



北京邮电大学研究生课程

数字超大规模集成电路分析与设计

第十讲 存储器和阵列结构（二）

梁利平

集成电路学院

2023年11月29日





第十一讲 存储器和阵列结构 (二)

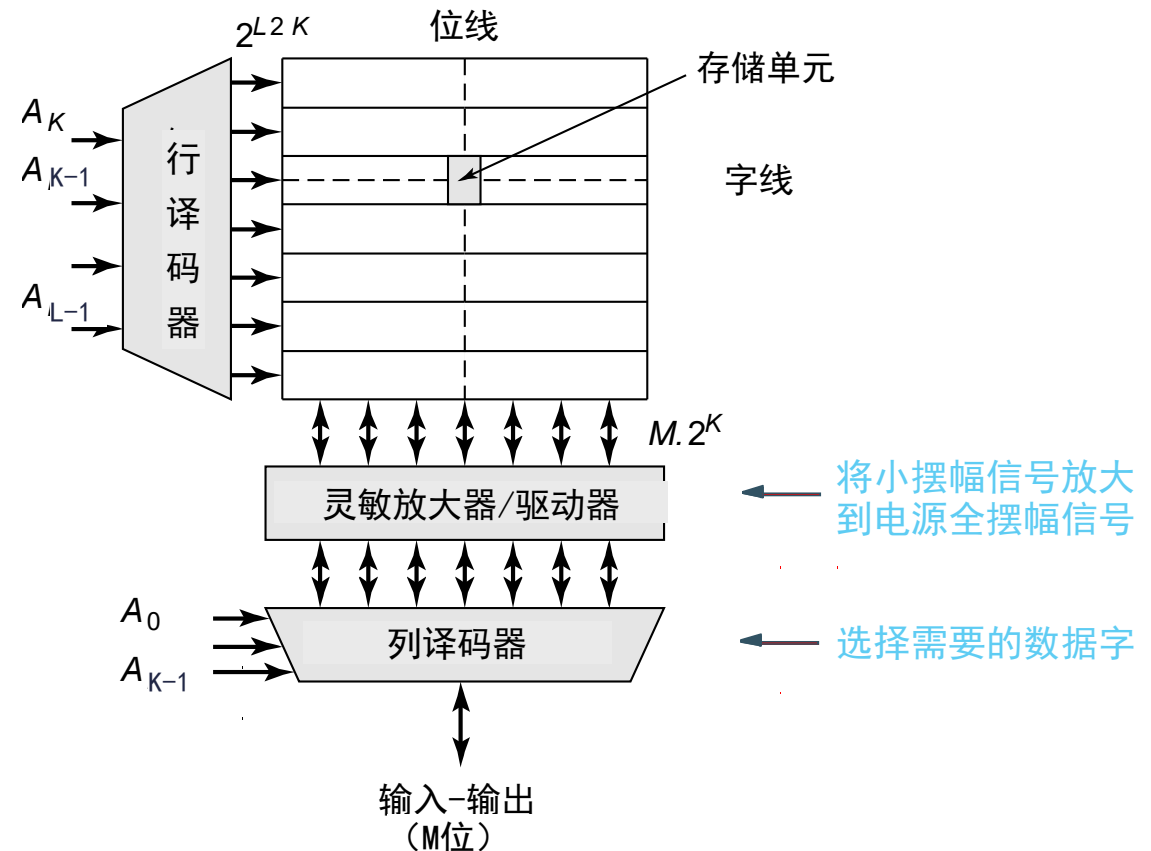
本讲主要内容

- (一) 存储器外围电路
- (二) 存储器设计实例

1.1 存储器外围电路

□ 存储器的外围电路

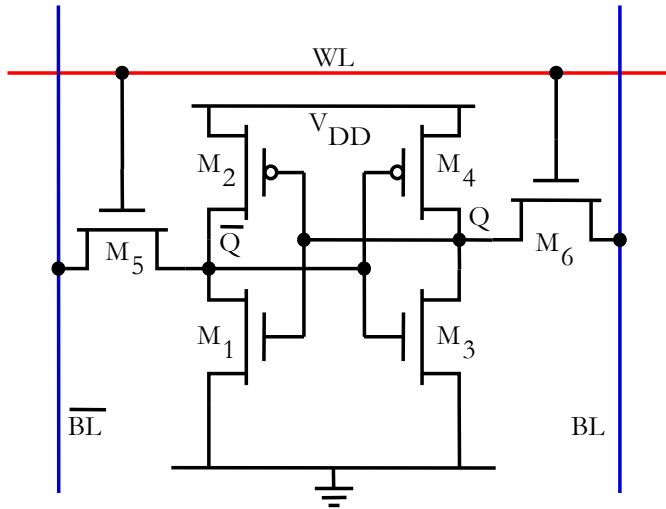
- 译码器
- 灵敏放大器
- 输入/输出缓冲器
- 控制 / 时序电路



1.2 存储器外围电路

□ 行地址译码器

- M个地址，需要 2^M 个复杂门；
- 组织方式：规则、高密度-与存储整列在尺寸上匹配。



(N)AND 译码器

$$WL_0 = A_0 A_1 A_2 A_3 A_4 A_5 A_6 A_7 A_8 A_9$$

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

NOR 译码器

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

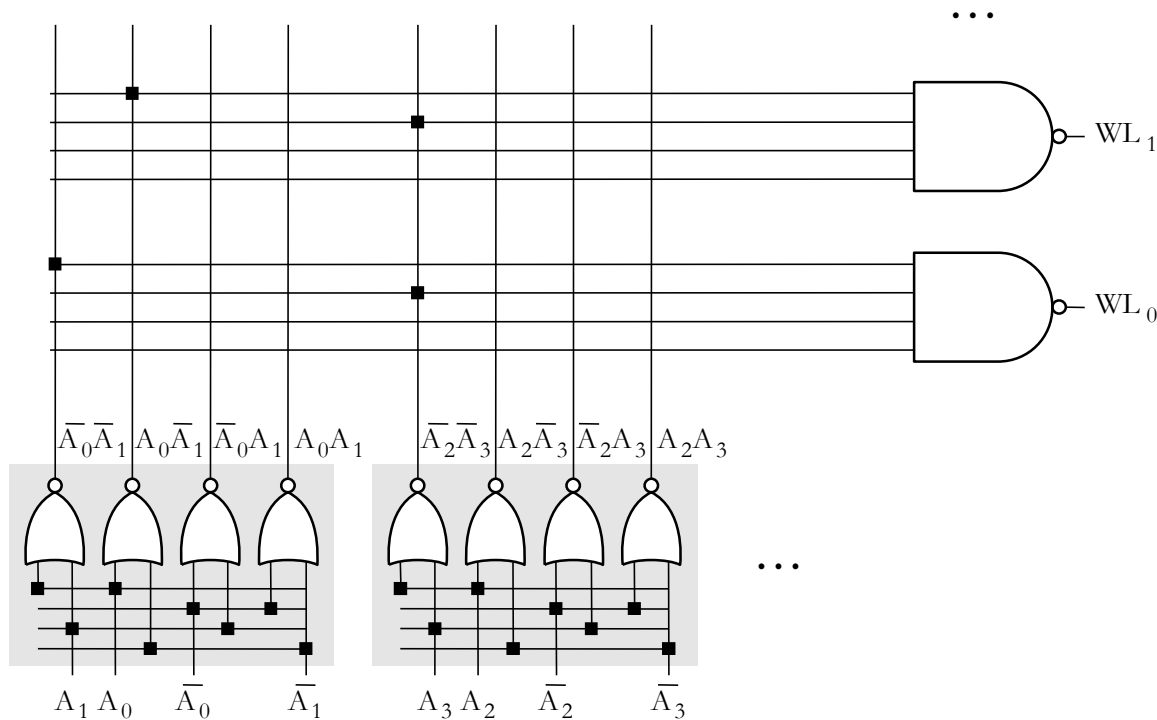
$$WL_{511} = \overline{A_0 + \bar{A}_1 + \bar{A}_2 + \bar{A}_3 + \bar{A}_4 + \bar{A}_5 + \bar{A}_6 + \bar{A}_7 + \bar{A}_8 + \bar{A}_9}$$

