



第七章 存储系统(一)

秦磊华 计算机学院

- 7.1 存储器概述
- 7.2 半导体存储器
- 7.3 主存的数据组织
- 7.4 主存与CPU的连接

CONTENT



1. 存储器分类

1) 按存储介质分

(1) 半导体存储器

◆ 双极型存储器 MOS存储器 : 速度快, 断电信息丢失;

(2) 纸介质存储器

◆ 纸带: 信息外露, 保存时间长;

(3) 磁介质存储器

◆ 磁芯、磁带、磁盘: 容量大, 速度慢、体积大、保存时间长

(4) 光介质存储器

◆ CD-ROM CD-RW CD-R / DVD-ROM DVD-RW DVD-R: 便于携带, 廉价, 易于保存、保存时间长



7.1 存储器概述

1. 存储器分类

2) 按存储方式分

(1) 随机存储器

◆ 存取时间与物理位置无关：磁芯、半导体存储器

(2) 顺序存储器

◆ 存取时间与物理位置无关：磁芯、磁带、光盘存储器等



1. 存储器分类

3) 按读/写功能分

(1) 只读存储器 (ROM)

◆ 内容采用特殊方法预先写入：不能采用常规写指令修改；

(2) 读/写存储器

◆ 可读可写的存储器：读写指令均可访问。



1. 存储器分类

4)按信息的可保存性分

(1)易失性存储器 (*Volatile Memories*)

◆断电后信息消失：SRAM、DRAM；

(2)非易失性存储器(*Non-Volatile Memories*)

◆断电后信息不消失：ROM、NVRAM、磁介质存储、光介质存储等。

1. 存储器分类

5)按在机器中的作用分

- (1)控制存储器：CPU内部，用于存放控制信号(微程序控制器)
- (2)寄存器：CPU内部的暂存单元，速度快、容量有限
- (3)高速缓冲存储器：处于CPU和主存之间，缓解二者间速度差异
- (4)主存储器：存放指令和数据，CPU按地址直接随机存取的大容量存储器
- (5)辅助存储器：磁盘、磁带等大容量存储器，CPU不能直接访问，速度慢

2. 存储系统主要技术指标

◆存储时间

接收到读写命令到从存储器中读出或开始写入信息所经历的时间

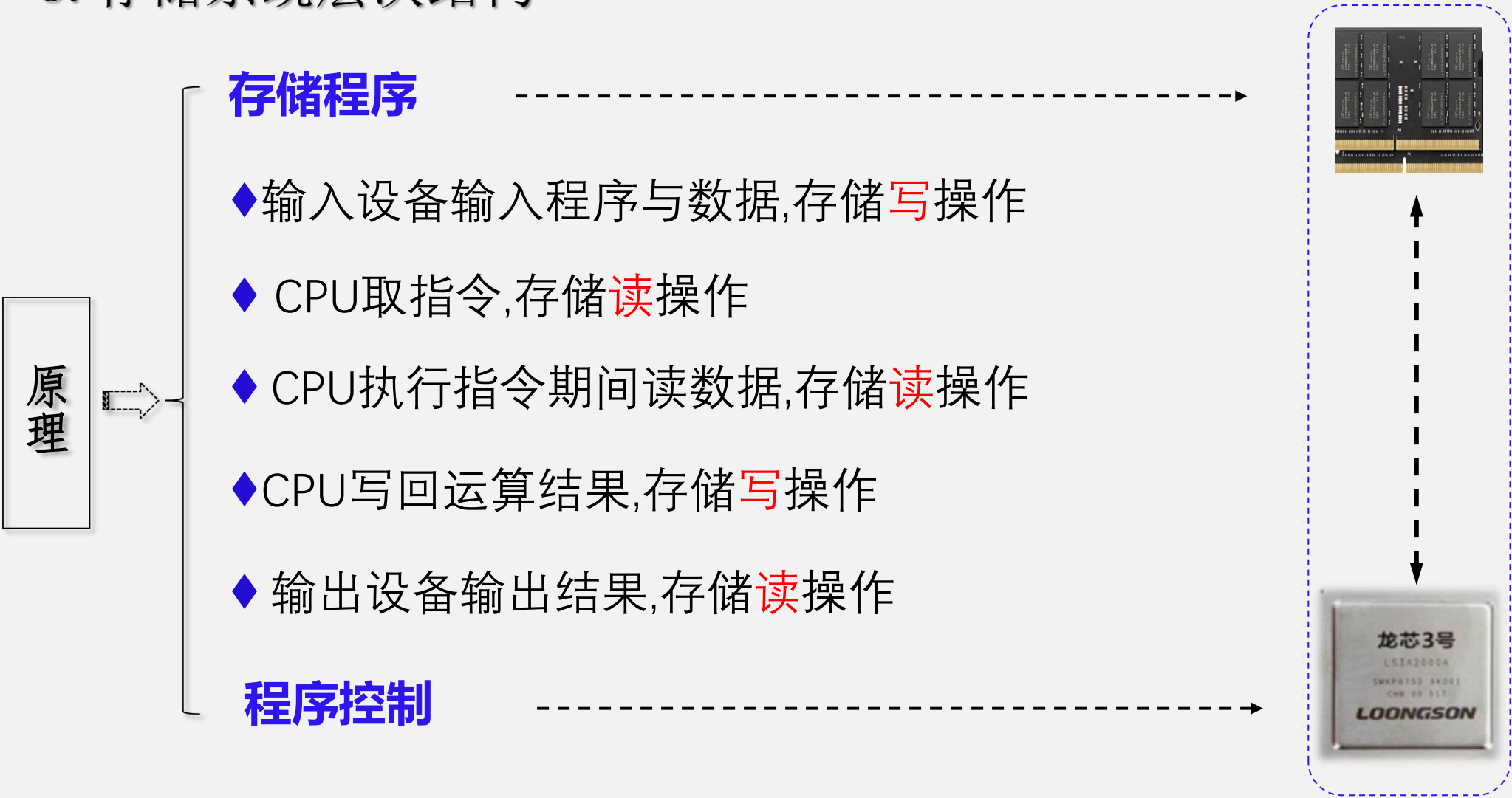
◆存储周期

连续两次访问存储器所需要的最小时间间隔

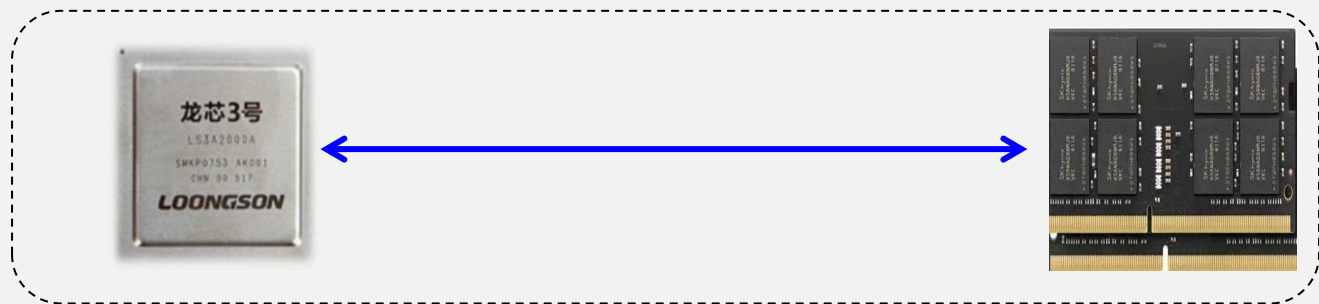
◆存储器带宽

单位时间内存储器存取的信息量, Byte/s

3. 存储系统层次结构

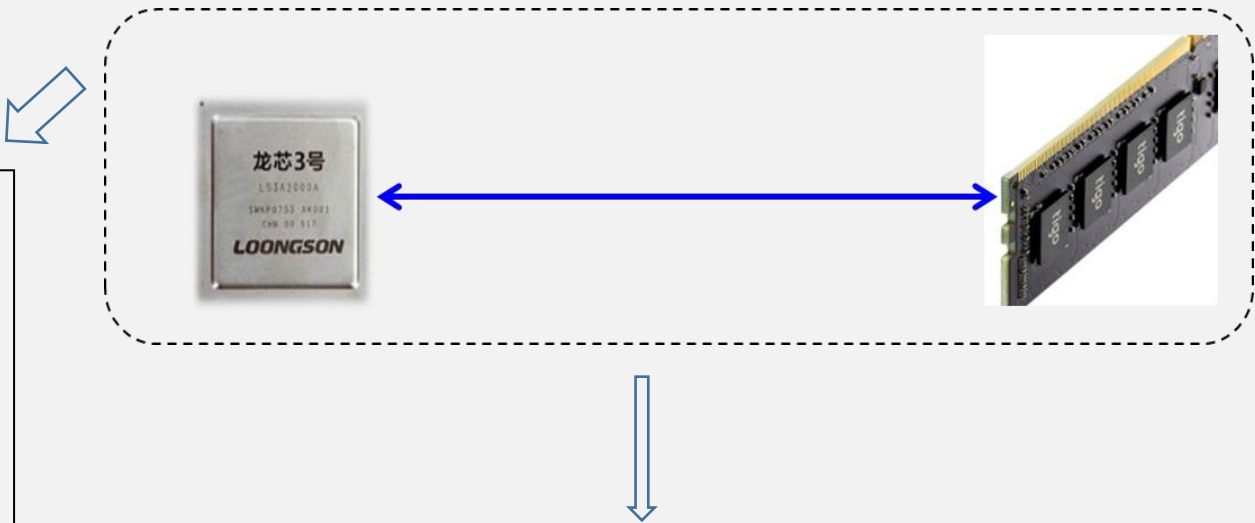
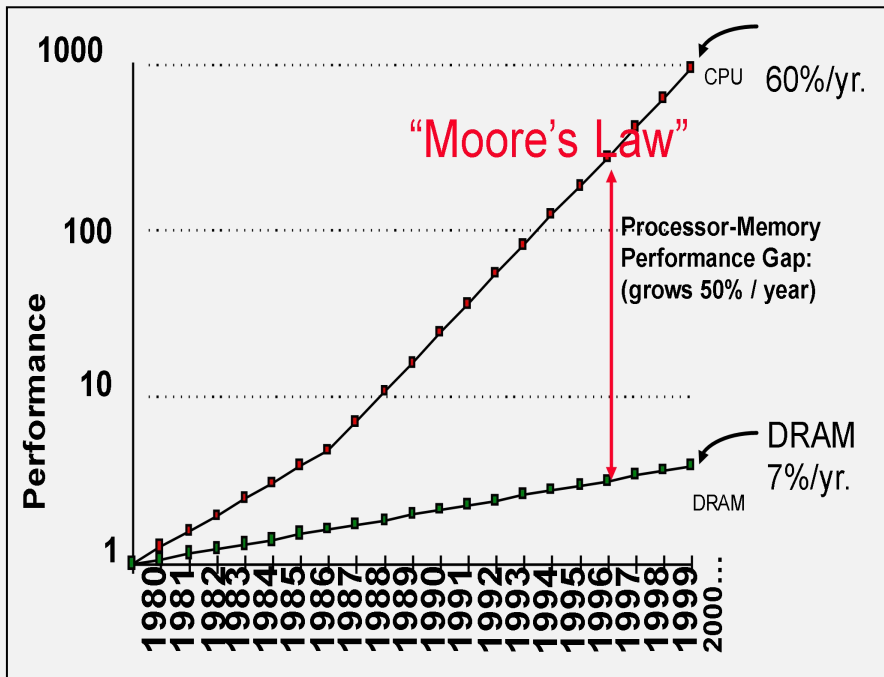


3. 存储系统层次结构



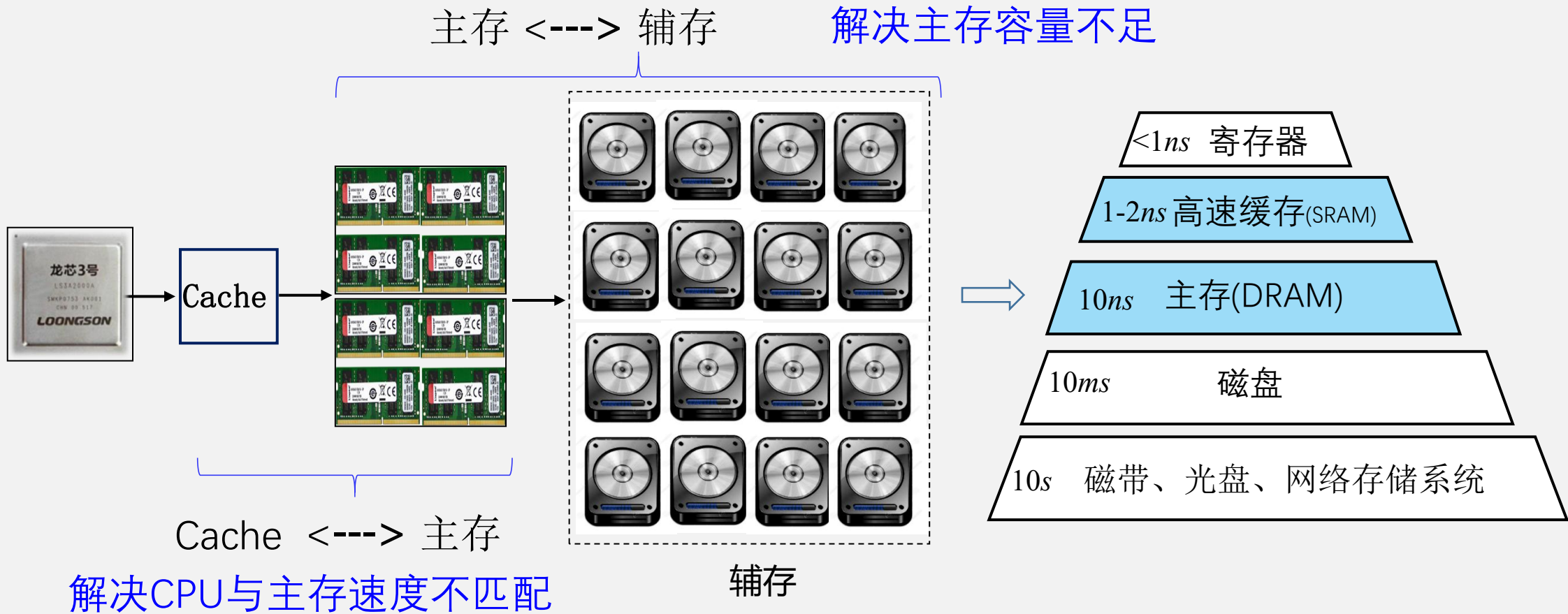
- 1) 主存速度够快吗?
- 2) 主存容量够大吗?
- 3) 存储器便宜吗?
- 4) 能速度既快、容量又大, 价格还便宜吗?

3. 存储系统层次结构



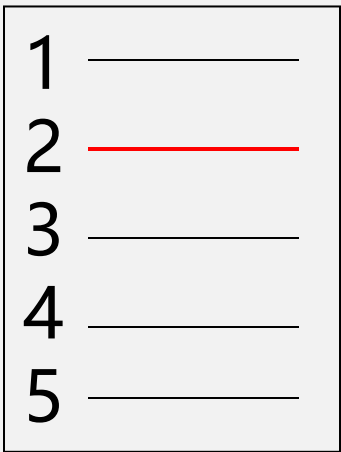
- ◆主存增速与CPU增速不同步
- ◆指令执行期间多次访问存储器

3. 存储系统层次结构



CPU访问到的存储系统具有Cache的速度，辅存的容量和摊平后的价格

3. 存储系统层次结构



局部性原理

◆时间局部性:

现在被访问的信息2在不久的将来还将再次被访问;

时间局部性的程序结构体现: 循环结构

◆空间局部性:

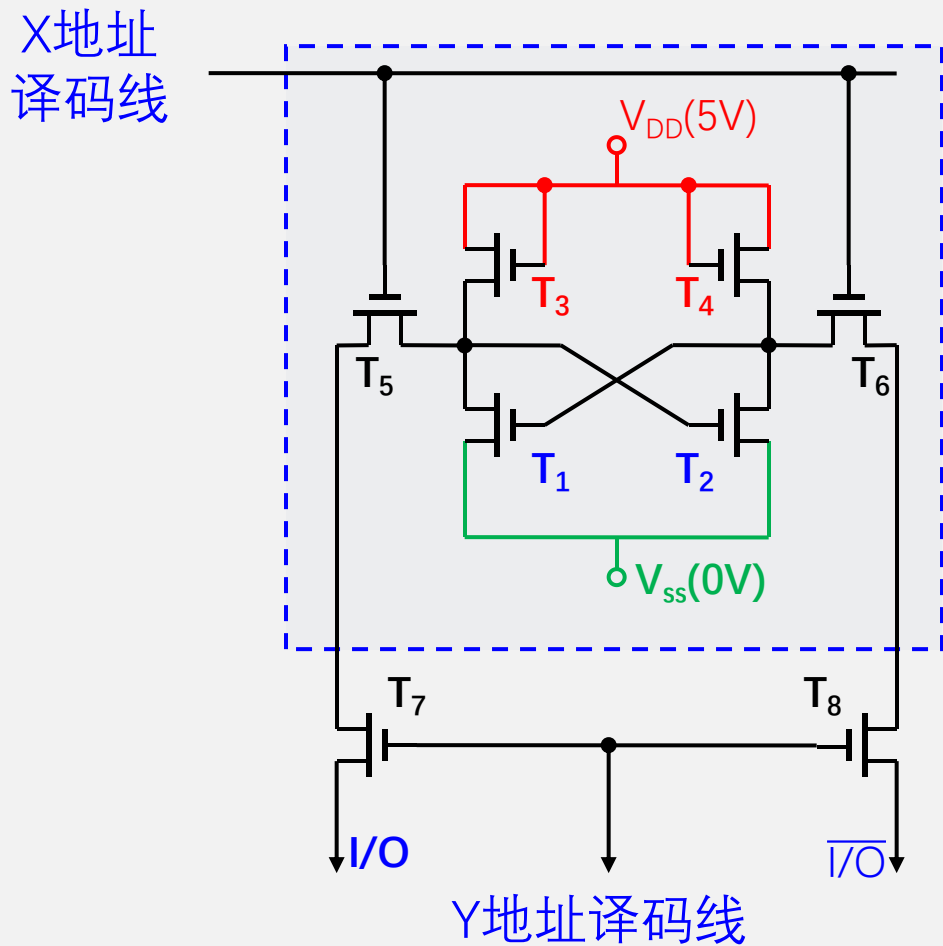
现访问信息2, 下一次访问2附近的其它信息。

空间局部性的程序结构体现: 顺序结构



7.2 半导体存储器

1. 六管SRAM存储单元 (SRAM Cell—存储 1 bit)

