

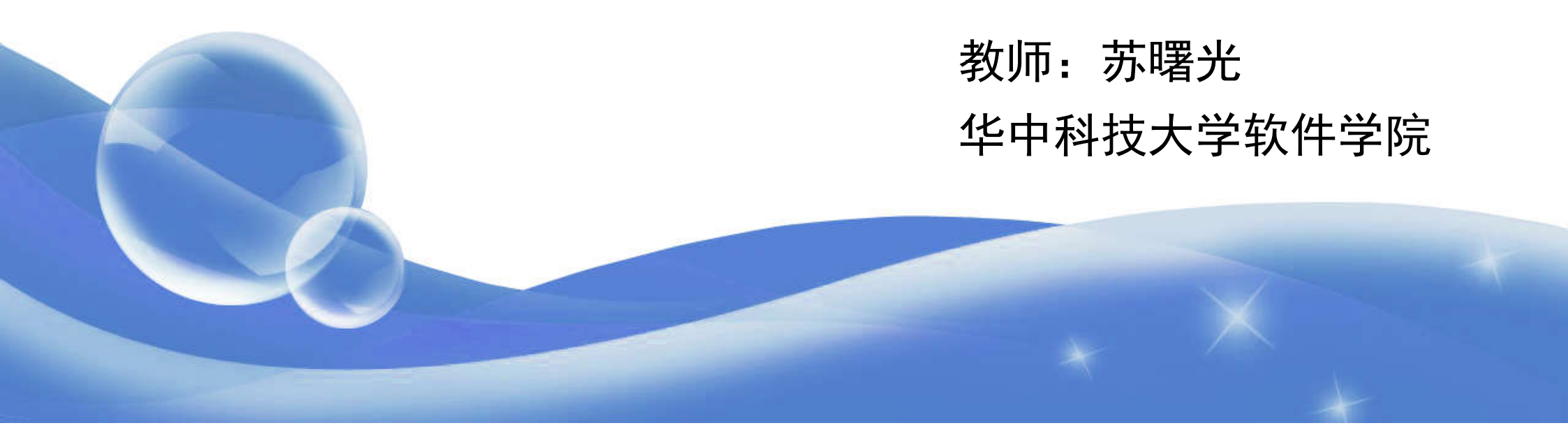


《微机原理与接口》

第4章 8088存储系统

教师：苏曙光

华中科技大学软件学院



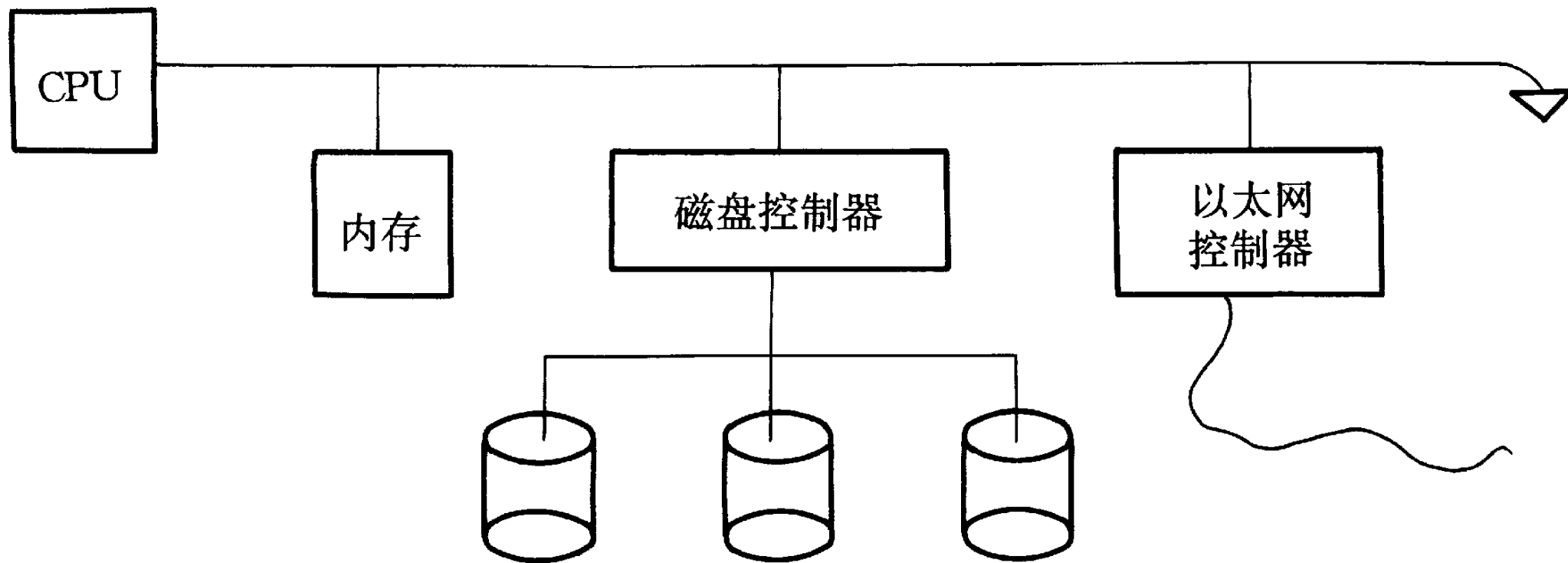
●第四章 8088存储系统

- 1. 存储器的设计
- 2. IBM PC/XT的结构
- 3. 8088最小系统设计



第1节 8088的存储器

● 存储的层次



● 存储器（Memory）

■ 存储器是计算机的**记忆设备**，用来存放程序和数据。

■ 存储介质

◆ 半导体器件和磁性材料。

◆ **最小存储单位**就是一个**双稳态半导体电路**或一个**CMOS晶体管**或**磁性材料存储元**，它可存储一位二进制代码（**BIT**）。

存储器的分类

- 按存储介质分

- 半导体存储器：用半导体器件组成的存储器。

- 磁表面存储器：用磁性材料做成的存储器。

- 按存储方式分

- 随机存储器：任何存储单元的存取时间和位置无关。

- 顺序存储器：只能按某种顺序来存取，存取时间和位置有关。

- 按存储器的读写方式

- 只读存储器(ROM)：半导体存储器。

- 随机读写存储器(RAM)：半导体存储器。

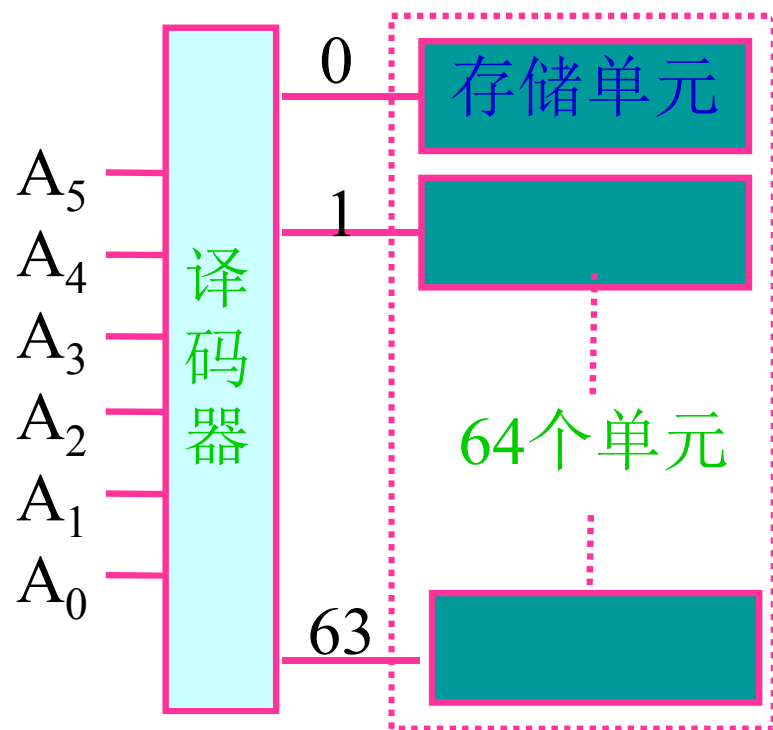
- 按信息的可保存性分

- 非永久记忆的存储器：断电后信息即消失的存储器。

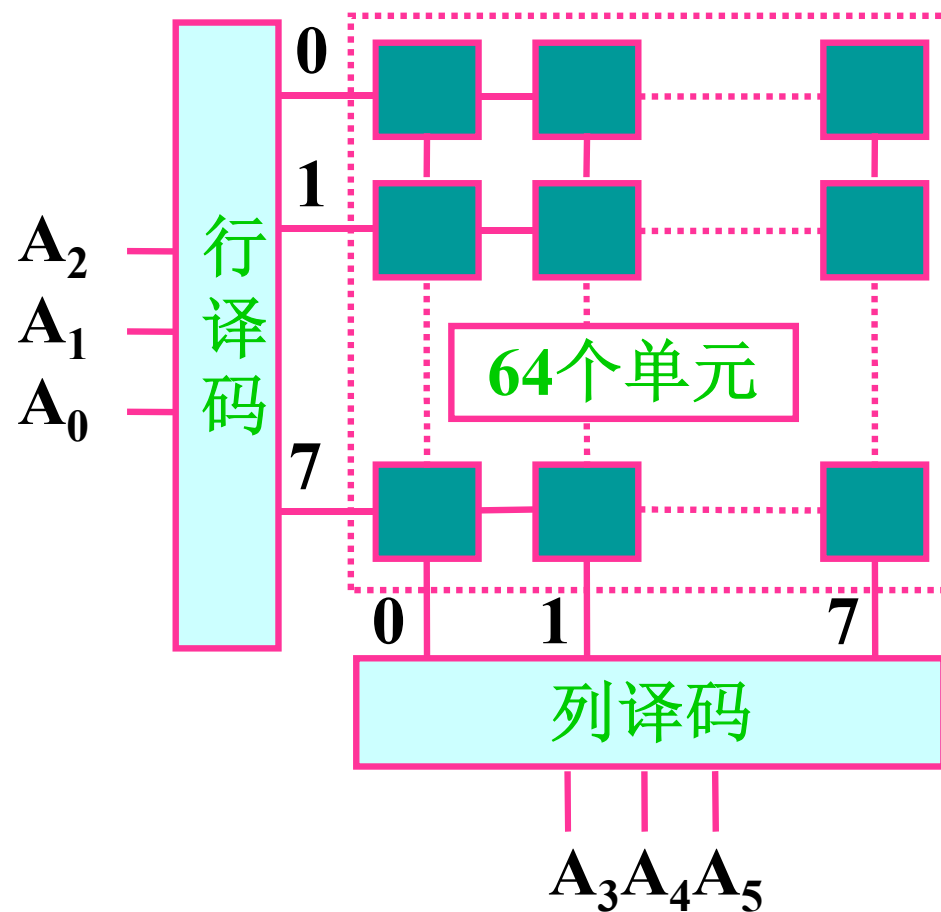
- 永久记忆性存储器：断电后仍能保存信息的存储器。

存储芯片中存储单元的两种排列方式

- 以64个存储单元为例



线性排列



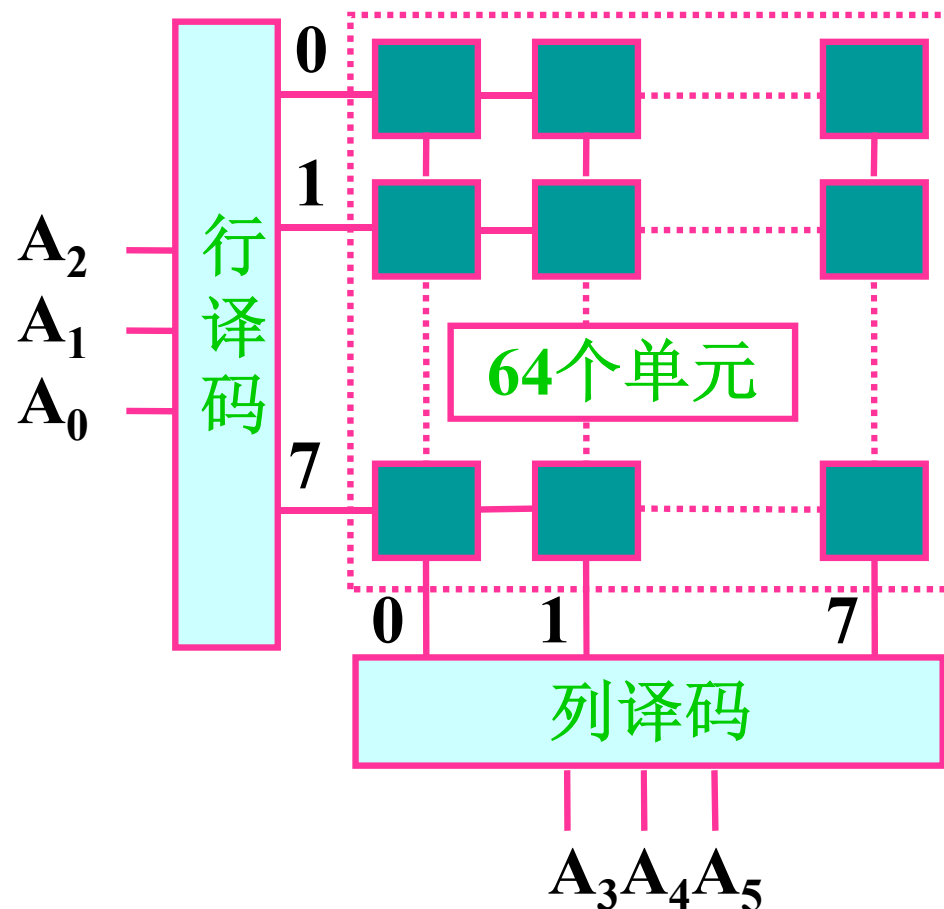
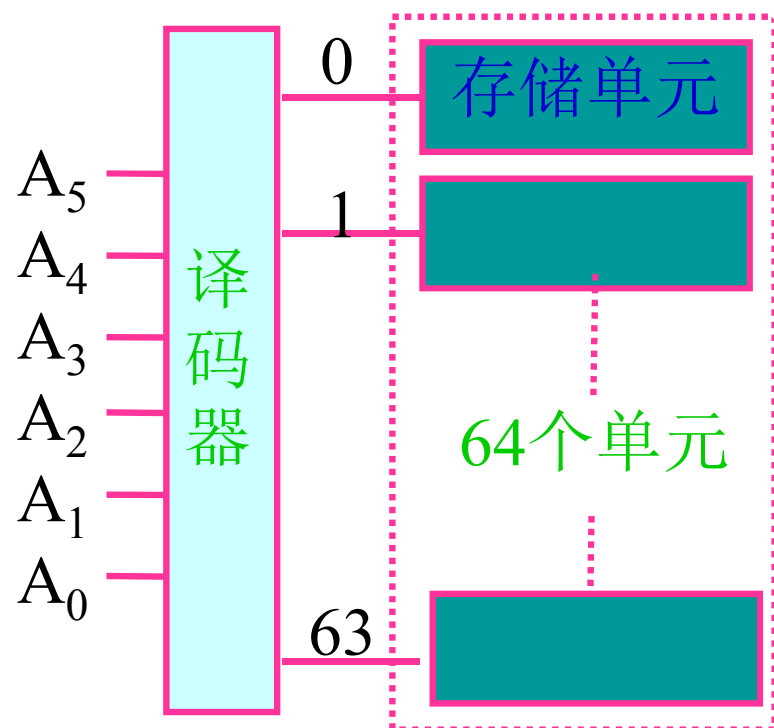
矩阵排列

存储单元的两种排列方式

- 两种地址译码方式

- 单译码方式 • 64根选择线

- 双译码方式 • 16根选择线



半导体存储器芯片的结构

- 组成:

