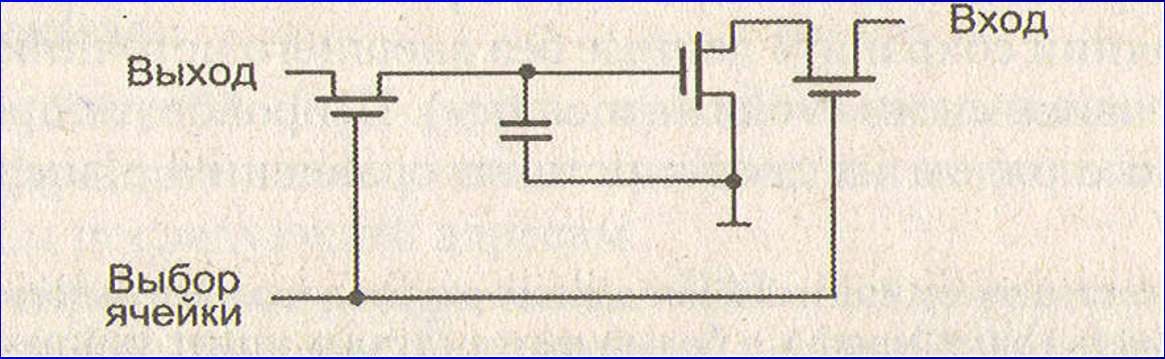
***Слайд*** 23



## Статическое и динамическое ОЗУ

SRAM, на 6

транзисторах – быстрее, менее чувствительны к внешним излучениям

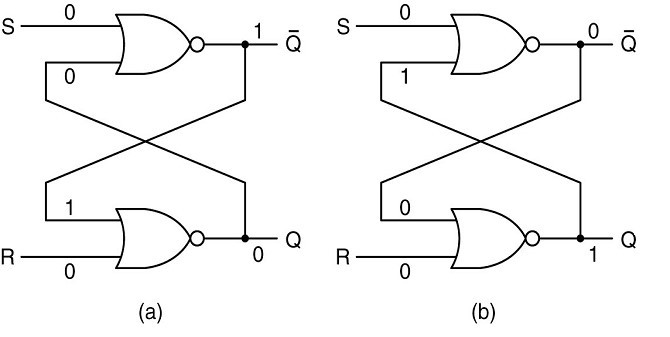
DRAM

Простота схемы DRAM позволяет достичь высокой плотности размещения, в итоге, *снизитьстоимость.Главныйнедостаток-что накапливаемыйнаконденсаторезарядсовременемтеряется.*

*СреднеевремяутечкизарядаDRAMсоставляетсотни(десятки) миллисекунд,поэтому,зарядследуетрегенерировать.*

***Слайд*** 24

# Latches - Защёлки



При R=S=0 два стабильных состояния:



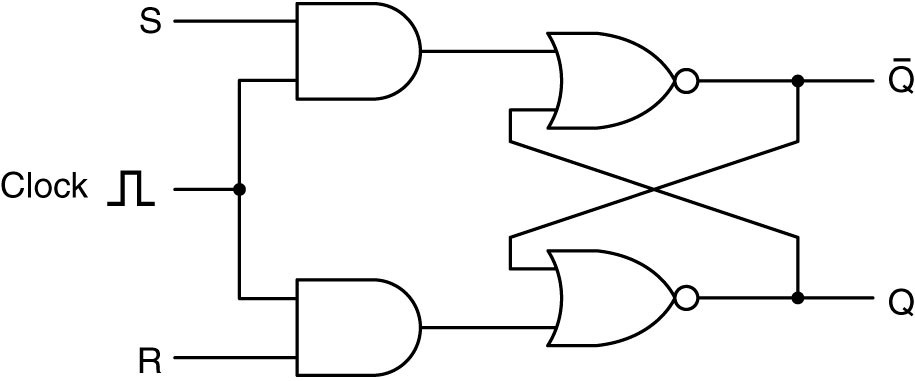
1. Защёлка ИЛИ-НЕ (NOR) в состоянии 0.
2. Защёлка ИЛИ-НЕ (NOR) в состоянии 1.