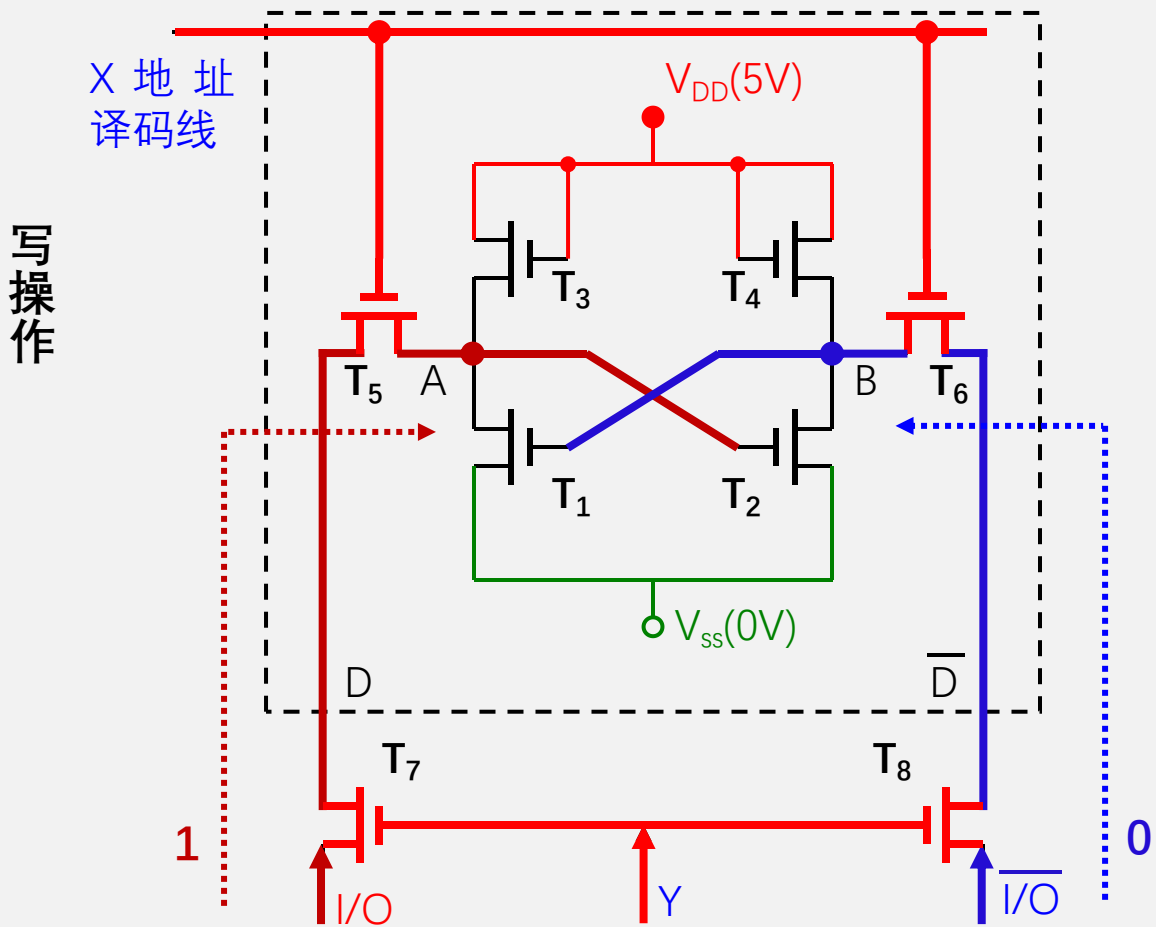


工作管：T₁、T₂

负载管：T₃、T₄

门控管：T₅、T₆、T₇、T₈

1. 六管SRAM存储单元 (SRAM Cell—存储 1 bit)



◆ X地址选通

T₅、T₆管导通

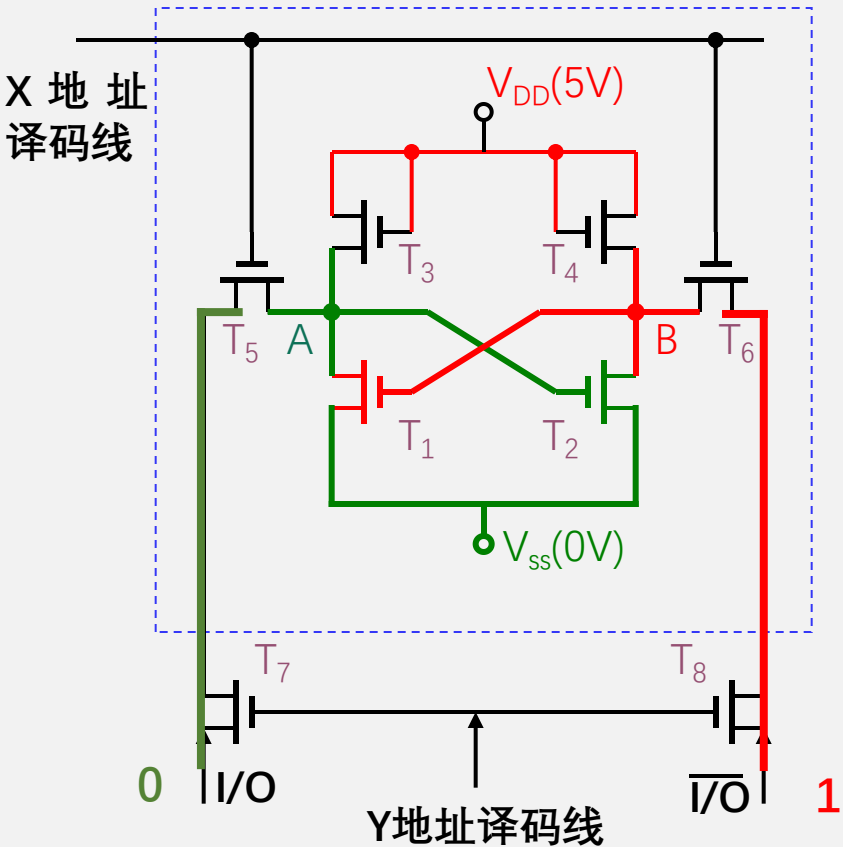
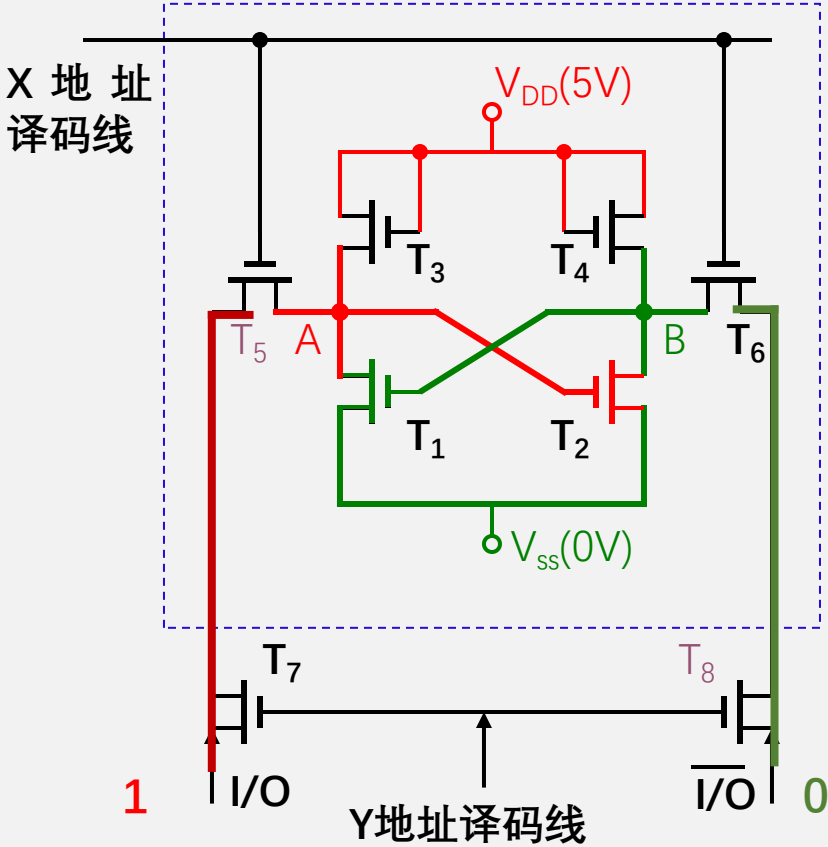
A点与位线相连

◆ Y地址选通

T₇、T₈管导通

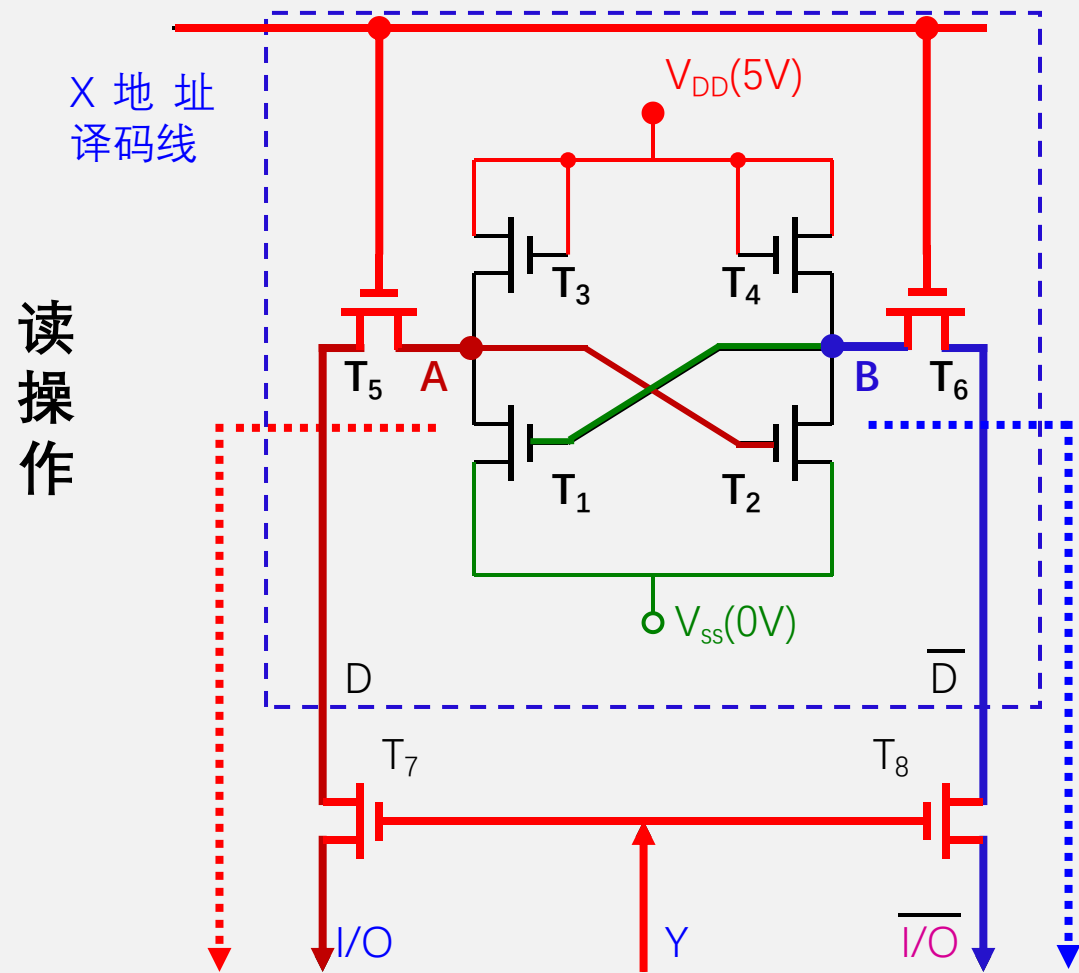
A点电位输出到I/O端

1. 六管SRAM存储单元 (SRAM Cell—存储 1 bit)



T_1 、 T_2 构成的稳定互锁状态来保存1 bit 信息

1. 六管SRAM存储单元 (SRAM Cell—存储 1 bit)



◆X地址选通

T5、T6管导通

A点与位线相连

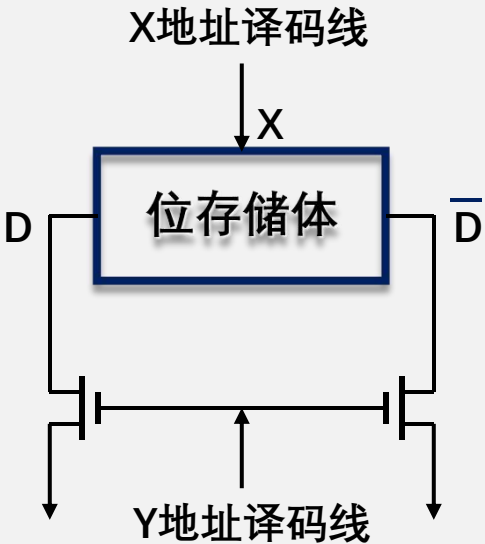
◆Y地址选通

T7、T8管导通

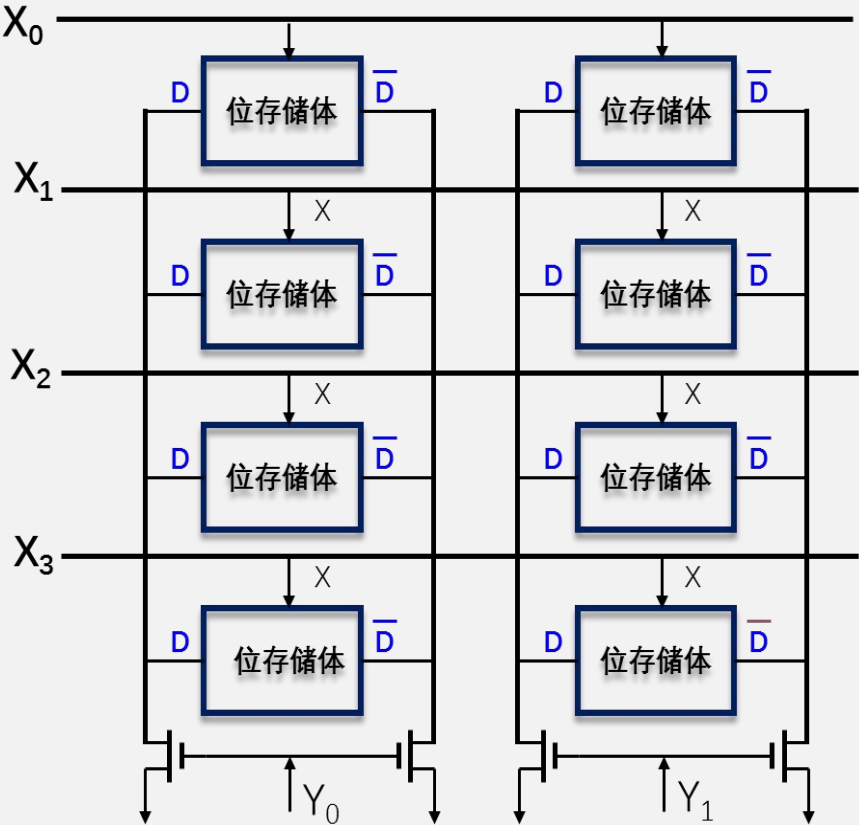
A点电位输出到I/O端

通过外接于I/O和 $\overline{I/O}$ 间电流放大器中
电流方向判断读出的是1还是0

2. SRAM存储单元封装与扩展



- ◆ X: 行选择线
- ◆ Y: 列选择线
- ◆ D: 数据输出口



8×1位的存储单元(M ×N位)