- 存储体

- ◆ 多个存储单元的有机集合就是存储体。

- 线性排列或矩阵排列

- 外围控制电路

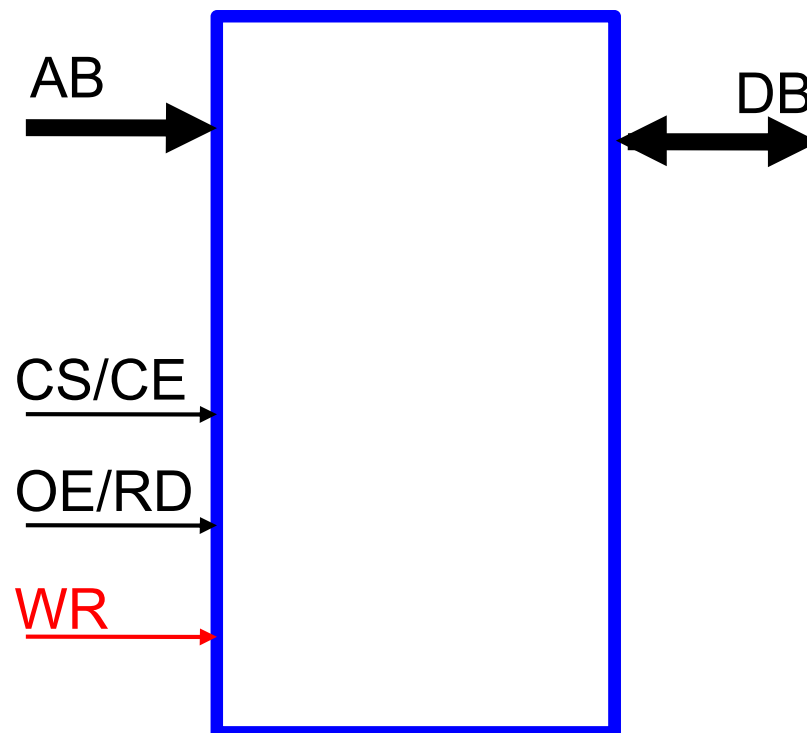
- ◆ 地址引脚（地址总线）

- ◆ 数据总线（输入和/输出总线）

- ◆ 片选端（CS/CE）

- ◆ 读控制端OE/RD

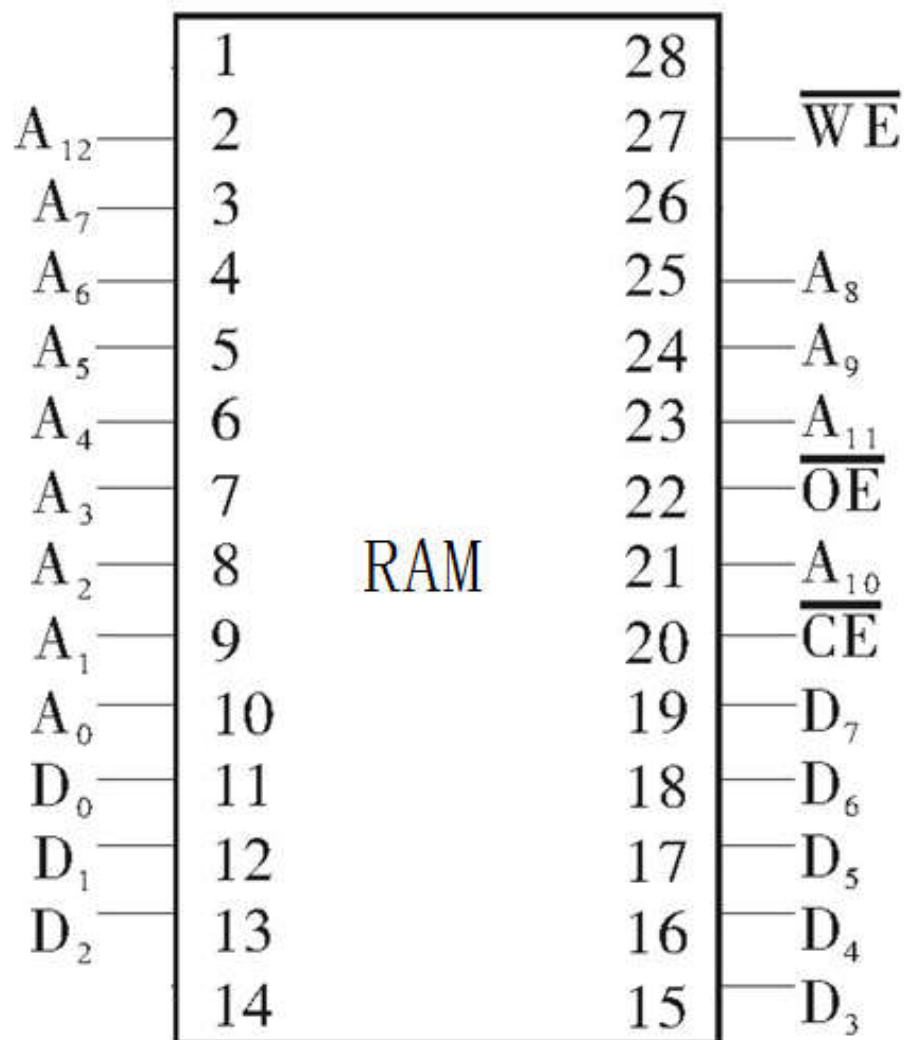
- ◆ 写控制端WR



RAM芯片的典型结构

- RAM芯片的典型结构

- 一组地址线
- 一组数据线
- 片选线
- 读线
- 写线



典型SRAM芯片

● 典型SRAM芯片