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

***Слайд*** 25



# Latches - Защёлки



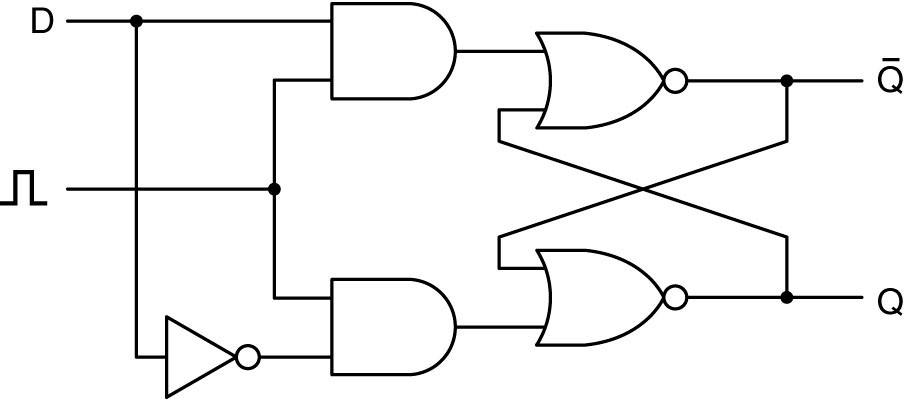
Синхронные RS-защёлки.

A clocked SR latch.

***Слайд*** 26



# Latches - Защёлки



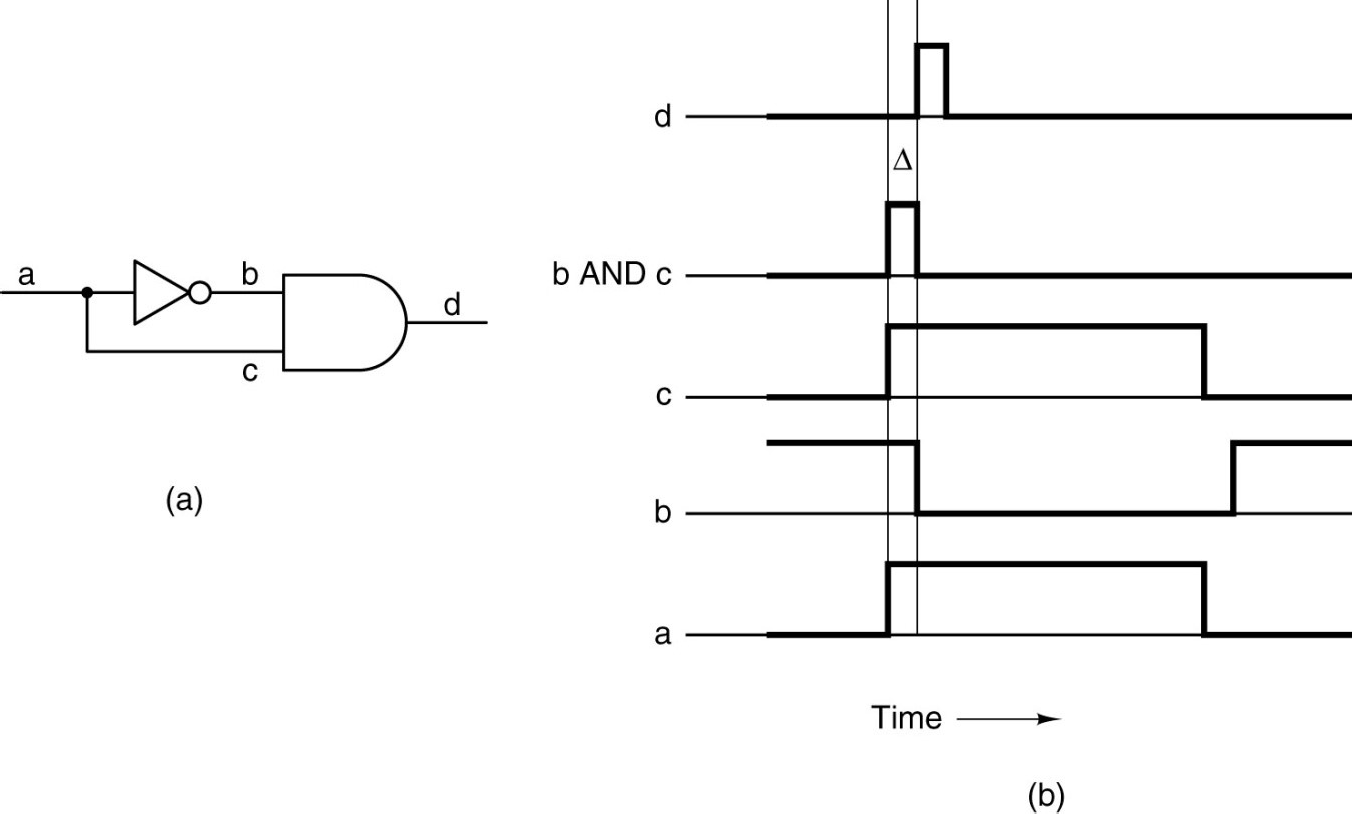
### Синхронная D защёлка - A clocked D latch. (Используется 11 транзисторов).