1.3 存储器外围电路

□ 层次化的行译码器

- 速度：利用多级结构来提高性能
- 功耗：预译码器一个选择信号就可关闭。

通过再一级信号
降低延时

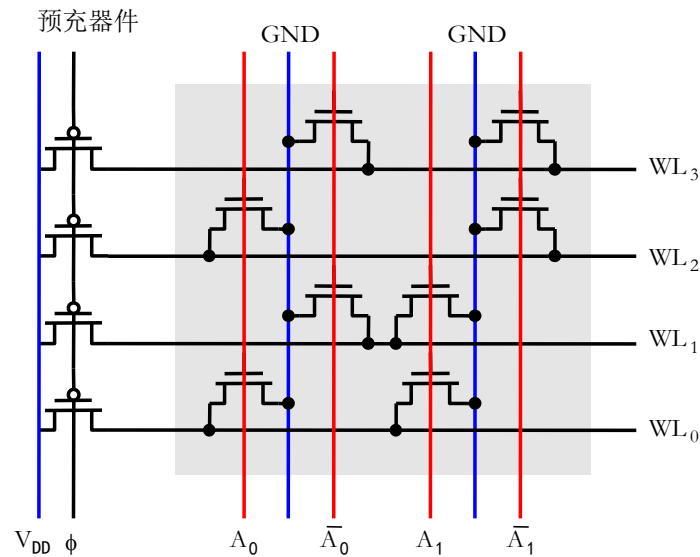


使用2输入预译码器的
NAND 译码器

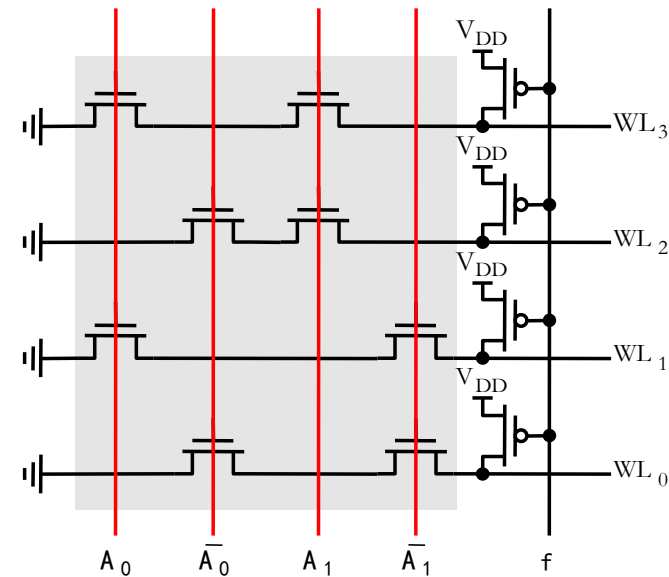
1.4 存储器外围电路

电容小, 充放电时间短快

□ 动态行译码器



2输入 NOR 译码器



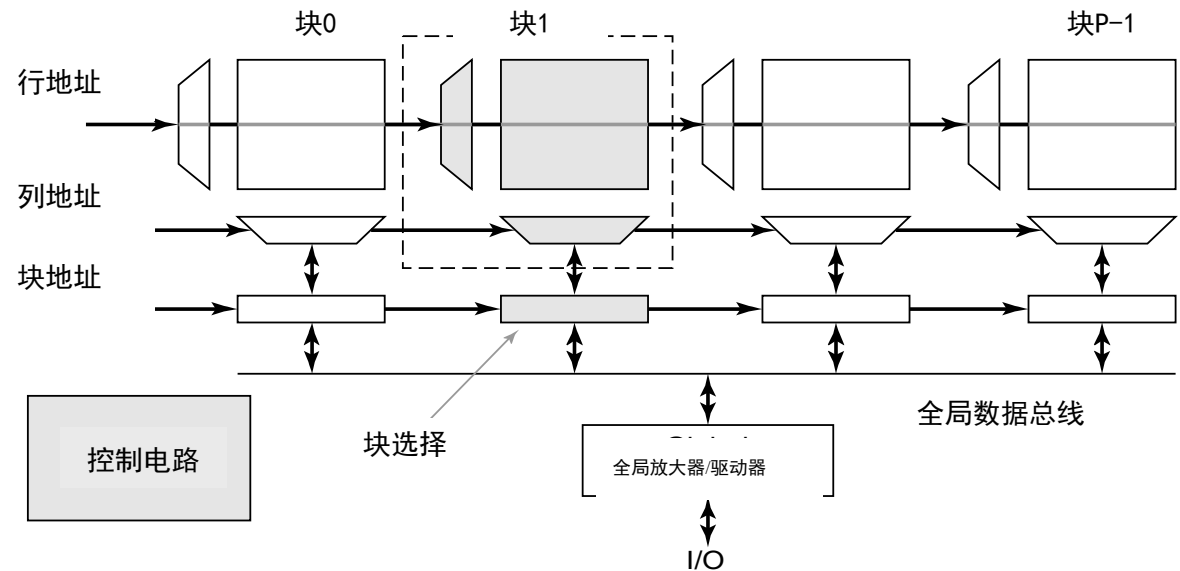
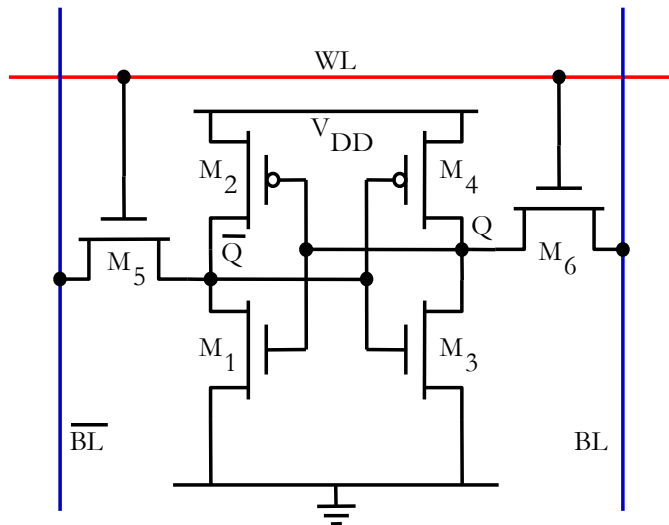
2输入 NAND 译码器

译码器速度-一个方向的过渡决定译码器速度。

1.5 存储器外围电路

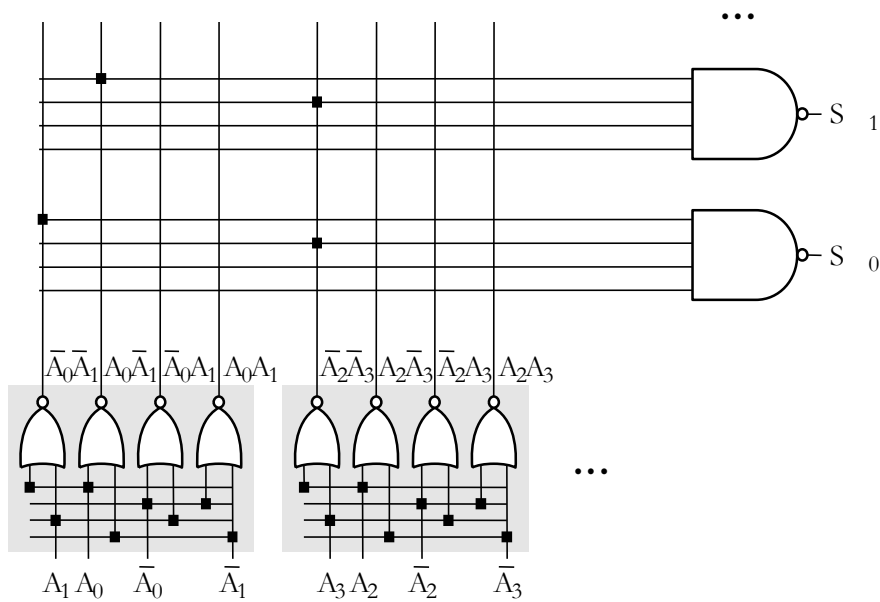
□ 列(块)地址译码器

- 列译码器与存储阵列的位线节距匹配；
- K 位的列（块）地址译码器，是一个具有 $2K$ 个输入的多路开关；
- 读操作：提供预充电位线至灵敏放大器的放电路径。
- 写操作：能驱动位线至低电平。

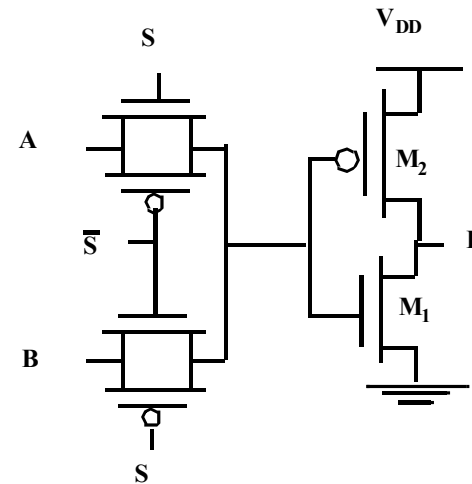


1.6 存储器外围电路

□ 列（块）译码器



层次化的译码器

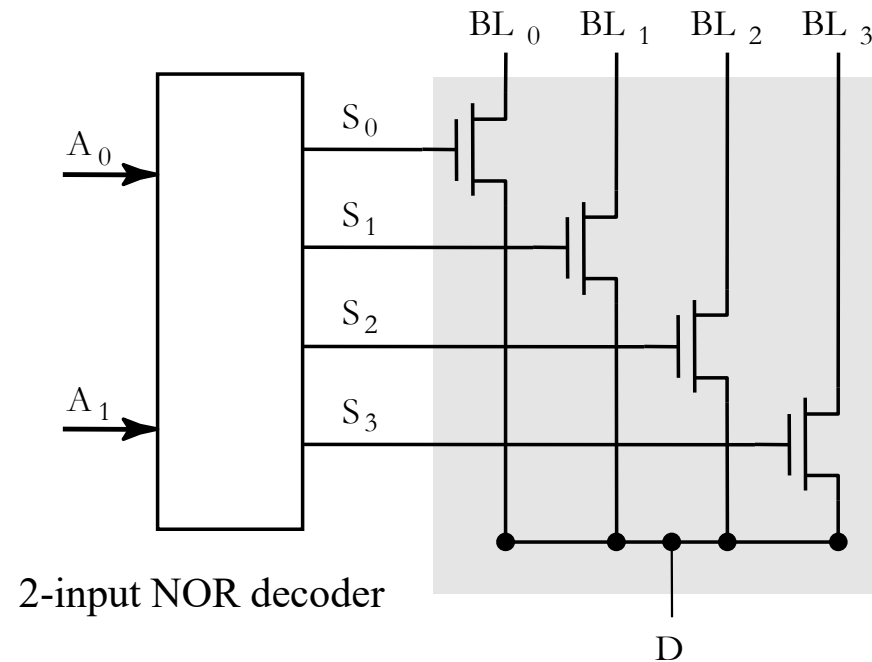


传输门复用器

1.7 存储器外围电路

□ 传输管列译码器-4输入

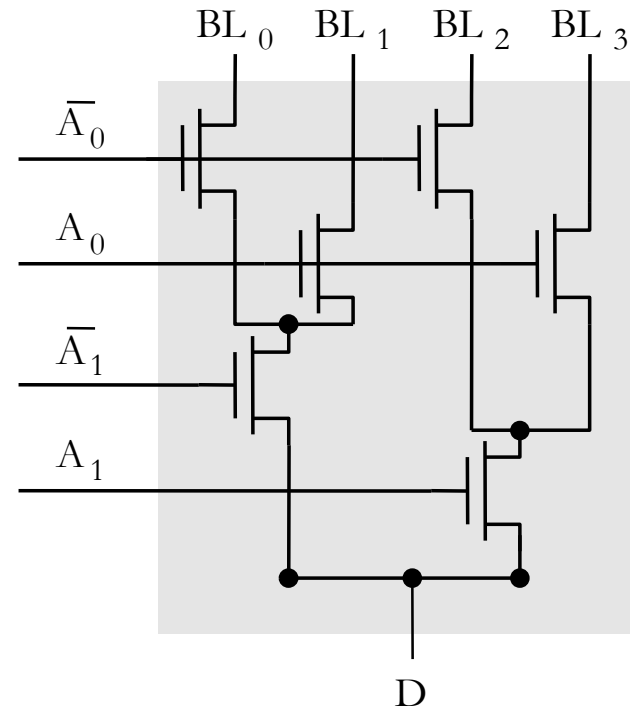
- 优点: 速度快 (t_{pd} 不会叠加到整体存储器的存取时间中)
数据路径上只有一个串联的晶体管。
- 缺点: 晶体管数量多。



1.8 存储器外围电路

□ 树状列译码器-4-1

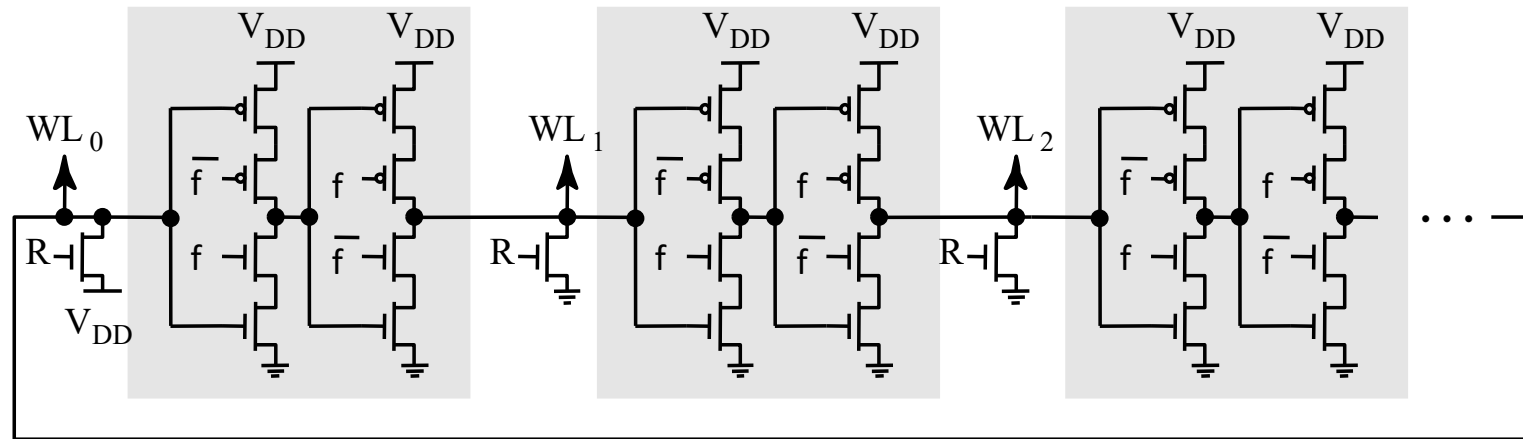
- 优点：器件数量减少
- 缺点：延迟显著增加
- 解决方法
 - ✓ 插入缓冲器buffer
 - ✓ 递进地尺寸调整
 - ✓ 树状与传输管相结合的方式



