



# 第七章 MOS存储器

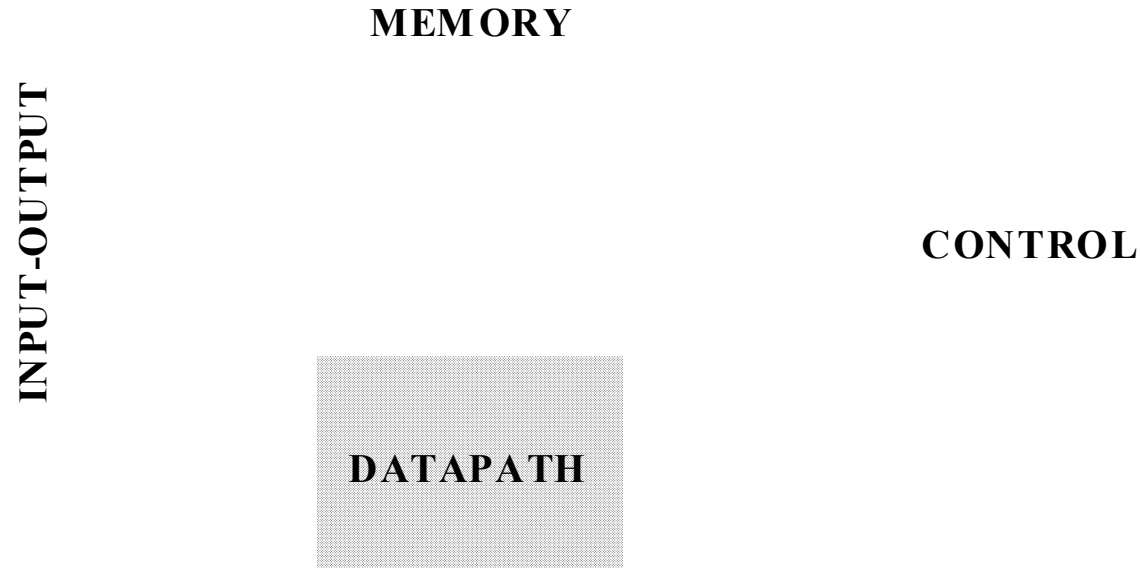
---

## 7.1 MOS存储器结构



# 计算机组成

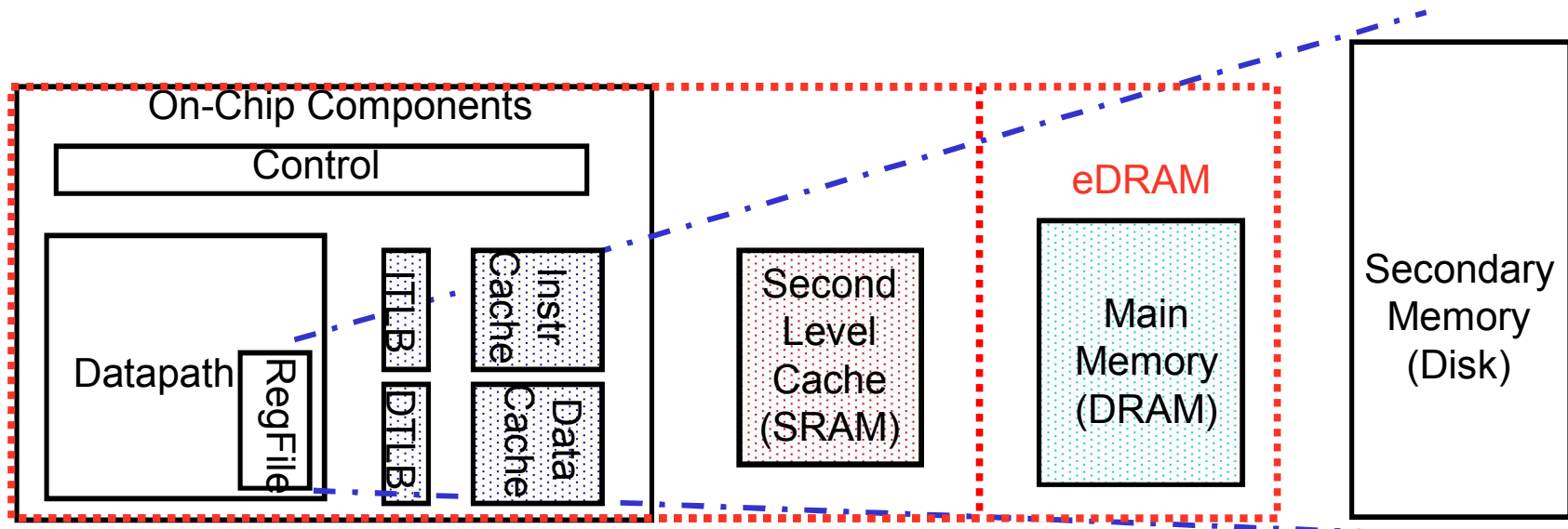
---



■存储器是数字系统中重要的组成部分，直接影响着系统的面积，速度和功耗

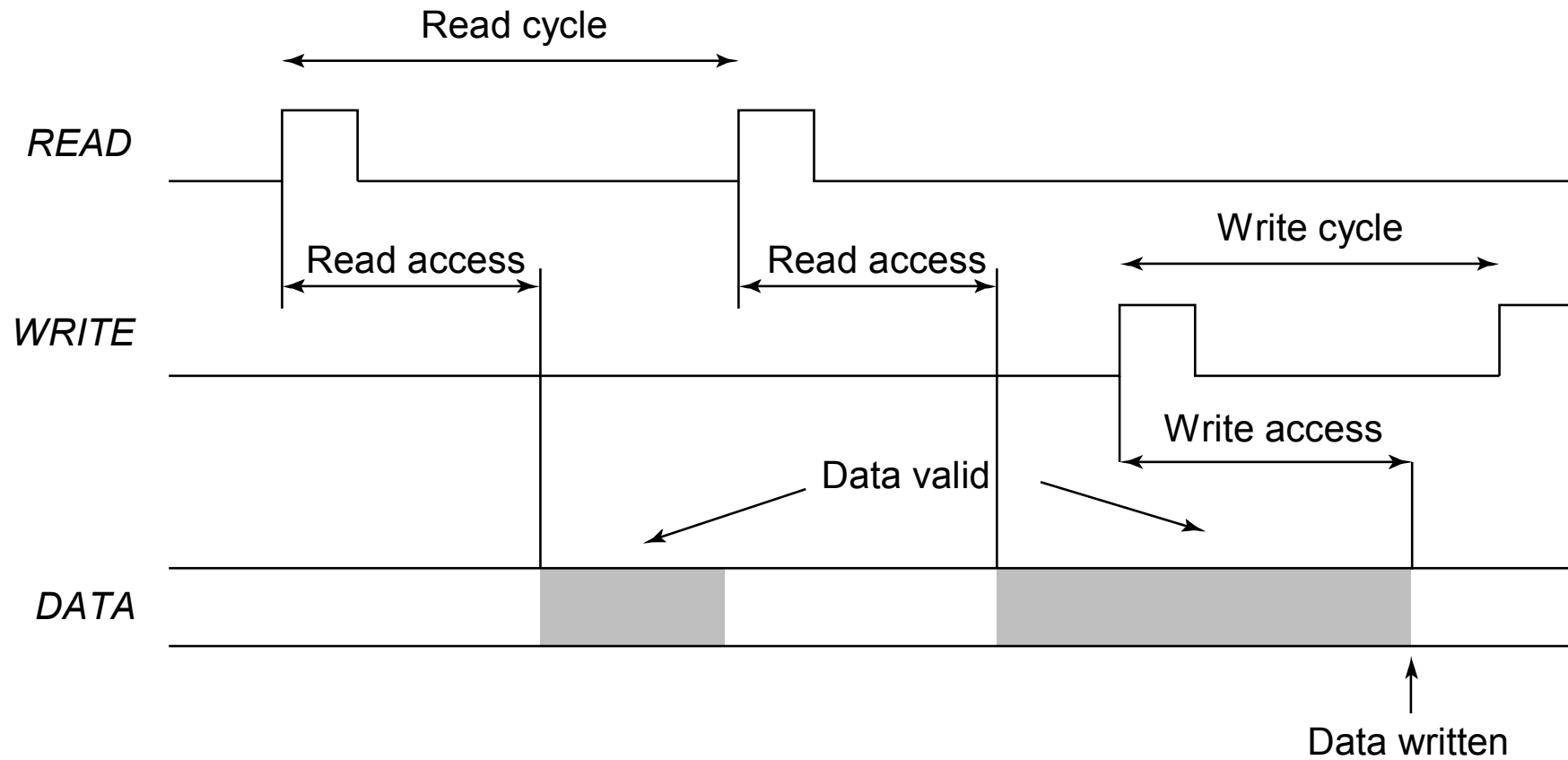
# 存储器层次

## 速度（带宽）与成本的折中



<b>Speed (ns):</b>	.1's	1's	10's	100's	1,000's
<b>Size (bytes):</b>	100's	K's	10K's	M's	T's
<b>Cost:</b>	highest				lowest

# Memory Timing: Definitions





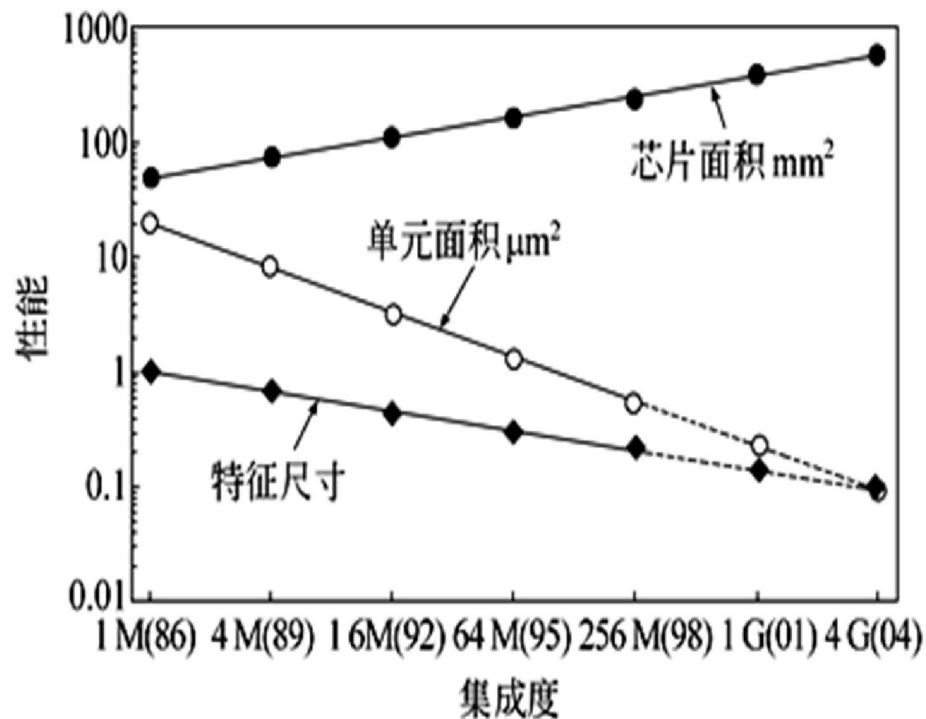
# MOS存储器结构

---

- 存储器分类
- **MOS**存储器结构
- 外围电路

# 摩尔定律引导的MOS存储器发展

- **摩尔定律**: 单个芯片上所能容纳的器件数量, 每12-18个月翻一番。



摩尔定律得以保持的途径:

**缩小特征尺寸**

**增大芯片面积**

**单元结构的改进**

# MOS存储器的分类

- MOS存储器主要分为两大类
  - 随机存取存储器 (Random Access Memory, RAM) : 挥发性存储, 断电后存储内容消失。
  - 只读存储器 (Read Only Memory, ROM) : 不挥发性存储, 存储内容可以长期保持, 至少保持10年以上。
  - 不挥发性随机存取存储器 (FeRAM, MRAM)

# 随机存取存储器RAM分类

动态随机存取存储器（Dynamic Random Access Memory, DRAM）：

存储原理：依靠电容存储，保持时间短，必须定期刷新。

特点：单元电路简单，面积小，有利于提高集成密度

用途：集成度高、功耗低，适合于计算机的内存。

静态随机存取存储器（Static Random Access Memory, SRAM）：

存储原理：双稳态电路存储，只要不断电存储信息不会丢失。

特点：电路复杂，占用面积大，集成度不如DRAM高。

用途：工作速度快，常用来作高速缓冲存储器（cache）。



## 只读存储器ROM的分类

1. 掩模编程的只读存储器 (Mask Rom) : 真正意义的只读存储器, 存储信息由制作时的某一块掩模版确定, 产品生产出来存储内容就不能再改变。适合于存储固定程序、常数、字符等固定内容。
2. 基于熔丝或反熔丝的可编程只读存储器 (programmable ROM, PROM) : 存储内容由用户编程确定, 一般只能编程一次, 相当于固定内容的只读存储器, 但是比Mask ROM在应用上有一定灵活性。

## 只读存储器ROM的分类(续)

3. 可擦除的可编程只读存储器 (erasable and programmable ROM, EPROM) : 可以随机改写, 擦除和写入时间较长, 耗能较大, 不如RAM的写入方便, 因此归入只读存储器类。

(1) 紫外光擦除 UVEPROM (ultraviolet EPROM)

只能在断电情况下全片统一擦除。

(2) 电擦除 EEPROM (electrical EPROM) :

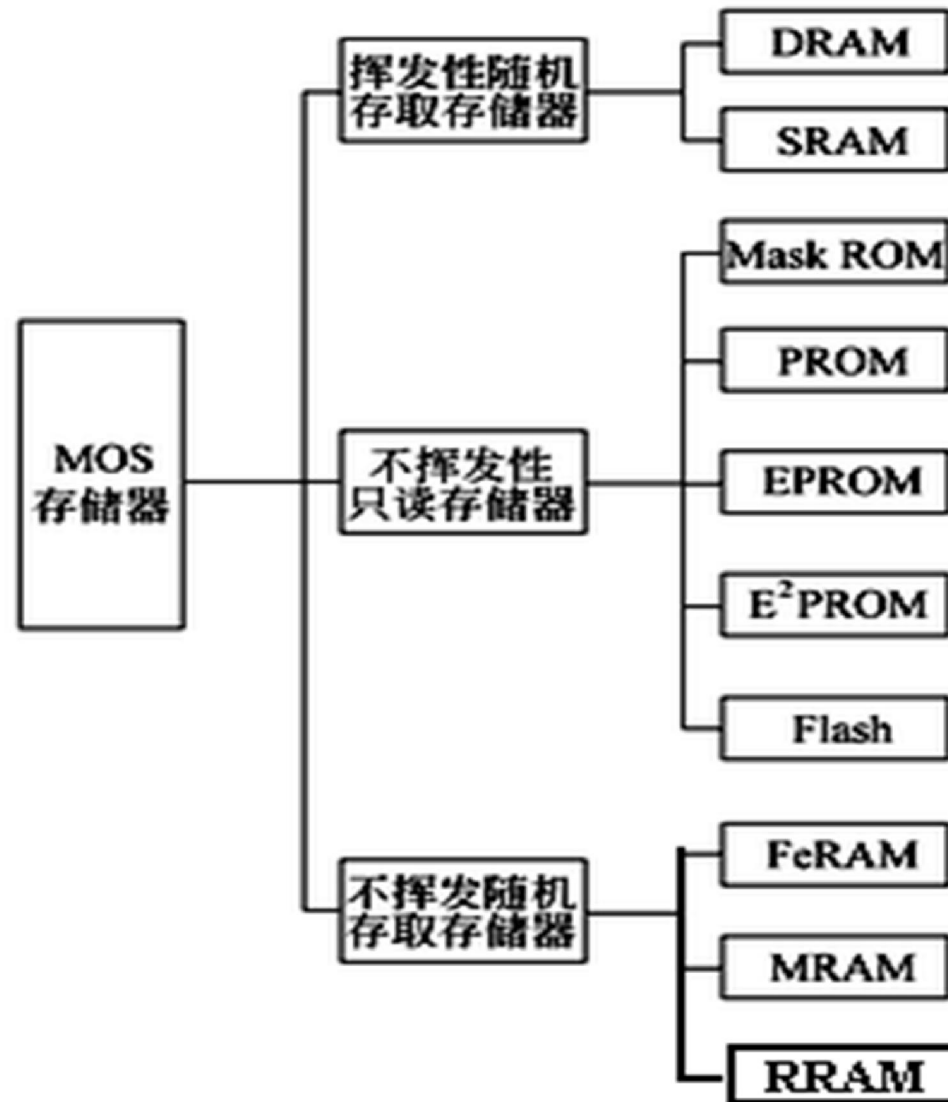
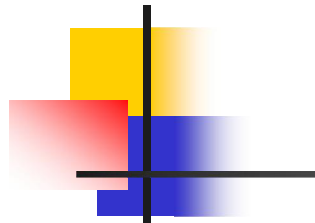
按位擦除和改写

Flash Memory——一种可全片或按扇区快速擦除的  
EEPROM

## 不挥发性随机存取存储器

- FeRAM (ferroelectric RAM),
- MRAM (magnetic RAM),
- RRAM (Resistive RAM)
  - **优点：**具有DRAM高密度和RAM随机读/写的特点，不挥发性，保持时间长，耐久性好，功耗小、工作电压低、读写速度快、以及抗辐射、抗干扰等。
  - **缺点：**制作成本高，和常规集成电路工艺的不兼容性。
  - **前景：**取代硬盘实现大容量存储器。