***Слайд*** 27



# Flip-Flops - Триггеры

#### *Защёлкизапускаютсяуровнемсигнала,триггеры–* фронтом!

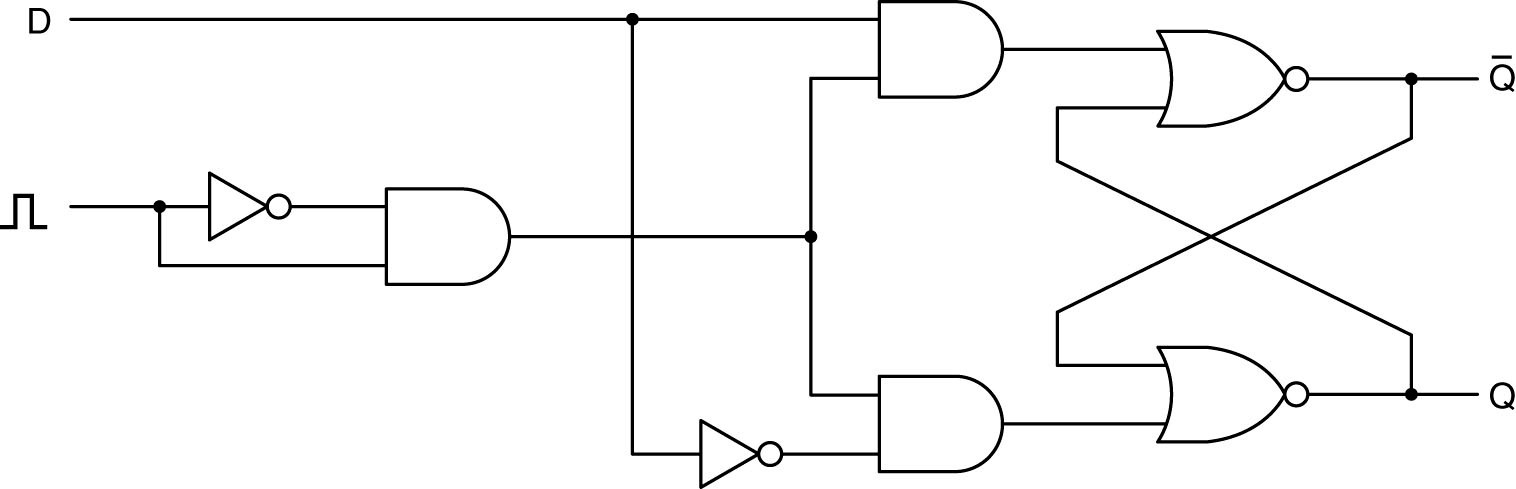


* 1. A pulse generator.
  2. Timing at four points in the circuit.

***Слайд*** 28



# Flip-Flops



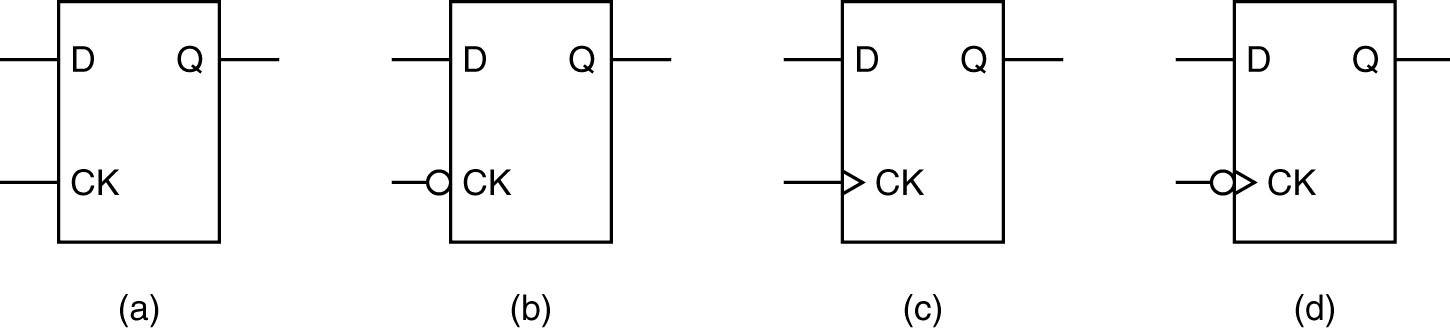
A D flip-flop.