1.9 存储器外围电路

□ 非随机存储器的译码器

- 非随机存储类存储器并需要一个全译码器



循环移位寄存器的译码器
R信号使指针回到第一个位置

1.10 存储器外围电路

□ 灵敏放大器 (Sense Amplifiers)

$$t_p = \frac{C \times \Delta V}{I_{av}}$$

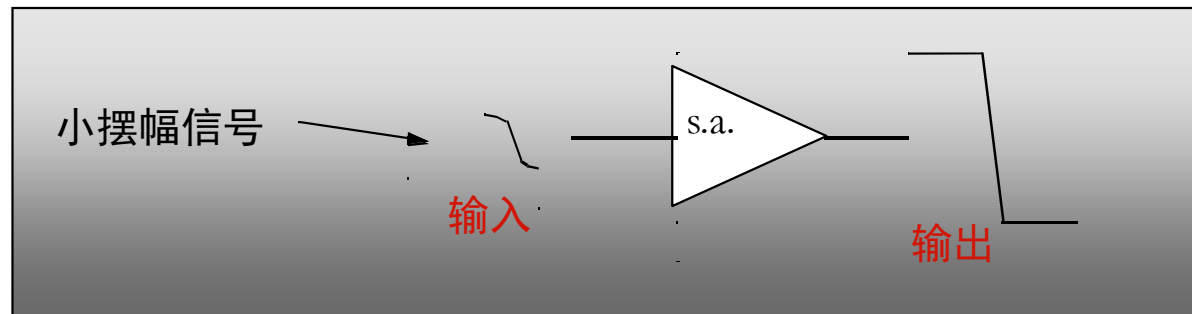
使 ΔV 尽可能的小

大

小

电荷
平均电流

方法：使用灵敏放大器 (Sense Amplifier)



放大： DRAM通过电荷共享产生电压差，需要放大才能正确工作；

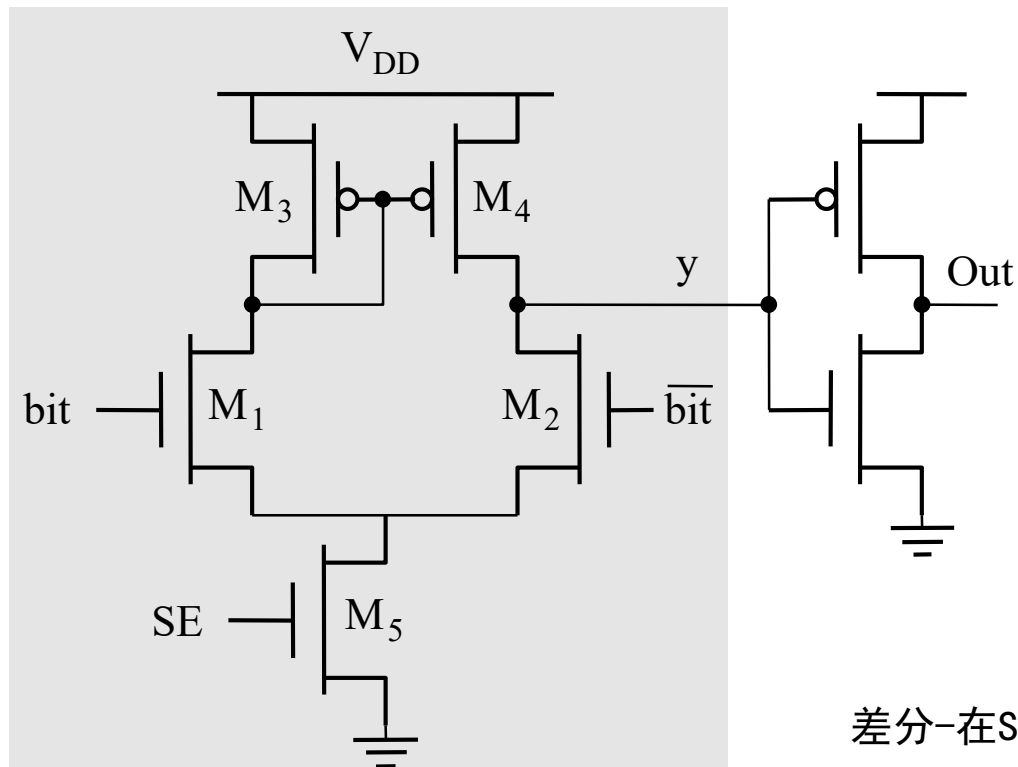
减小延迟： 检测位线上很小的过渡变化并放大到较大的输出摆幅，弥补存储器单元优先的扇出驱动能力；加速位线过渡过程。

降低功耗： 减小位线上信号摆幅，降低充电放电功耗；

恢复信号： DRAM中的刷新功能，需要灵敏放大后把位线驱动至全信号摆幅。

1.11 存储器外围电路

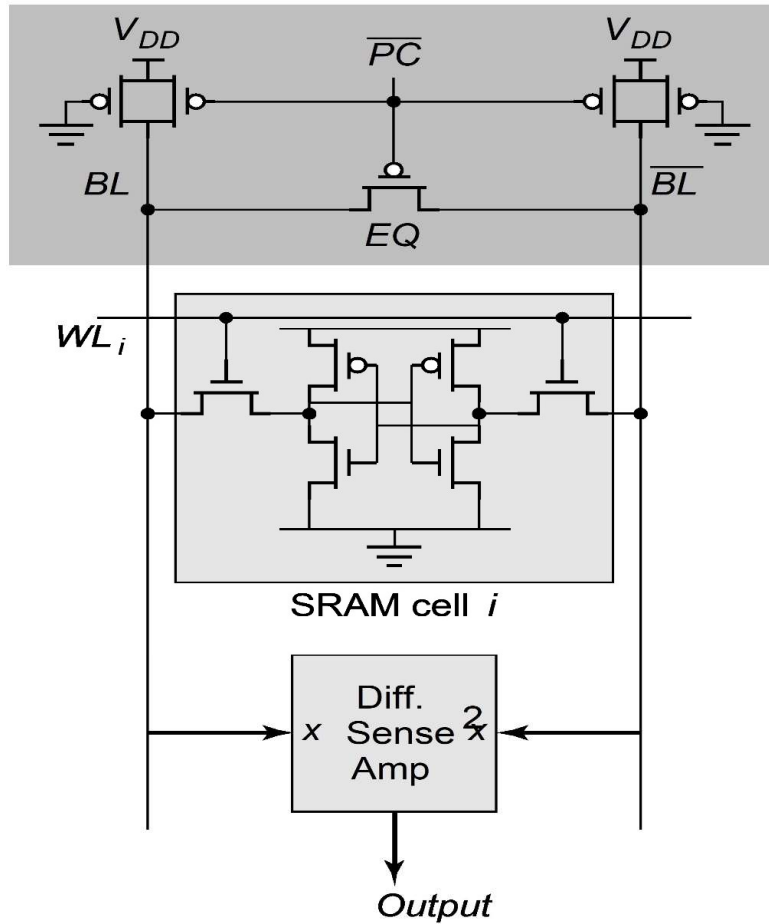
□ 差分灵敏放大器 (Sense Amplifier)



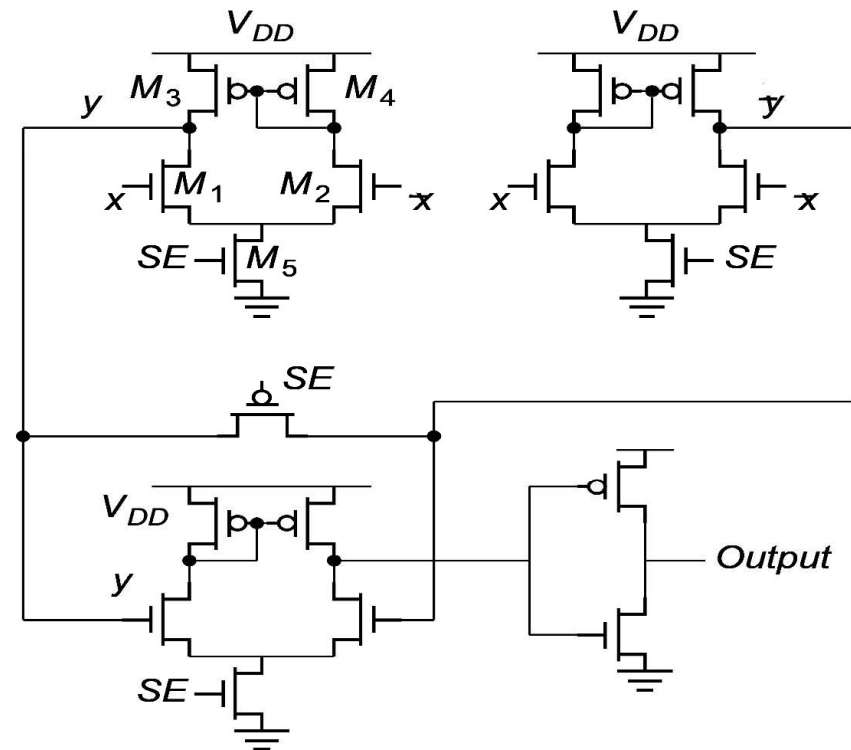
差分-在SRAM中直接使用

1.12 存储器外围电路

□ 差分灵敏放大 – SRAM



(a) SRAM sensing scheme



(b) two stage differential amplifier

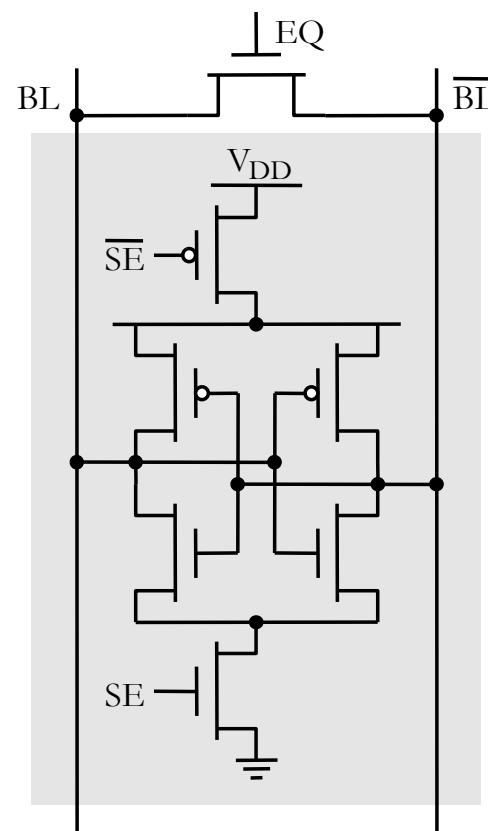
电压差 0.25mV

1.13 存储器外围电路

以小博大——正反馈

□ 基于Latch的灵敏放大器 (DRAM)

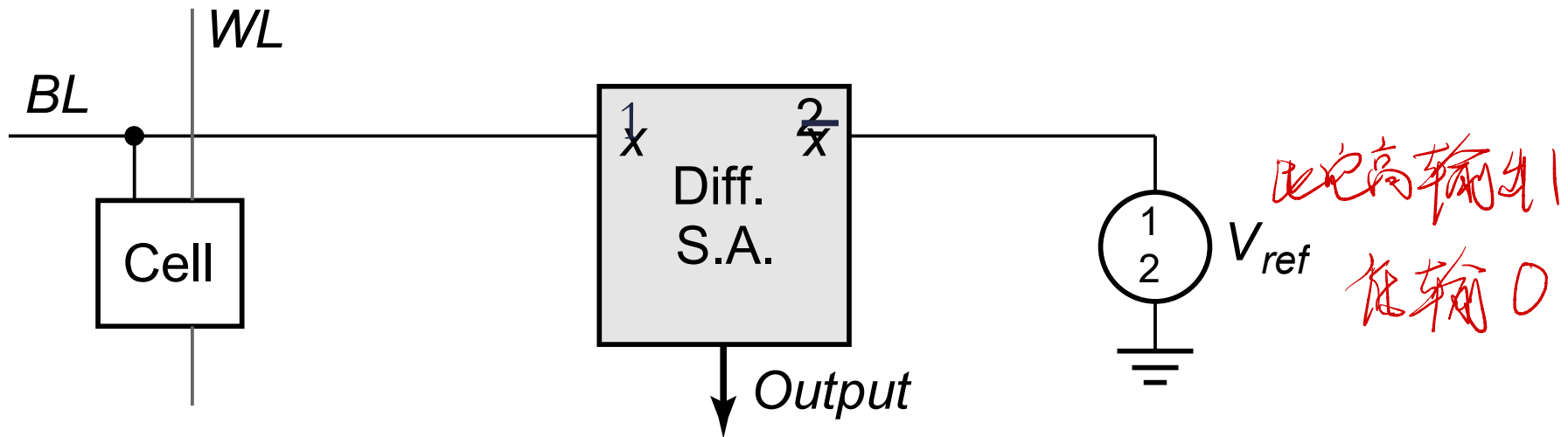
- 利用EQ将两个位线预充到Latch的亚稳态。
- 一旦位线上有足够的电压差，通过给SE信号使灵敏放大器开始工作，正反馈会快速地将输出推到稳定状态。



1.14 存储器外围电路

□ 单端到差分转换电路

单端输出的DRAM如何使用差分灵敏放大器？



如何得到好的 V_{ref} ?