# MOS存储器的分类



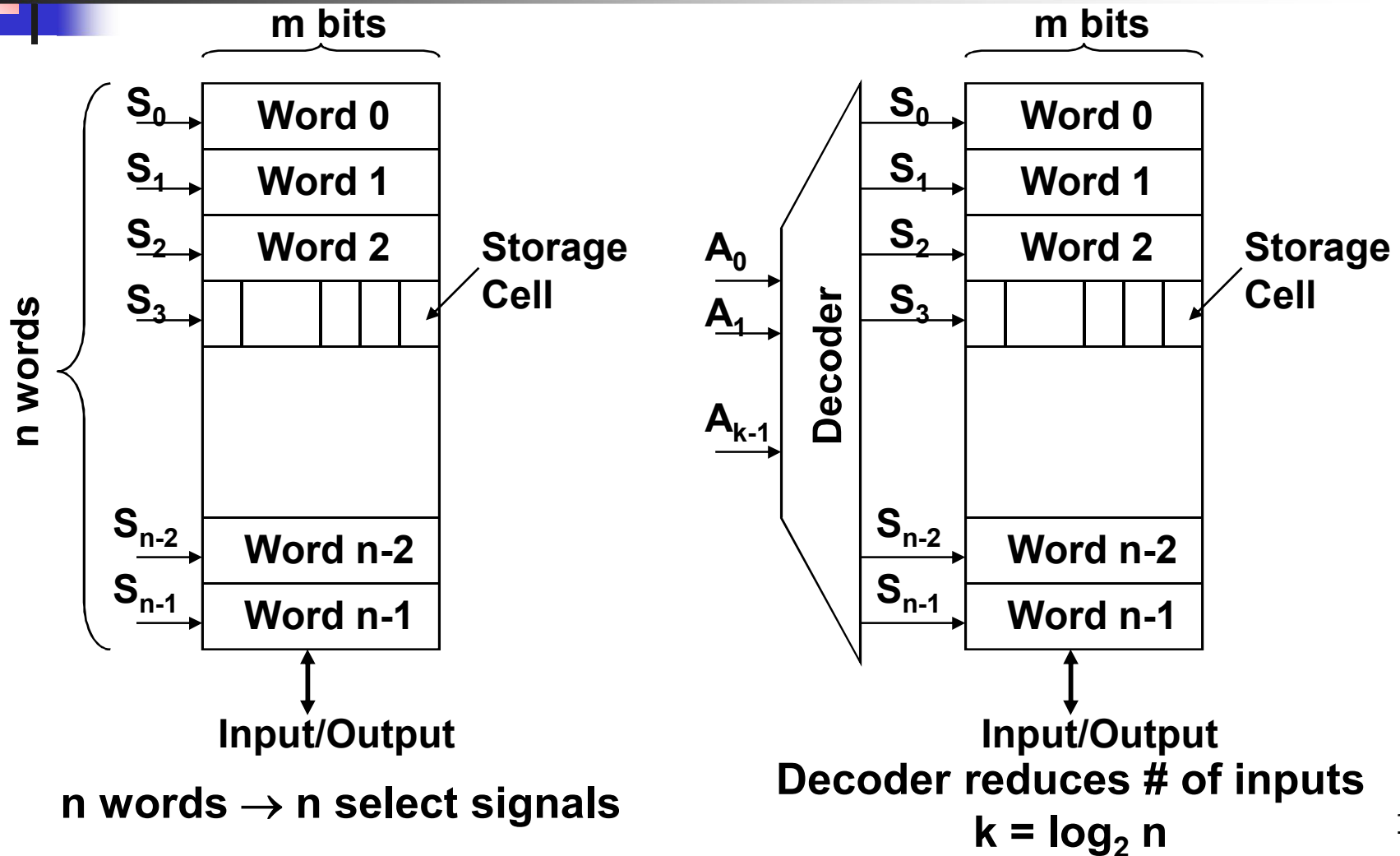


# MOS存储器结构

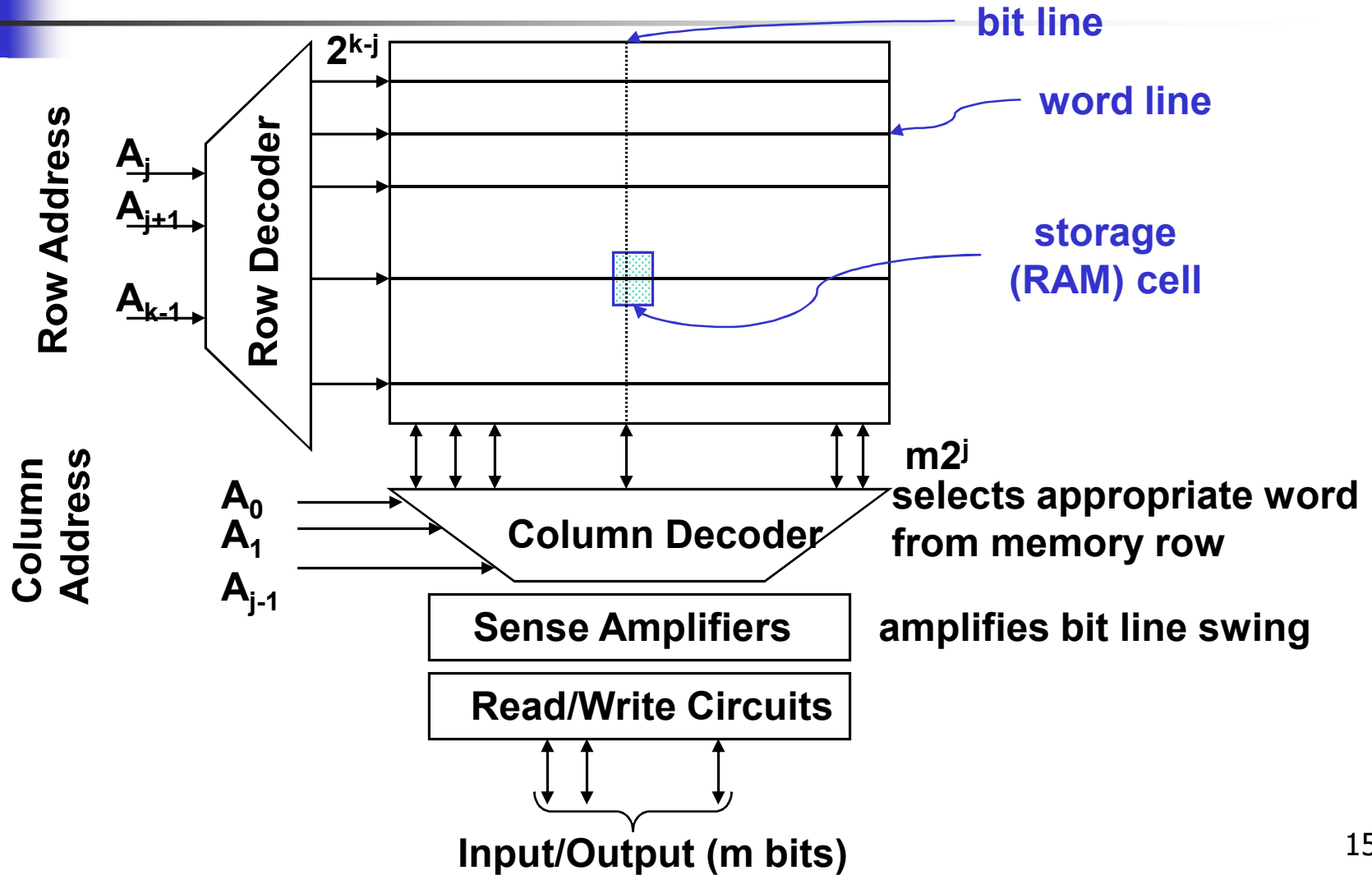
---

- 存储器分类
- **MOS**存储器结构
- 外围电路

# 一维存储器结构



# 二维存储器结构



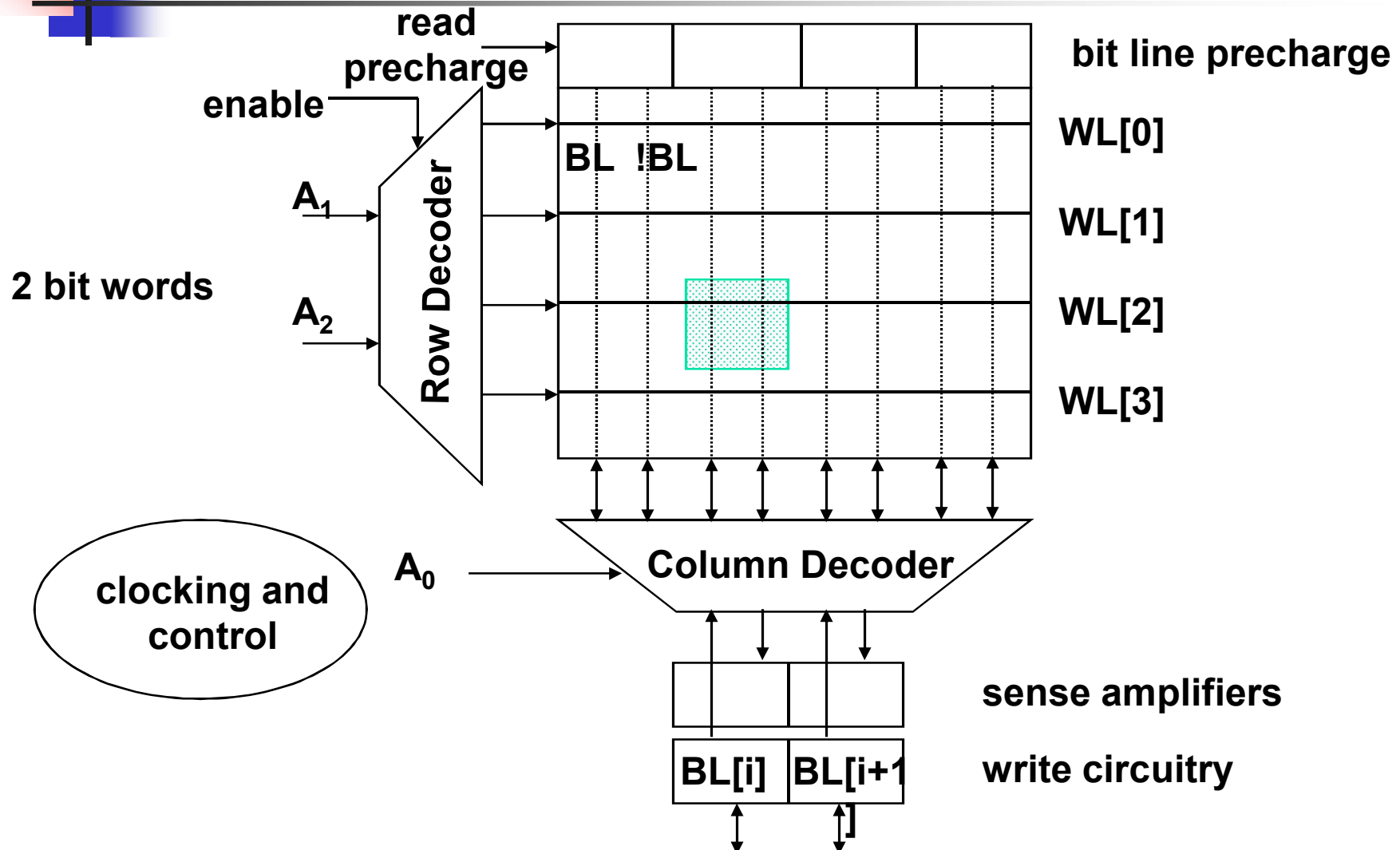


## 存储单元阵列

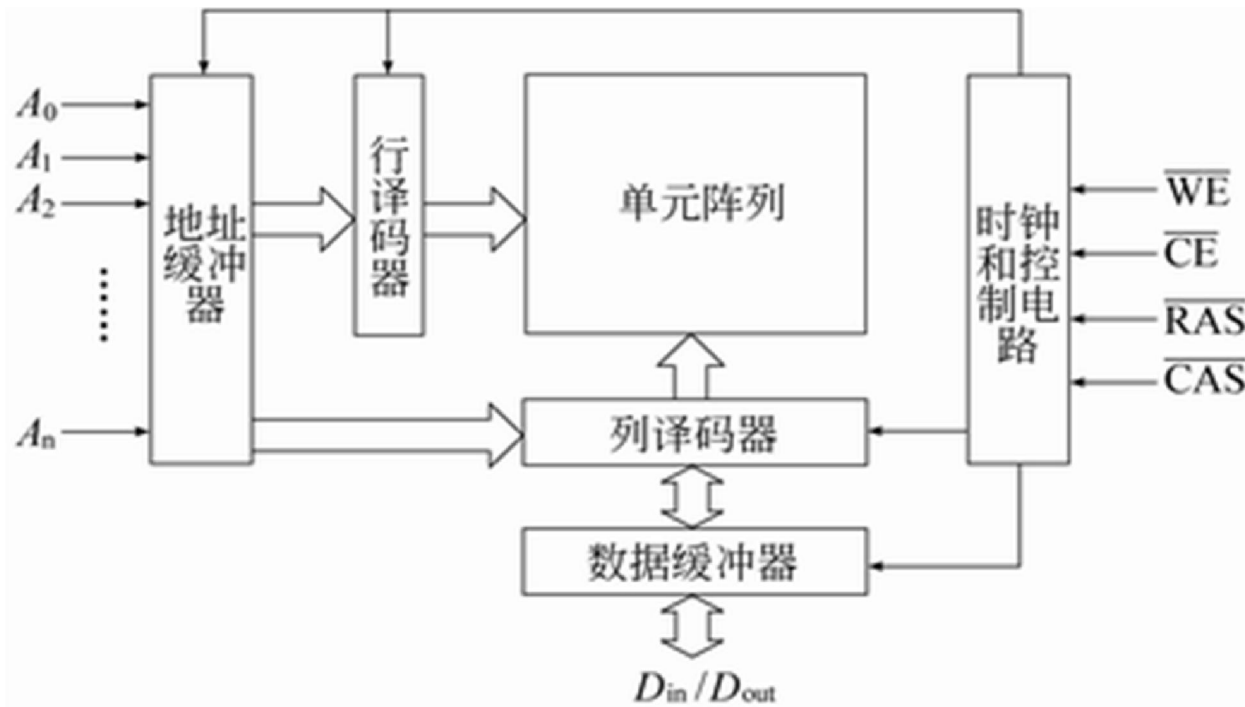
- 存储器的集成度就是指存储单元的数量，也就是存储器的容量。
- 存储单元一般都排成方阵。
- 例：一个4kb的存储器有4096个存储单元，这些单元可以排成64行×64列（=4096）的方阵。
- 一字多位：如一个1k×4b的存储器可以存储1024个字，每个字有4位。因此总的存储容量仍是4096。