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

***Слайд*** 29



# Flip-Flops



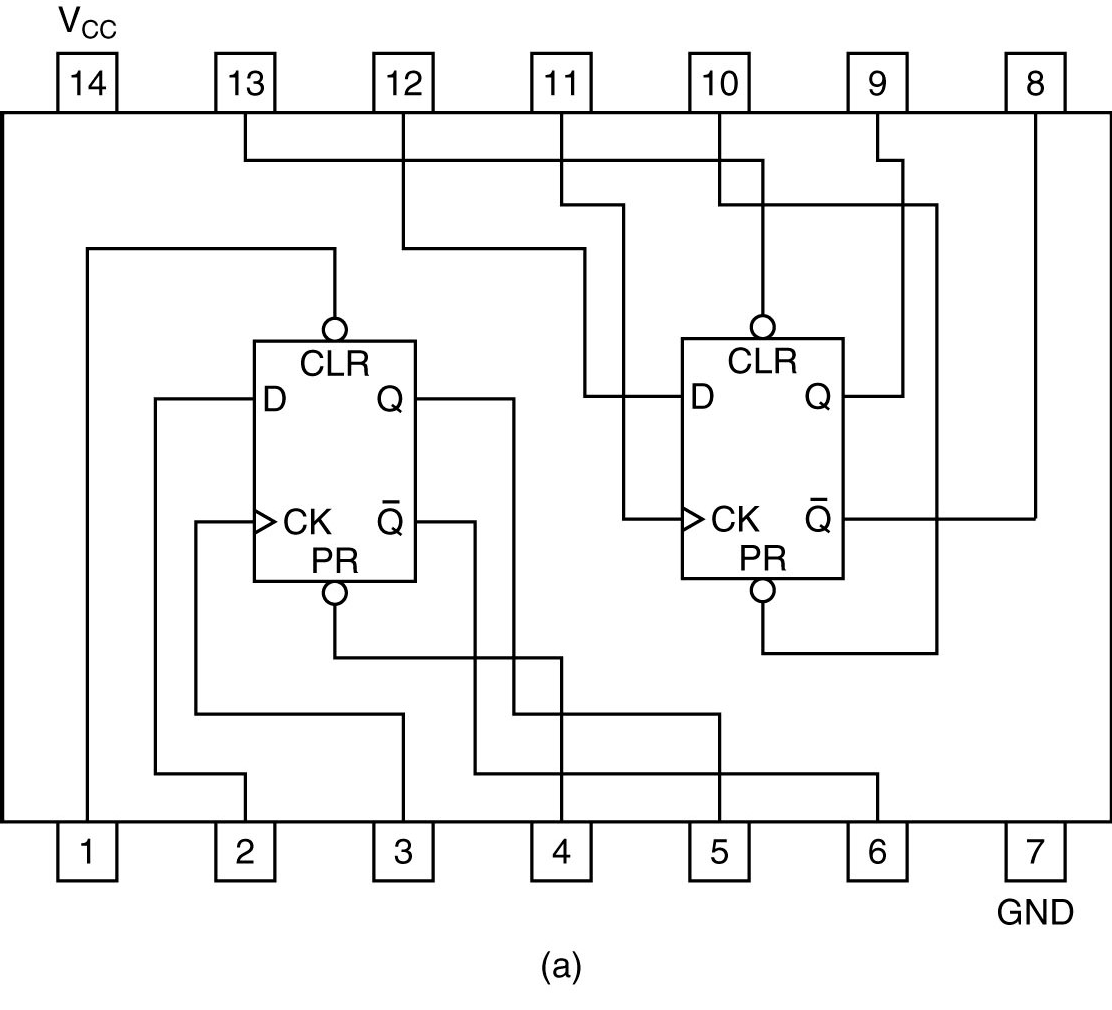
D latches and flip-flops.