- ◆ X地址选通
- ◆ Y地址选通

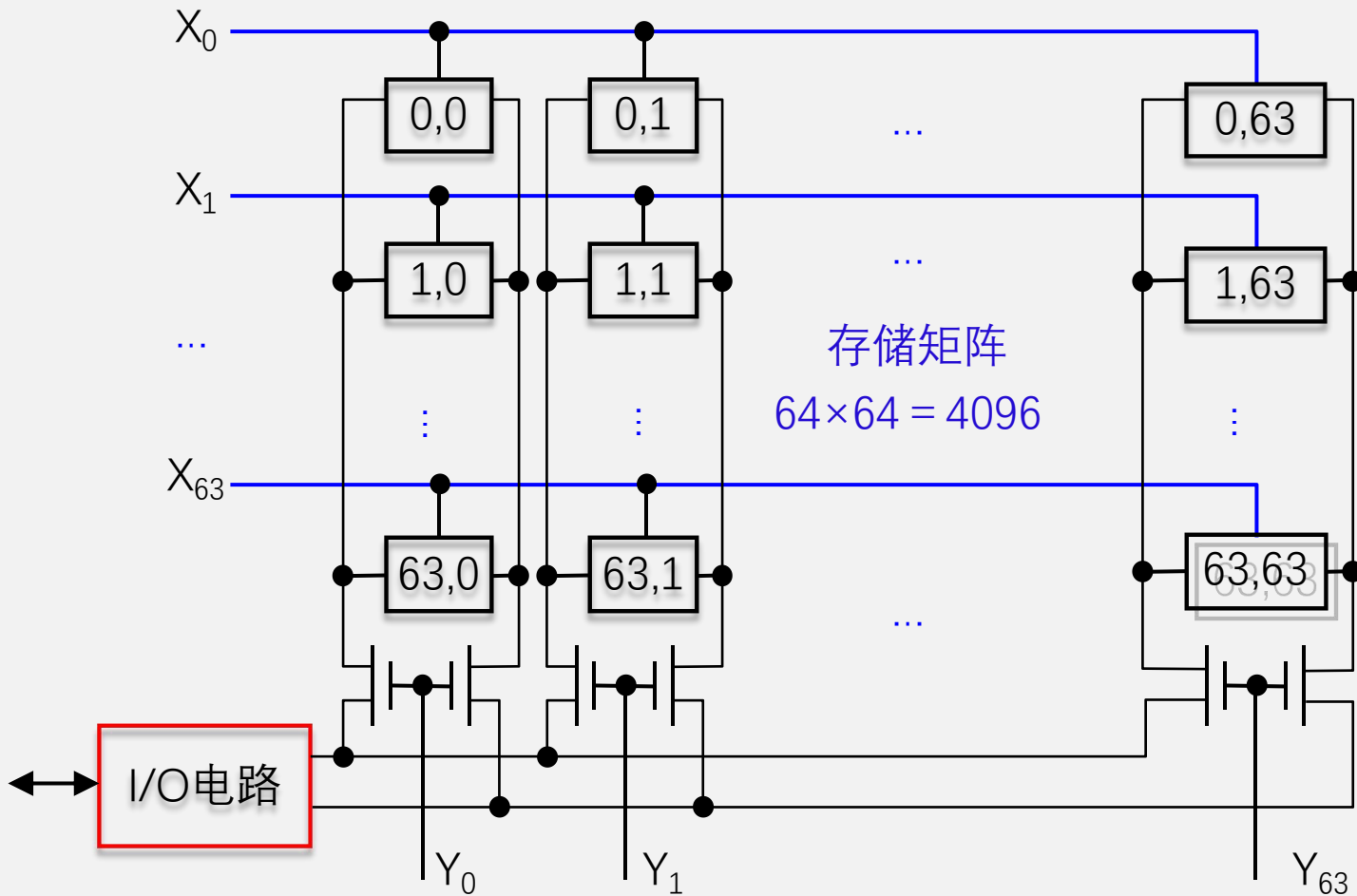
存储器地址线

8单元需3根地址线

行译码输入2根
列译码输入1根

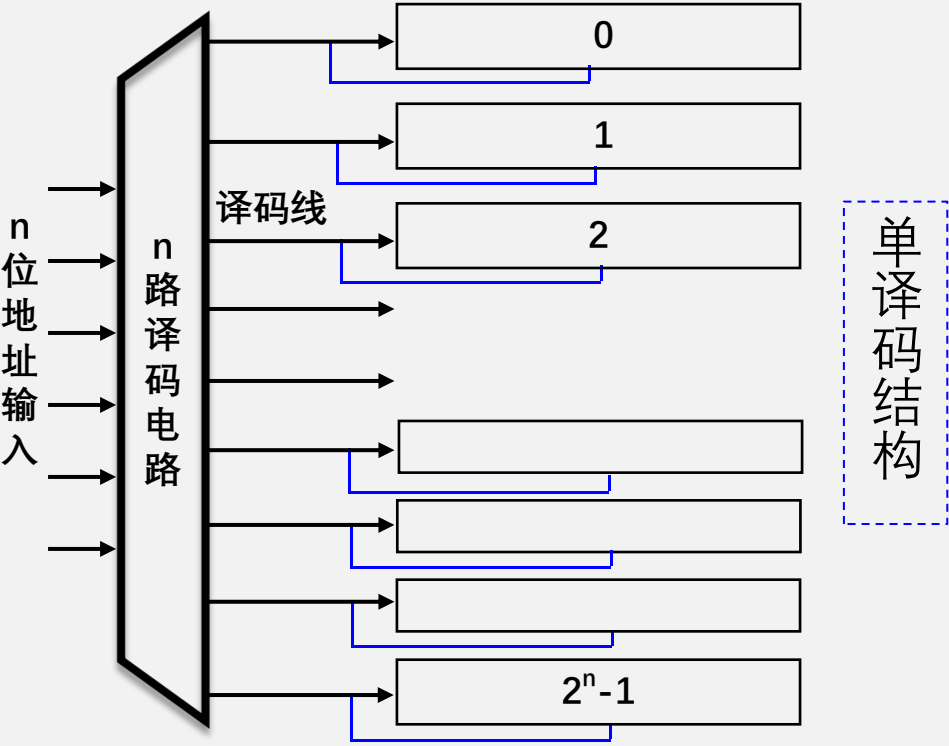


2. SRAM存储单元封装与扩展

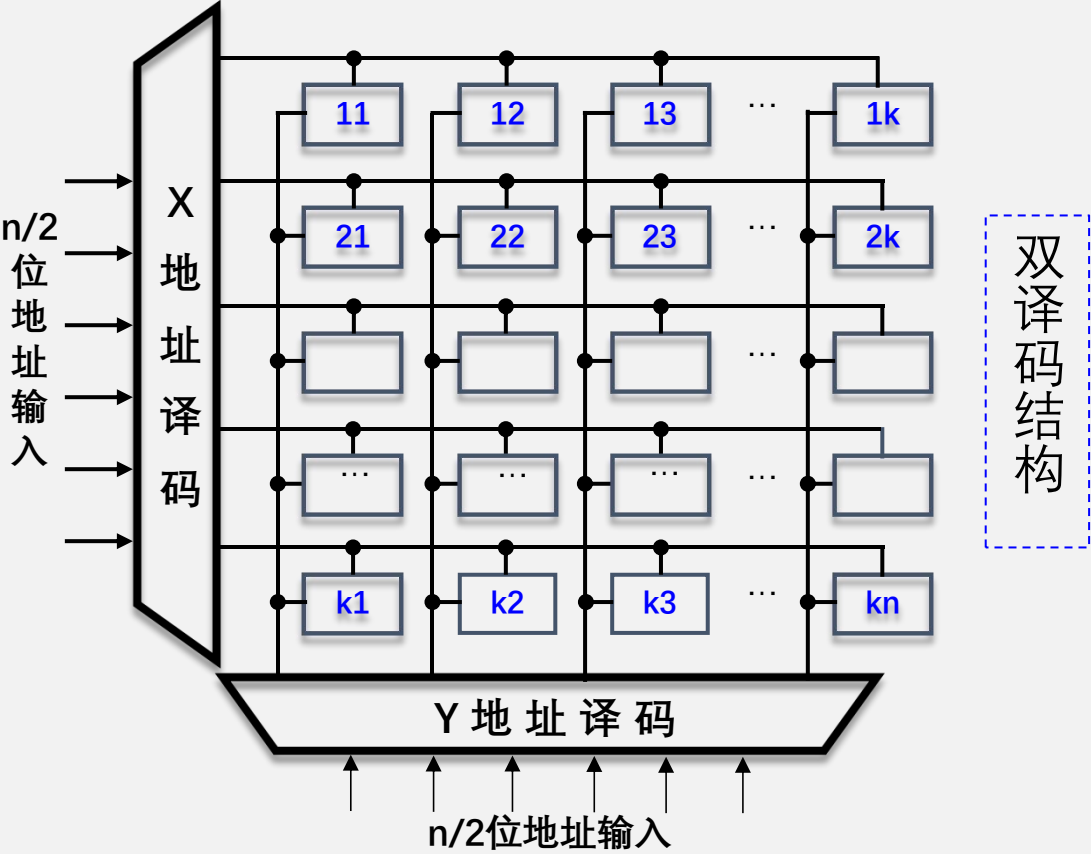




2. SRAM存储单元封装与扩展

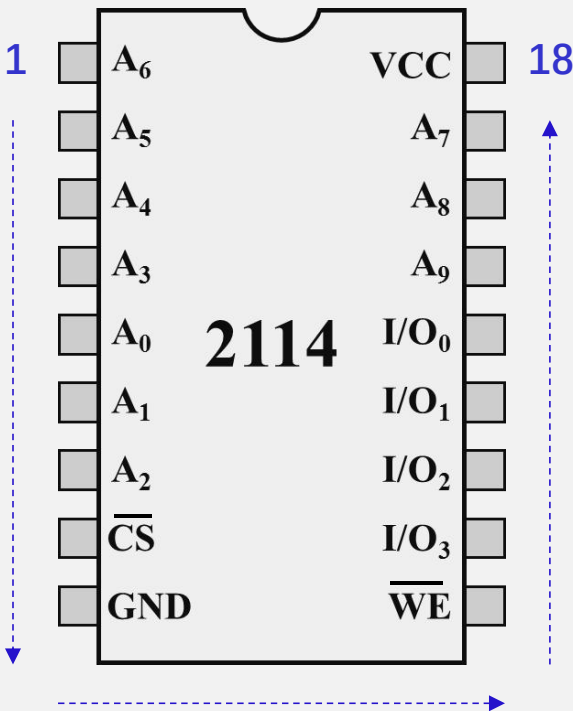


n位地址， 2^n 根译码线



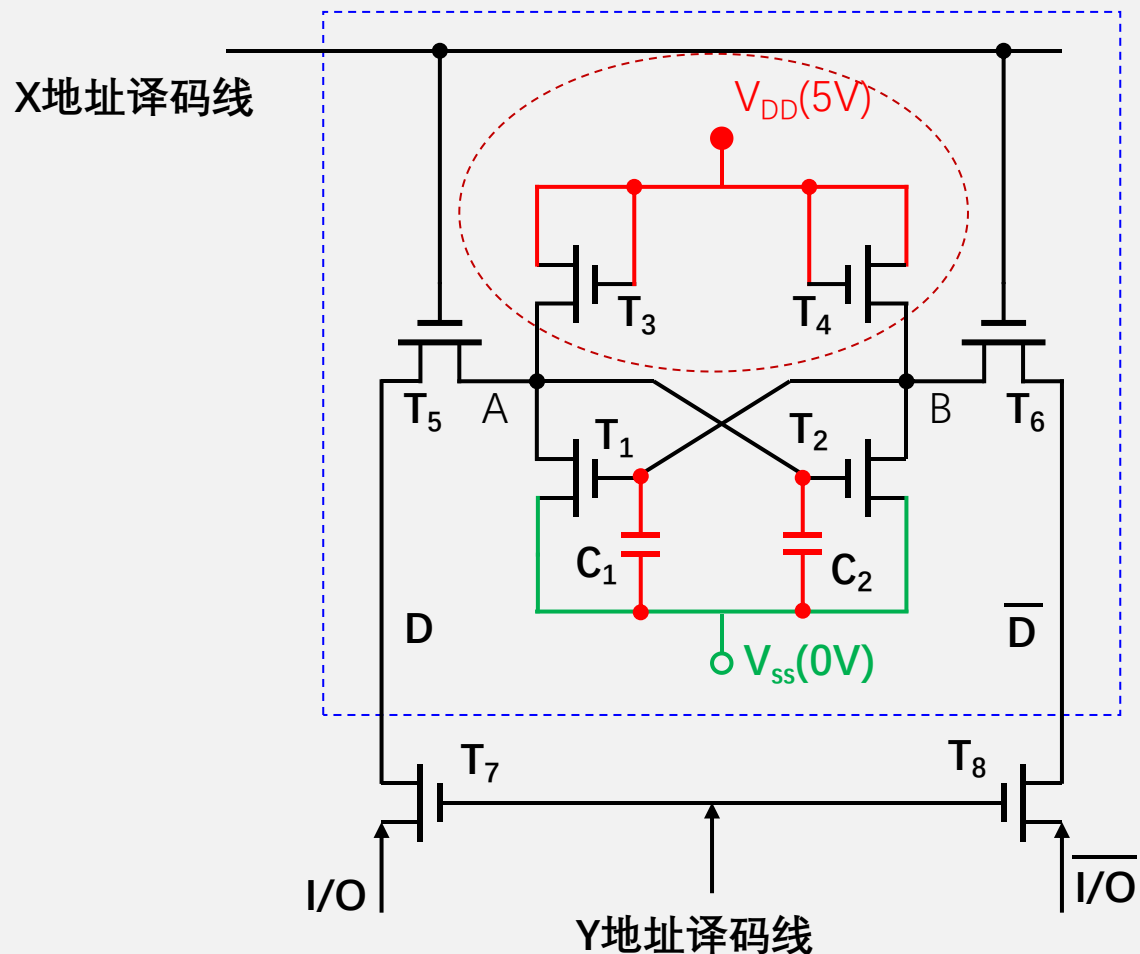
n位地址， $2 \times 2^{n/2}$ 根译码线

3. SRAM存储芯片举例

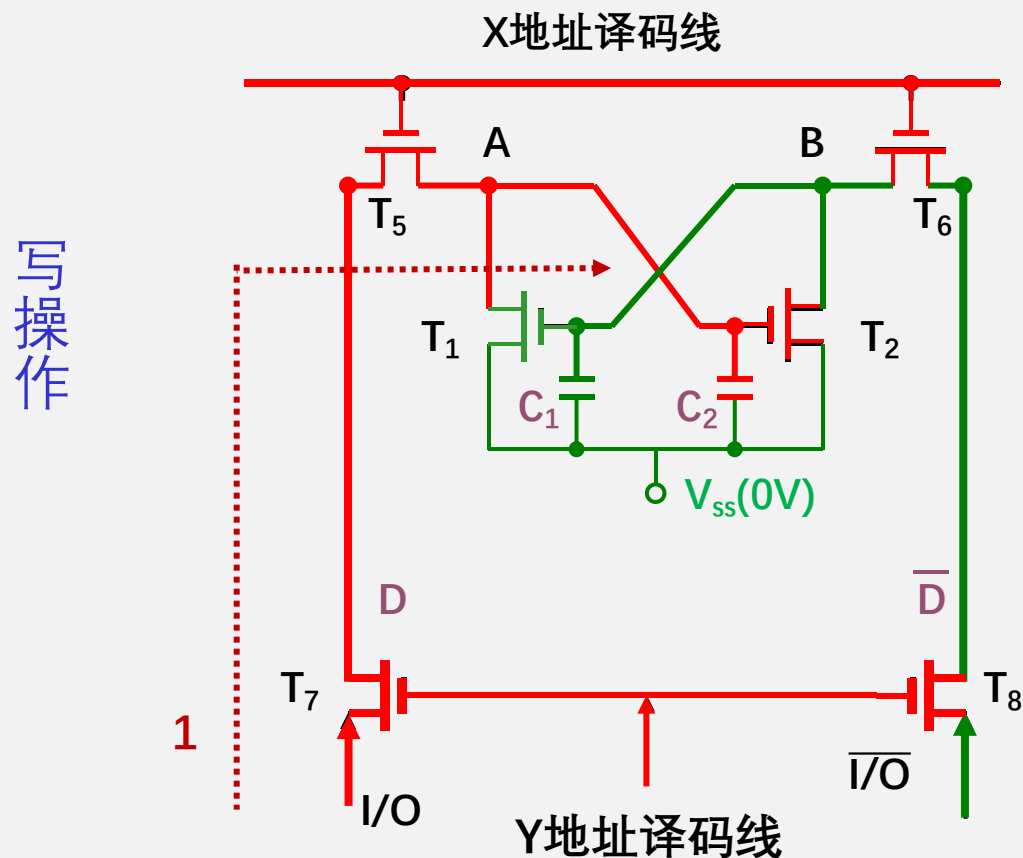


- 双列直插式引脚部署
- 地址线: 10根 1K 个存储单元
- 数据线(双向): 4根 每个存储单元4bit
- 读写控制线(Write Enable) 0:写 1:读
- 片选线(Chip Select) 0:选中 1: 不选中
- 电源线/地线

4. DRAM存储单元 (SRAM Cell—存储 1 bit)



- 去掉两个负载管
- 提升存储密度
- 减少功耗
- 降低成本
- 关注栅极分布电容的作用



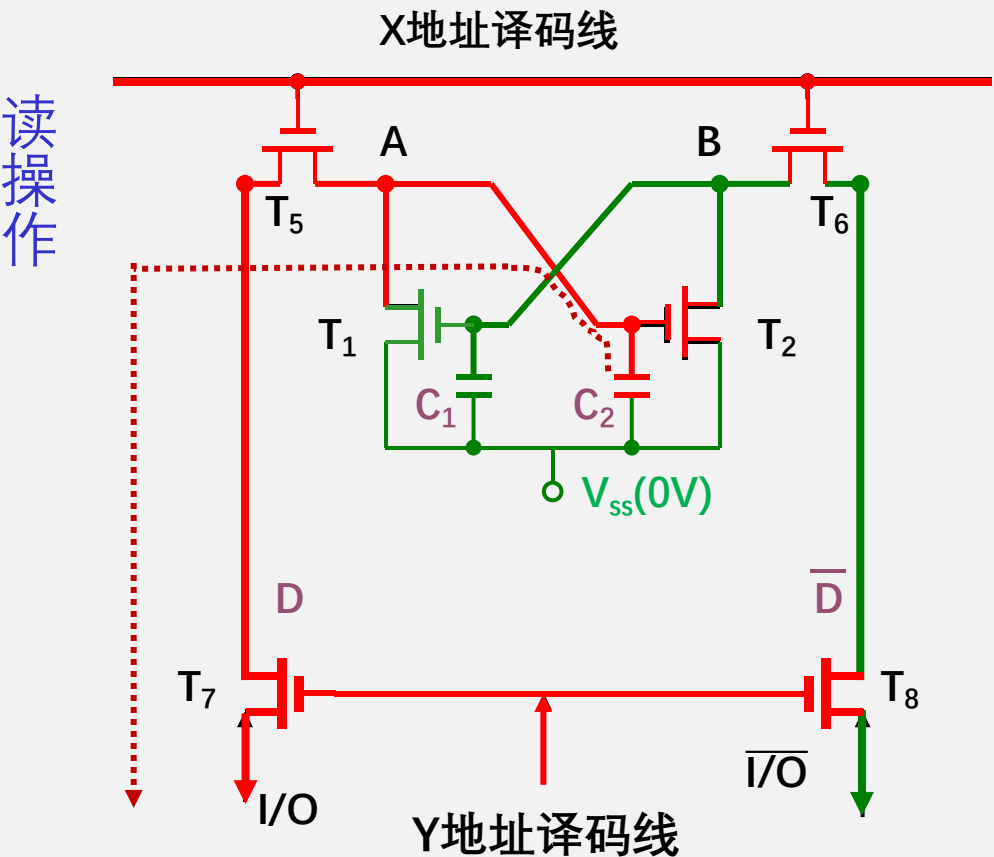
- Y地址选通
 - T7、T8管导通
 - I/O端数据写入到位线
- X地址选通
 - T5、T6管导通

写入数据向C₂和C₁充放电

C₂和C₁上电荷使T₁、 T₂构成稳定互锁态

利用电容电荷存储数据

4. DRAM存储单元 (SRAM Cell—存储 1 bit)



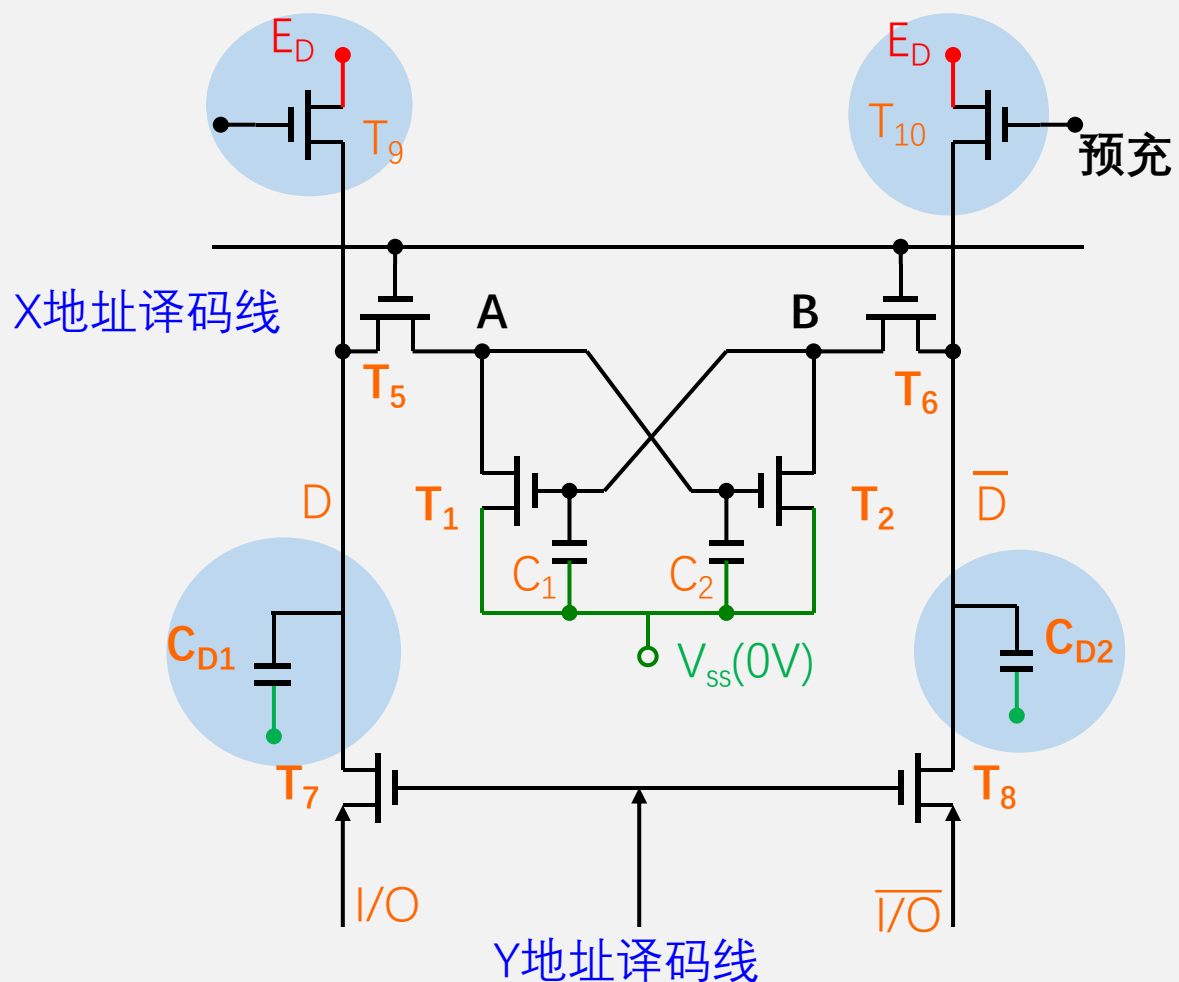
- Y地址选通
- T7、T8管导通
- I/O端数据与位线相连
- X地址选通
- T5、T6管导通

电容放电读出存储的信息

会有什么问题吗?