# 4x4 SRAM Memory



# 存储器的总体结构



1. 存储单元阵列
2. 译码器
3. 输入/输出缓冲器
4. 时钟和控制电路

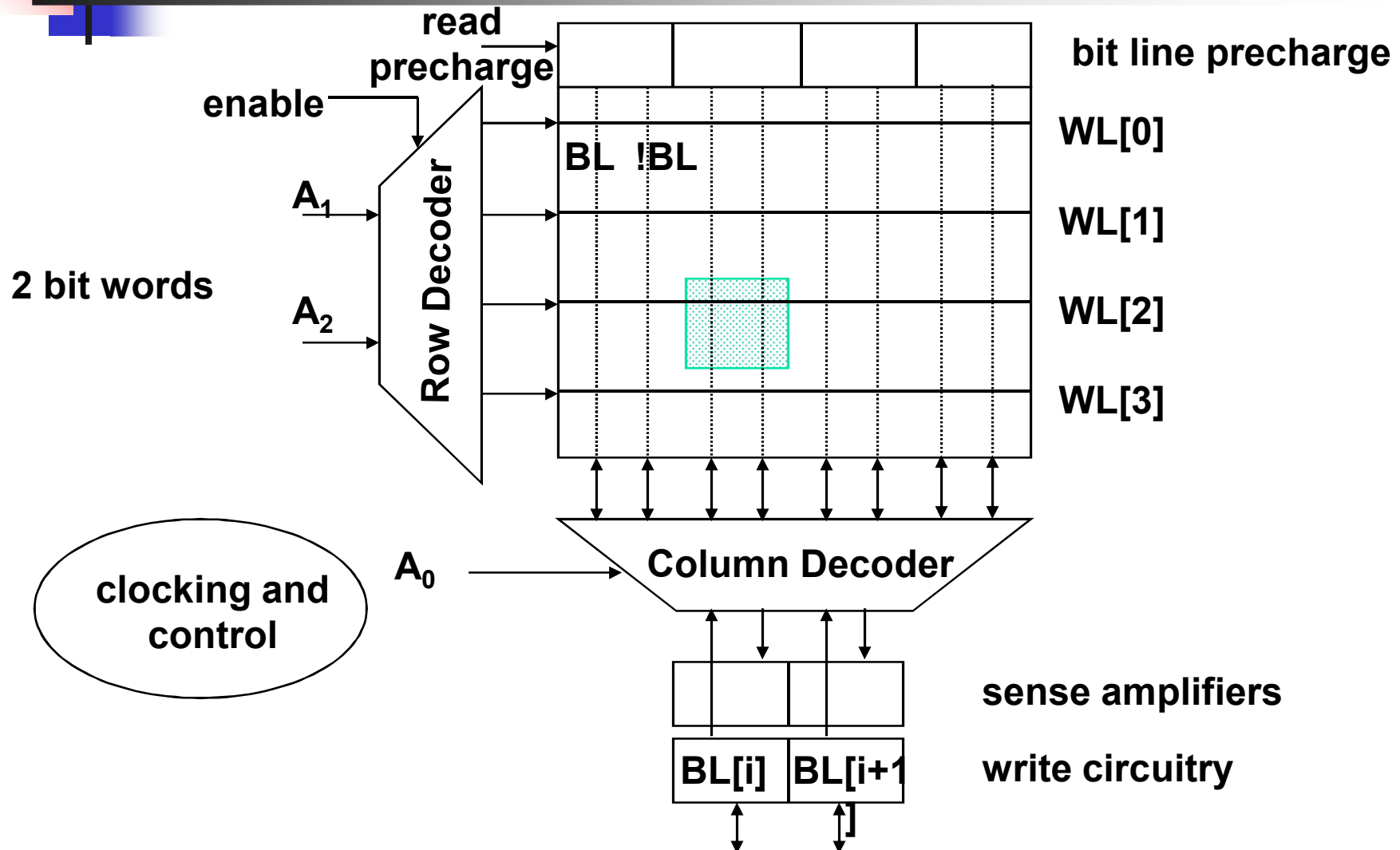


# MOS存储器结构

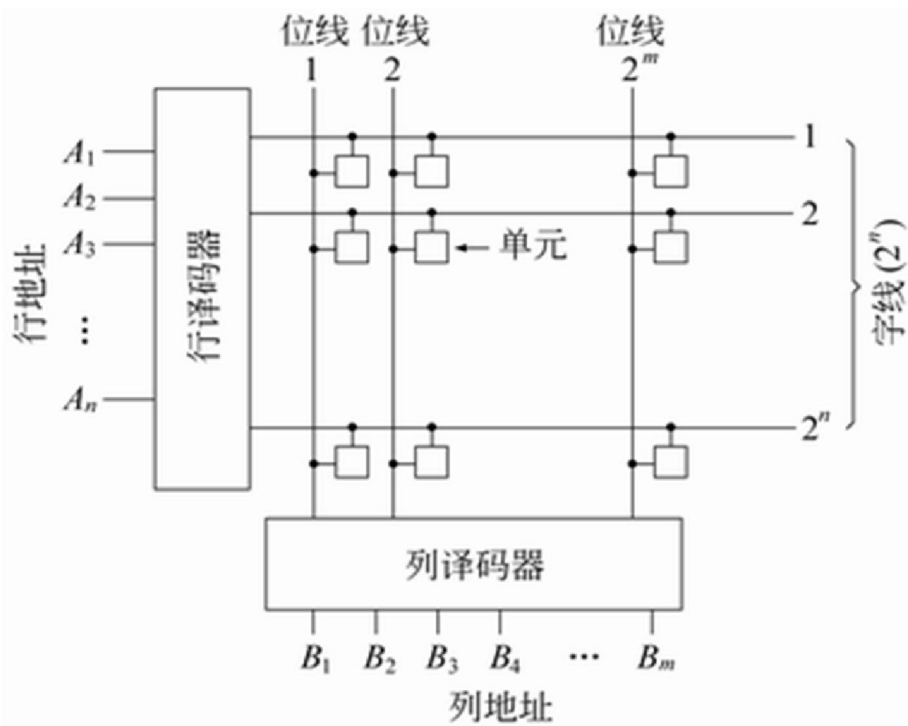
---

- 存储器分类
- **MOS**存储器结构
- 外围电路

# 4x4 SRAM Memory



# 外围电路：译码器



**4Kb**存储器为例：

■每字**1**位：行地址**6**位

（ $2^6=64$ ），列地址**6**位

■每字**4**位：行地址**6**位，列地址**4**位（同时选中**4**列）

# 行译码器

Collection of  $2^M$  complex logic gates  
Organized in regular and dense fashion

## (N)AND Decoder

$$WL_0 = A_0A_1A_2A_3A_4A_5A_6A_7A_8A_9$$

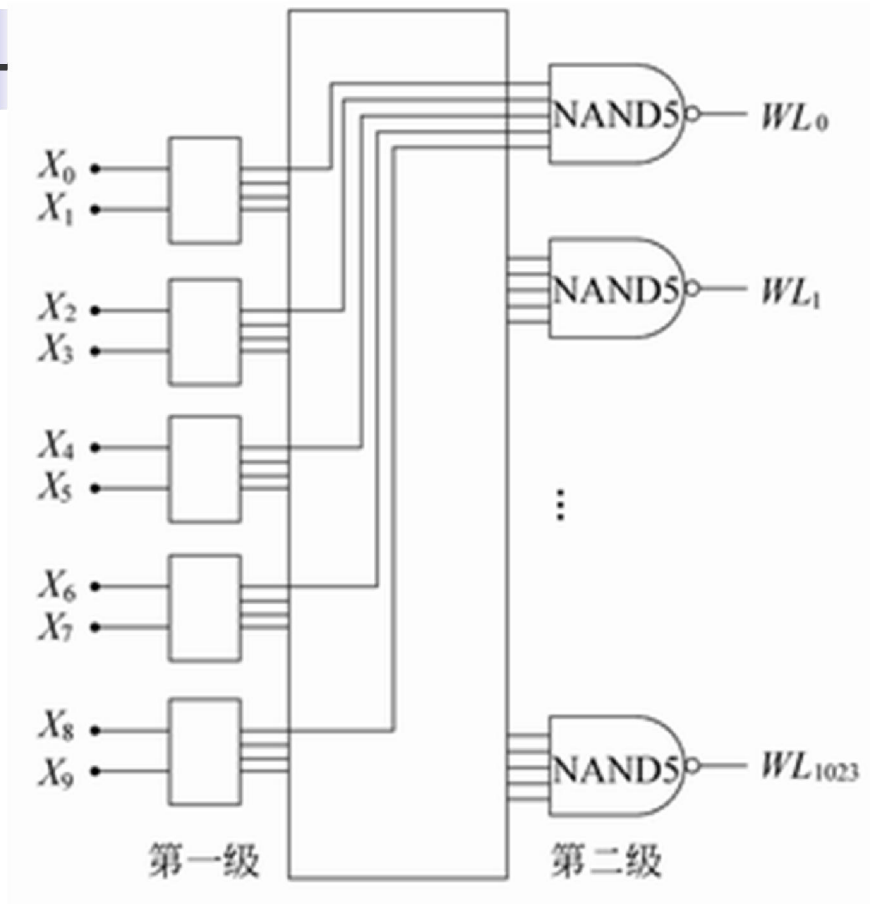
$$WL_{511} = \bar{A}_0\bar{A}_1\bar{A}_2\bar{A}_3\bar{A}_4\bar{A}_5\bar{A}_6\bar{A}_7\bar{A}_8\bar{A}_9$$

## NOR Decoder

$$WL_0 = \overline{A_0 + A_1 + A_2 + A_3 + A_4 + A_5 + A_6 + A_7 + A_8 + A_9}$$

$$WL_{511} = \overline{A_0 + \bar{A}_1 + \bar{A}_2 + \bar{A}_3 + \bar{A}_4 + \bar{A}_5 + \bar{A}_6 + \bar{A}_7 + \bar{A}_8 + \bar{A}_9}$$