D – защёлки (а,b) и триггеры (c,d).

Триггер (с) 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)*-> Q=1, *Reset(Clear)*-> Q=0

***Слайд*** 30

# Flip-Flops

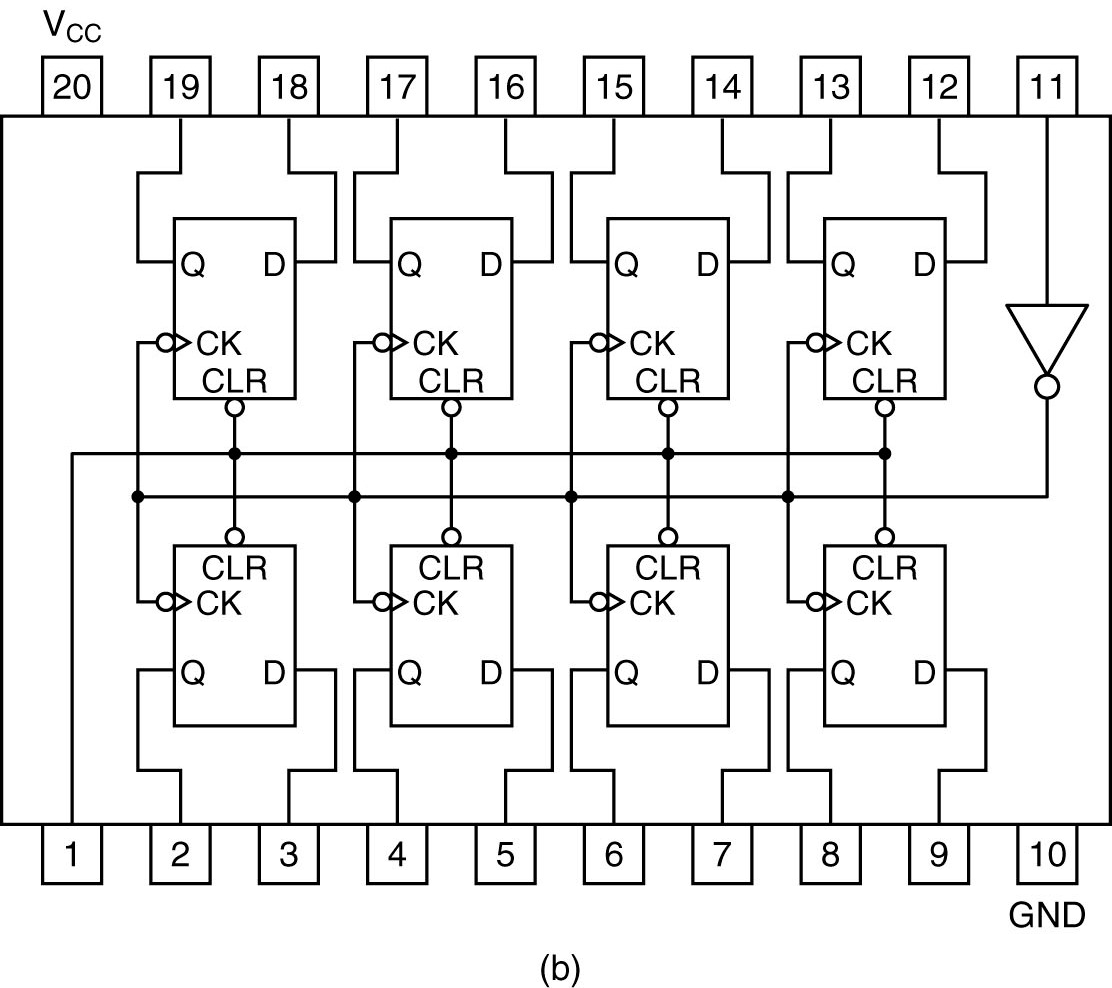


### Два независимых D-триггера на одной микросхеме (Dual D flip-flop).