■ Intel2114\6116\6232\6264\621

◆ 6264=8KB=8K×8=2¹³ × 8

◆ 13根地址线

◆ 8根数据线

◆ 片选线、读线、写线

● 引脚名称

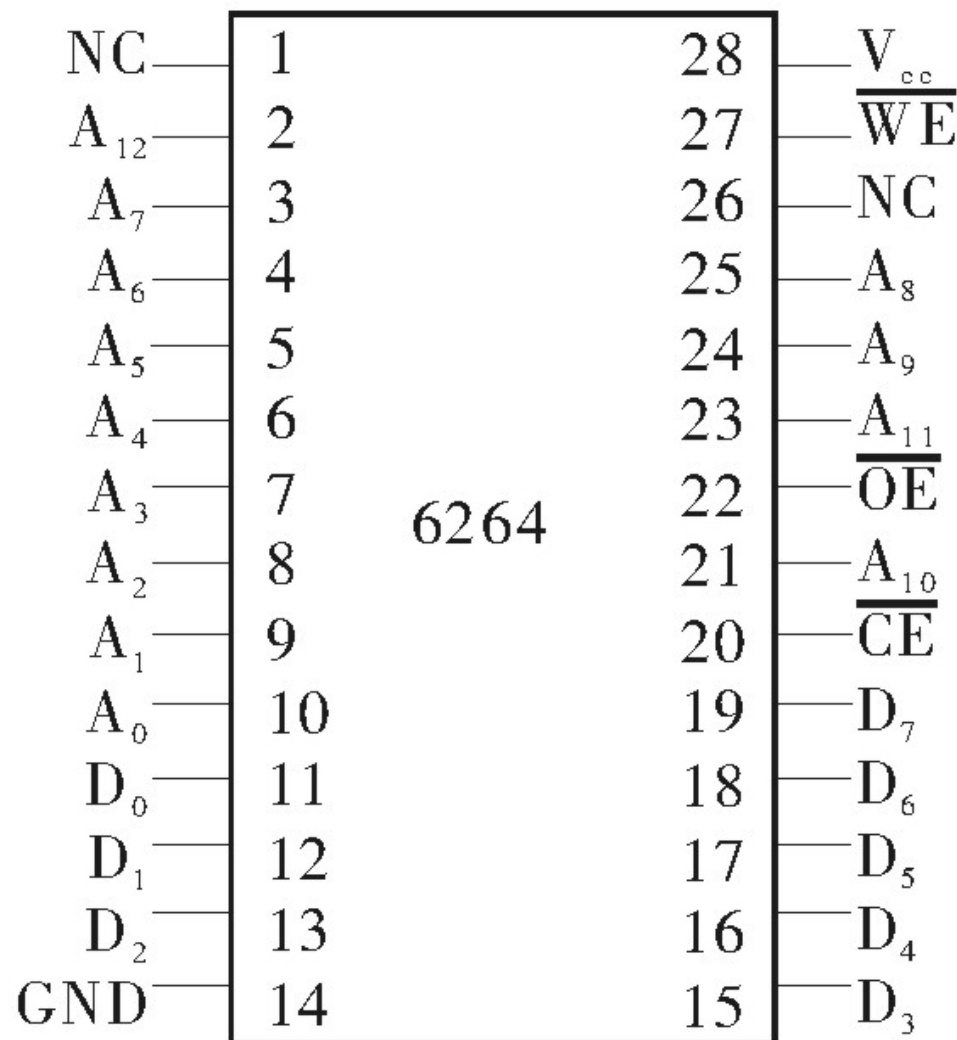
■ 地址线：A0～A12

■ 数据线：D0～ D7

■ 输出允许信号： \overline{OE}

■ 写允许信号： \overline{WE}

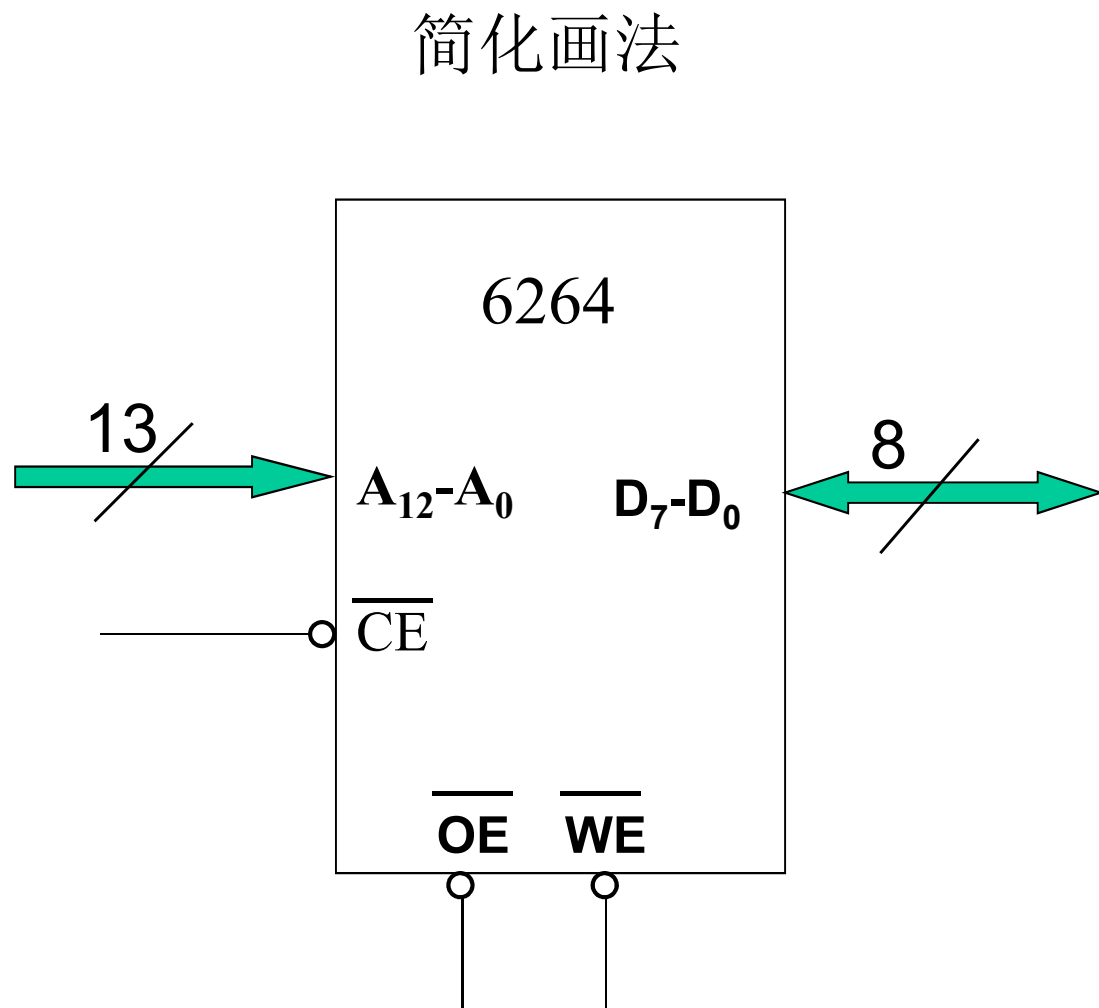
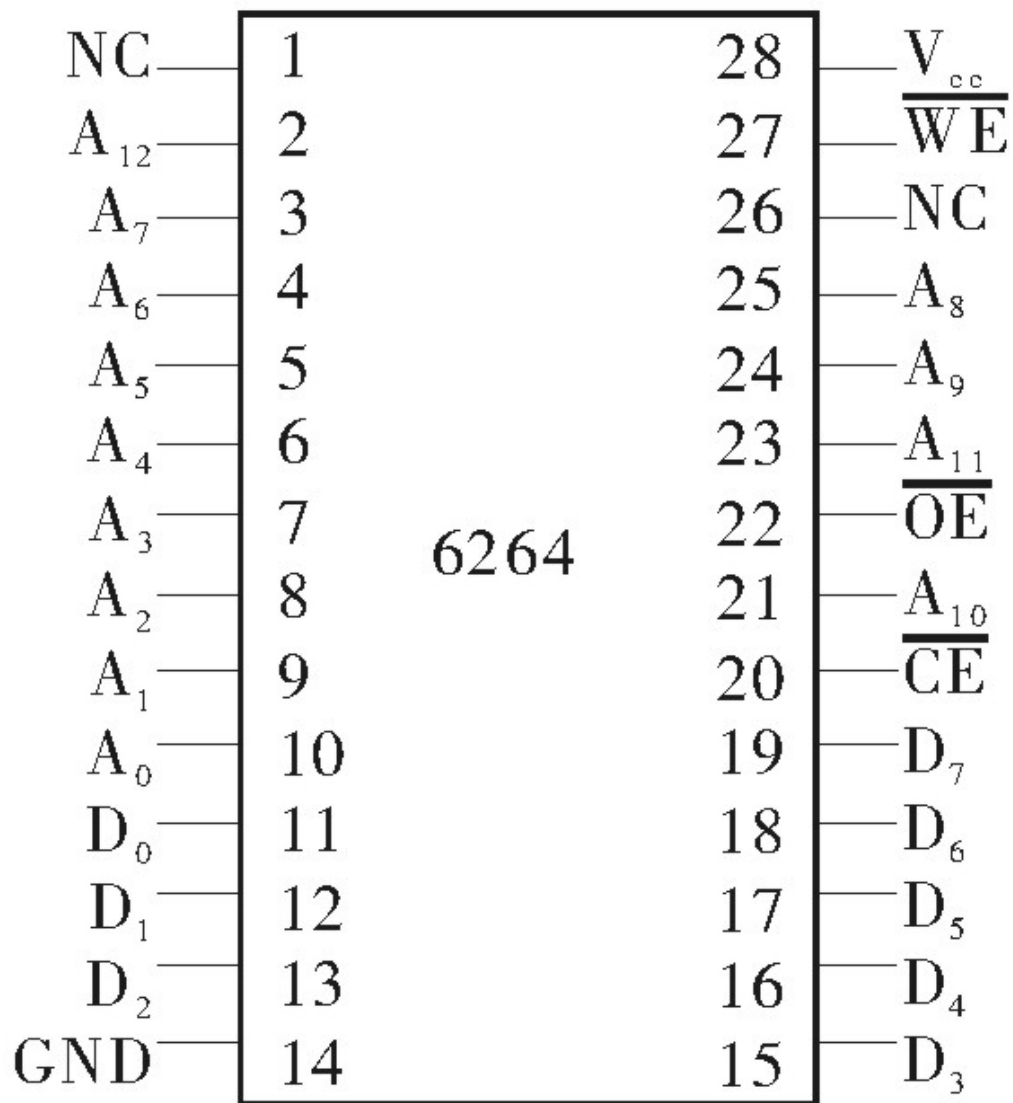
■ 选片信号： \overline{CE}