# 行译码器：多级译码

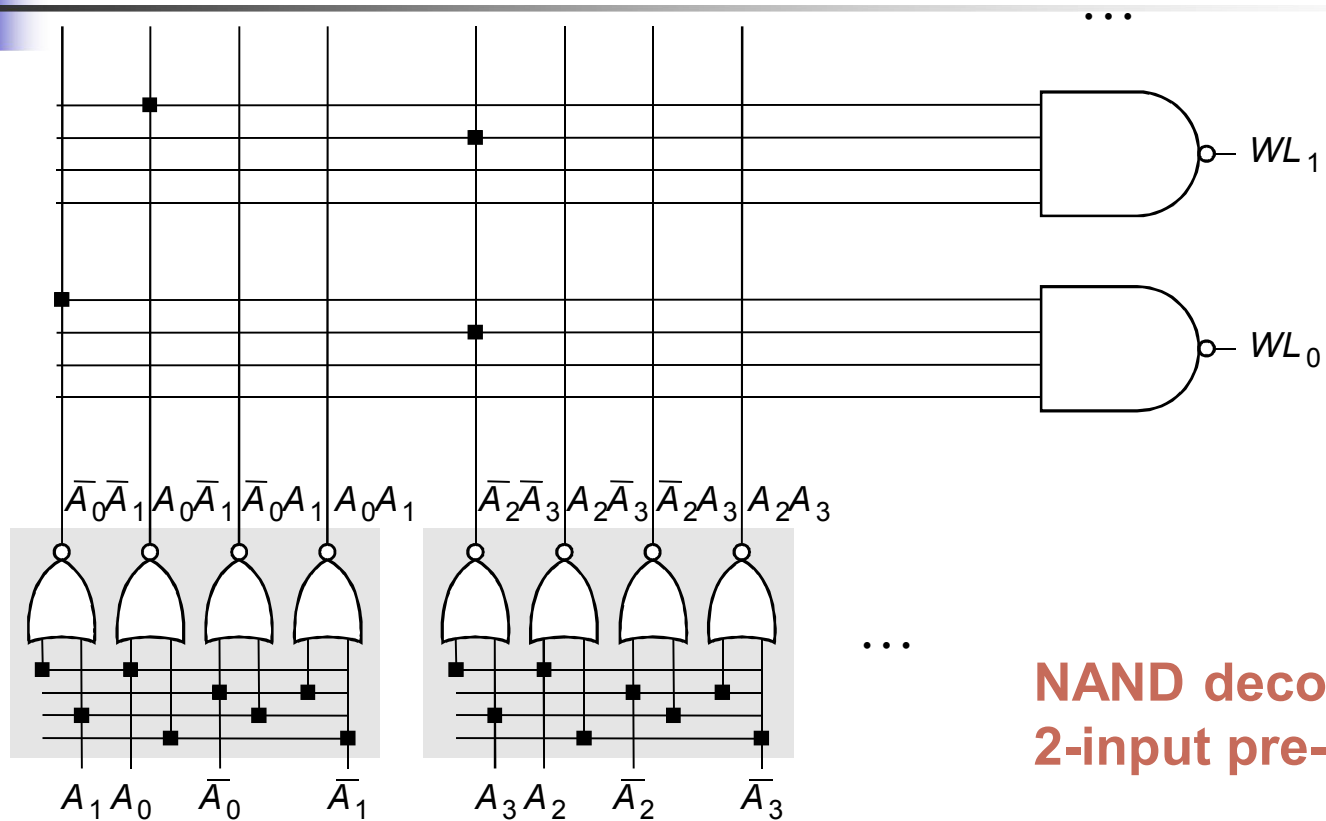


第一级：5个两输入  
与非门，把10个行地址  
分成5组译码

第二级：组合送入  
1024个五输入与非门  
译码

# 行译码器：多级译码

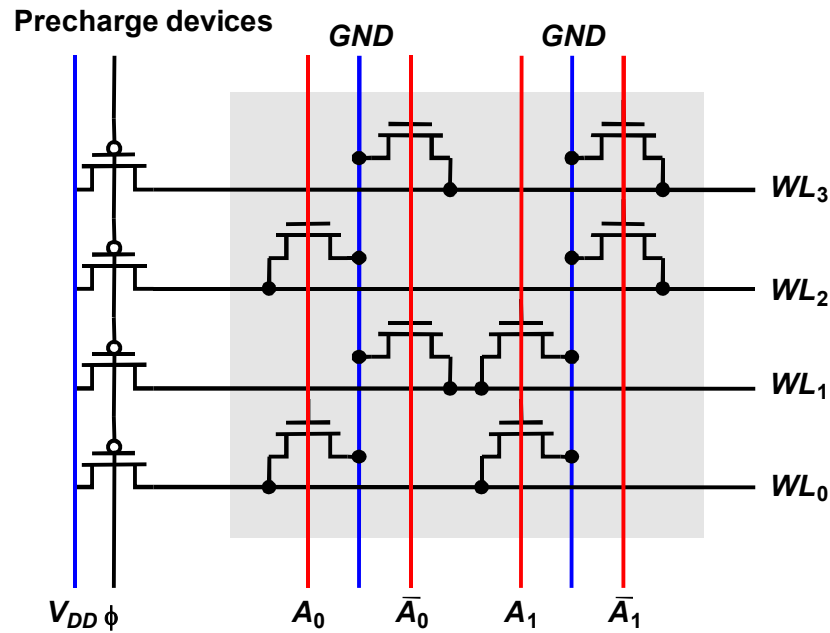
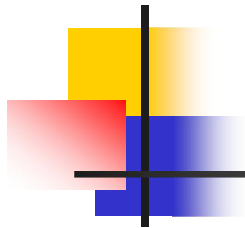
*Multi-stage implementation improves performance*



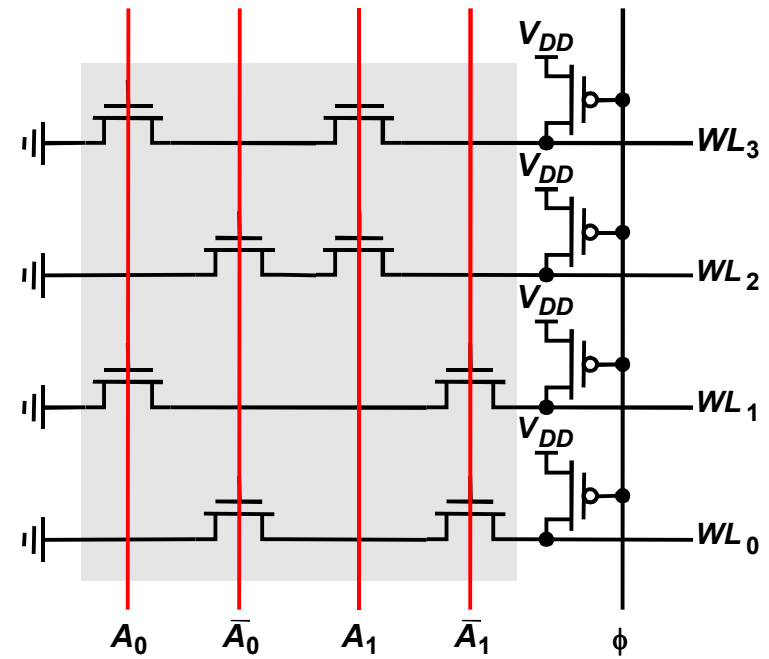
**NAND decoder using  
2-input pre-decoders**