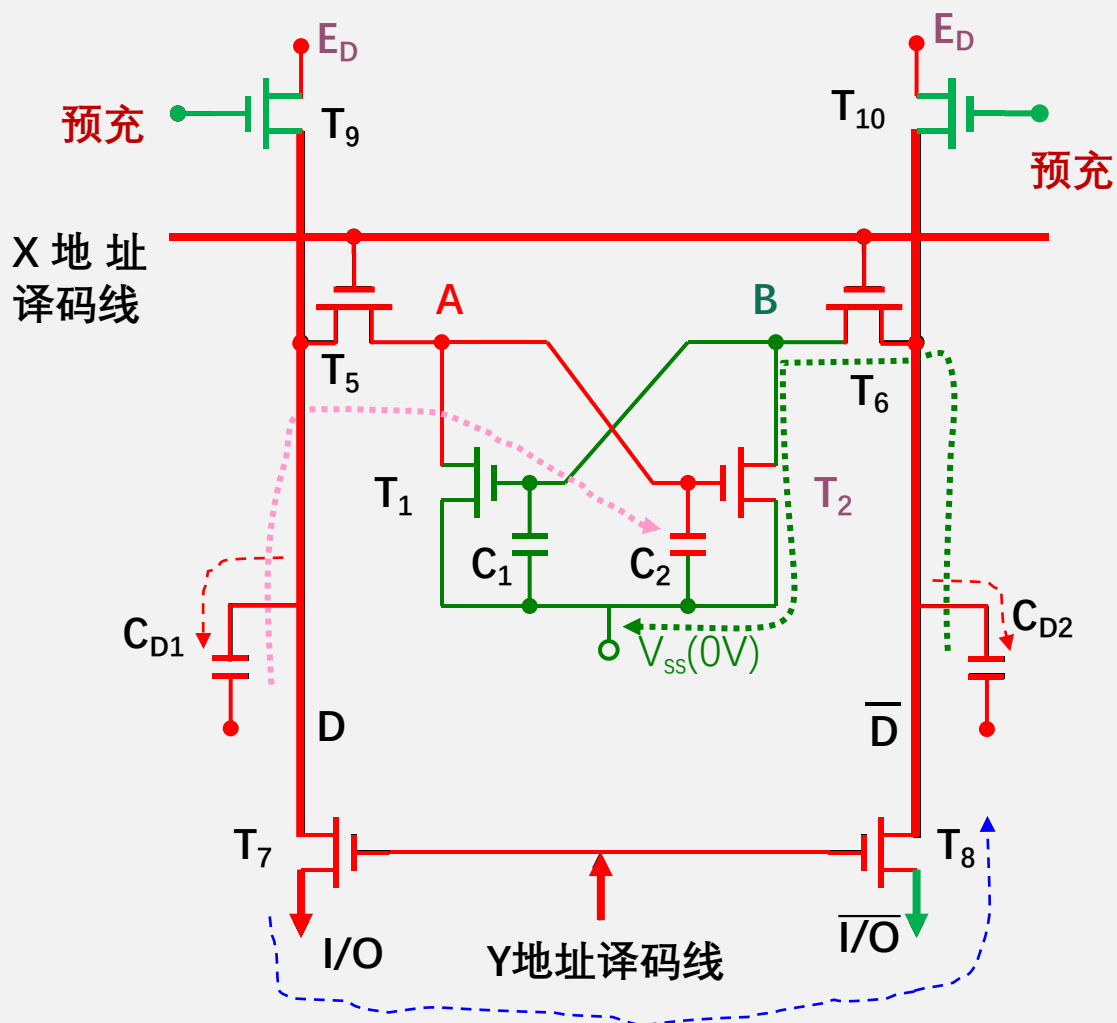
内容破坏

4. DRAM存储单元 (SRAM Cell—存储 1 bit)



- ◆ 外接大容量电容
- ◆ 增设预充电控制电路

4. DRAM存储单元 (SRAM Cell—存储 1 bit)



◆ 给出预充信号

T_9 、 T_{10} 导通、 E_D 给 C_D 充电

◆ 撤除预充信号

◆ X地址选通、Y地址选通

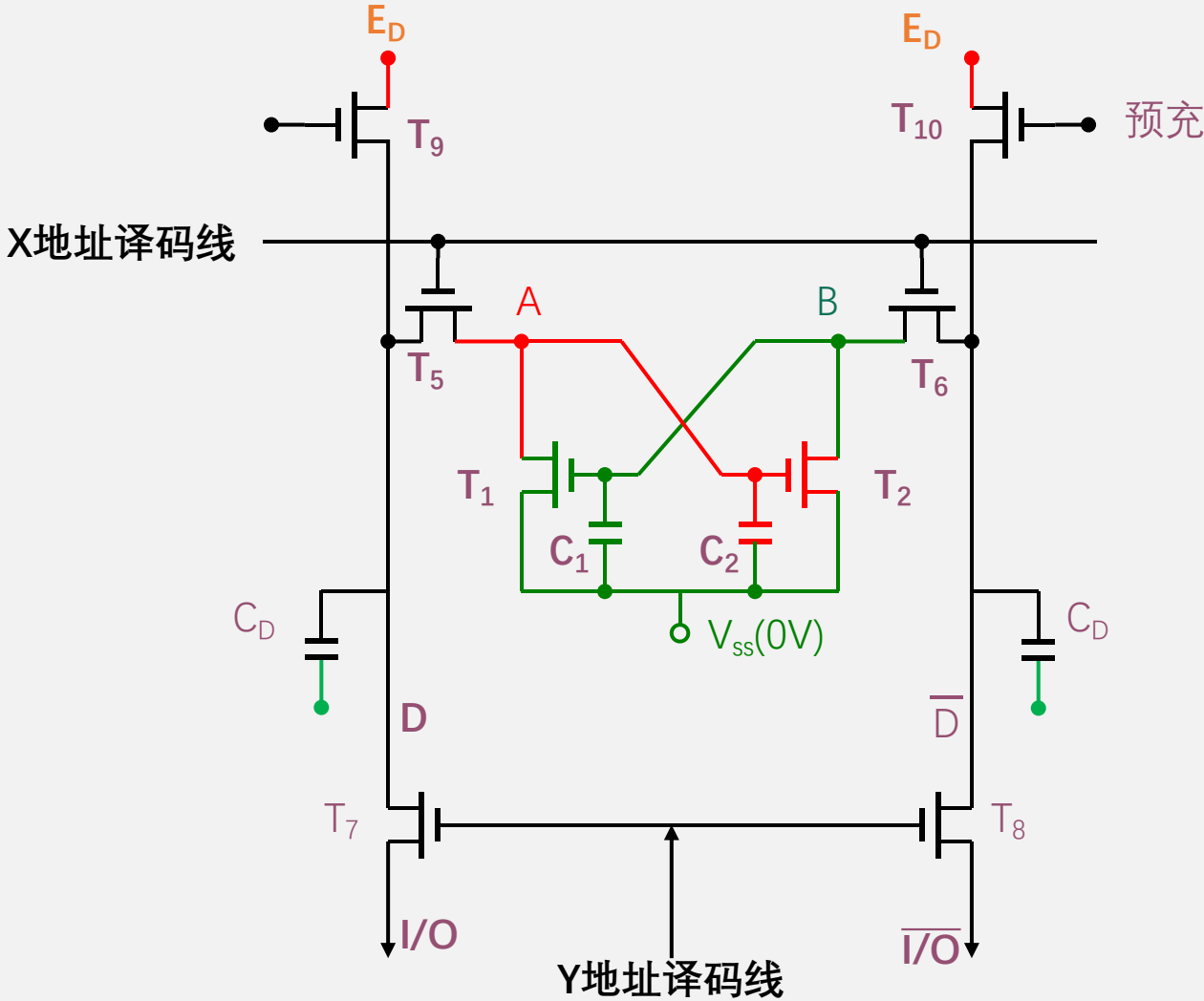
★ T_5 、 T_6 、 T_7 、 T_8 导通

★ C_{D1} 给 C_2 充电, C_{D2} 、 C_1 被放电

▲ 外接于I/O 和 I/O间的电流放大器
判断读出1还是0

过程繁多!

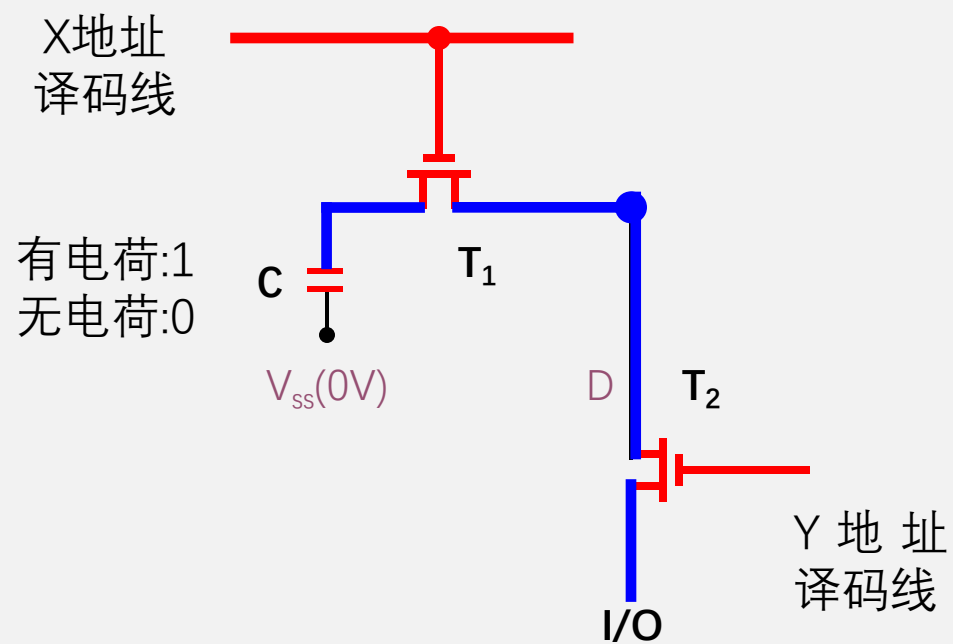
4. DRAM存储单元 (SRAM Cell—存储 1 bit)



◆ 如何进一步提高存储密度?

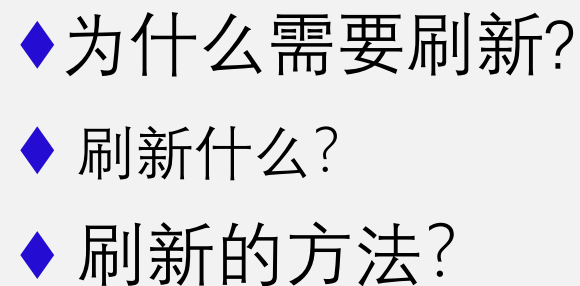
保留核心, 裁剪冗余

4. DRAM存储单元 (SRAM Cell—存储 1 bit)



读出后立即回写来保证破坏性读出

5. DRAM存储器的刷新



5. DRAM存储器的刷新

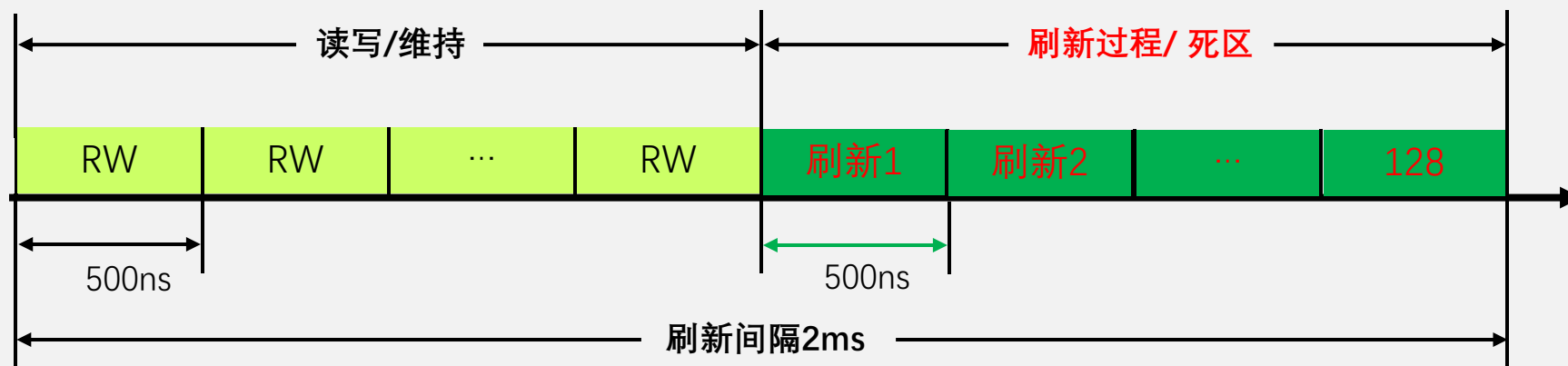
- **刷新：** 定期给栅极分布电容补充电荷，避免其电荷消耗引起信息丢失
 - 维持工作管 T_1 、 T_2 的工作状态，消耗栅极分布电容电荷
- **按行刷新(不完整的都操作)**
 - 存储体采用双译码结构，刷新地址计数器给出刷新行地址
- **刷新周期：** 存储器两次完整刷新之间的时间间隔
 - 栅极分布电容电荷泄漏临界值前必须进行刷新，称为**最大刷新周期**
- **刷新方式**
 - 集中式、分散式、异步式

5. DRAM存储器的刷新

设刷新周期2ms，存储矩阵采用128行×128列，存储体读/写周期为0.5 μs

◆ 集中刷新

2ms内共有 4000个读写周期 ($2000/0.5 = 4000$)



集中刷新的存储器平均读写周期

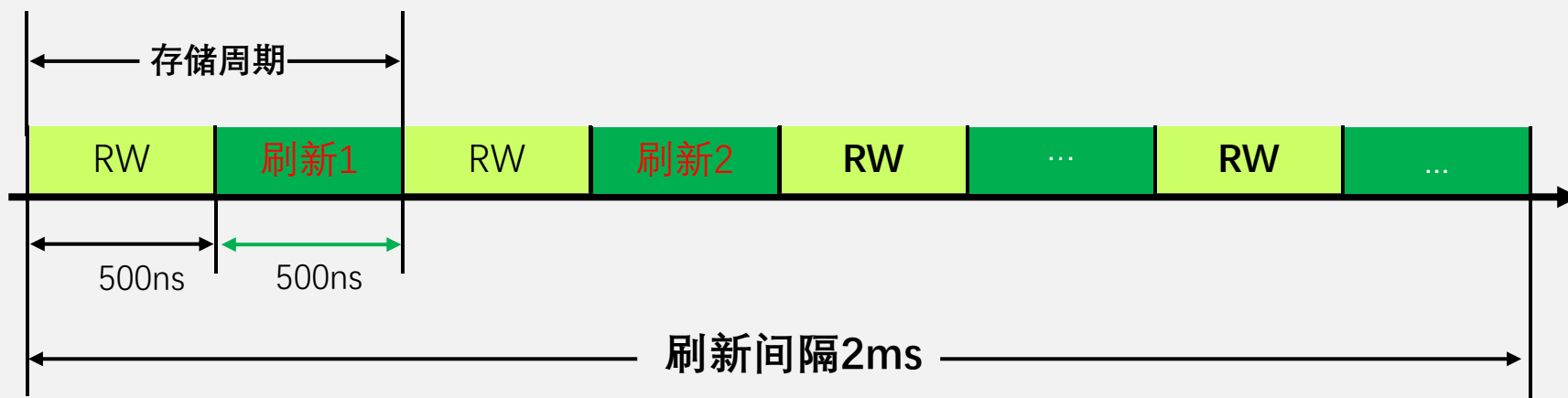
$$\bar{T} = 2\text{ms} / (4000 - 128) = 0.5165\mu\text{s}$$

5. DRAM存储器的刷新

设刷新周期2ms，存储矩阵采用128行×128列，存储体读/写周期为0.5 μs

◆ 分散刷新

各刷**分散**分布于各读/写周期中，存储周期=读写时间+刷新时间 = 1 μs



刷新周期内刷新了 2000次

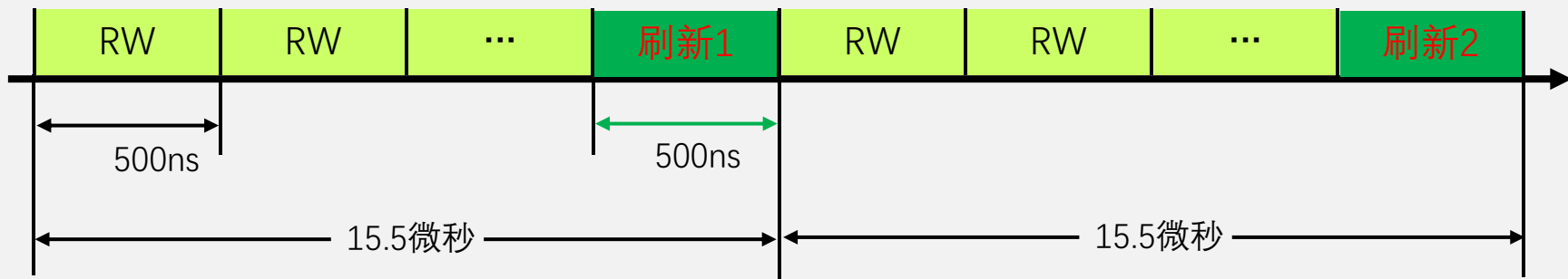
5. DRAM存储器的刷新

设刷新周期2ms，存储矩阵采用128行×128列，存储体读/写周期为0.5 μs

◆ 异步刷新

将刷新分布在不同时间段，每时间段刷新一次；

$2\text{ms}/128=15.5$ 微秒,即将128次刷新分散在128个15.5 微秒的时间段

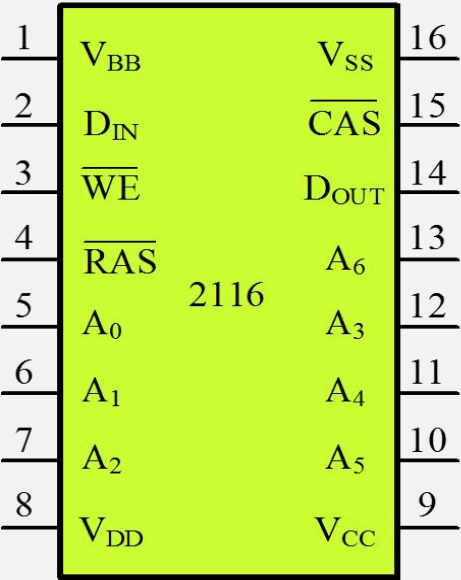


$$\bar{T} = 2\text{ms} / (4000 - 128) = 0.5165\mu\text{s}$$

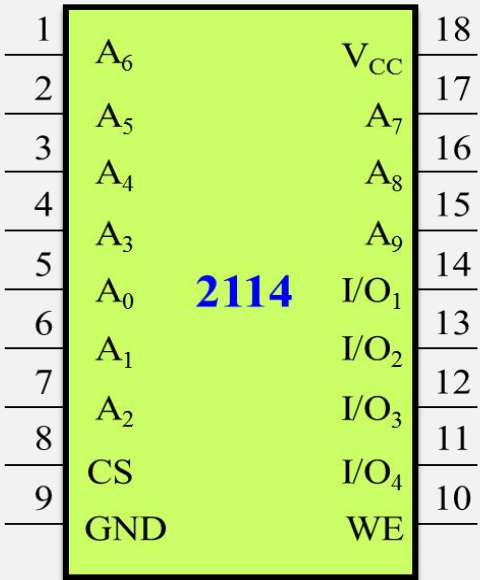


7.2 半导体存储器

6. DRAM芯片举例



- 地址线
- 数据线
- 读写控制线
- RAS CAS
- 电源线
- 地线



2116 的容量?