6264的引脚名称和工作过程

$\overline{\text{CE}}$	$\overline{\text{WE}}$	$\overline{\text{OE}}$	方式	操 作
0	0	0	非法	不允许 $\overline{\text{WE}}$ 与 $\overline{\text{OE}}$ 同时为低电平
0	1	0	读出	从RAM中读出数据
0	0	1	写入	将数据写入RAM中
0	1	1	选中	内部I/O三态门均处于高阻
1	×	×	未选中	内部I/O三态门均处于高阻

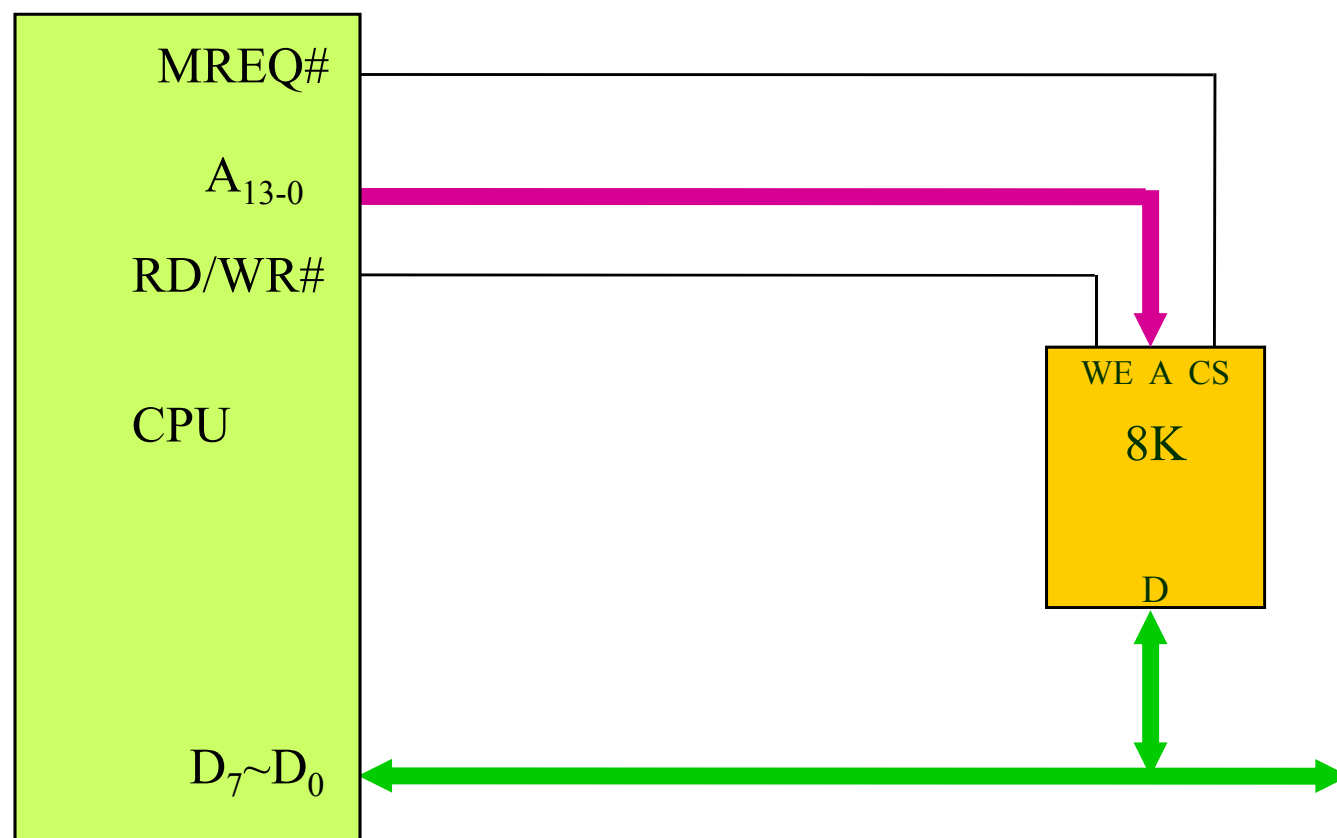
INTEL6264的封装和简化画法



● 内存与CPU的连接

- 地址线的连接
- 数据线的连接
- 控制线的连接

MREQ#: Mmemory REQUEST : MEMR 或MEMW



- 内存与CPU的连接

- 地址线的连接

- 数据线的连接

- 控制线的连接

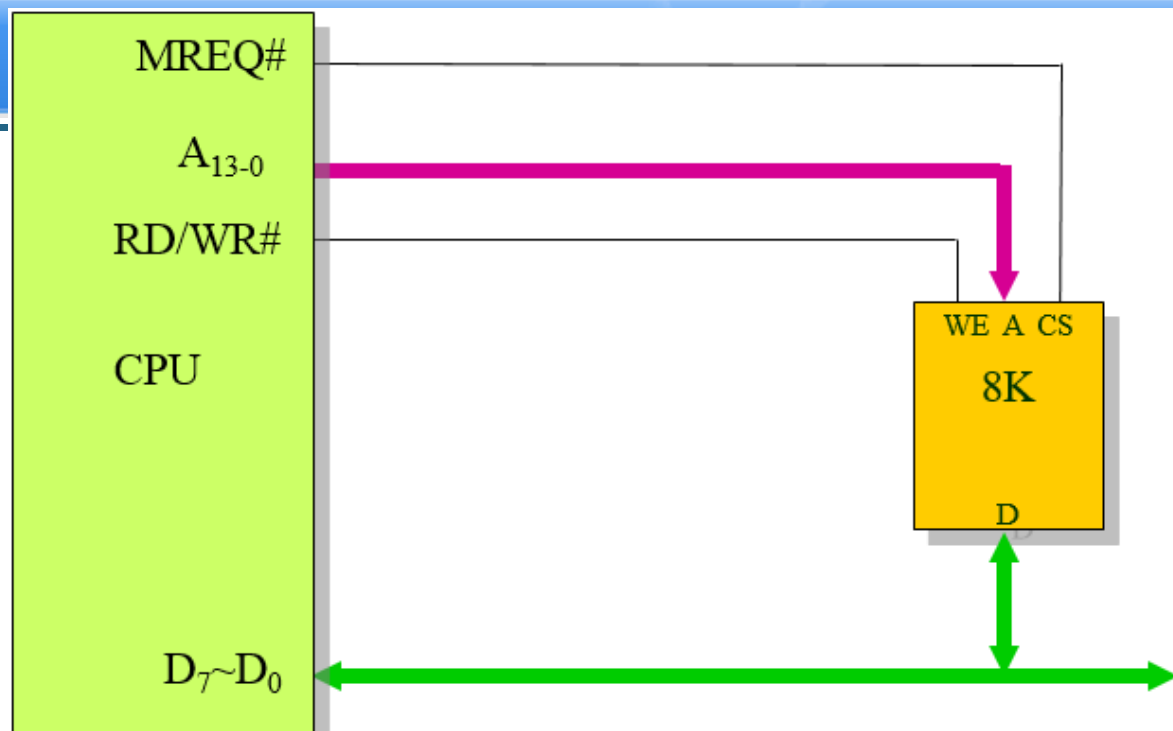
- 在连接中要考虑的问题

- CPU DB位数和存储芯片DB位数匹配. (位扩展/横向扩展)

- ◆ CPU DB位数: 即指字长

- 用户需求容量与存储芯片容量的匹配. (字扩展/纵向扩展)

- 每个存储芯片在地址空间中的范围 (片选CE的地址范围设计)