# 动态电路行译码器

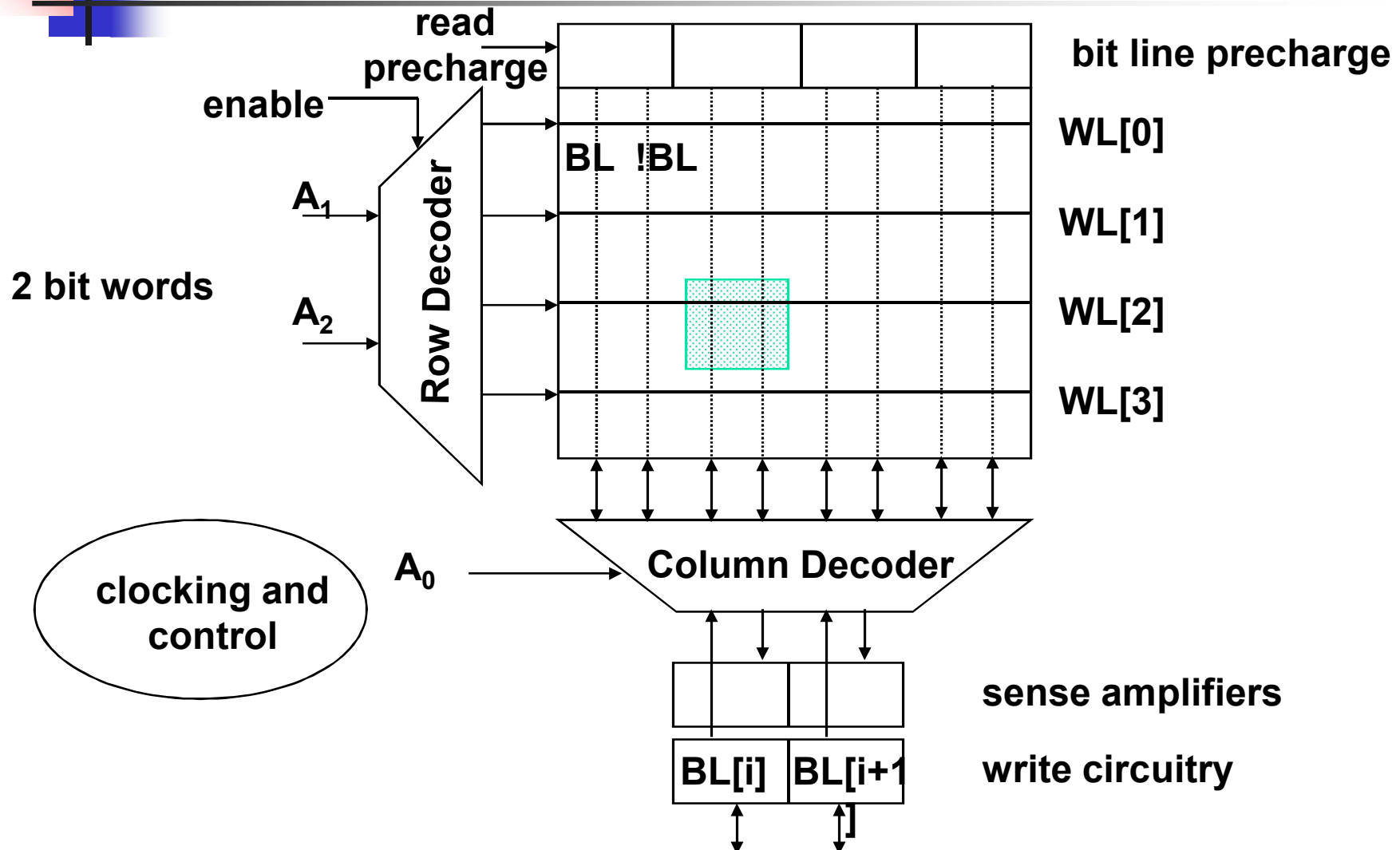


2-input NOR decoder

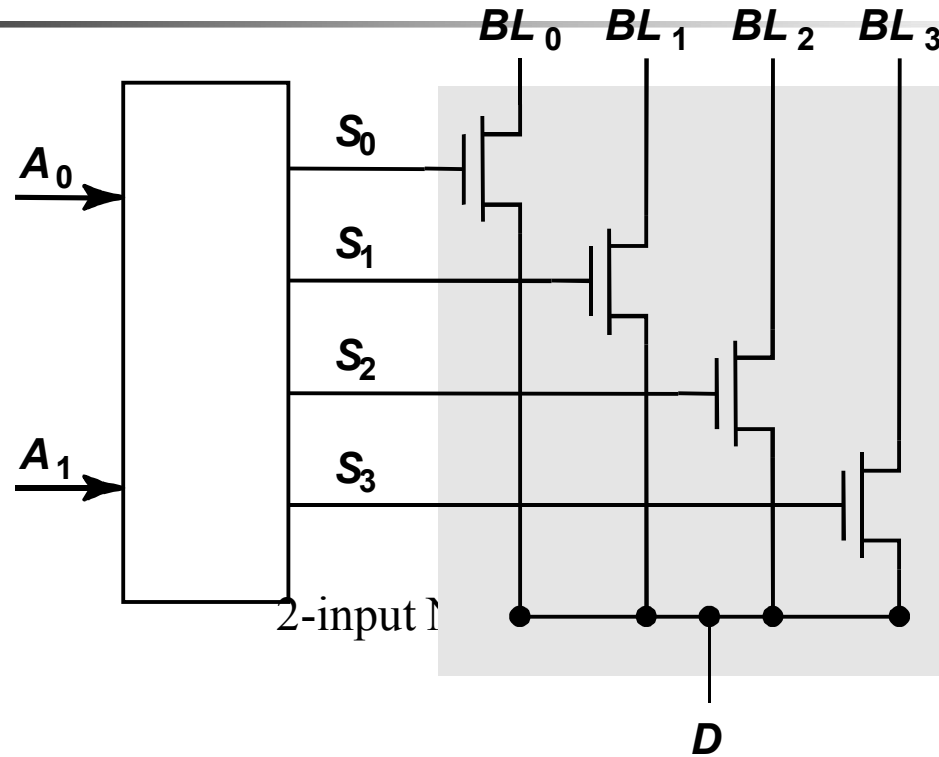


2-input NAND decoder

# 4x4 SRAM Memory



## 列译码器：4-input pass-transistor

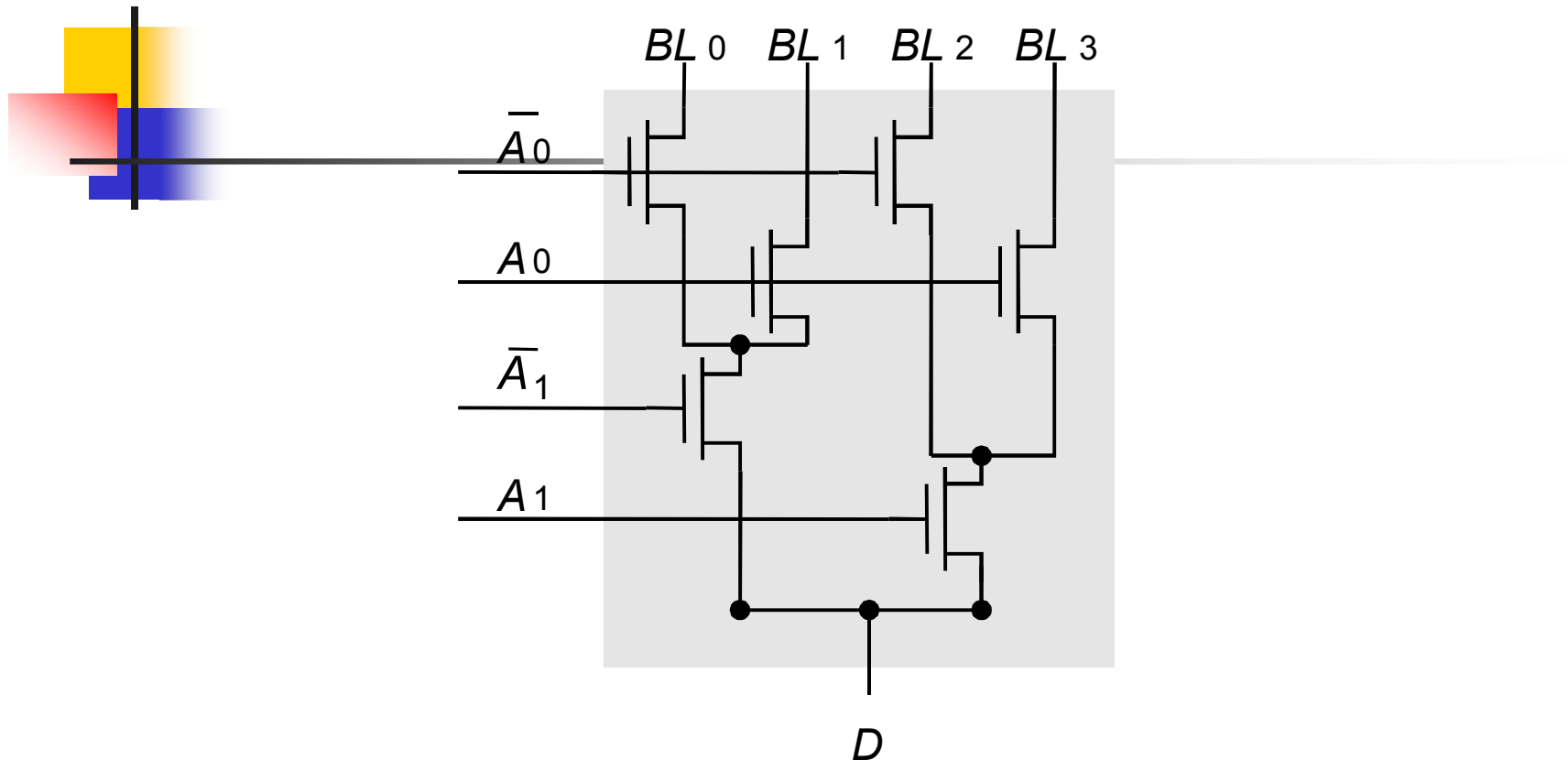


**Advantages: speed ( $t_{pd}$  does not add to overall memory access time)**

**Only one extra transistor in signal path**

**Disadvantage: Large transistor count**

# 4-to-1 tree based column decoder



Number of devices drastically reduced

Delay increases quadratically with # of sections; prohibitive for large decoders

Solutions: buffers

progressive sizing

combination of tree and pass transistor approaches

## 外围电路：输入/输出缓冲器

### ■ 地址输入缓冲器的要求和作用：

- 输入信号缓冲的作用
- 产生地址信号的正、反码
- 足够大的驱动能力
- 地址信号保持

$$WL_0 = A_0A_1A_2A_3A_4A_5A_6A_7A_8A_9$$

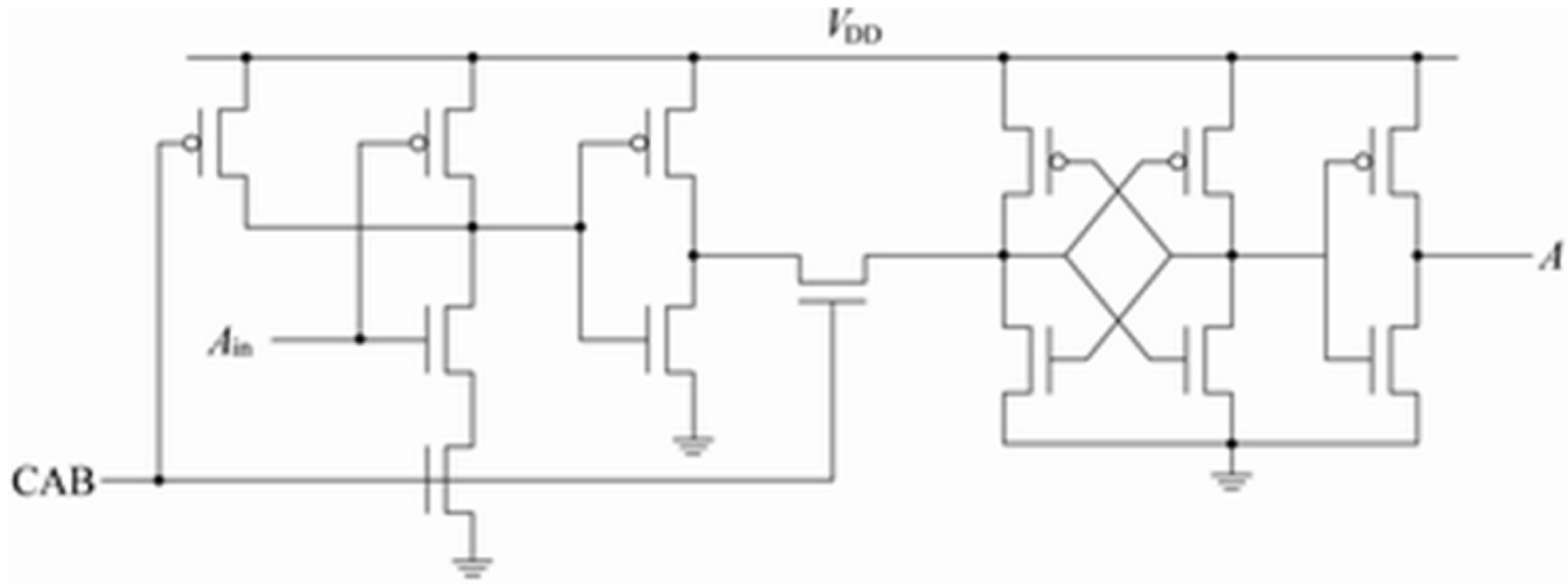
$$WL_{511} = \bar{A}_0\bar{A}_1\bar{A}_2\bar{A}_3\bar{A}_4\bar{A}_5\bar{A}_6\bar{A}_7\bar{A}_8\bar{A}_9$$

### ■ 数据输入缓冲器

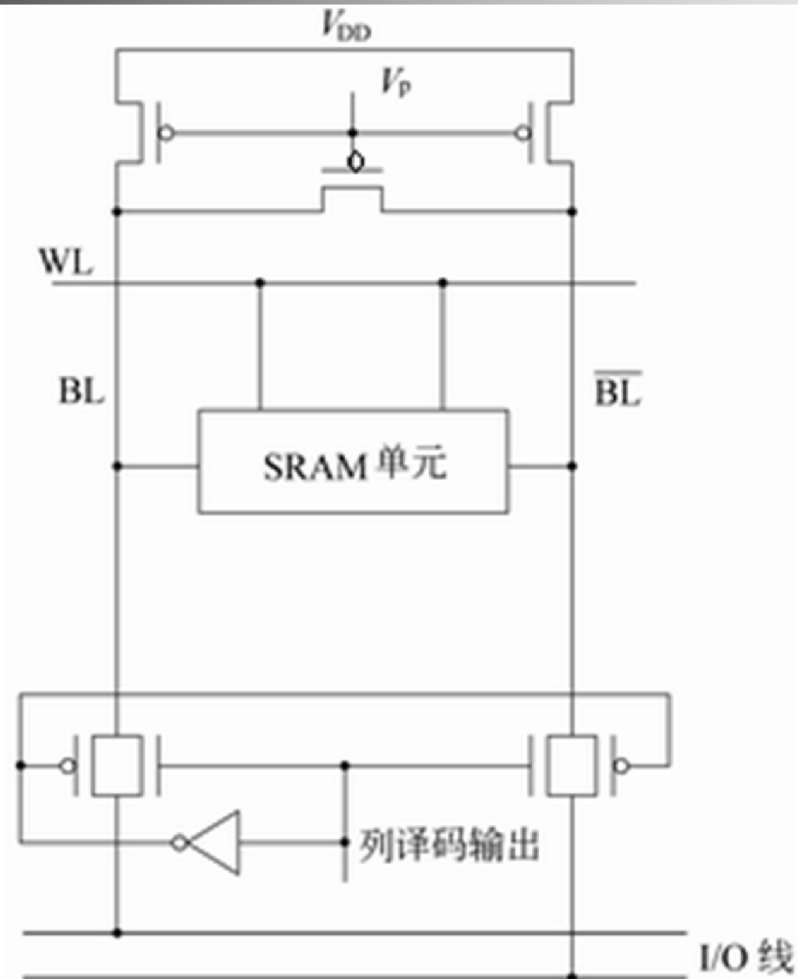
### ■ 数据输出缓冲器

$$WL_0 = \overline{A_0 + A_1 + A_2 + A_3 + A_4 + A_5 + A_6 + A_7 + A_8 + A_9}$$
$$WL_{511} = \overline{A_0 + \bar{A}_1 + \bar{A}_2 + \bar{A}_3 + \bar{A}_4 + \bar{A}_5 + \bar{A}_6 + \bar{A}_7 + \bar{A}_8 + \bar{A}_9}$$