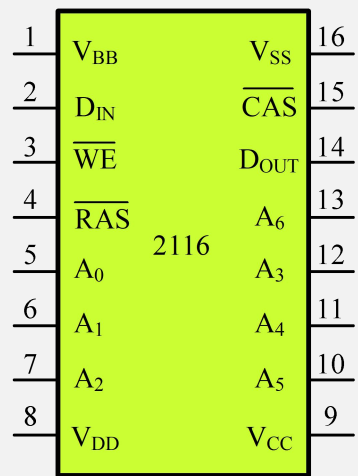
6. DRAM芯片举例

关于动态存储器刷新的几点说明：

- ◆不同材料不同生产工艺的动态存储器刷新周期不同，常见有2ms、4ms、8ms，刷新时间间隔不能超过刷新周期。
- ◆刷新按行进行，故需了解存储体行、列结构，即行译码器输出线数，同时考虑**双译码原则**。如存储器由256K的存储单体构成时，行译码输出线为 2^9 ；若存储器由512K的单体构成，则行译码输出可以是 2^9 或 2^{10} ，但要说明，具体情况要查存储器手册。
- ◆刷新地址由专门器件-----刷新地址计数器给出。

7.2 半导体存储器

6. DRAM芯片举例



思考: DRAM比SRAM慢的原因?

$\overline{\text{RAS}}$: 行地址选通信号输入引脚，低电平有效，兼作芯片选择信号。低电平时，表明芯片当前接收的是行地址；

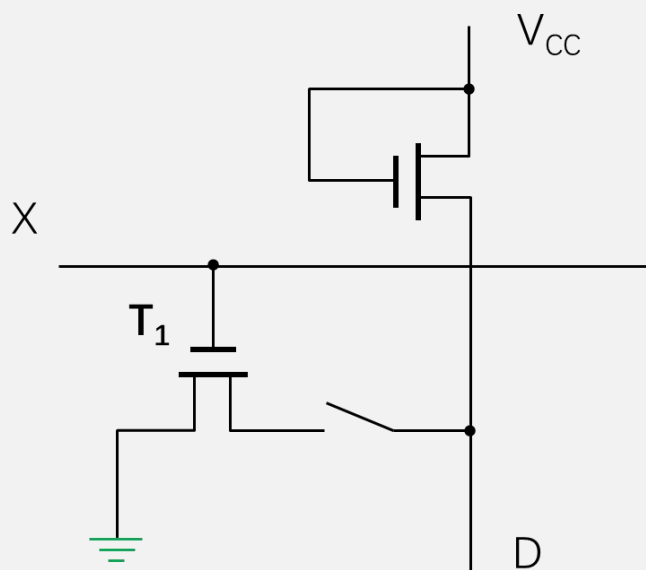
$\overline{\text{CAS}}$: 列地址选通信号输入引脚，低电平有效，表明当前正在接收的是列地址（此时 行选择 应保持为低电平）



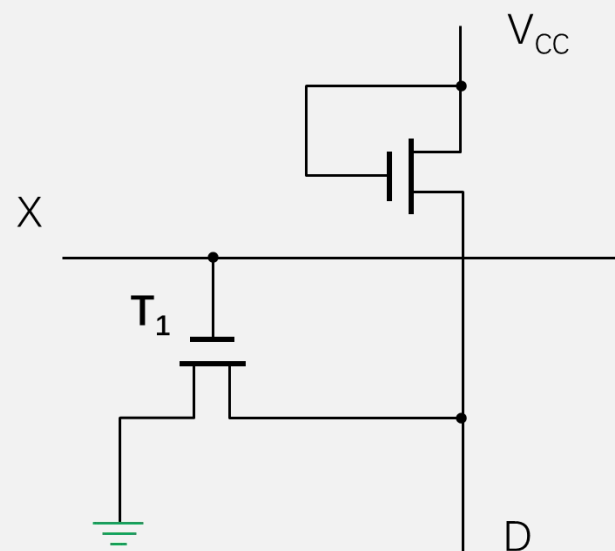
7.2 半导体存储器

7. 只读存储器ROM

(1) 掩模式只读存储器



1 单元

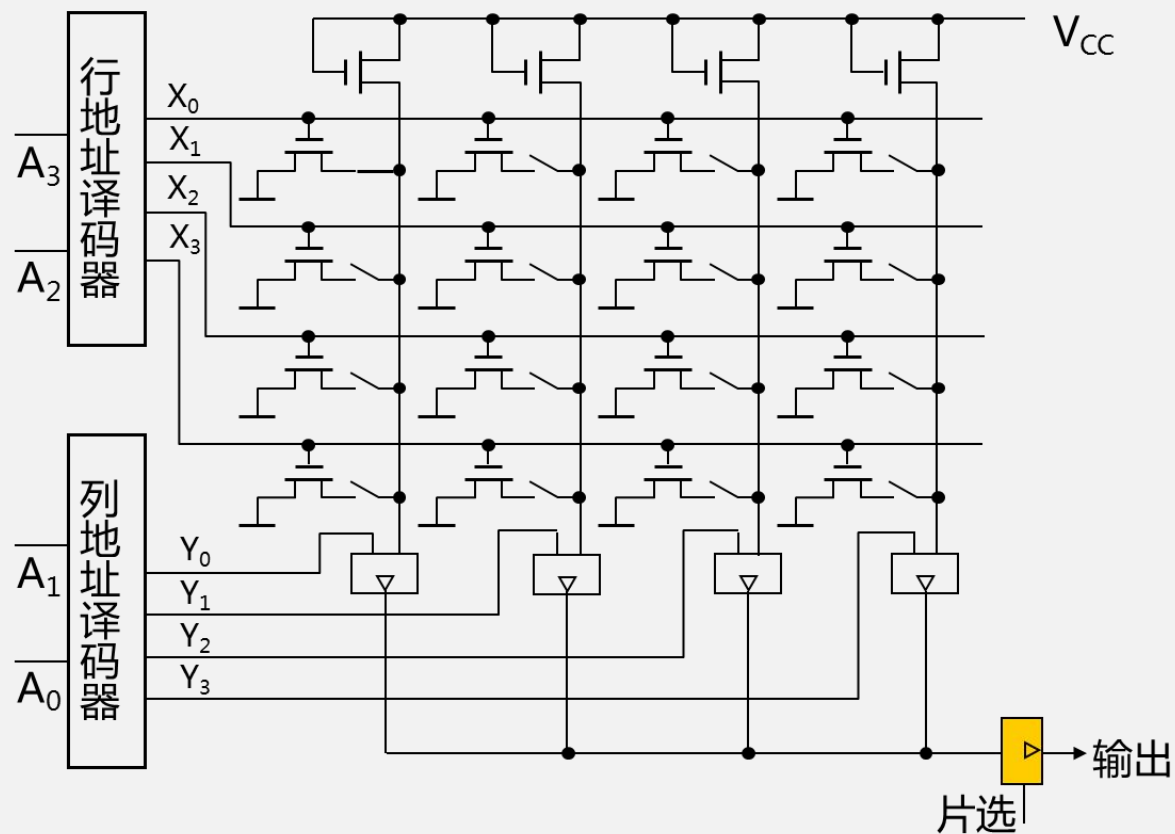


0 单元

7.2 半导体存储器

7. 只读存储器ROM

(1) 掩模式只读存储器

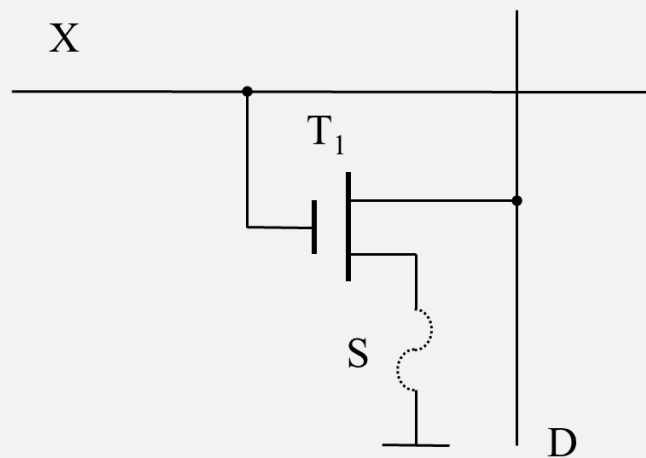




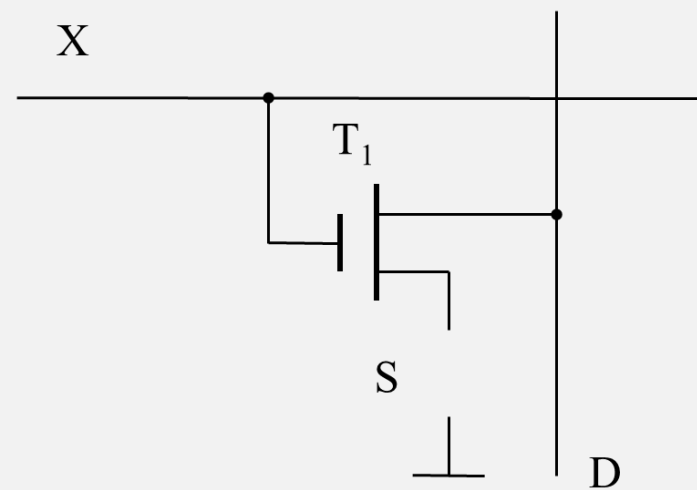
7.2 半导体存储器

7. 只读存储器ROM

(2) 熔丝式只读存储器(可编程一次)



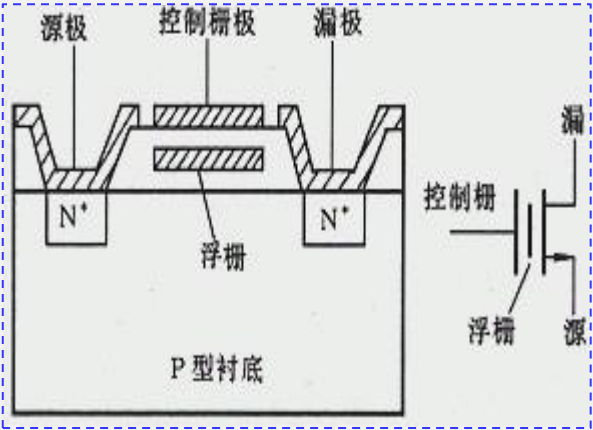
0 单元



1 单元

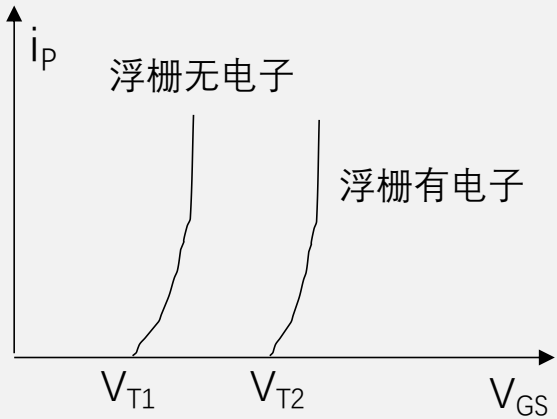
7. 只读存储器ROM

(3) EPROM: Erasable Programmable Read Only Memory



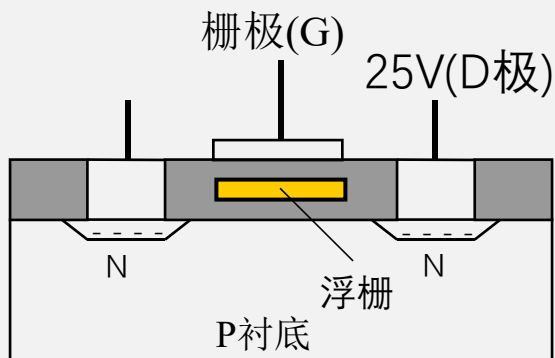
◆ EPROM: 采用**浮栅技术**的可编程存储器，多采用N沟道叠栅MOS管(SIMOS)，

◆ 浮栅上无电荷时，栅极加控制电压，MOS管导通；浮栅上带负电荷时，衬底表面感应出正电荷，MOS管开启电压变高，若栅极加同样控制电压，MOS管处于截止状态。故SIMOS管可以利用浮栅是否积累有负电荷来存储二值数据。

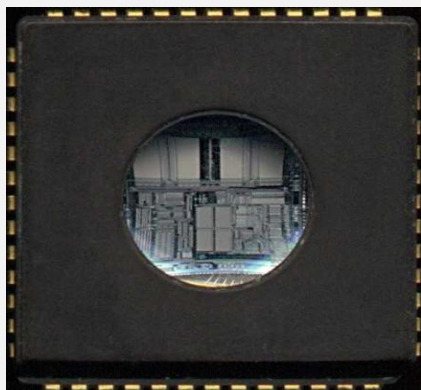


7. 只读存储器ROM

(3) EPROM: Erasable Programmable Read Only Memory



EPROM结构

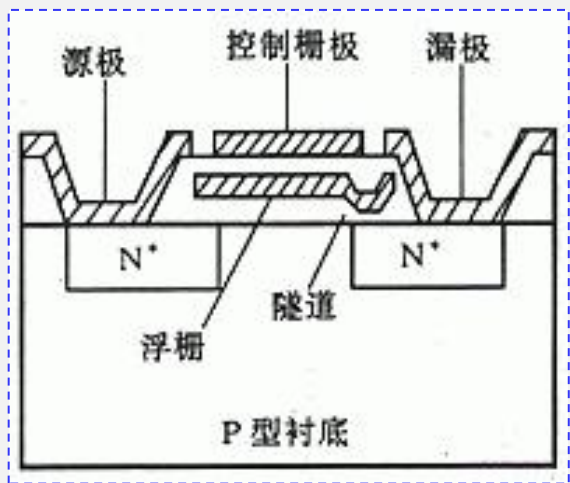


◆写入数据前，浮栅不带电，要使浮栅带负电荷，须在其D、G极加上**高压**(如25V),使漏极及衬底间PN结反向击穿，产生大量高能电子，穿过薄氧化绝缘层堆积在浮栅上，从而使浮栅带有负电荷。浮栅上的电子没有放电回路，能够长期保存。

◆用**紫外线或X射线照射**时，浮栅上的电子形成光电流而泄放，从而恢复写入前的状态。照射一般需要15至20分钟。

7. 只读存储器ROM

(4) EEPROM: Electrically Erasable Programmable read only memory



隧道MOS管EEPROM结构图

◆ E²PROM采用**浮栅技术**，其浮栅延长区与漏区间形成一个厚度约为80埃的薄绝缘层，当**漏极接地、控制栅加高电压**时，交叠区将产生强电场，使电子通过绝缘层到达浮栅。称为“隧道效应”，故**该MOS管也称为隧道MOS管**。

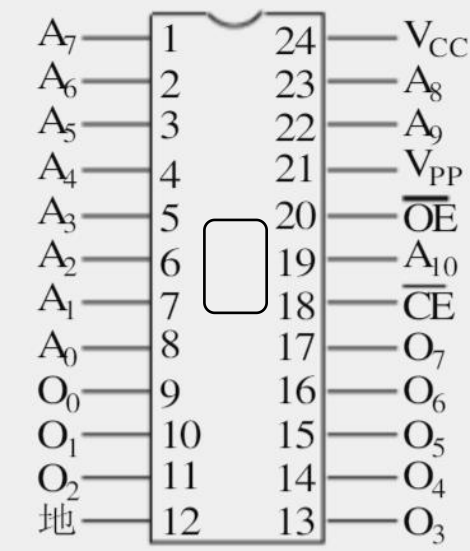
◆ 当**控制栅接地、漏极加正电压**，则浮栅放电。隧道MOS管也是利用浮栅是否积累有负电荷来存储二值数据。与SIMOS(叠栅注入MOS)不同的是**隧道MOS管利用电擦除，且擦除速度快**。

◆ E²PROM具有ROM的非易失性和类似RAM的功能，可随时改写(重复上万次)。该芯片内部备有升压电路,只需提供单电源即可，**为数字系统设计和在线调试提供极大方便**。



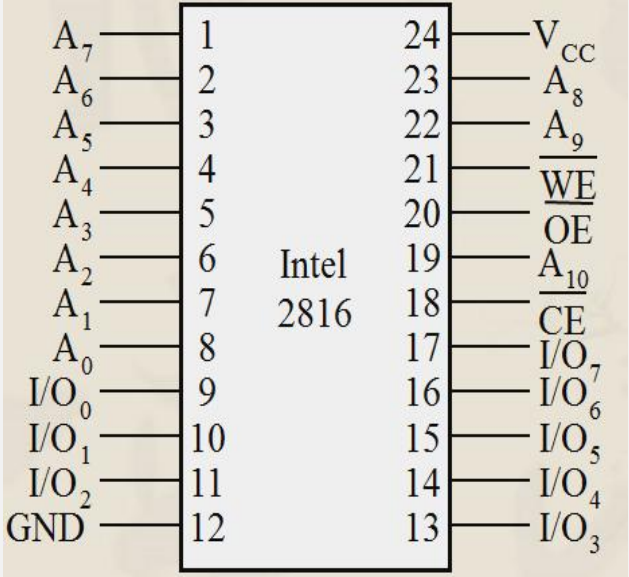
7.2 半导体存储器

7. 只读存储器ROM



EPROM2716

V_{pp}- 25V专用设备上的写操作



E²PROM2816A

\overline{OE} :输入允许
 \overline{CE} :片选信号
 \overline{WE} :写允许



7.2 半导体存储器

7. 只读存储器ROM

常见E²PROM及其参数

	2816	2816A	2817	2817A	2864A
读取时间ns	250	200-250	250	200-250	250
读电压V _{pp} /V	5	5	5	5	5
写/擦电压V _{pp} /V	21	5	21	5	5
字节擦写时间ms	10	9-15	10	10	10
写入时间ms	10	9-15	10	10	10
封装	DIP24	DIP24	DIP28	DIP28	DIP28