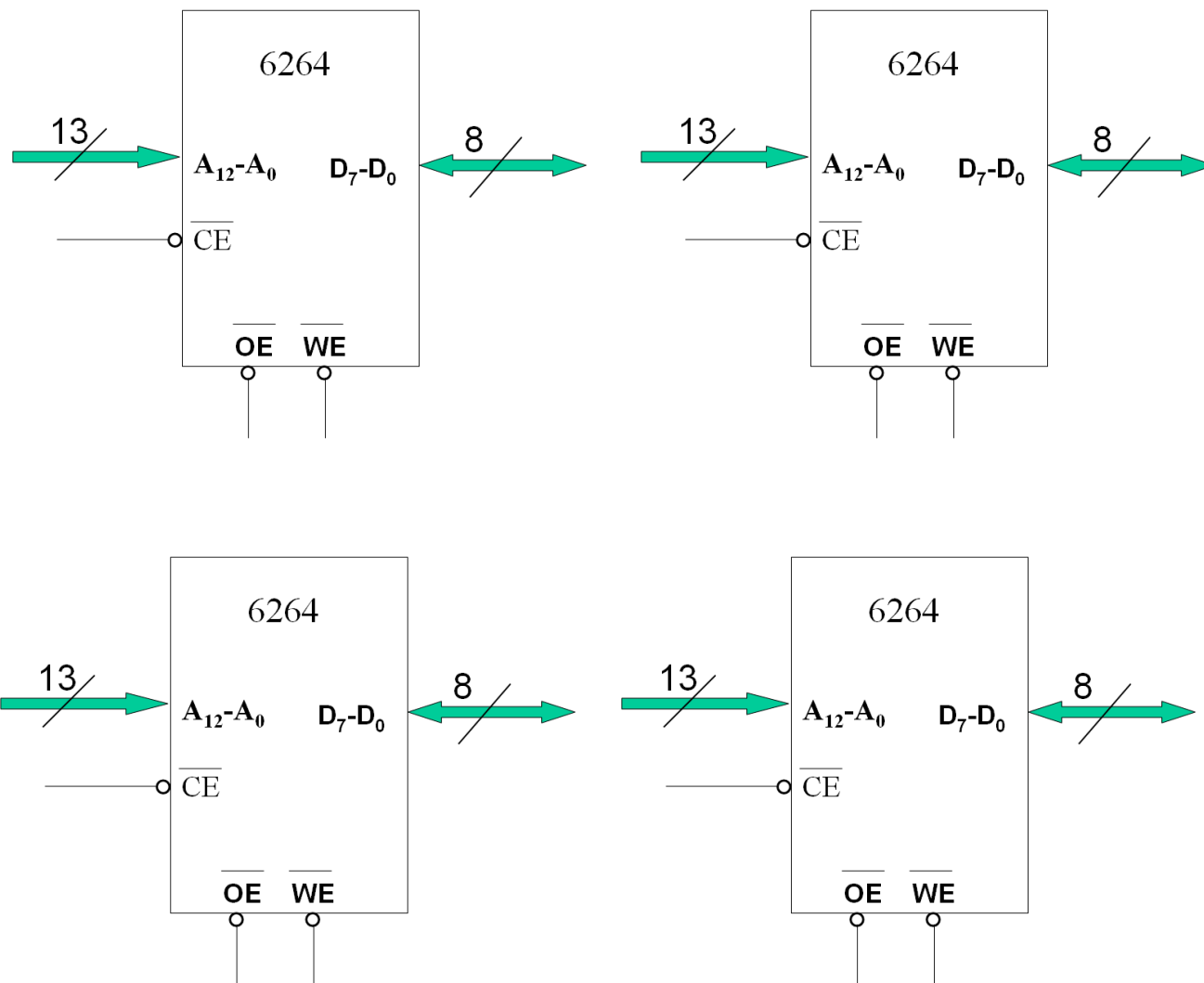
存储器系统地址译码的设计

- 存储空间的设计

- 字扩展

- 位扩展

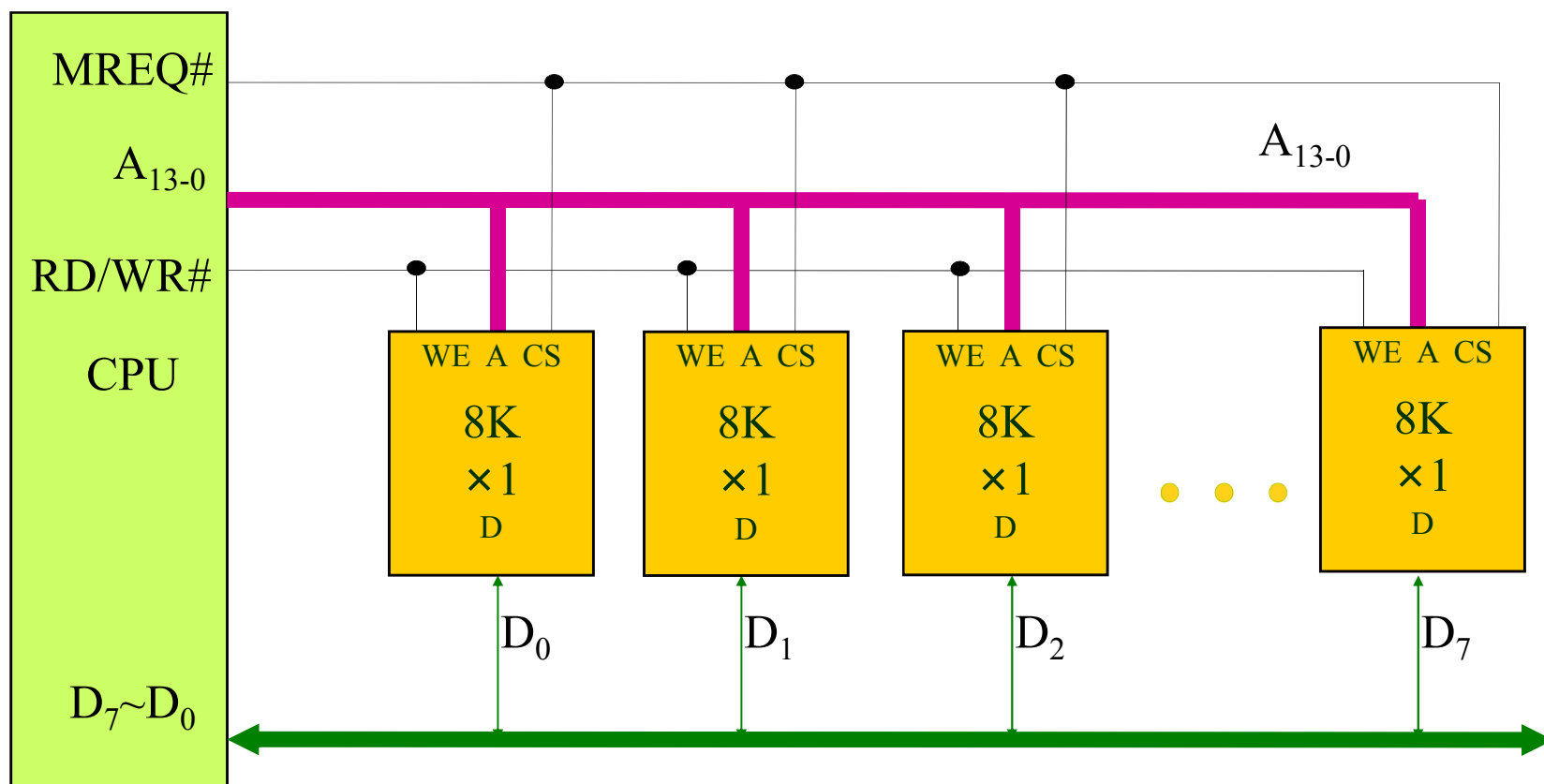
- 字扩展和位扩展



位扩展

- 存储芯片的数据总线扩展
- 各个存储芯片并行工作
- 当存储芯片数据位数 < CPU数据位数（或字长）时，进行位扩展，用多个芯片扩展起来提供一个完整的字数据。
 - 芯片数量 = CPU数据位数（或字长） / 存储芯片数据位数
- 扩展原则
 - 所有存储芯片同名的地址线、控制线、片选端相连后与CPU连接，每个存储芯片的数据总线分别连接CPU数据总线的不同位段，使得每个芯片提供同一个字的不同位段。