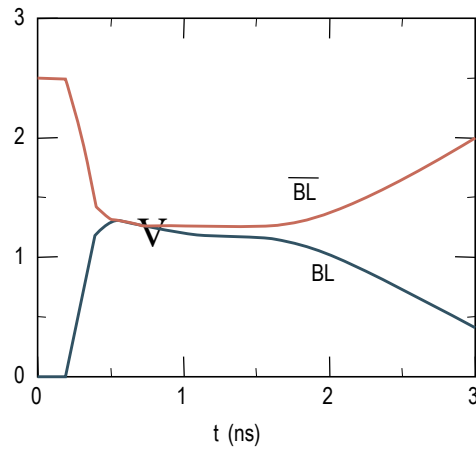
□ 具有冗余单元的开放式位线结构



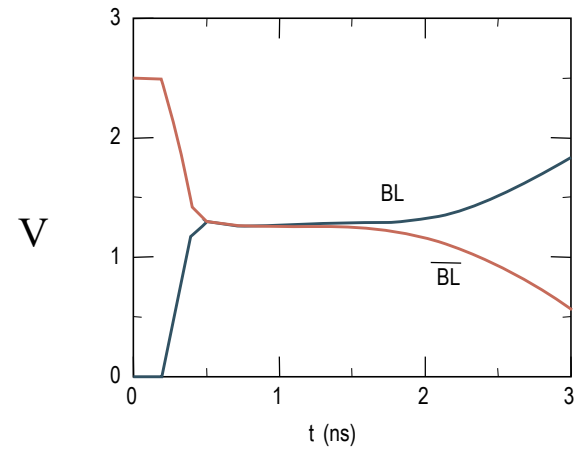
分割存储阵列两个部分，差分放大器在中间；
每一边增加一列冗余单元（dummy cell）；

1.16 存储器外围电路

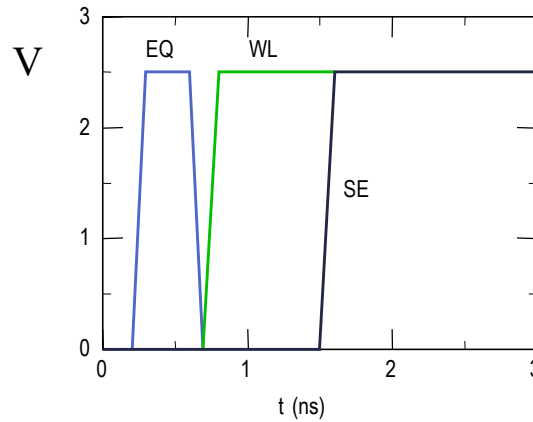
□ 具有冗余单元结构的DRAM 读出过程



读0



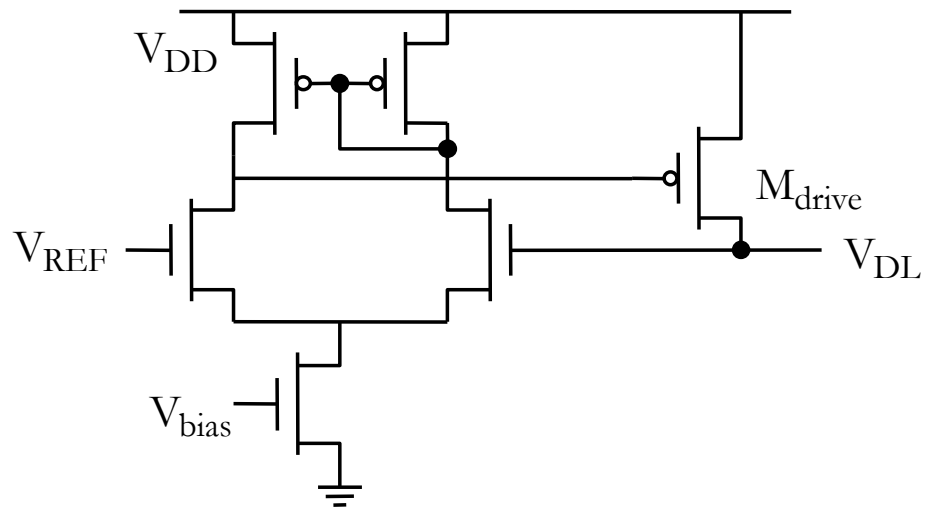
读 1



控制信号

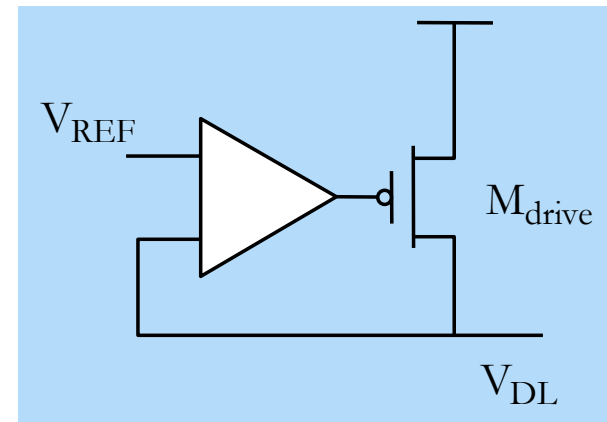
1.17 存储器外围电路

□ 电压转换器



LD0模型

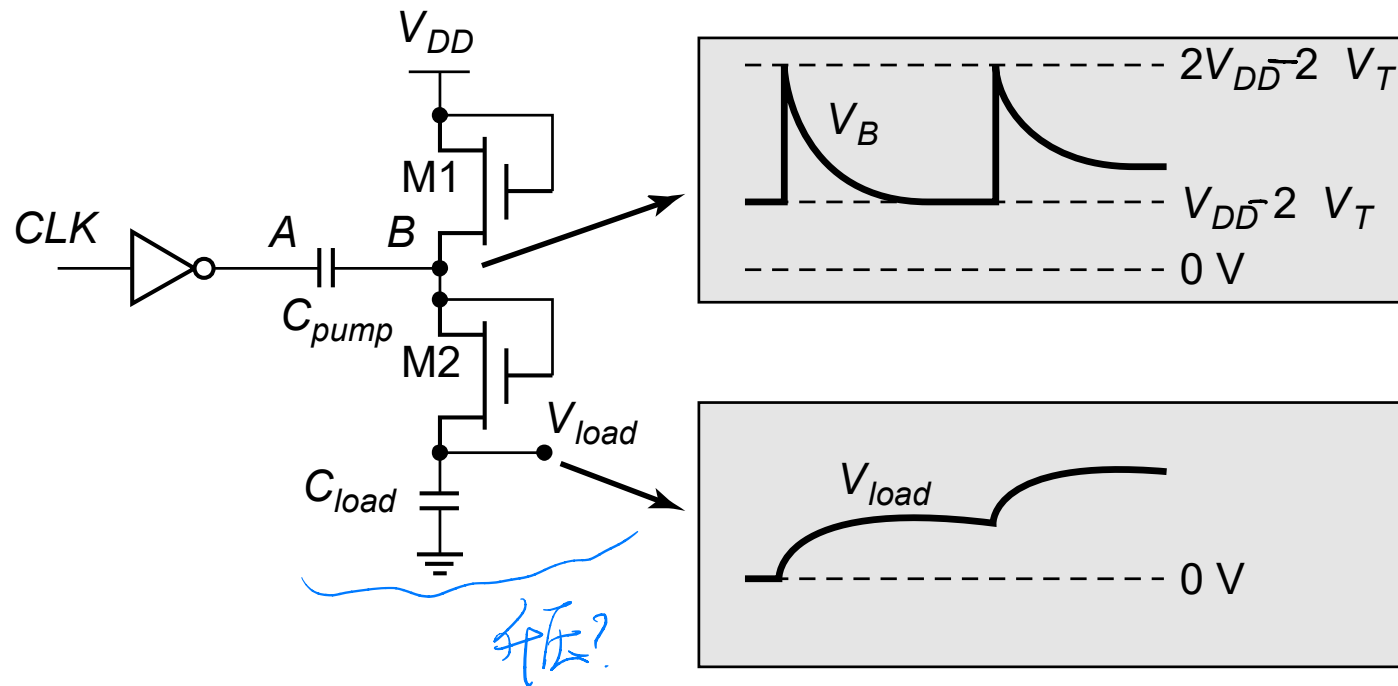
等效模型



1.18 存储器外围电路

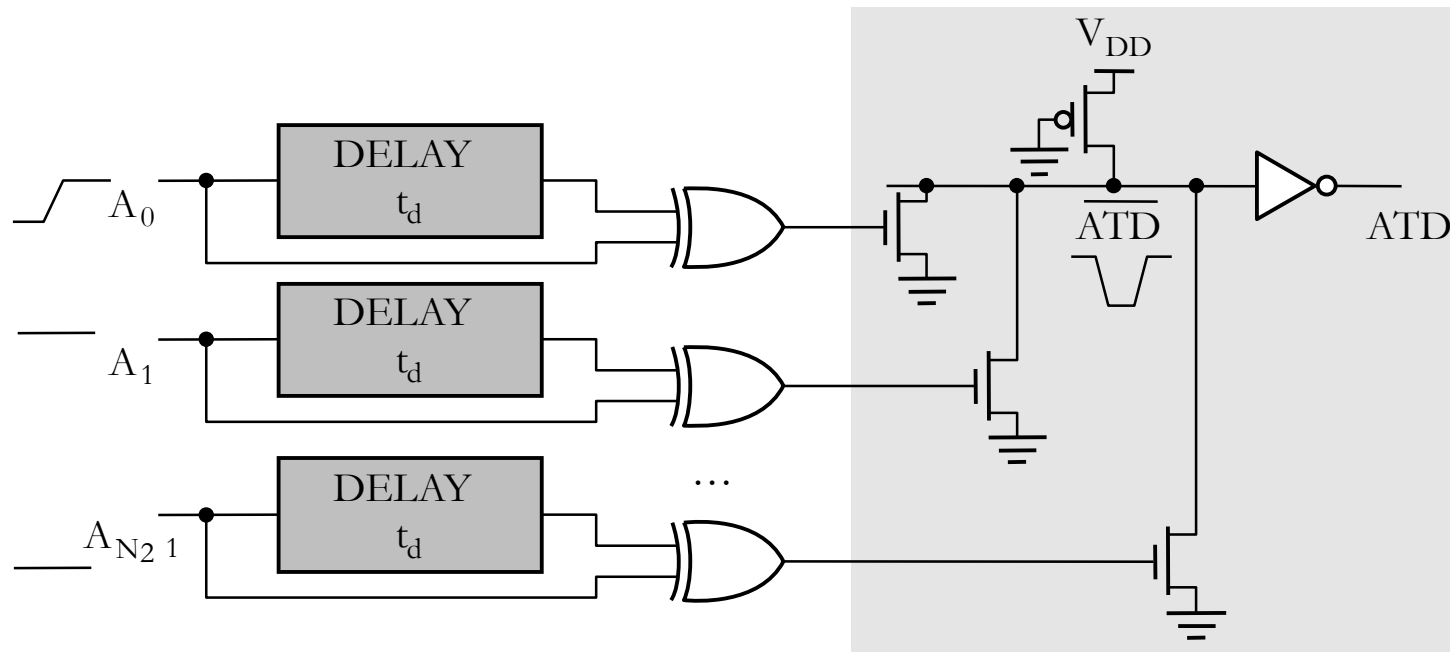
$$V_B = V_{DD} - V_T$$

□ 电荷泵 (Charge Pump)