1)存储字长与数据字长的概念

- ◆ 存储字长：主存一个存储单元能存储的二进制位数的最大值；
- ◆ 数据字长(字长)：计算机一次能处理的二进制数位数的最大值；

2)主存编址！

计算机主存一般**按字节编址**,而计算机的字长也往往包含多个字节,如16、32和64位字长，分别为2、4、8个字节。

0	1	2	3
4	5	6	7
8	9	10	11
12	13	14	15

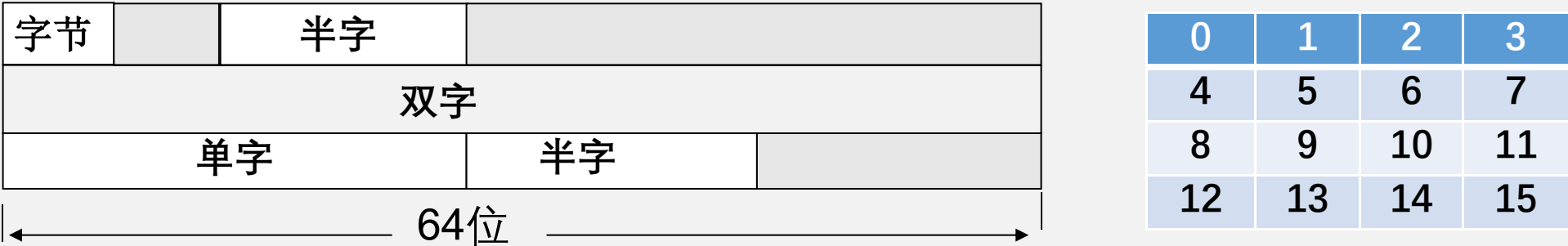
主存

思考：你的主机主存容量为4G，以下哪项描述正确？

A. 4GB B. 4G×16bit C. 4G×32bit D. 4G×64bit

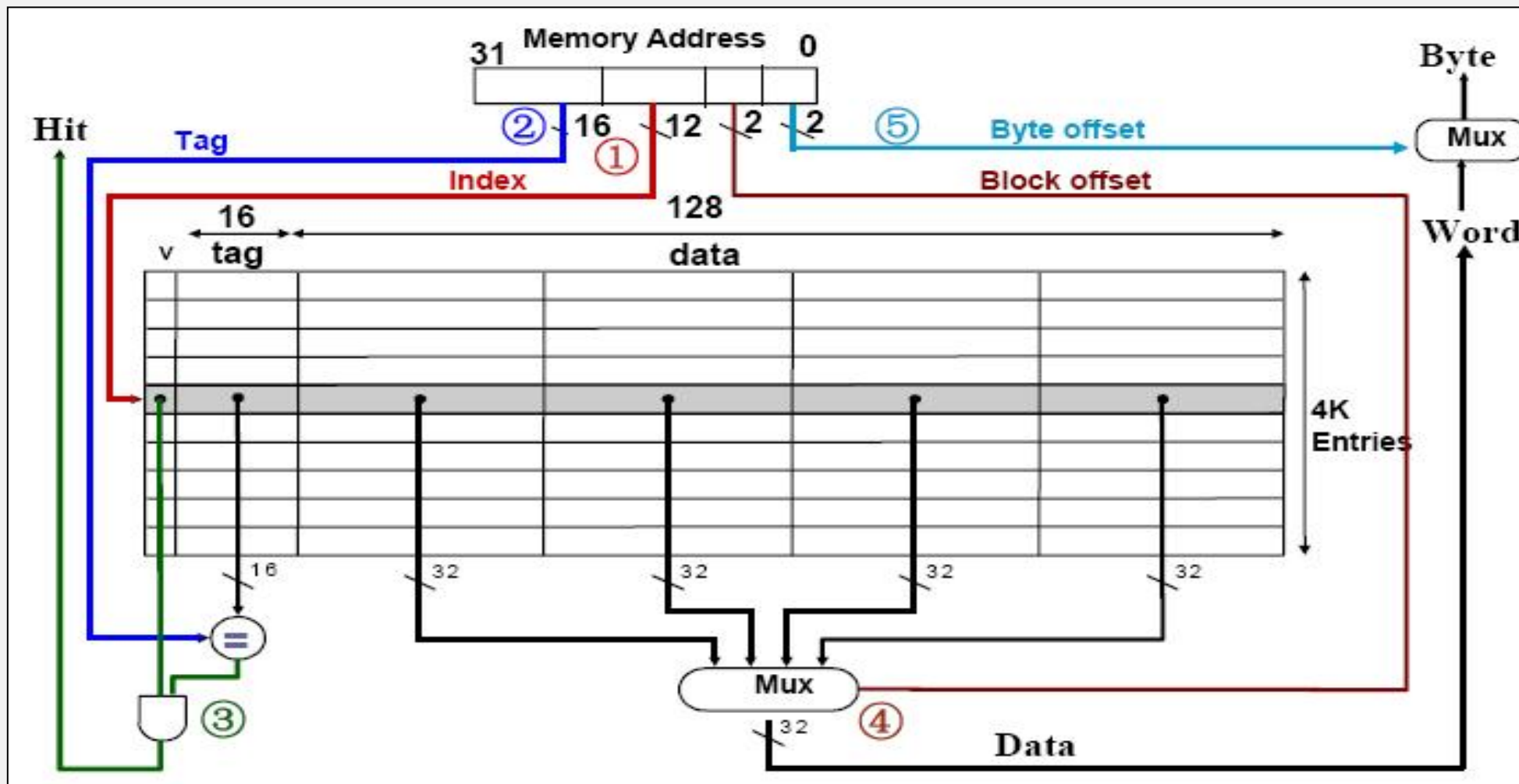
3)整数边界存放

- ◆ 数据字长32位，则半字长16位，双字长64位
- ◆ 存储字长64位，则各类按整数边界对齐的数据存放如下：



- ◆ 双字数据起始地址的最末三位为000(8字节整数倍),表示访问一个64位字；若访问其中的字节或半字则用低三位地址中的部分位来选择。
- ◆ 单字长整数边界对齐的起始地址末二位为00(4字节整数倍)
- ◆ 半字起始地址的最末一位为0(2 字节的整数倍)。

7.3 主存的数据组织



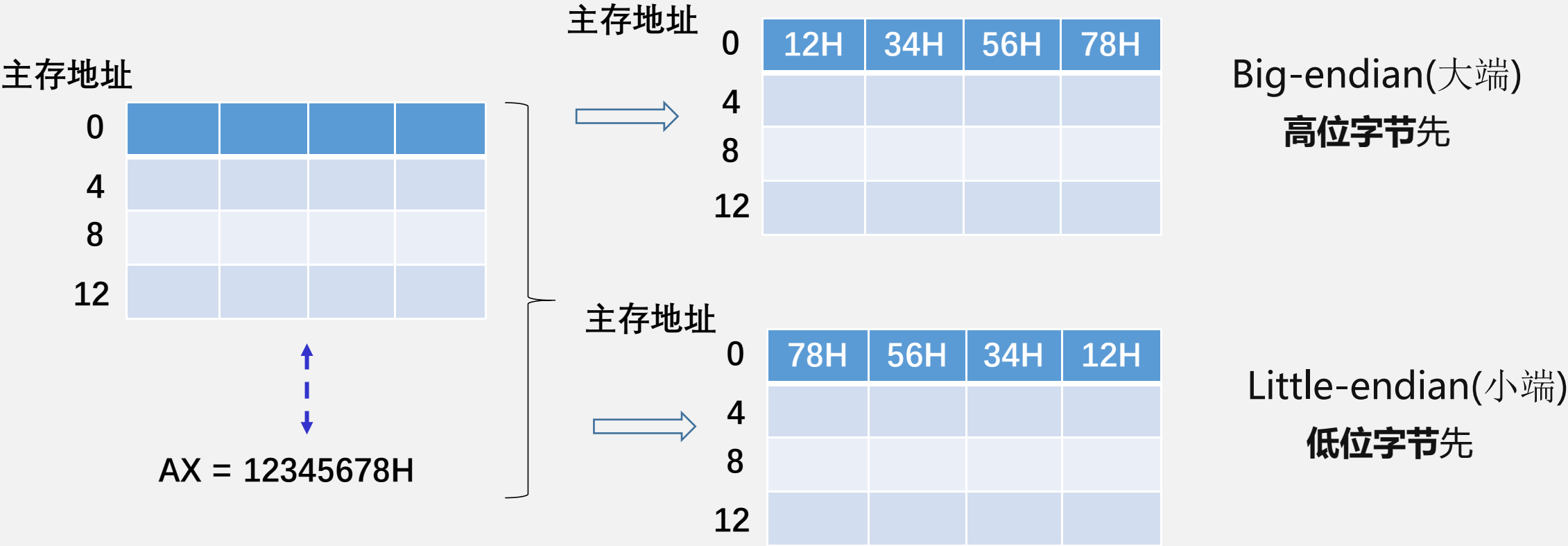


7.3主存的数据组织

4)大端和小端存放

◆ 数据有高、低位之分

◆ 主存地址有高、低地址之分





4)大端和小端存放

- ◆ 大小端模式各有优势：小端模式强制转换类型时不需要调整字节内容，直接截取低字节即可；大端模式由于符号位为第一个字节，方便判断正负。
- ◆ 网络协议都采用Big-endian方式传输数据，Big-endian也称为网络字节序。

4)大端和小端存放

◆ 如何知道你的机器采用大端还是小端模式？

```
#include<stdio.h>
#include<stdlib.h>
int CheckSystem()
{
    union check
    {
        int i;
        char ch;
    };
    c.i = 1;
    return (c.ch == 1);
}
int main()
{ int ret = CheckSystem();
  printf("ret : %d\n",ret);
  system("pause");
  return 0; }
```

ret: 1



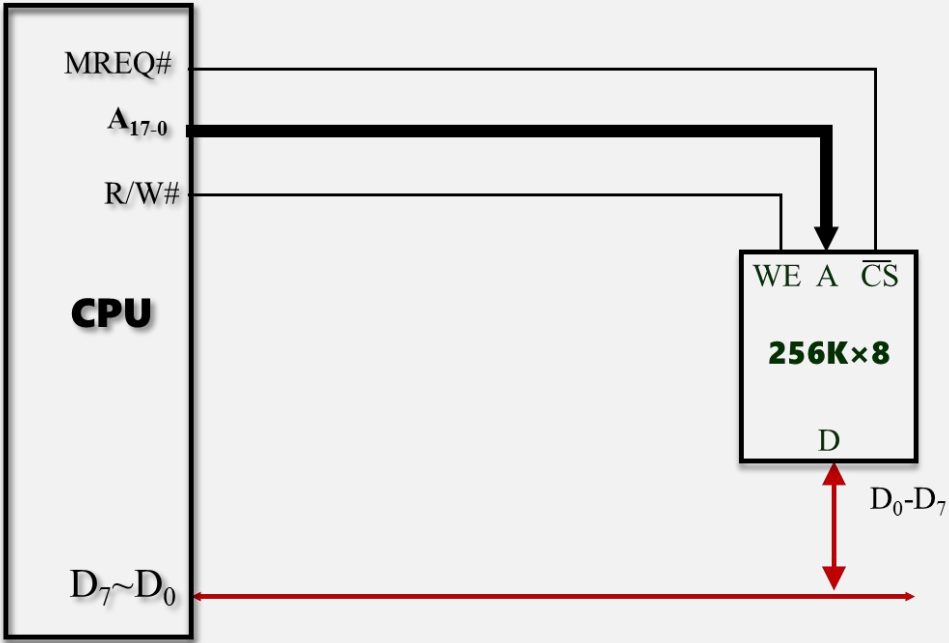
小端模式

若是大端模式



ret: ?

将CPU和存储器的地址线、数据线和控制信号线连接起来，CPU对存储器的读写操作,并满足CPU对主存容量和存储字长的要求。



存储器扩展的种类

- ◆位扩展 :当存储单体数据位不足时
- ◆ 字扩展:当存储单体容量不足时
- ◆ 字位同时扩展:当存储器单体数据位和容量均不满足时

1. 位扩展(存储器字长扩展)

- ◆位扩展时，多个存储单体**并行工作**，实现存储字长扩展。
- ◆所需存储单体数量: $K = \text{CPU数据总线位数} / \text{存储单体数据位}$

例1 用256K×8位的SRAM芯片，构造256K×32位的存储器？

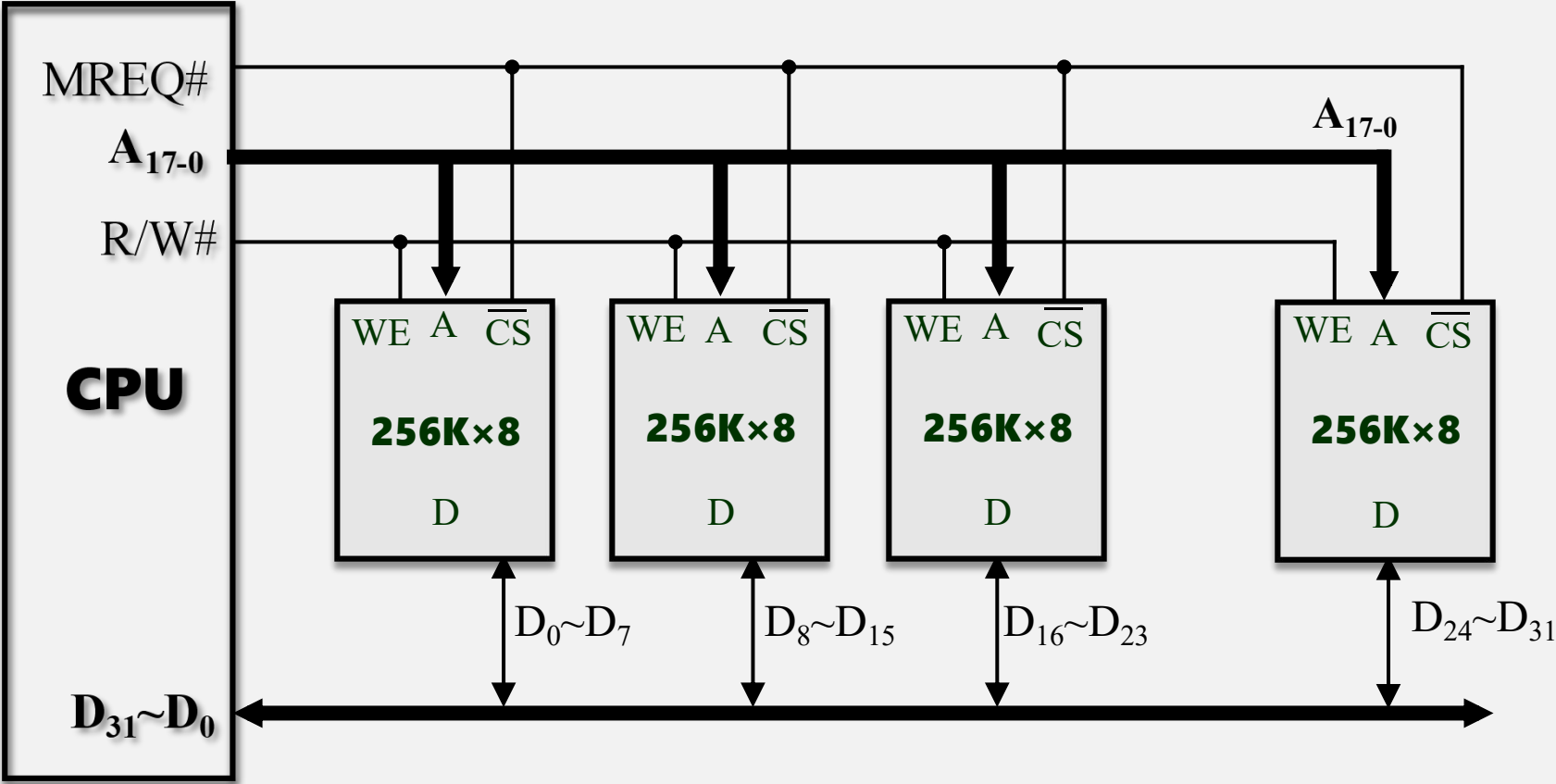
所需存储单体数量: $K = 32 / 8 = 4$

CPU与各存储单体的连接如下图所示。



7.4 主存与CPU的连接

1. 位扩展(存储器字长扩展)



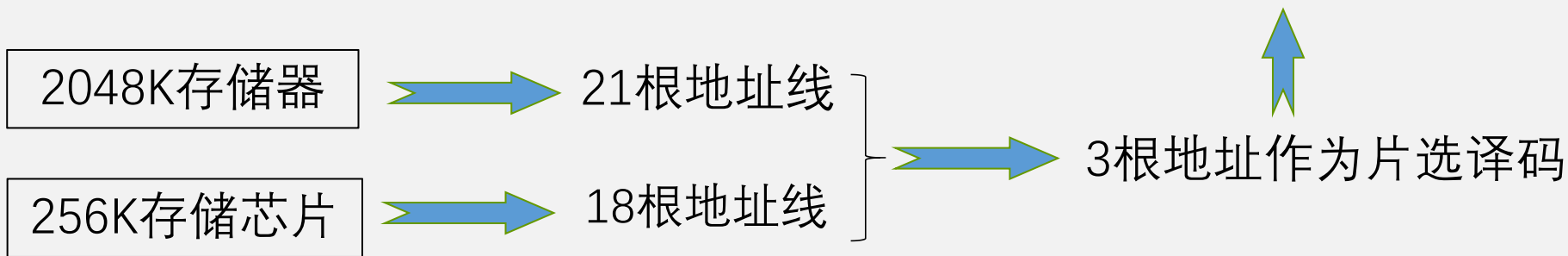
存储单体并行工作的实现方法

2. 字扩展(存储器容量扩展)

- ◆字扩展时，多个存储单体在片选信号的控制下**分时工作**，实现存储容量扩展。
- ◆所需存储单体数量: $K = \text{CPU地址线位数} / \text{存储单体地址线}$
 $= \text{目标存储器容量} / \text{存储单体存储容量}$ (注意单位统一)

例2 用256K×8位的SRAM芯片，设计2048K×8位的存储器。

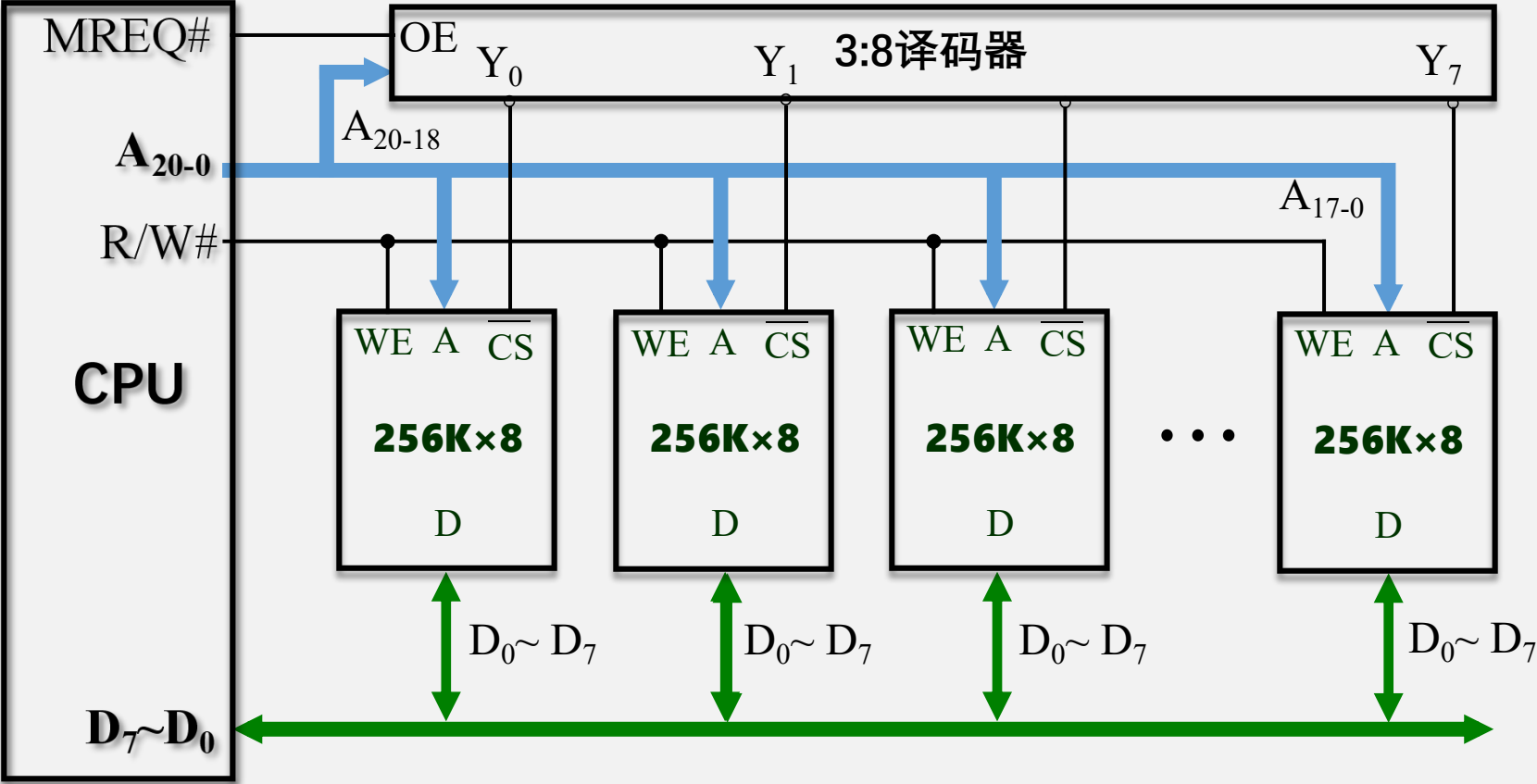
所需要的芯片数量: $2048K / 256K = 2^{(21-18)} = 8$ 每个片选信号选择256K存储空间