***Слайд*** 31

# Flip-Flops

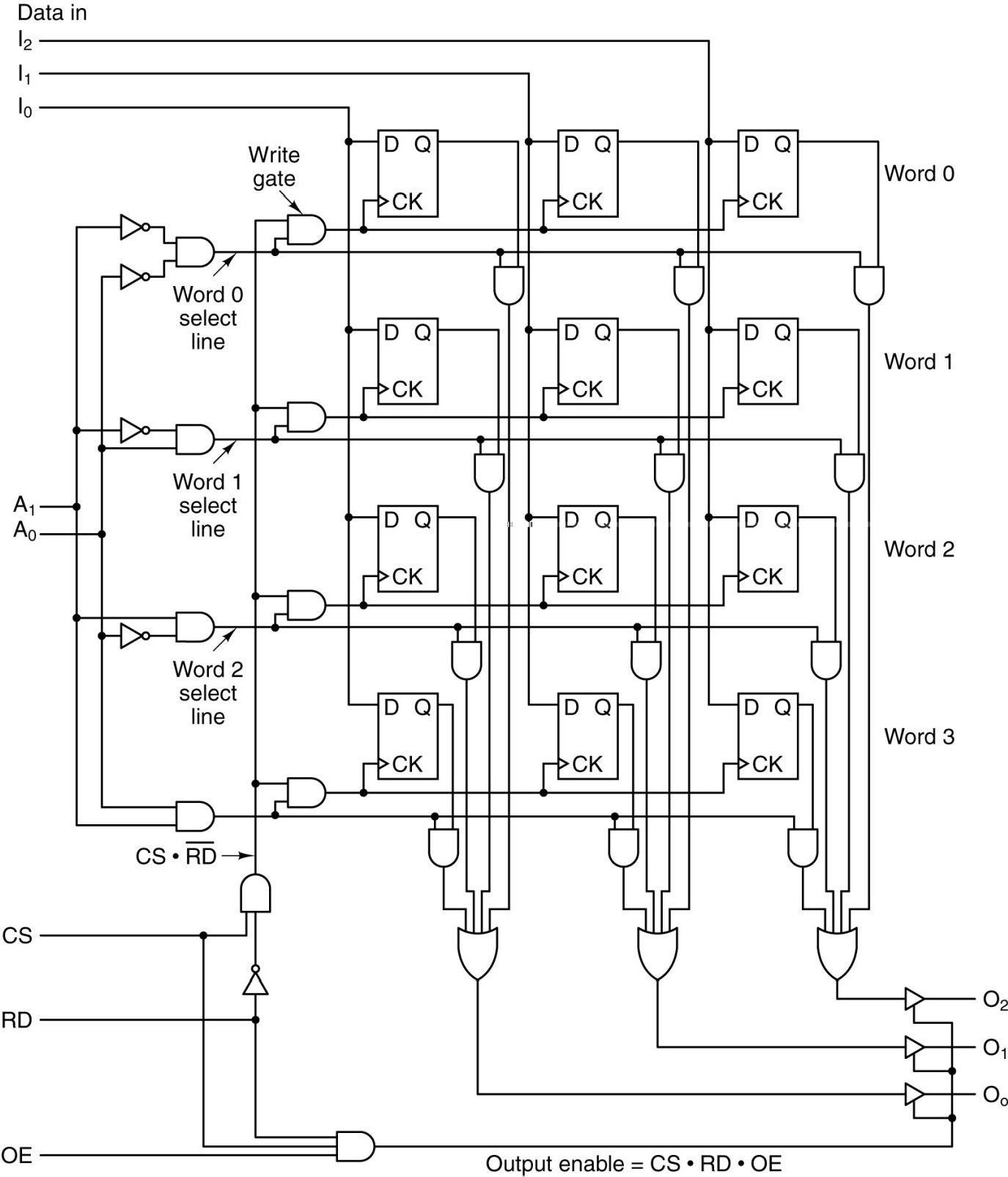


### Восьмиразрядный триггер (Octal flip-flop) – в качестве восьмиразрядного регистра.



***Слайд*** 32

# Memory Organization



3

Logic diagram for a 4 x 3 memory.