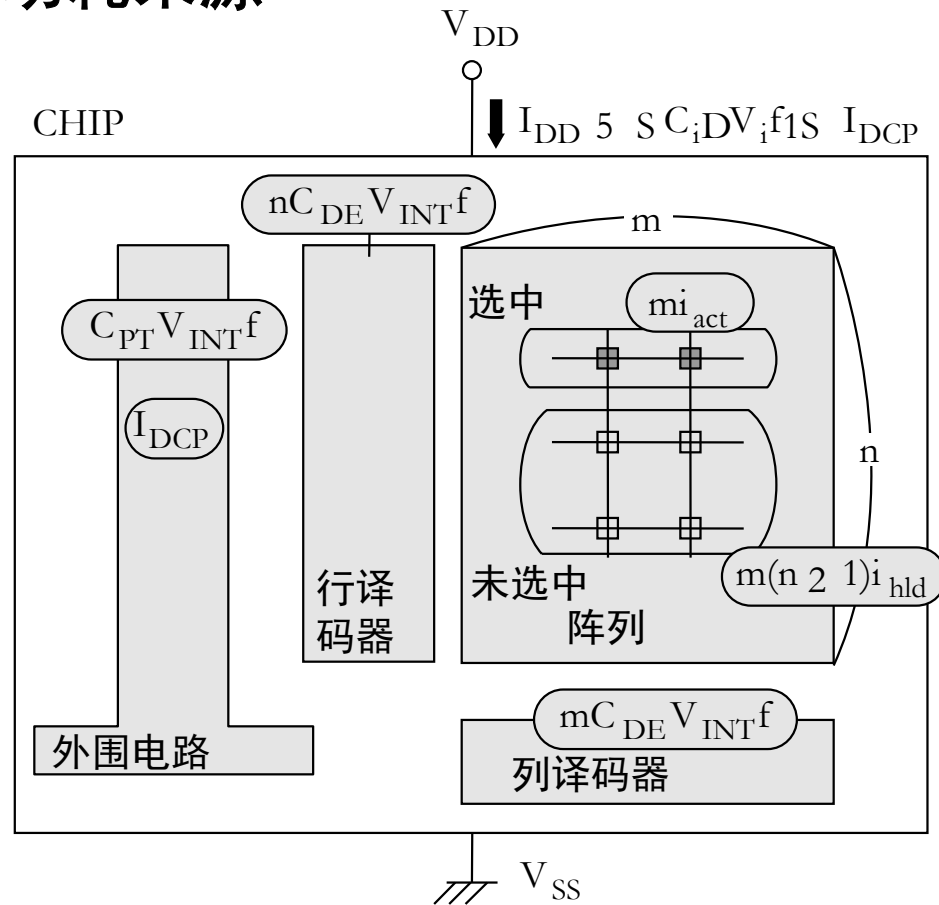
1.19 存储器外围电路

□ 地址变化检测(ATD)



1.20 存储器外围电路

□ 存储器中的功耗来源

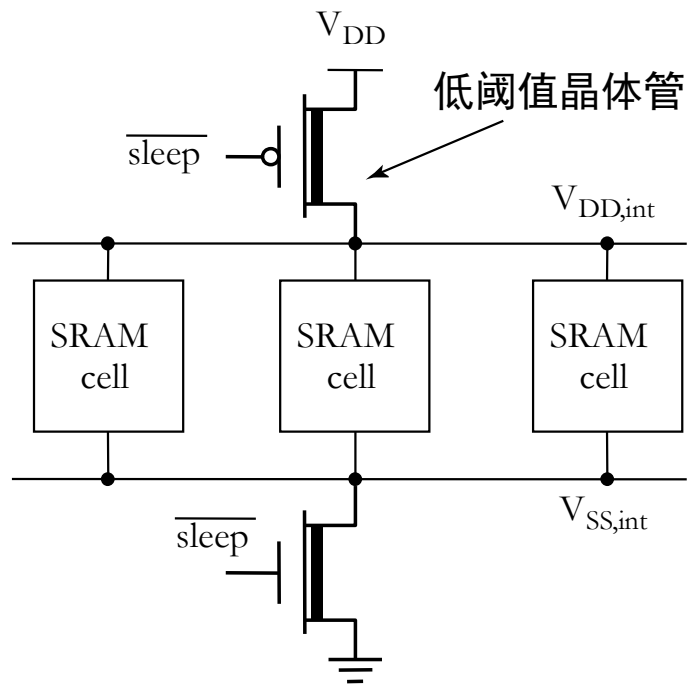


阵列功耗降低

From [Itoh00]

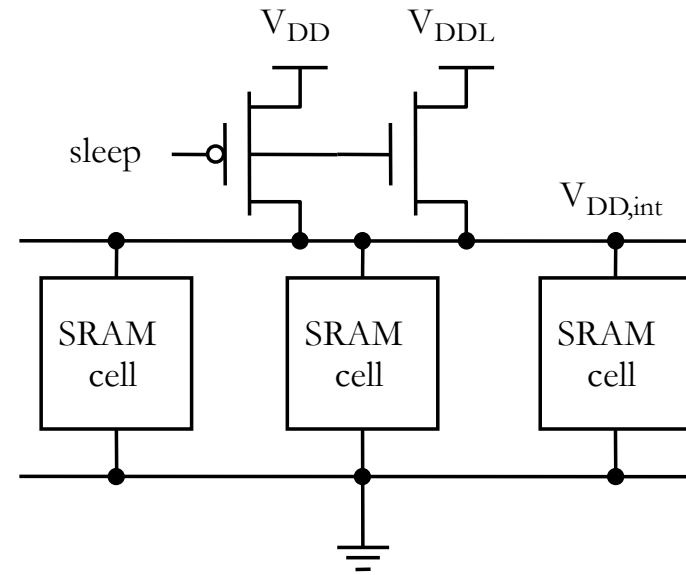
1.21 存储器外围电路

□ 减小 SRAM 的漏电



插入额外的电阻

降低漏电流



降低供电电压



第十一讲 存储器和阵列结构 (二)

本讲主要内容

- (一) 存储器外围电路
- (二) 存储器设计实例



2.1 存储器设计实例

- 可编程逻辑阵列(Programmable Logic Array)
- SRAM
- Flash Memory

2.2 存储器设计实例

□ PLA 与 ROM

□ Programmable Logic Array

实现随机逻辑的标准化方法
“两级逻辑实现”