7.4 主存与CPU的连接

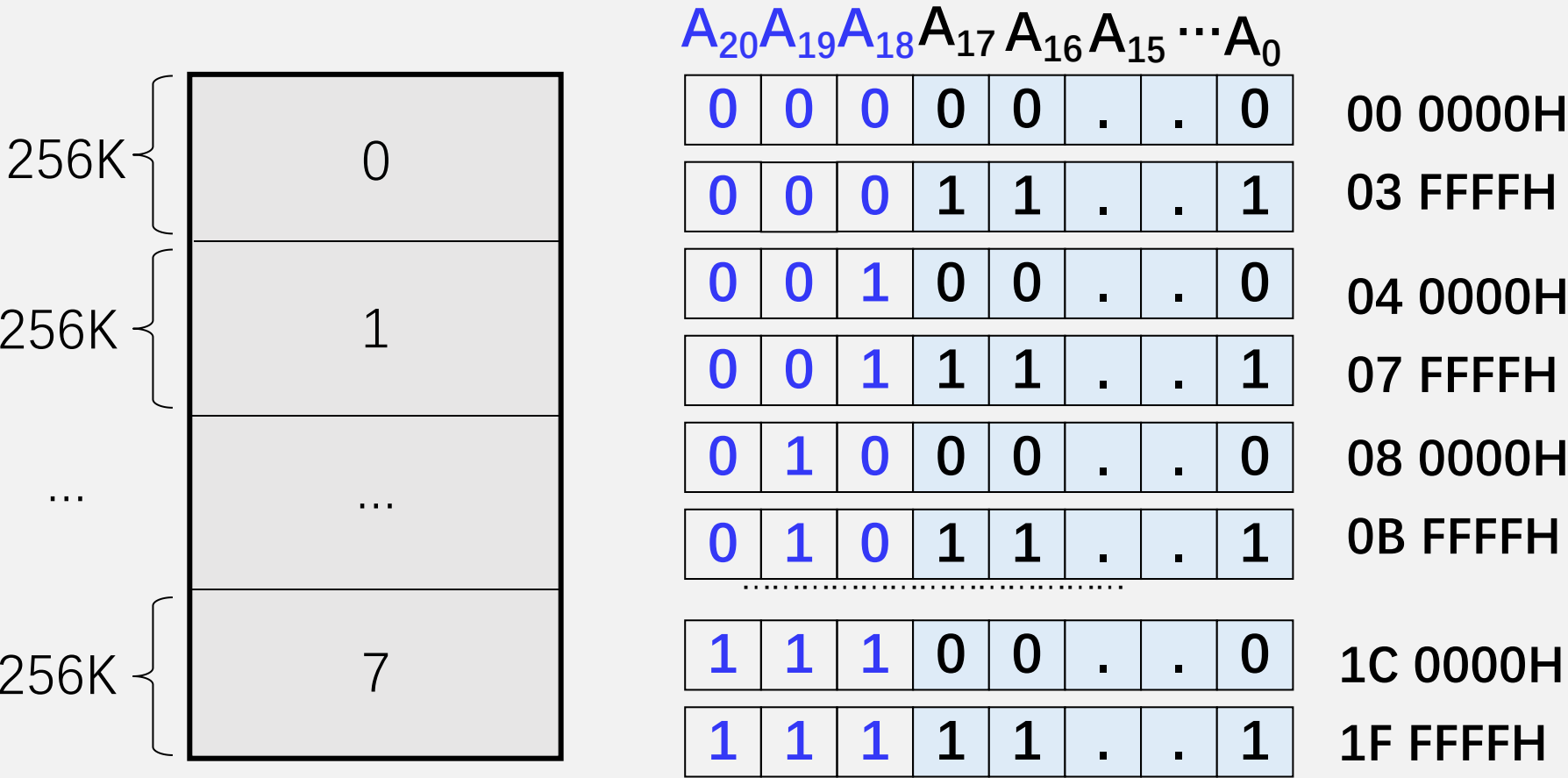
2. 字扩展(存储器容量扩展)





7.4 主存与CPU的连接

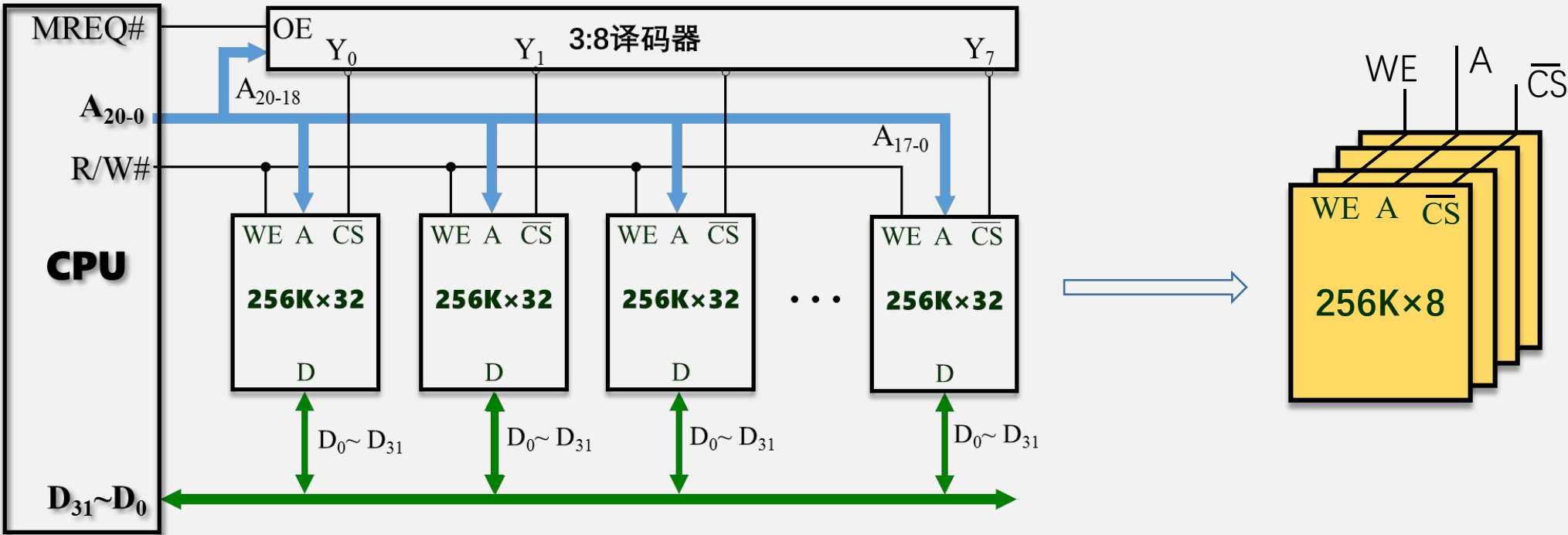
2. 字扩展(存储器容量扩展)



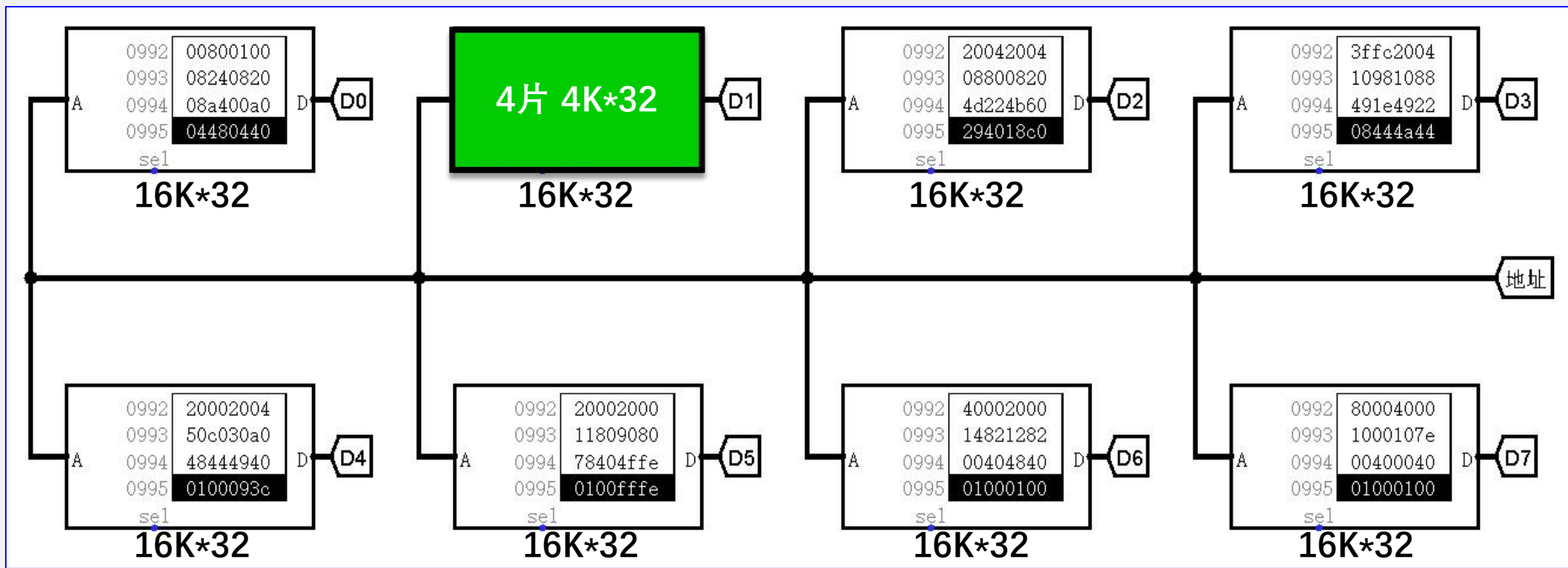
3. 字位同时扩展

例3 用256K×8位的SRAM芯片，设计2048K×32位的存储器？

所需要的芯片数量: $(2048K \times 32) / (256K \times 8) = 32$



3. 字位同时扩展

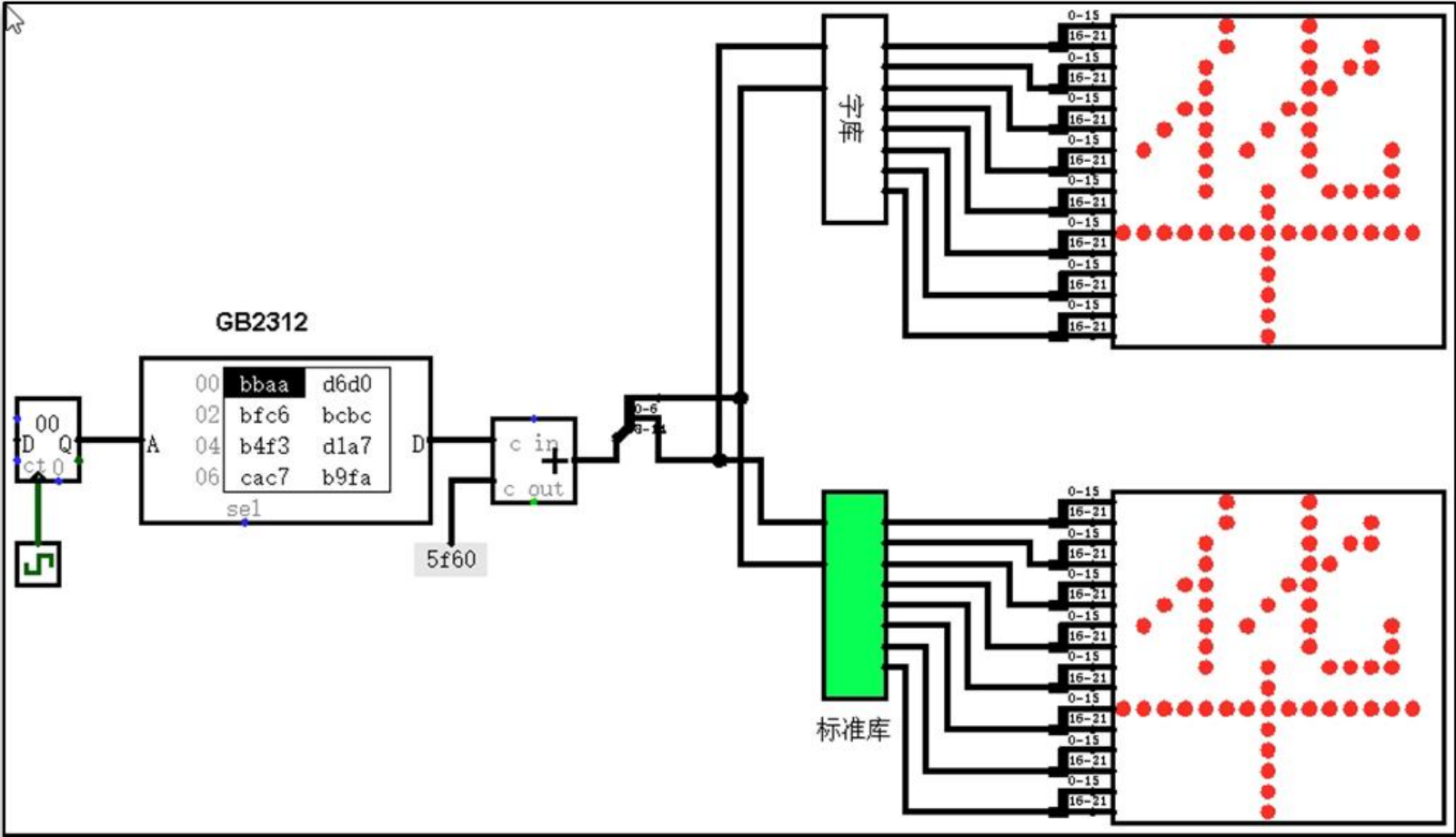


- ◆ 16×16 点阵需256位才能显示一个汉字，原字库文件数据如何分布
- ◆ 用4片 $4K \times 32$ 位的ROM 扩展并代替其中一片 $16K \times 32$ 位器件



7.4 主存与CPU的连接

3. 字位同时扩展



3. 字位同时扩展

例4 某计算机的主存地址空间中 0x0000到3FFF为ROM存储区域、 0x4000到0x5FFF为保留地址区域、0x6000到0xFFFF为RAM地址区域。

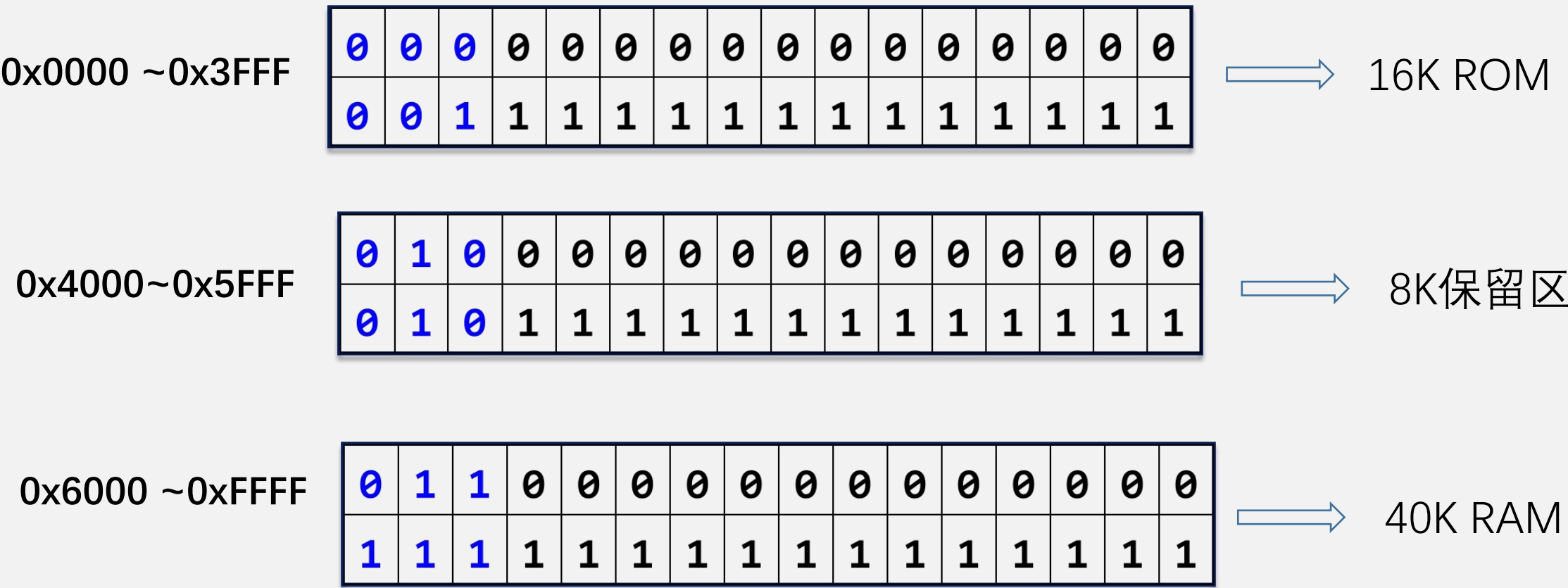
RAM的控制信号为CS#和WE#，CPU地址线 $A_{15} \sim A_0$ ，数据线 $D_7 \sim D_0$ ，控制信号有读写控制R/W#和访存请求MREQ#。

- 1)如ROM和RAM都采用8K×1芯片，画出与CPU的连接图
- 2)如ROM采用8K×8芯片，RAM芯片采用4K×8的芯片，画出与CPU的连接图
- 3)如ROM采用16K×8的芯片，RAM芯片采用8K×8的芯片，试画出与CPU连接图