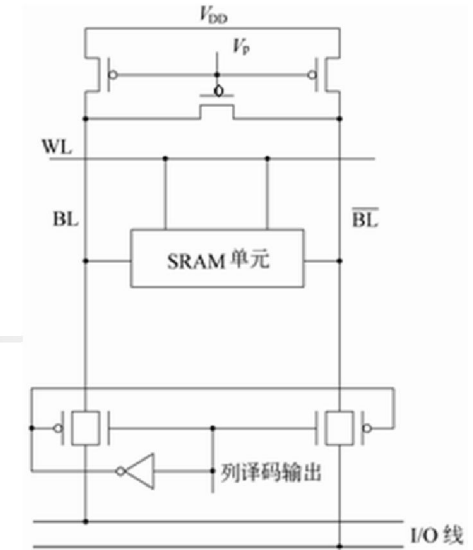
# 地址缓冲器：锁存、驱动



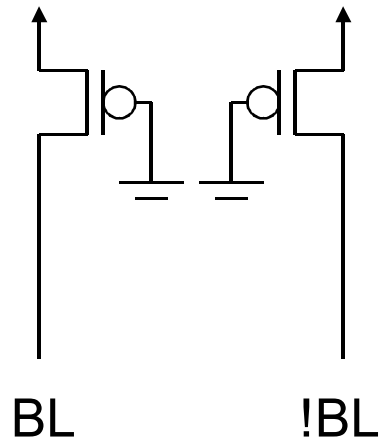
## SRAM的位线结构:位线预充



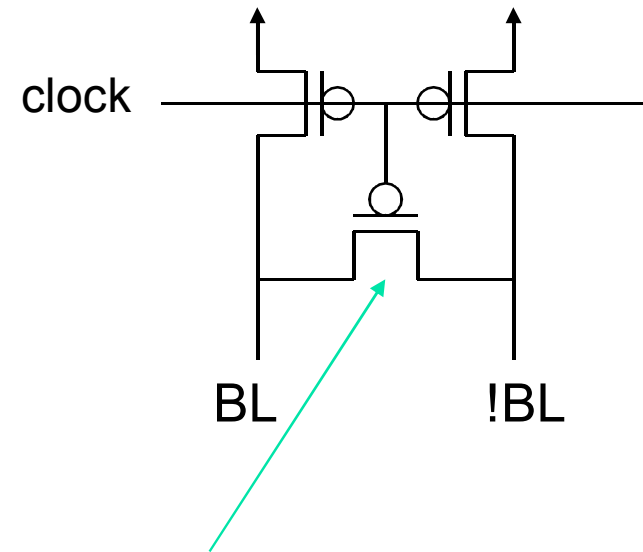
# Bit Line Precharging



### Static Pull-up Precharge



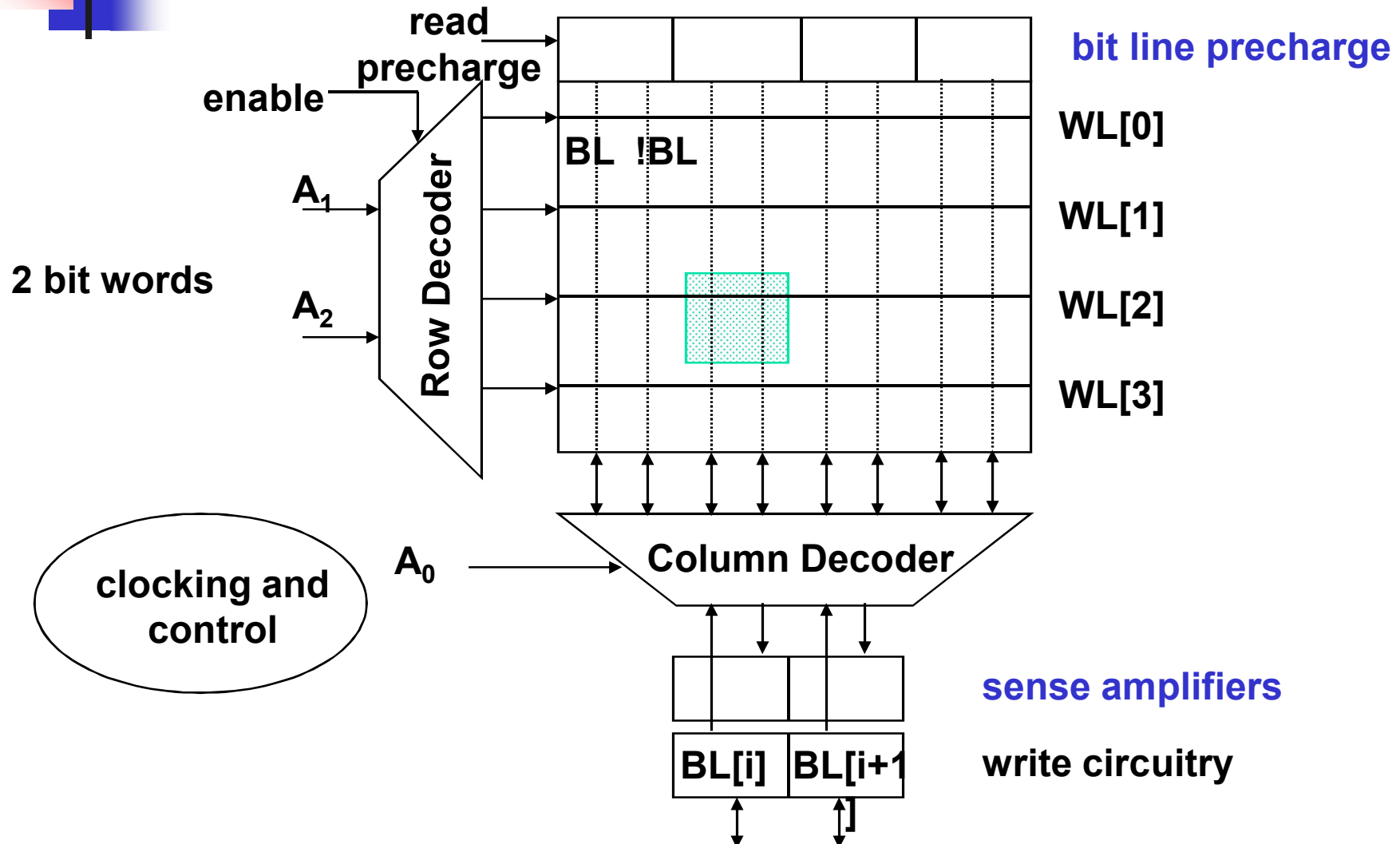
### Clocked Precharge



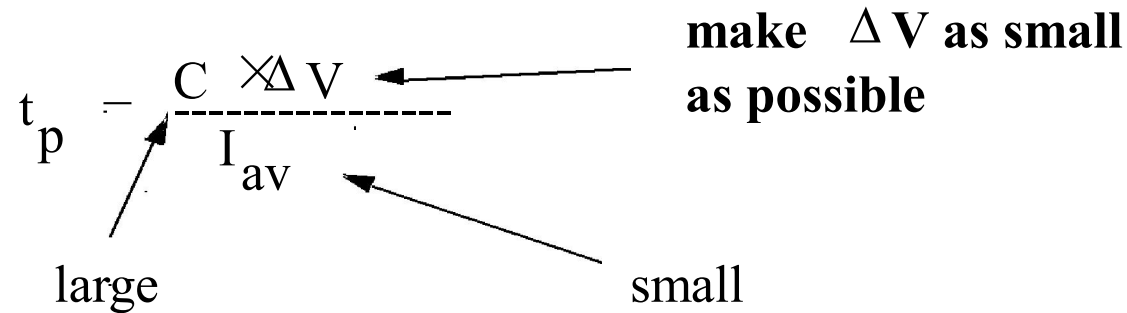
**equalization transistor**



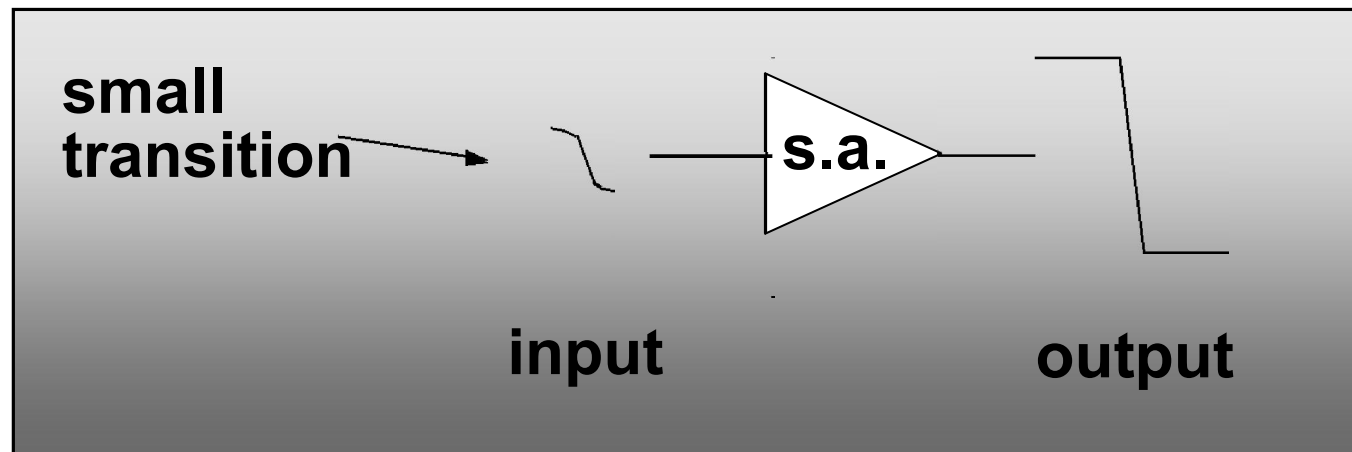
# 外围电路：灵敏放大器



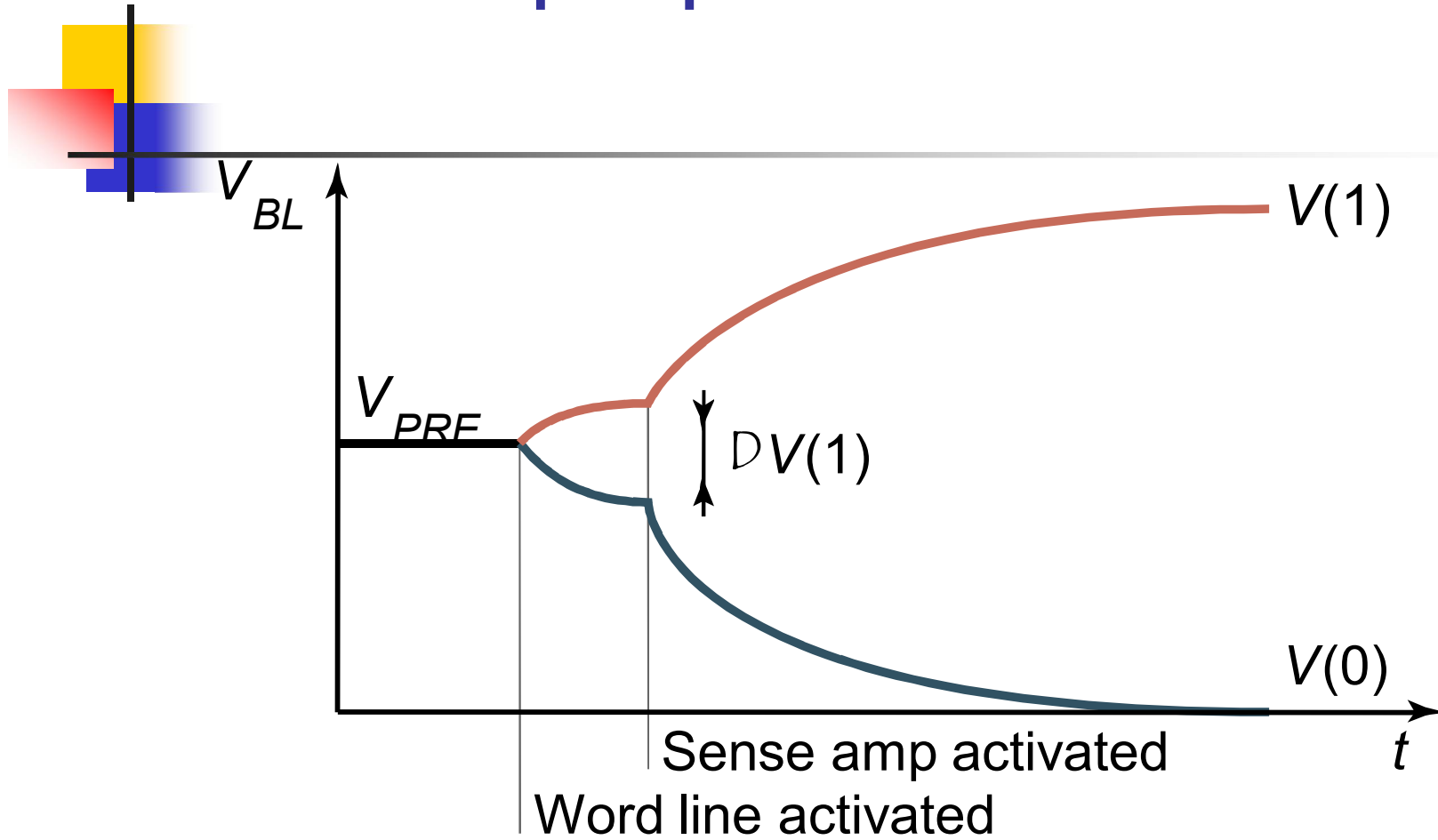
# Sense Amplifiers



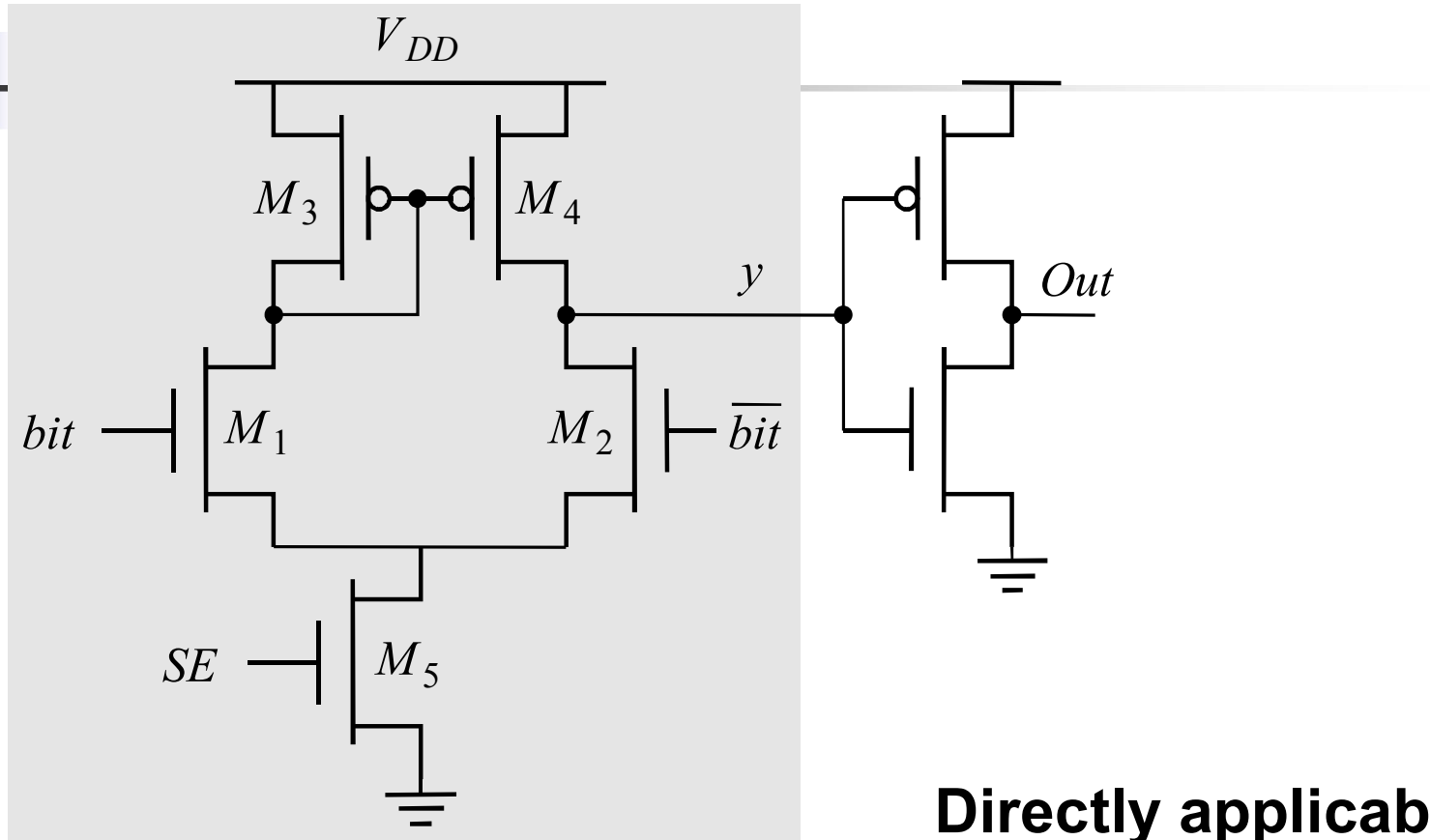