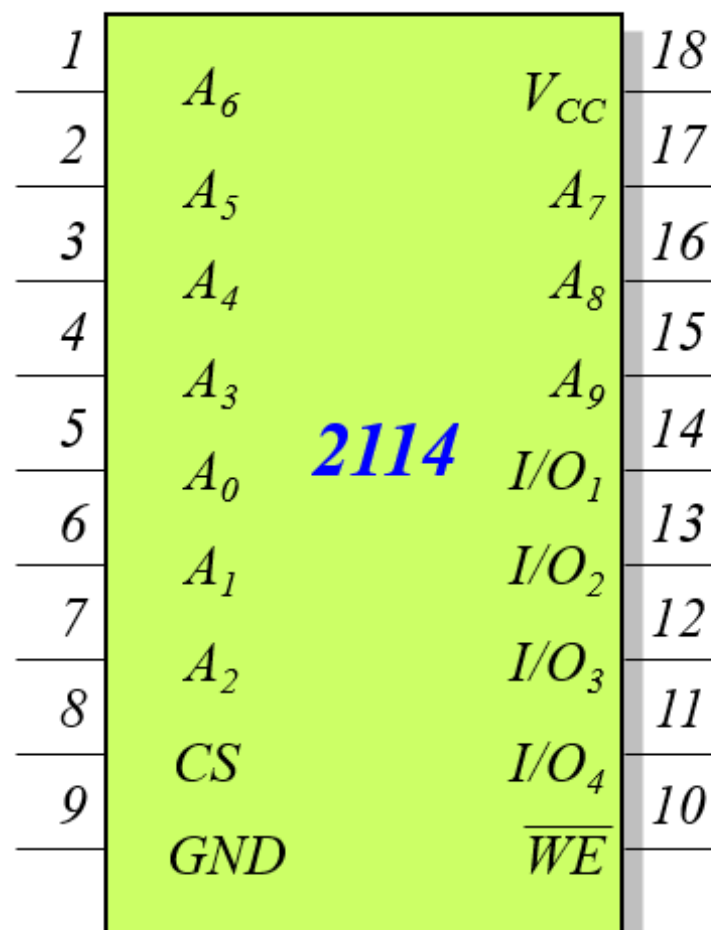
位扩展



CPU字宽 N 位，存储芯片 K 位： $K < N$ ，共需要 (N/K) 个芯片

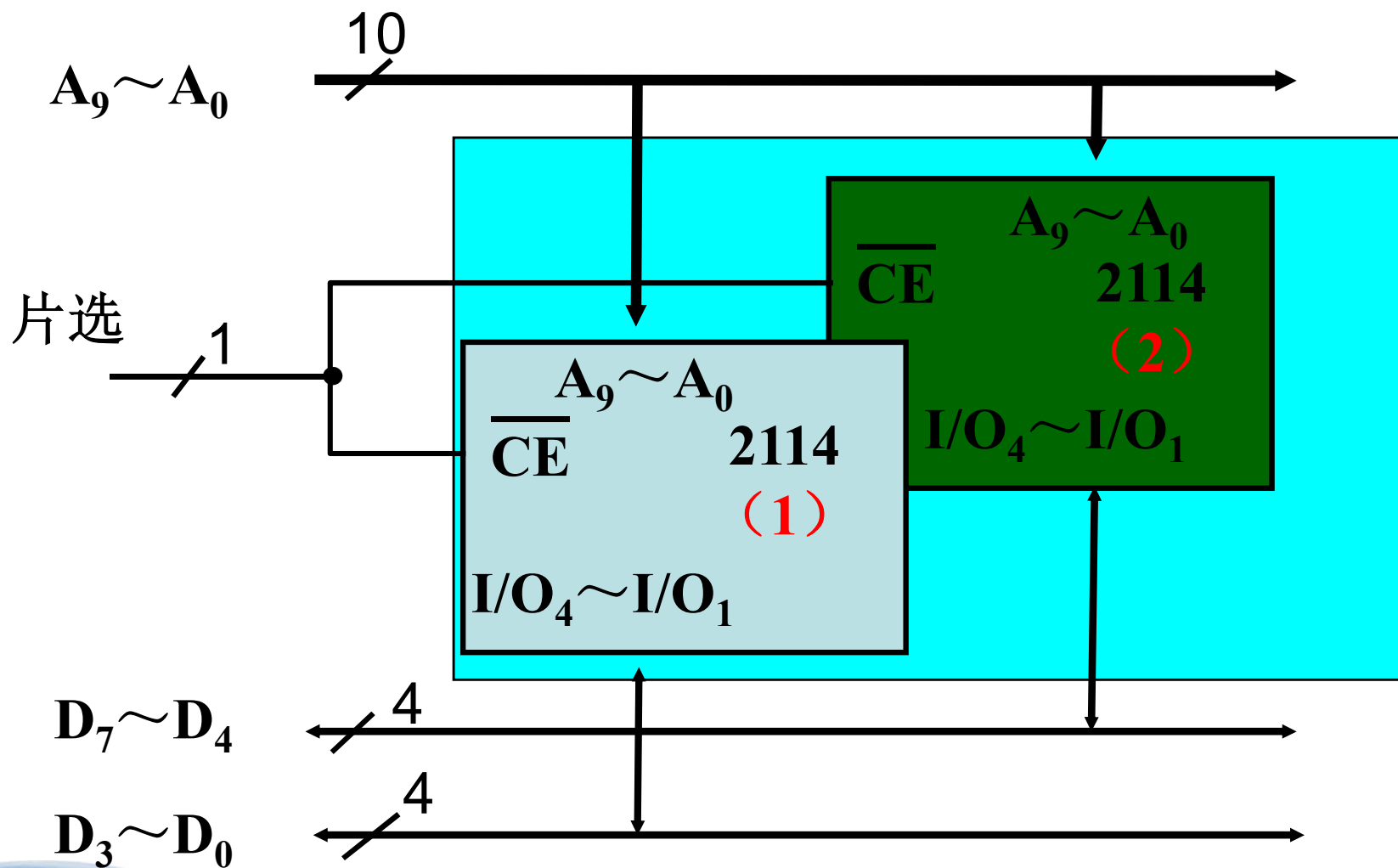
位扩展的例子

- 例: 用Intel 2114芯片 (1K x 4), 设计 1K * 8 的存储系统。



位扩展

- 用Intel2114芯片 (1K x 4), 设计一个1Kx8的存储系统



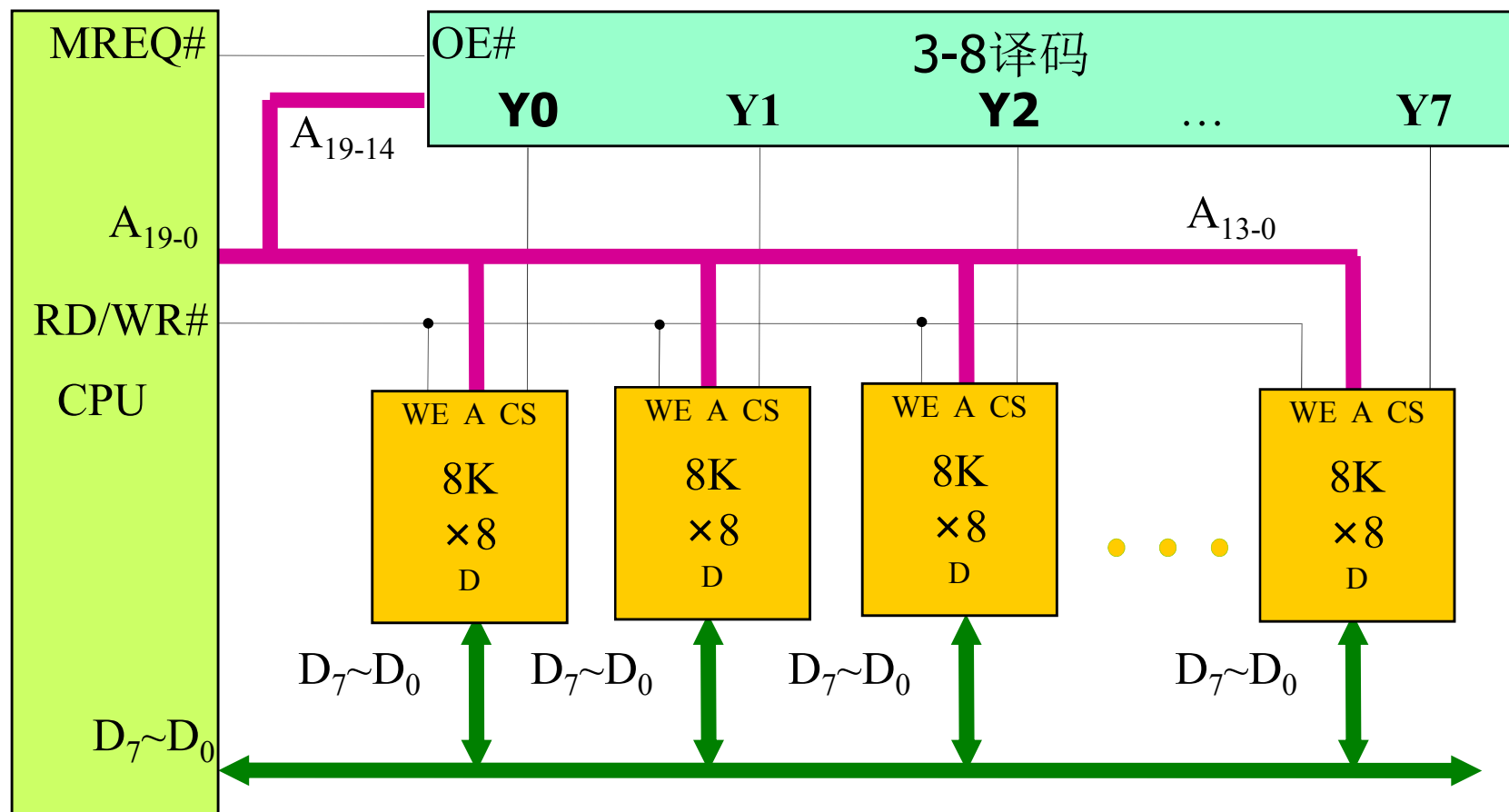
字扩展

● 功能

- 当储存芯片字数 < 用户需求容量时，进行字扩展，用多个芯片扩展起来提供一个更大容量的存储空间。
- $\text{芯片数量} = \text{用户需求容量} / \text{储存芯片字数}$

● 扩展原则

- 所有存储芯片同名的地址线、控制线、数据总线相连后与CPU连接，每个芯片的片选端分别连接地址译码器的不同输出，从而每个芯片占据存储空间的不同子空间。
 - ◆ CPU地址总线中剩余高位连接地址译码器的输入



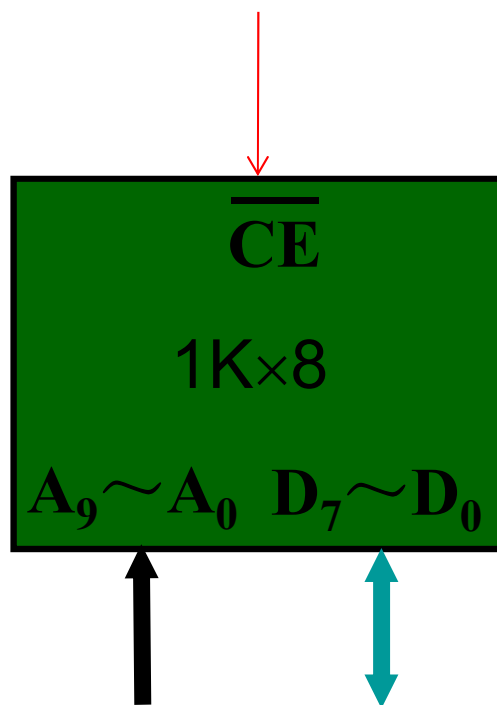
用户需求容量 M ，存储芯片容量 K ： $K < M$ ，共需 (M/K) 个芯片