□ 主要区别 等价于ROM

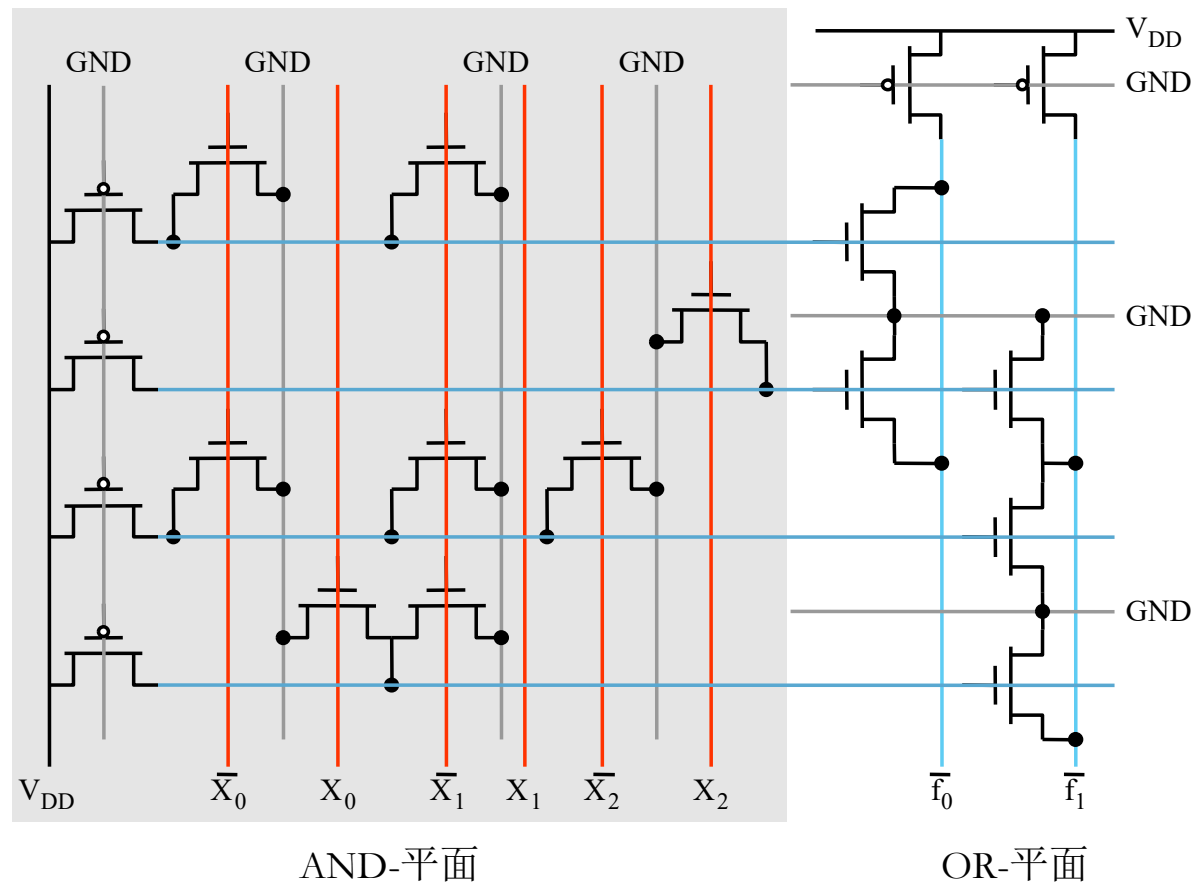
ROM: 全译码

PLA: one element per minterm

2.3 存储器设计实例

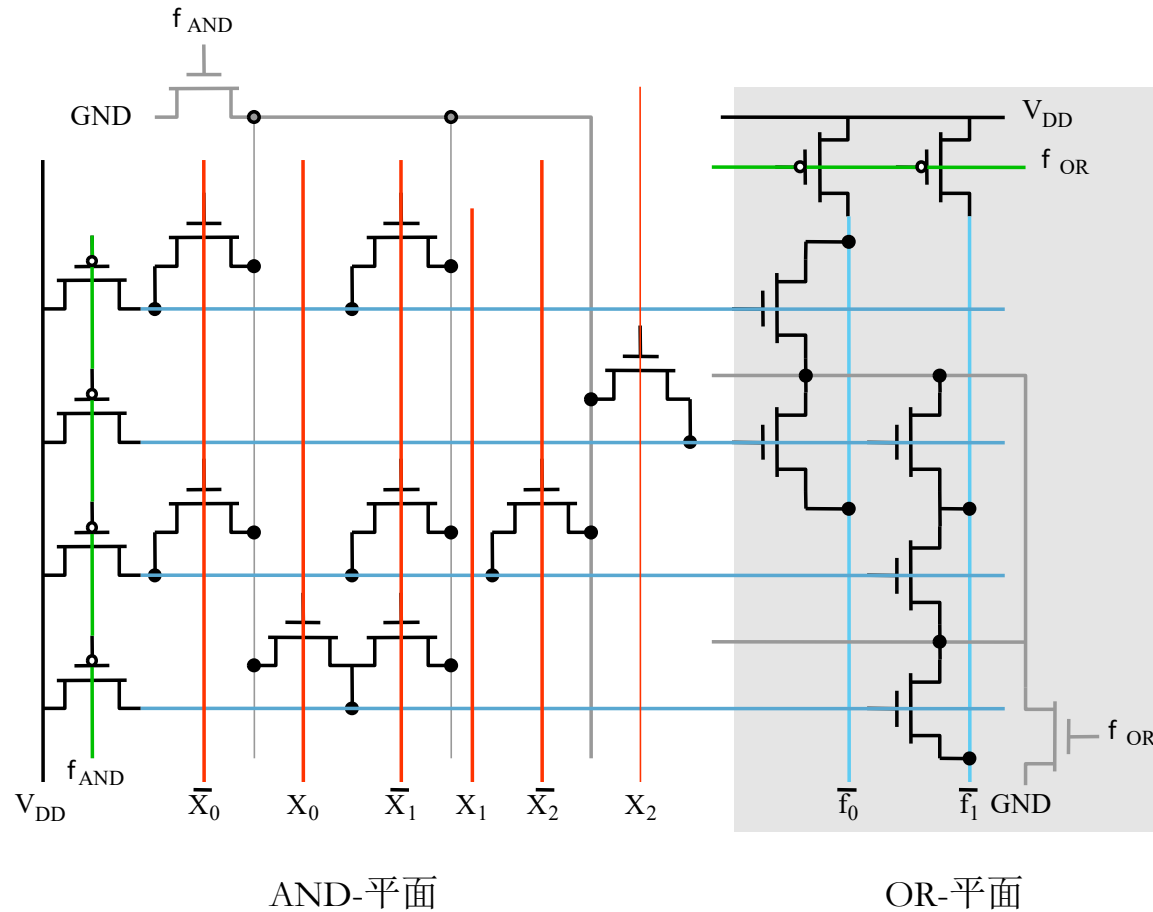
□ 可编程逻辑阵列PLA(Programmable Logic Array)

Pseudo-NMOS PLA



2.4 存储器设计实例

□ 动态 PLA



容量： 4MX1b
1MX4b

X地址（行）：10位
Y地址（列）：7位或5位
Z地址（块）：5位

块数：32块
灵敏放大器：16个

块阵列：1024行X128列
列地址：3位列选
4位数据总线选



2.6 存储器设计实例

□ SRAM的等效框图和电路图

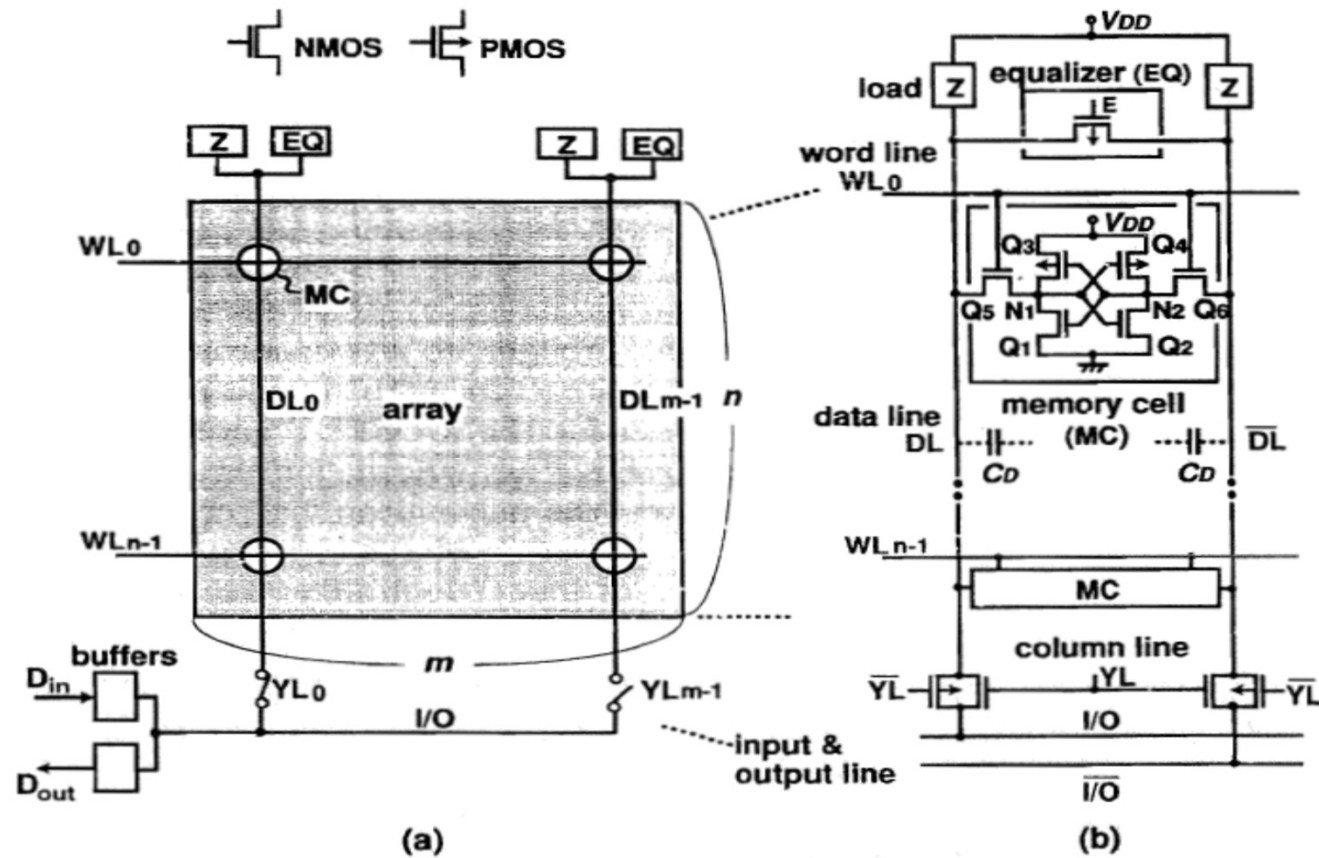


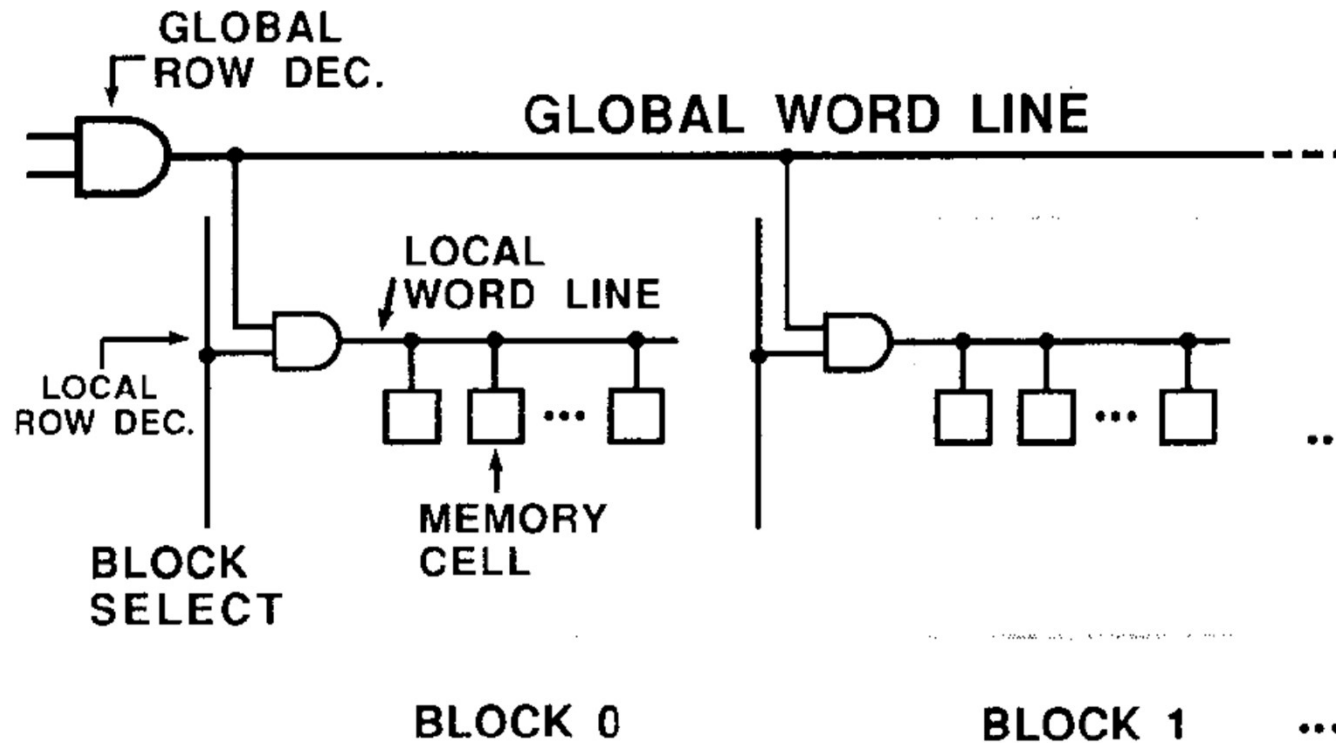
Fig. 1.23. A conceptual SRAM array (a) and an actual data-line configuration (b)

□ 存储阵列的结构



2.9 存储器设计实例

□ 字线电路结构-DWL(Devided-Word-Line)

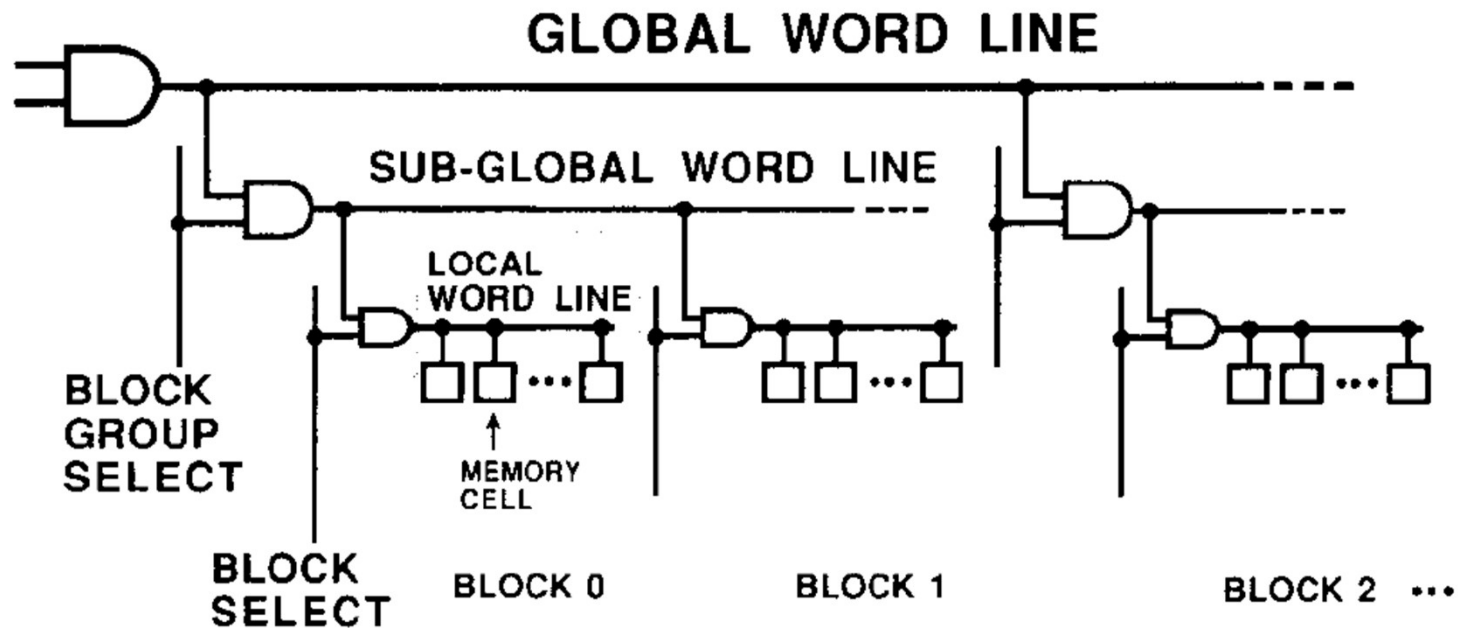


(a)