**Idea: Use Sense Amplifier**



# Sense Amp Operation

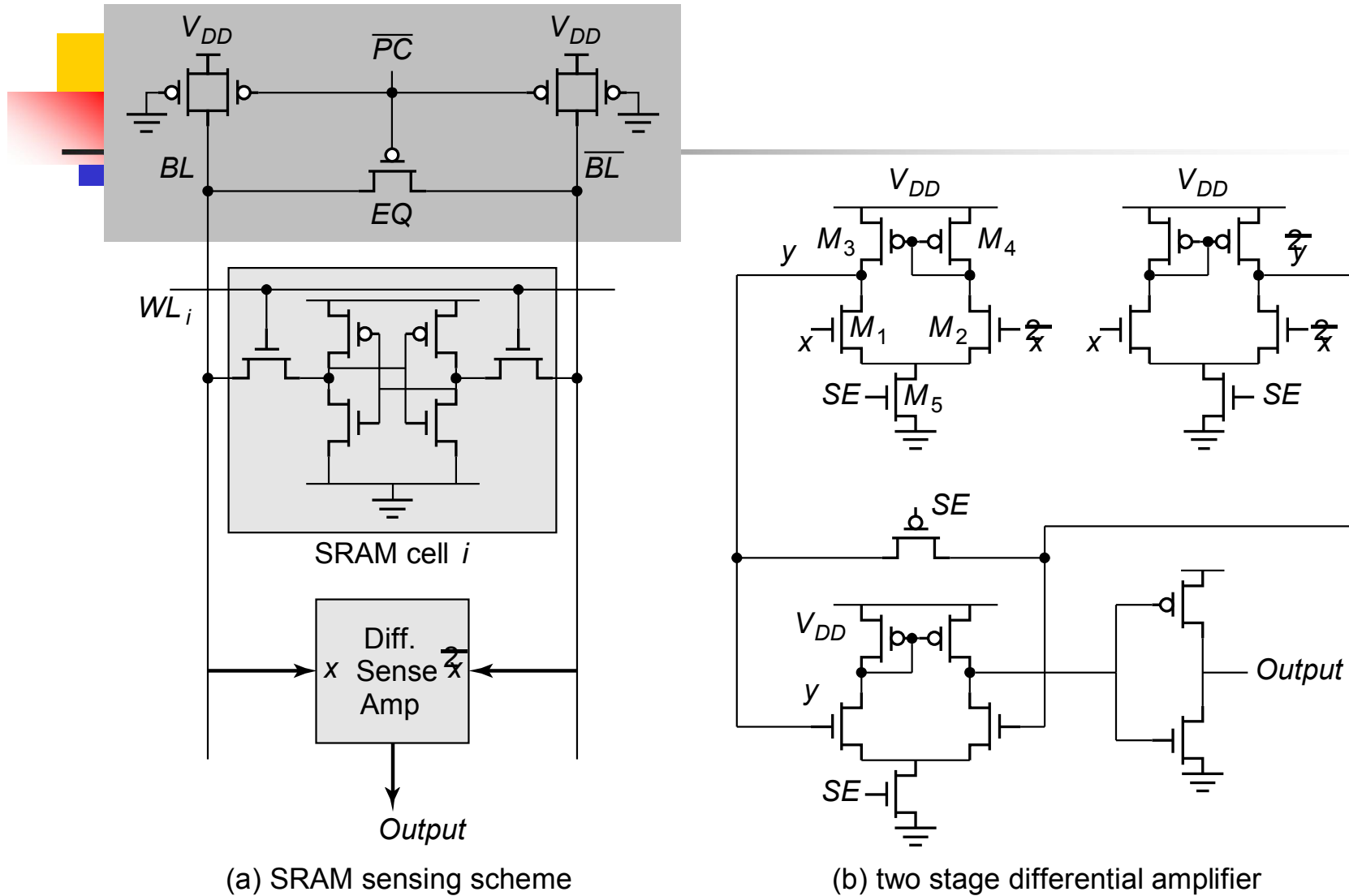


# Differential Sense Amplifier

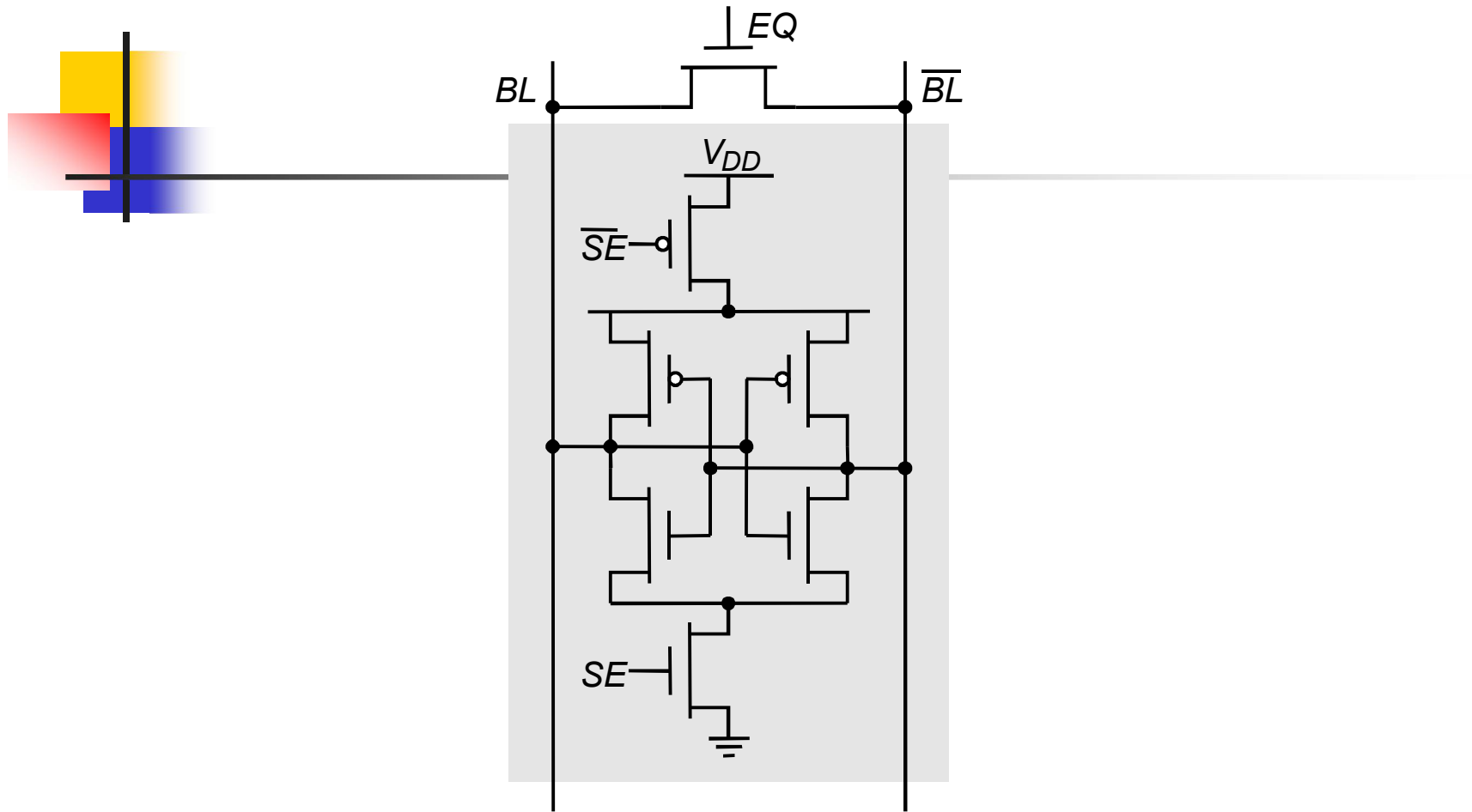


**Directly applicable to  
SRAMs**

# Differential Sensing — SRAM



# Latch-Based Sense Amplifier (DRAM)



Initialized in its meta-stable point with  $EQ$

Once adequate voltage gap created, sense amp enabled with  $SE$

Positive feedback quickly forces output to a stable operating point.

# 一个SRAM的总体结构