字扩展的例子

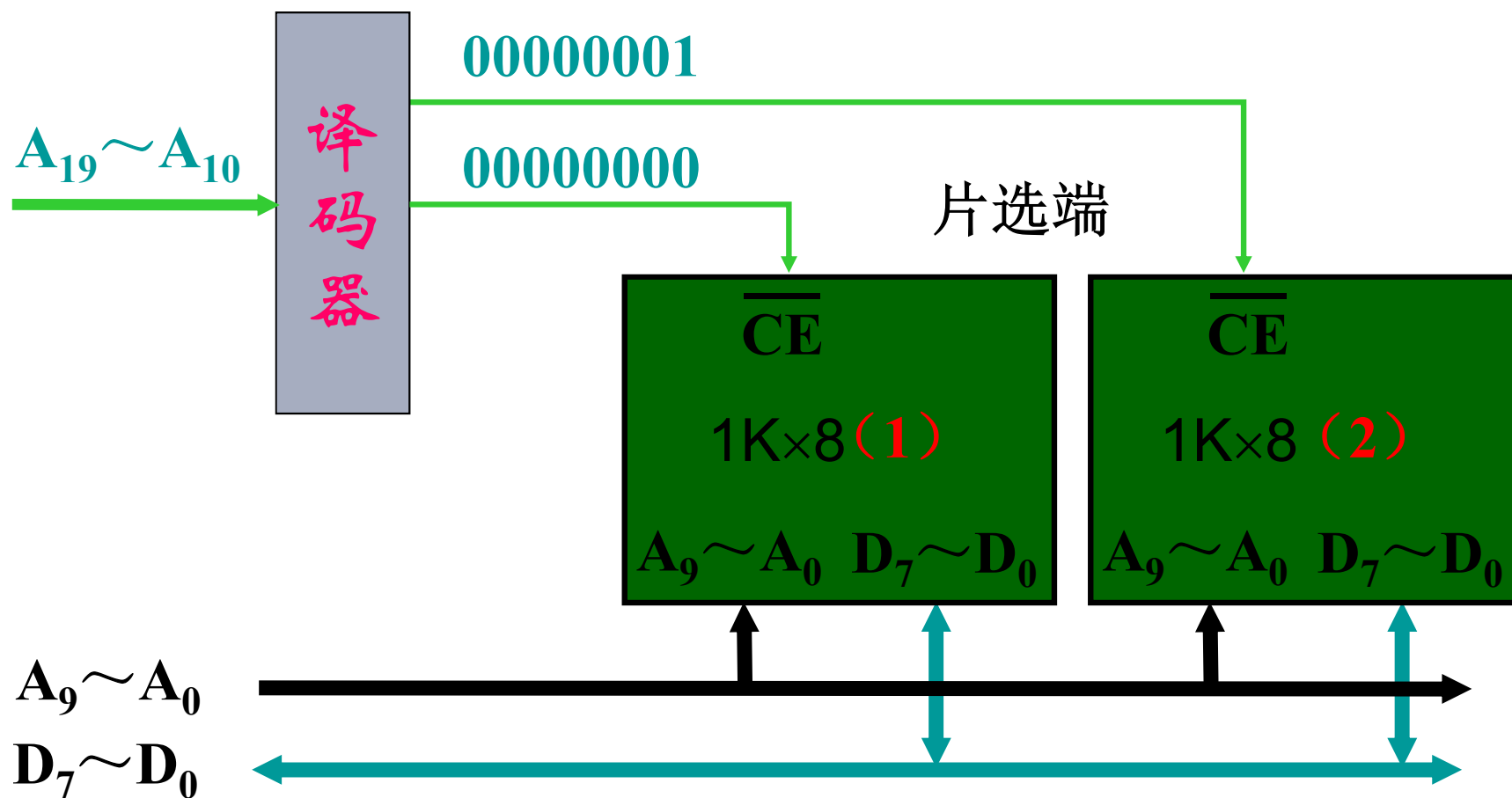
- 例子

- 用两片 $1\text{K} \times 8$ 位SRAM芯片构成 $2\text{K} \times 8$ 位的存储器



字扩展的例子

- 用两片 $1\text{K} \times 8$ 位SRAM芯片构成 $2\text{K} \times 8$ 位的存储器



字/位扩展

- 适用条件:

- 字向和位向均不满足用户要求时进行字/位扩展。

- 方法:

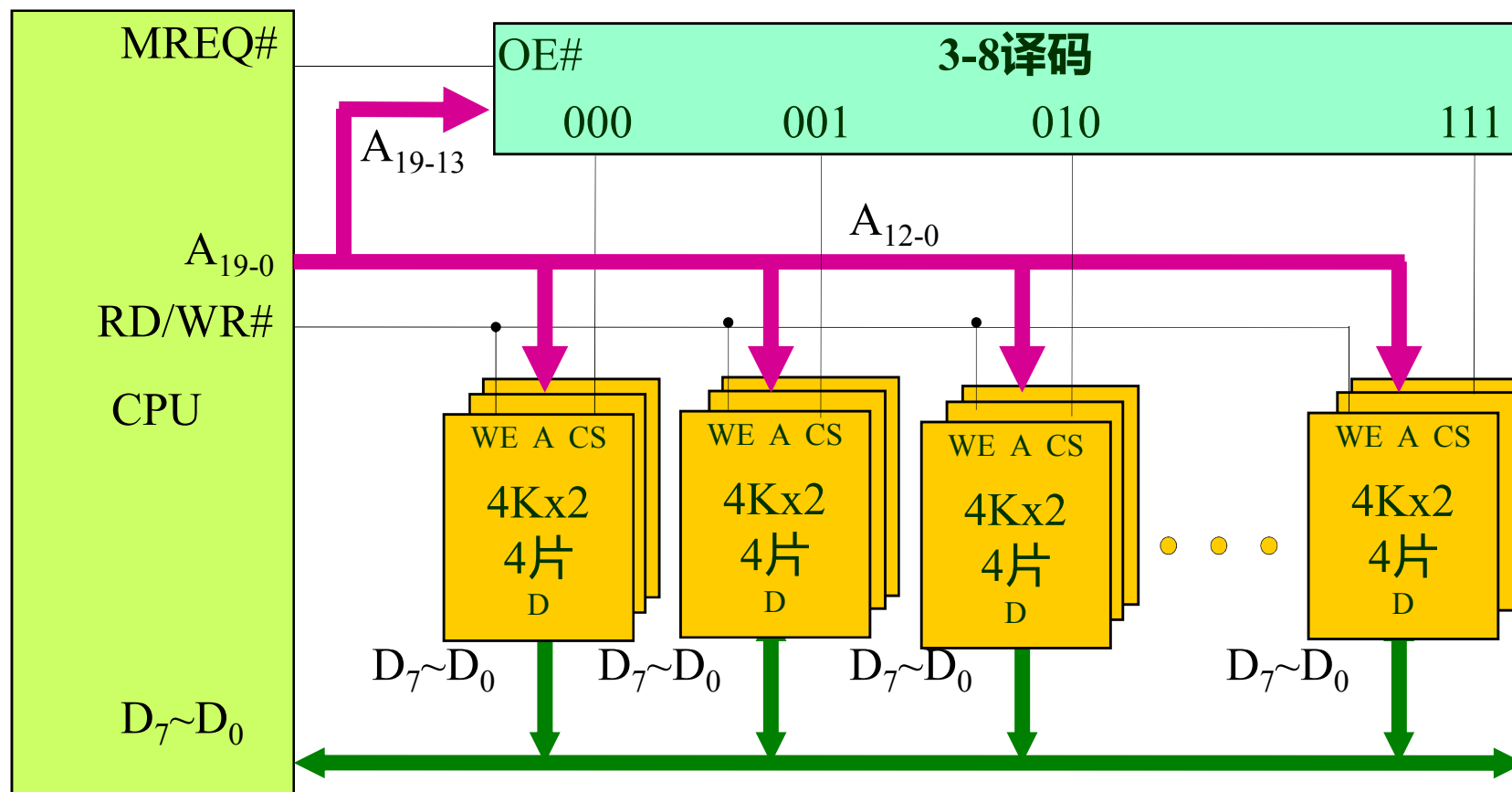
- 用户需求 M 字 \times N 位 且使用 L 字 \times K 位的芯片

- ◆ 需要 $(M / L) \times (N / K)$ 个存储芯片。

- 存储芯片分成: M / L 组, 每组 N / K 片

- ◆ 组内采用位扩展连接 (不同芯片的数据线分开连接)

- ◆ 组间采用字扩展连接 (不同组的片选线分开连接)



用户MxN位，芯片mxn位：需要 $(M/m) \times (N/n)$ 个芯片

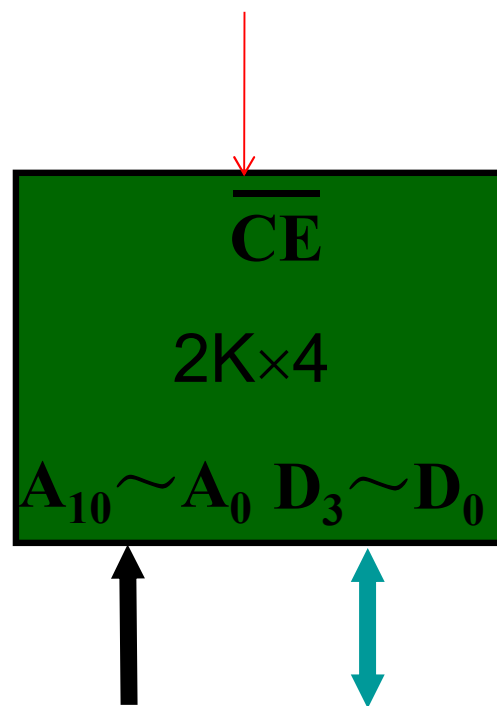
字位扩展的例子（课堂作业）

● 例3：用 $2K \times 4$ 位芯片组成 $8K \times 8$ 位的RAM存储器。

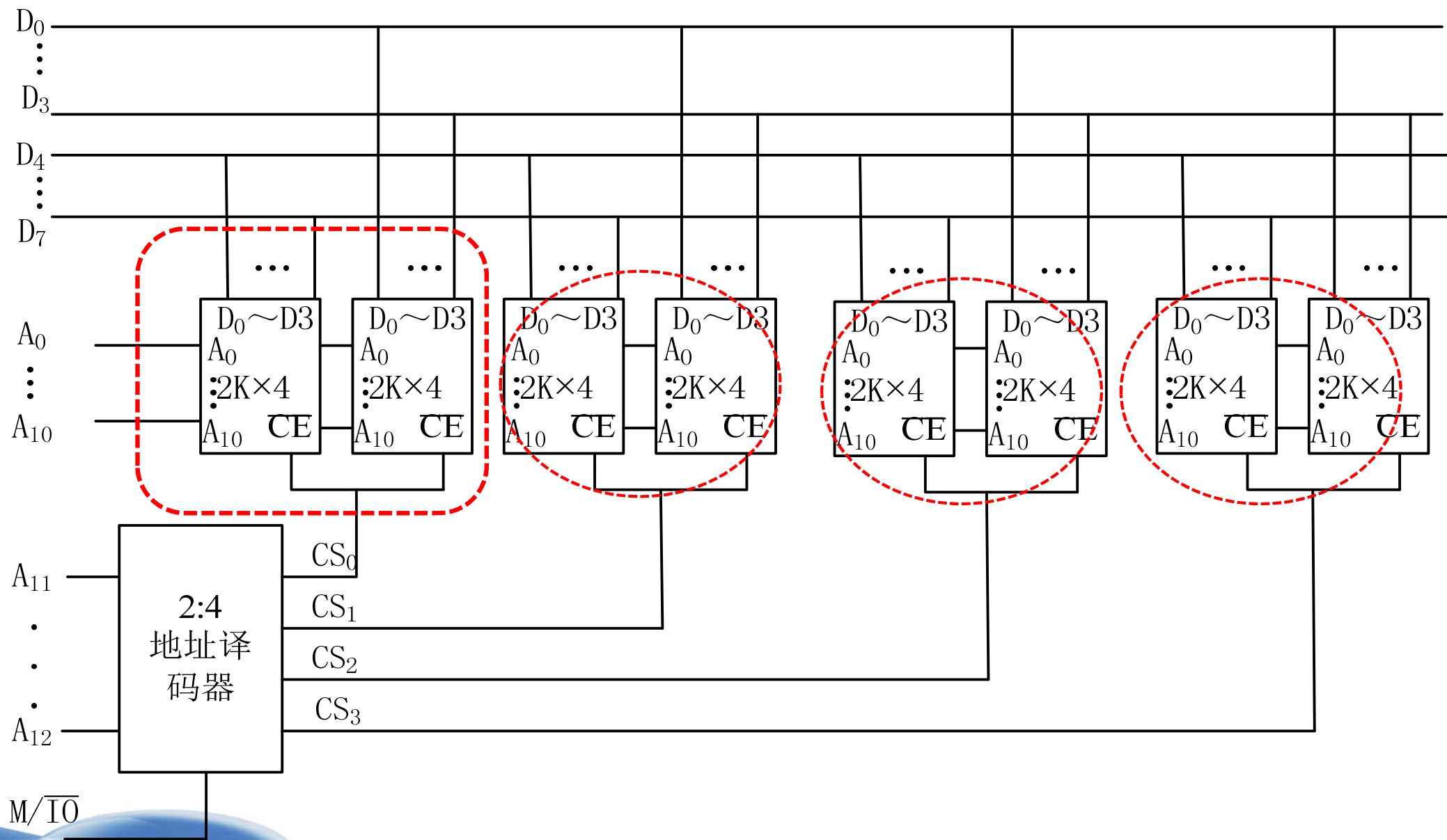
■ 需要芯片数量： $8K/2K \times 8\text{位}/4\text{位} = 8\text{片}$