[Hirose90]

2.10 存储器设计实例

□ 字线电路结构-HWD(Hierarchical-Word-Decoding)

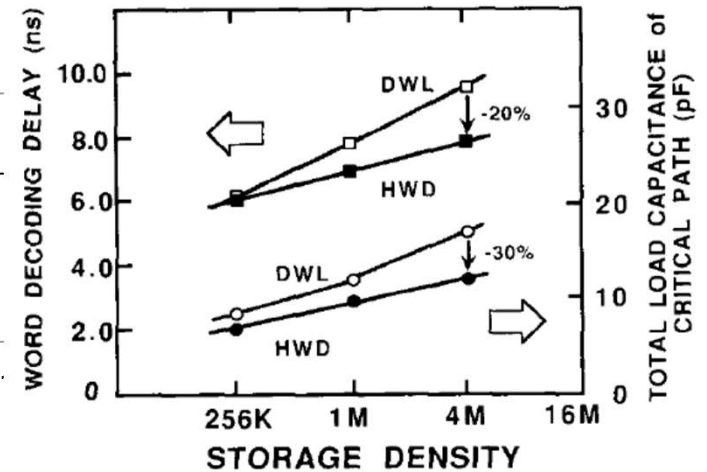
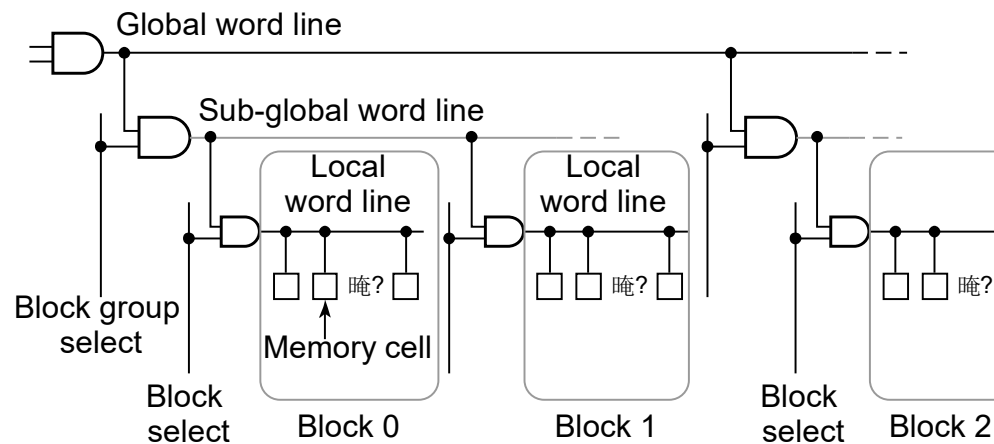


(b)

[Hirose90]

2.11 存储器设计实例

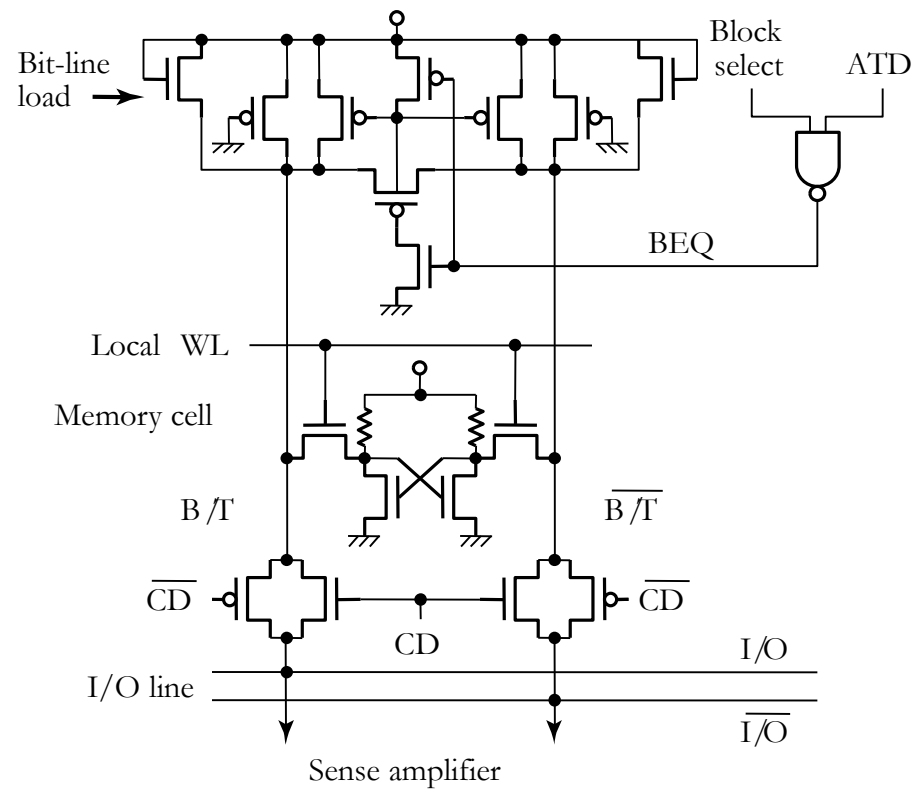
□ DWL与HWD的性能和功耗对比



[Hirose90]

2.12 存储器设计实例

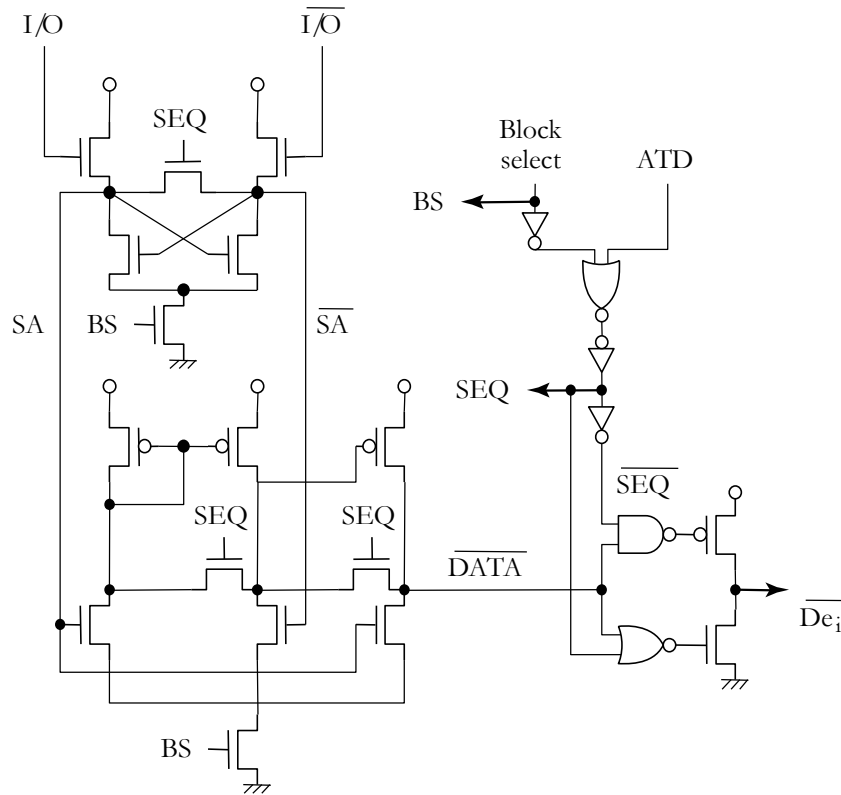
□ 位线（Bit-line）电路结构



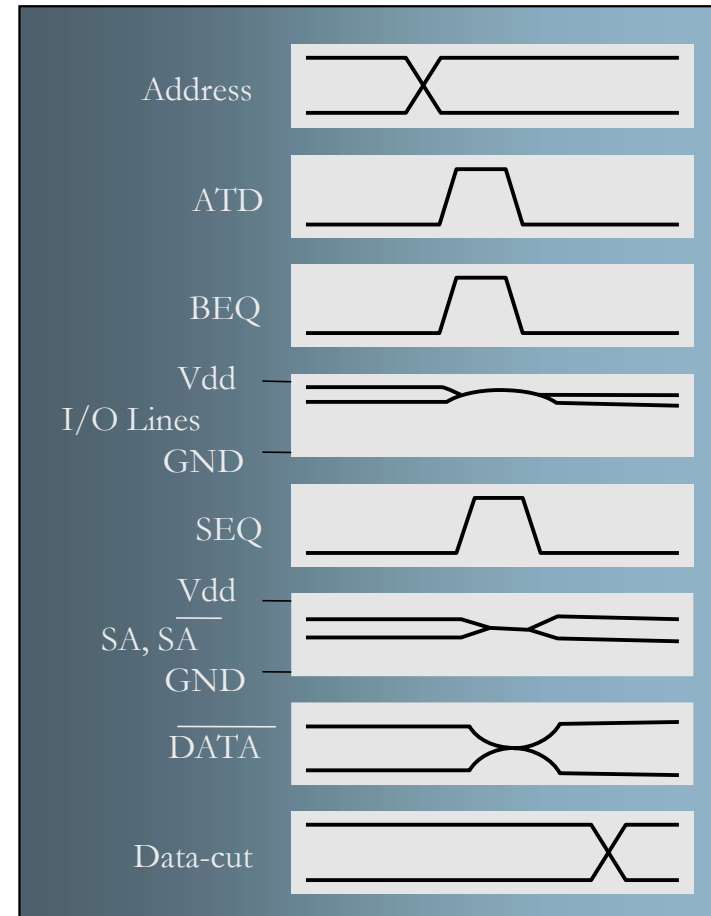
[Hirose90]

2.13 存储器设计实例

□ 灵敏放大器与波形



[Hirose90]