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

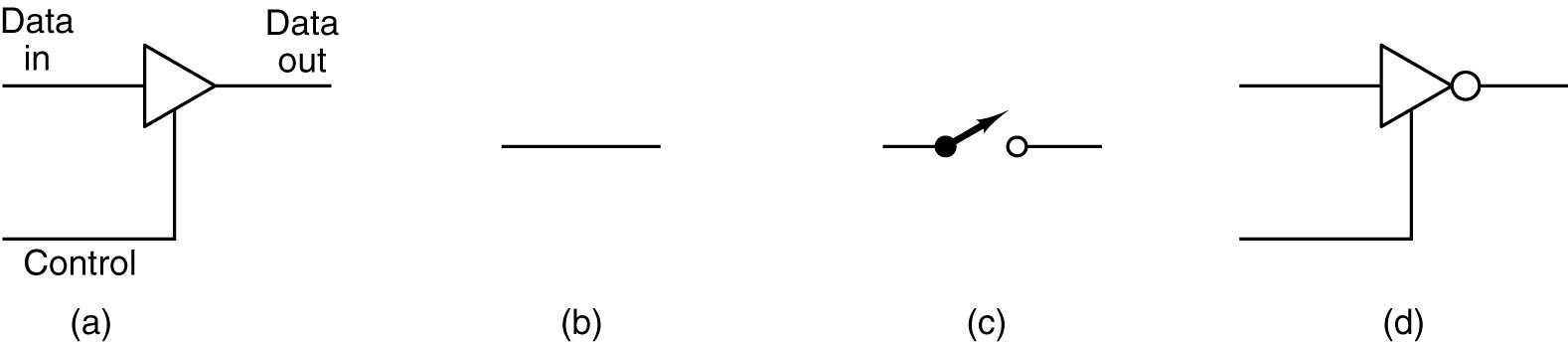
CS – chip select RD – read

OE – output еnable

***Слайд*** 3



# Memory Organization (2)



### A noninverting buffer – буферный элемент без инверсии.

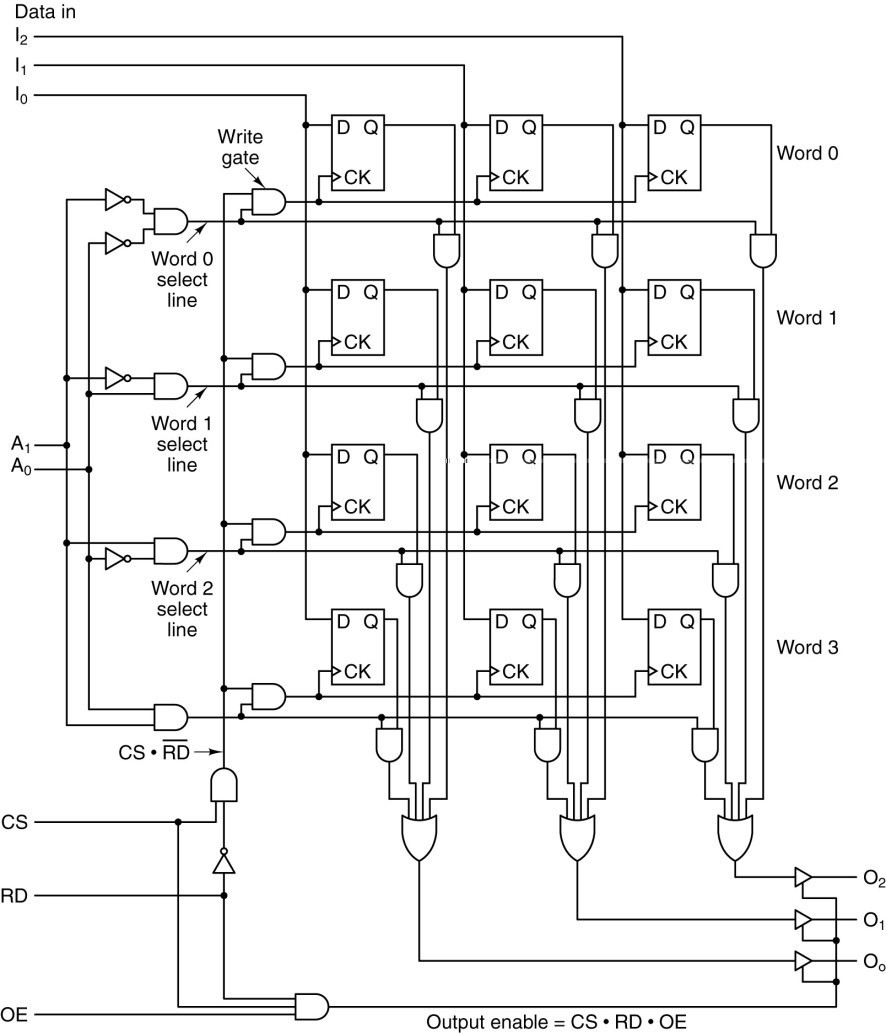
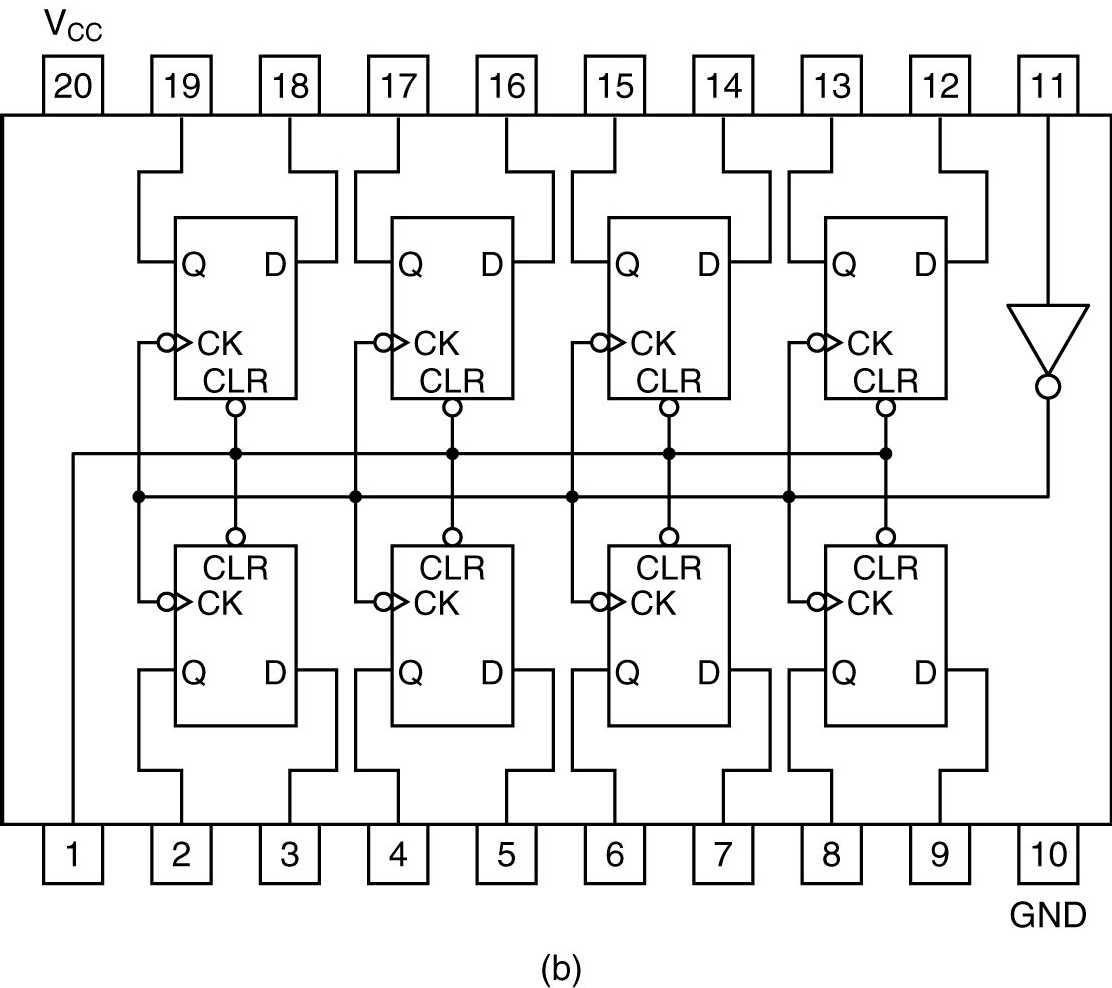
* + 1. Effect of (a) when control is high.

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

* + 1. An inverting buffer – буферный элемент с инверсией.

***Слайд*** 34

# Memory Оrganization



8-бит – 20 выводов, 12 бит – 14 выводов.



***Слайд*** 35



# 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 th*С*e*ла*m*йд*** 36



# 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”

***Слайд*** 37