7.4 主存与CPU的连接

3. 字位同时扩展

计算存储分布空间对应的存储容量：

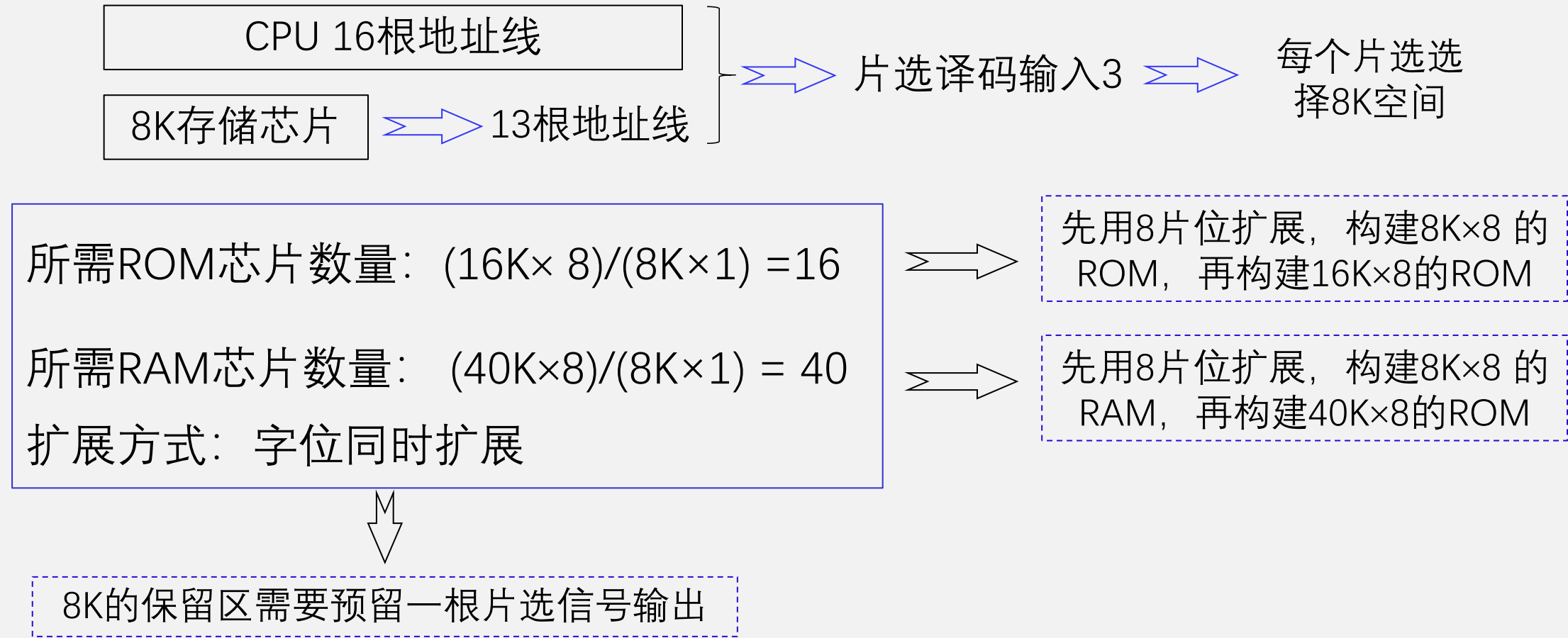




7.4 主存与CPU的连接

3. 字位同时扩展

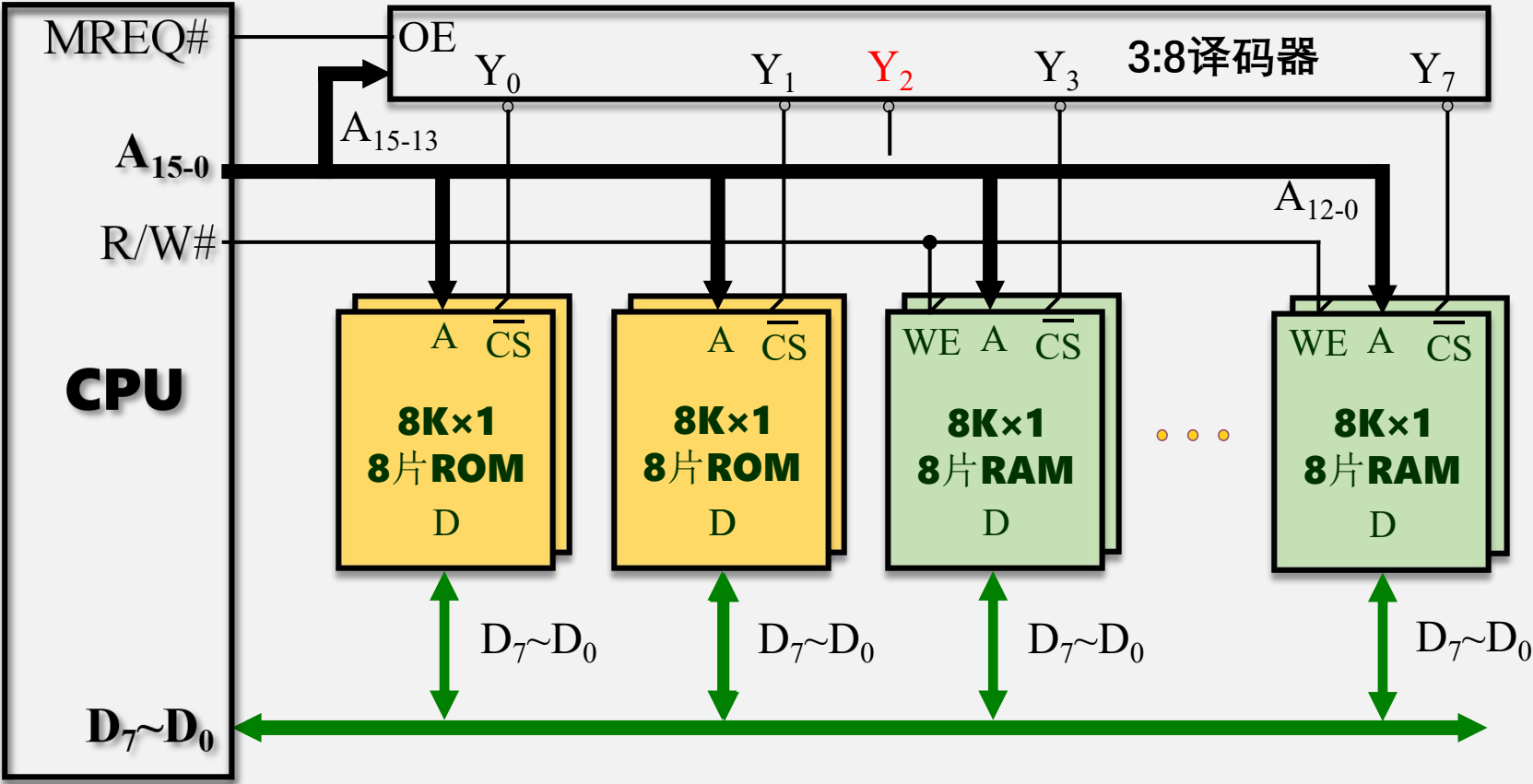
1)如ROM和RAM都采用8K×1芯片， 画出与CPU的连接图





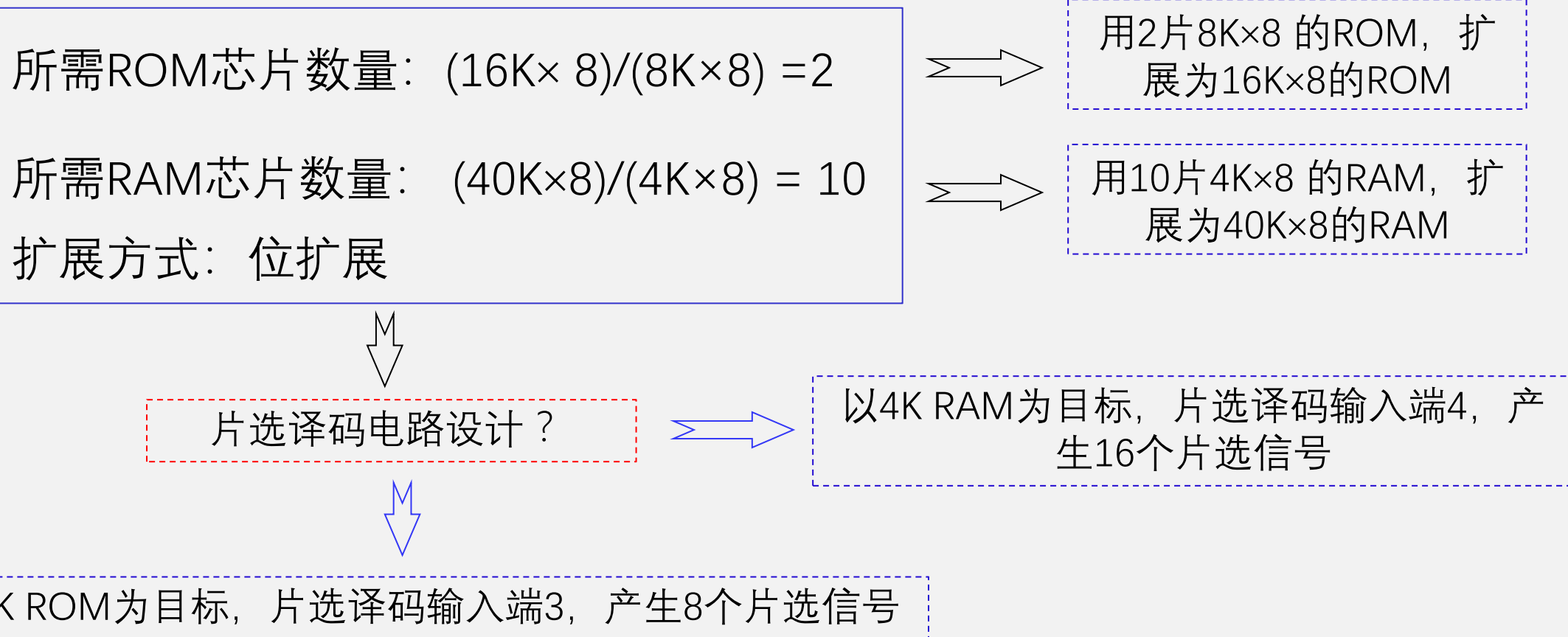
7.4 主存与CPU的连接

3. 字位同时扩展



3. 字位同时扩展

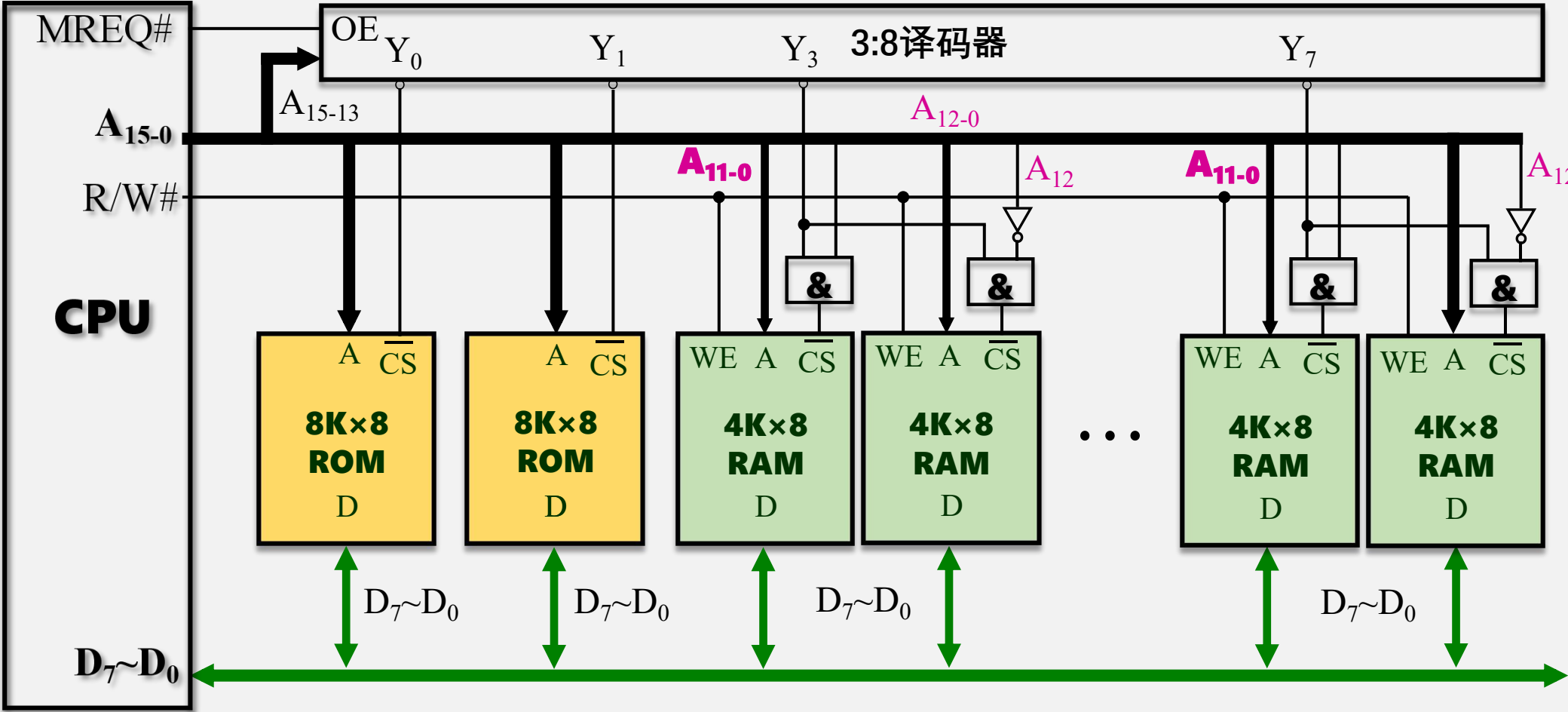
2)如ROM采用8K×8芯片， RAM芯片采用4K×8的芯片， 画出与CPU的连接图





7.4 主存与CPU的连接

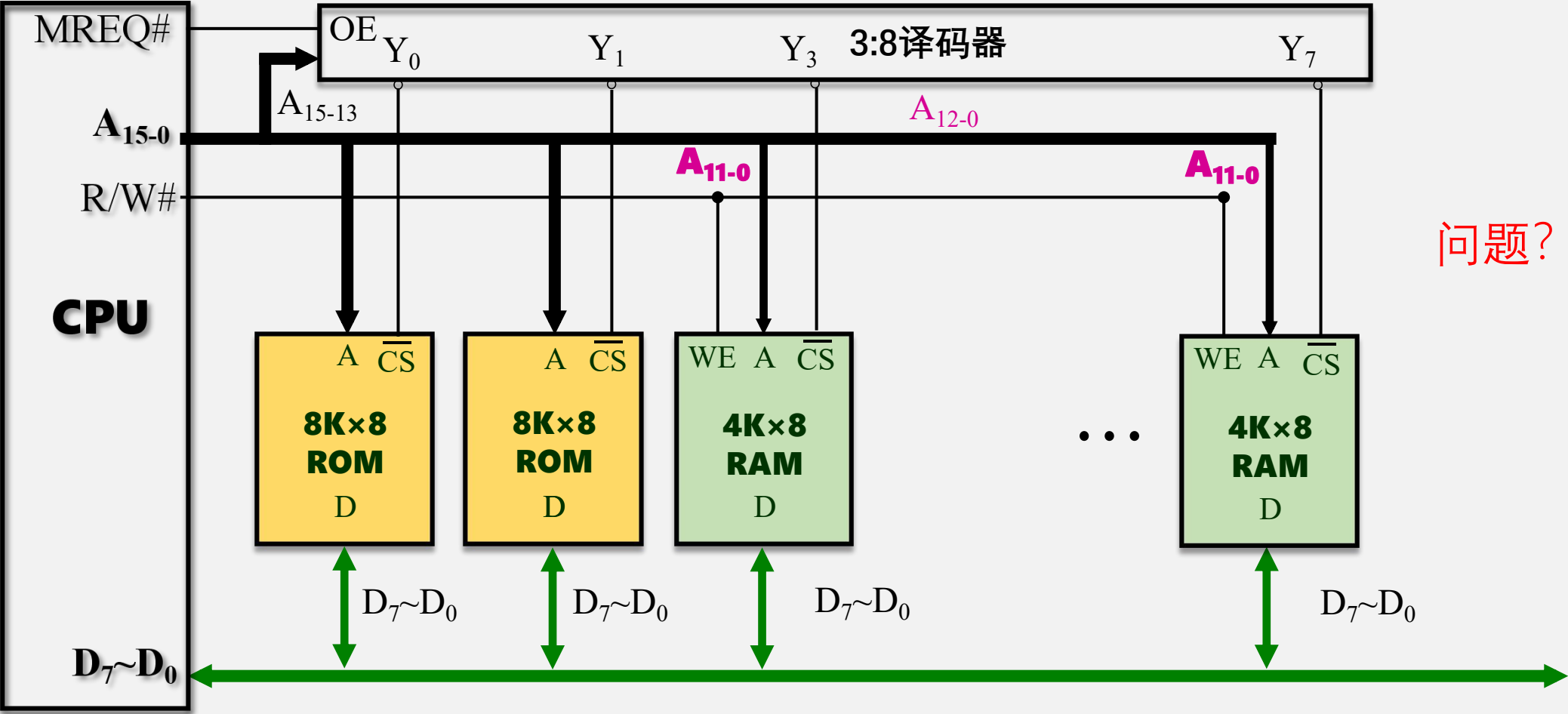
3. 字位同时扩展





7.4 主存与CPU的连接

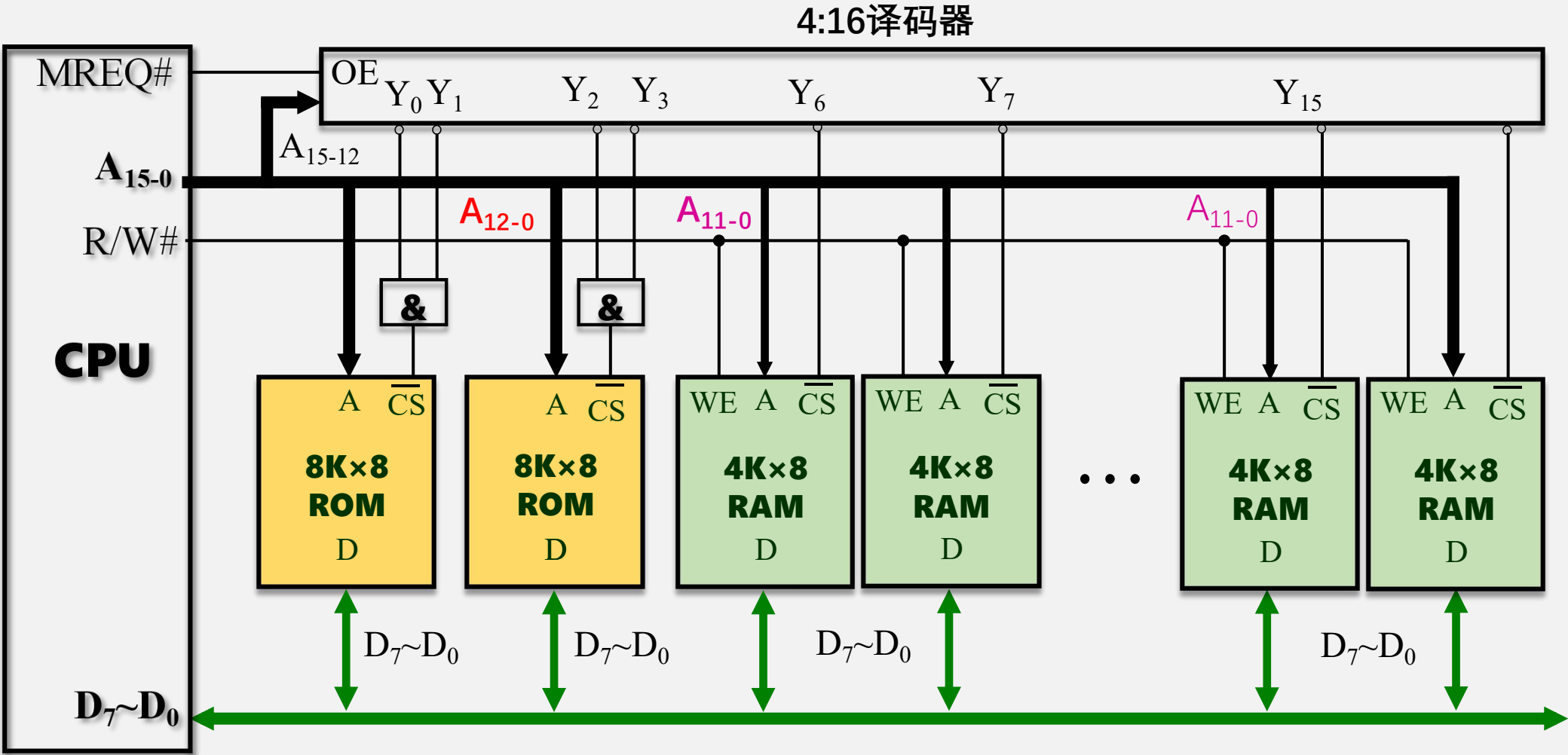
3. 字位同时扩展





7.4 主存与CPU的连接

3. 字位同时扩展



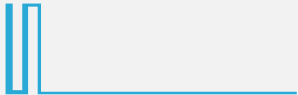


7.4 主存与CPU的连接

3. 字位同时扩展

3)如ROM采用16K×8的芯片， RAM芯片采用8K×8的芯片， 画出与CPU连接图

请参照2自己完成



第一部分完