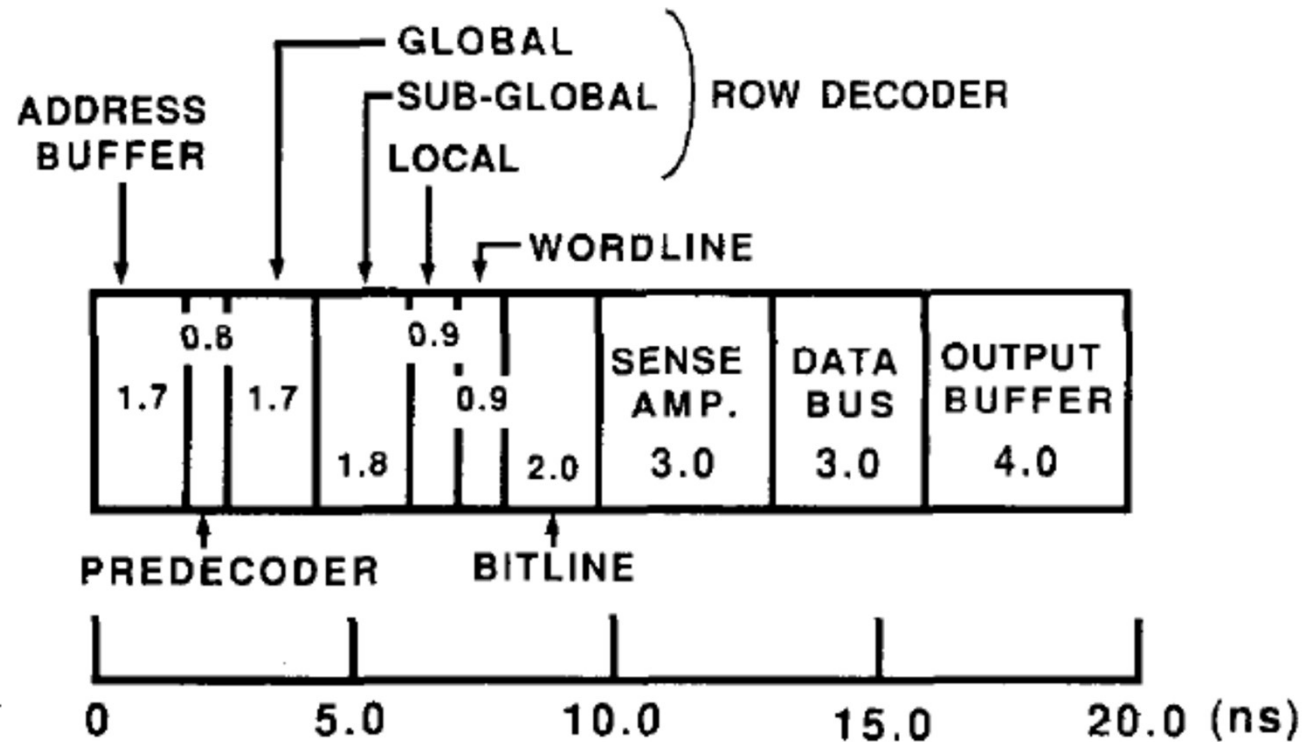


[illegible]

[Hirose90]

2.15 存储器设计实例

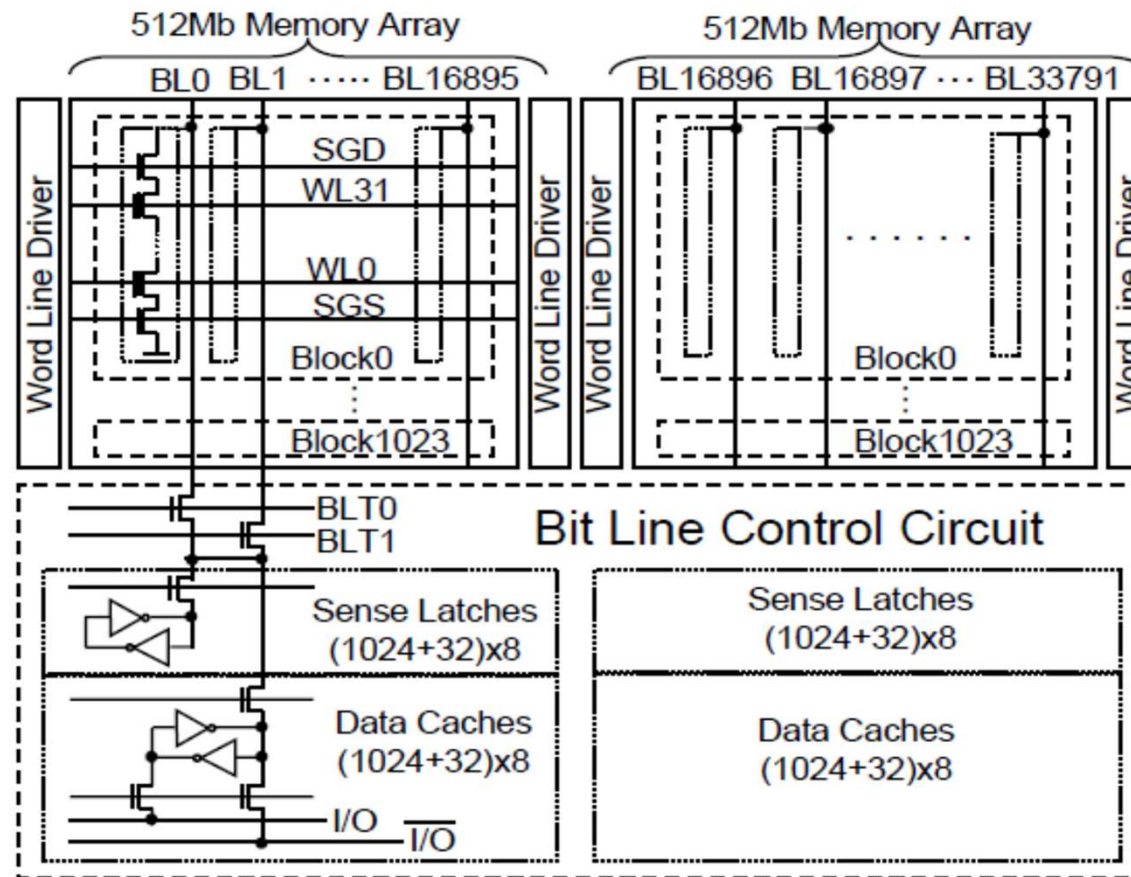
□ 存储时间的组成



[Hirose90]

2.16 存储器设计实例

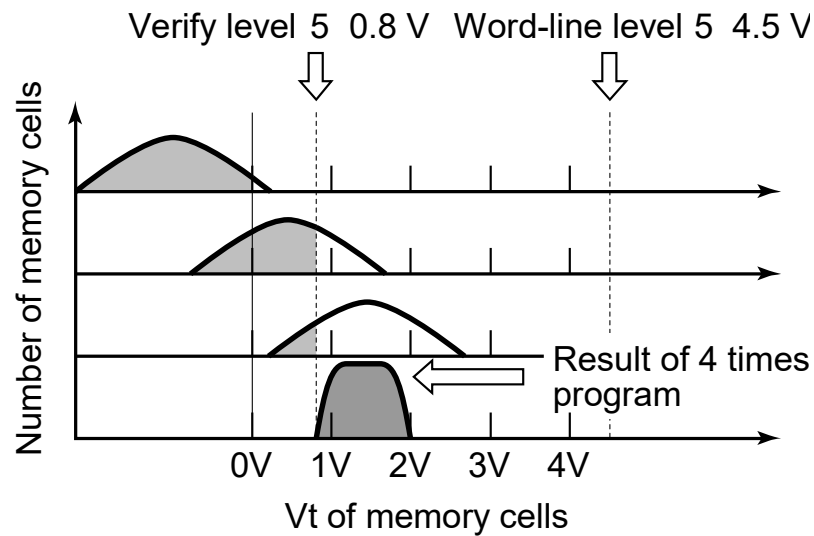
□ 1 Gbit NAND Flash 存储器



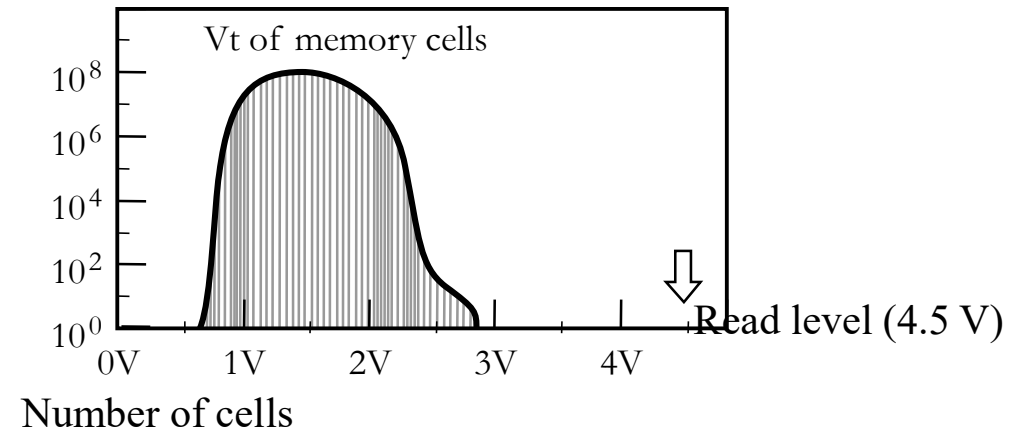
From [Nakamura02]

2.17 存储器设计实例

□ Flash 存储器写入



阈值的变化过程

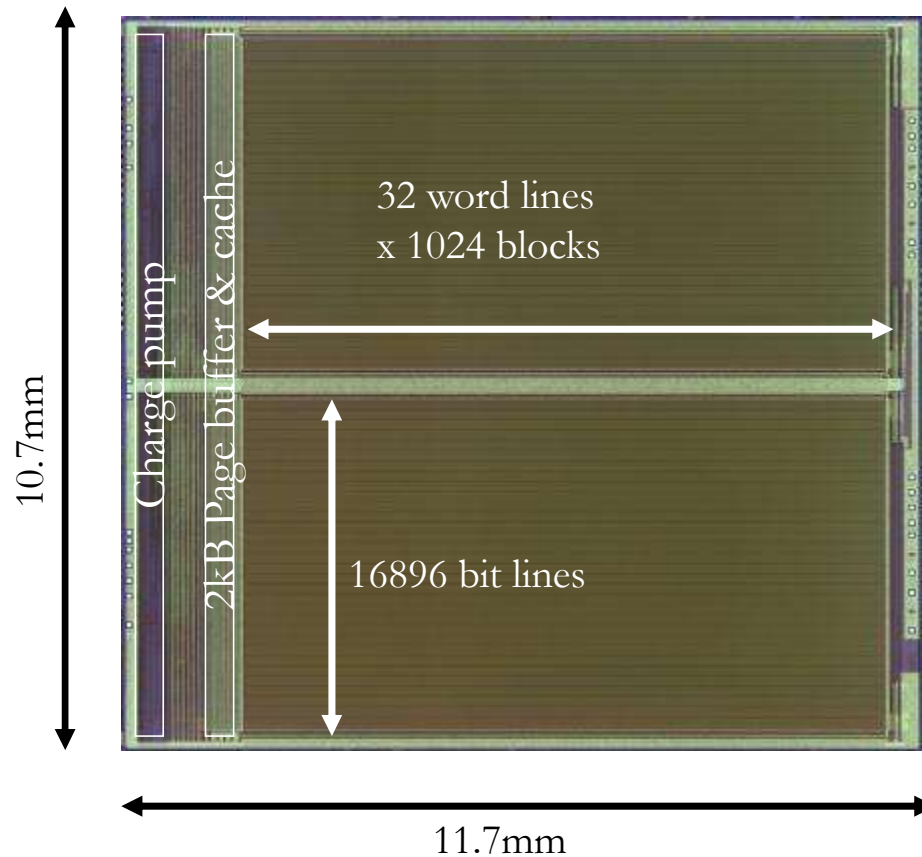


阈值最后的分布

From [Nakamura02]

2.18 存储器设计实例

□ 125mm² 1G位 NAND Flash 存储器



- Technology 0.13μm p-sub
CMOS triple-well 1poly,
1polycide, 1W, 2Al
- Cell size 0.077μm²
- Chip size 125.2mm²
- Organization 2112 x 8b x 64
page x 1k block
- Power supply 2.7V-3.6V
- Cycle time 50ns
- Read time 25μs
- Program time 200μs / page
- Erase time 2ms / block

From [Nakamura02]

小结

- 存储器被组织成单元阵列。每一个单元通过块、列和行地址来寻址。
- 存储单元应当设计成能以最小面积获得最大的信号。
- 当单元的弱信号特性给定时，外围电路对存储器操作的可靠性以及合理的性能非常重要。
- 译码器、灵敏放大器和I/O缓冲器是每一个存储器设计整体必不可少的部分。
- 一个存储器必须能在各种工作和制造条件下正确工作。识别错误操作的可能来源并提供一个合适的模型是解决存储器可靠性首要解决的问题。

谢谢！

欢迎指正！