■ 分成： $8K/2K = 4\text{组}$

■ 每组片数： $8\text{位}/4\text{位} = 2\text{片}$



字位扩展的例子：8片=4组X2片/组



内存单元地址译码的过程

- CPU对存储单元的访问过程

- 假如CPU AB = 20位，且存储芯片容量= 2^n

- **第一步：片选：**选择存储芯片。

- ◆ 由CPU地址线的高若干位得到片选信号。

- ◆ 用**地址高位**实现片选（**片选译码**）（用户设计）

- 高位位数 = $A_n, A_{n+1}, \dots, A_{19}$

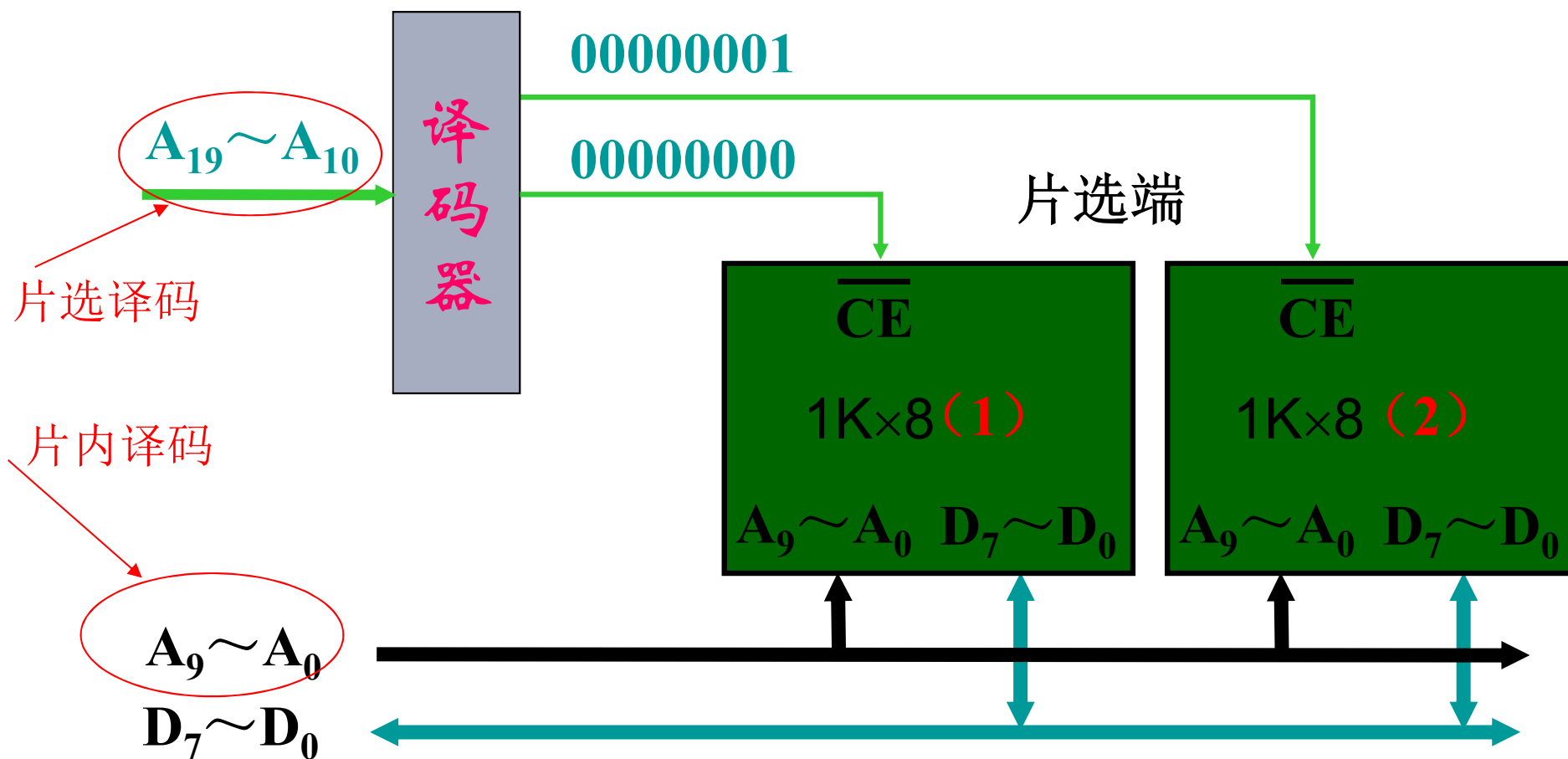
- **第二步：字选：**在存储芯片内选择存储单元。

- ◆ 由CPU地址总线低n位选择。

- ◆ 用**地址低位**实现字选（**片内译码**）（芯片实现）

字扩展的例子

- 用两片 $1\text{K} \times 8$ 位SRAM芯片构成 $2\text{K} \times 8$ 位的存储器



片选译码方法（片选译码电路的设计）

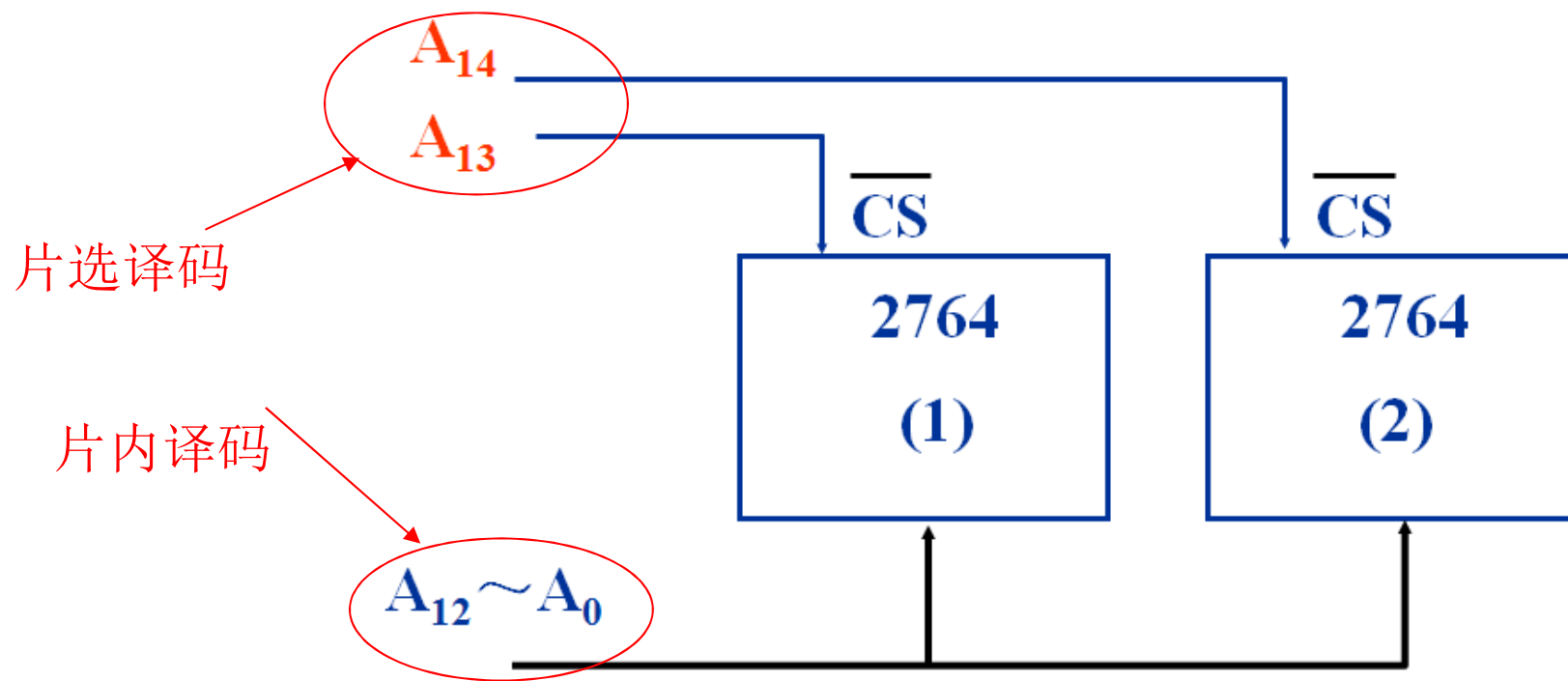
- 存储系统片选信号的产生方法
 - 线选法
 - 全译码法
 - 部分译码法



线选法

- 方法:

- 用高位地址线直接（/反向）连存储芯片片选信号。



- 优点

- 电路简单，不需外加额外电路。

- 缺点

- 存储芯片地址空间不连续。

- 存储单元具有多个地址（地址重叠）。

- 适用场合

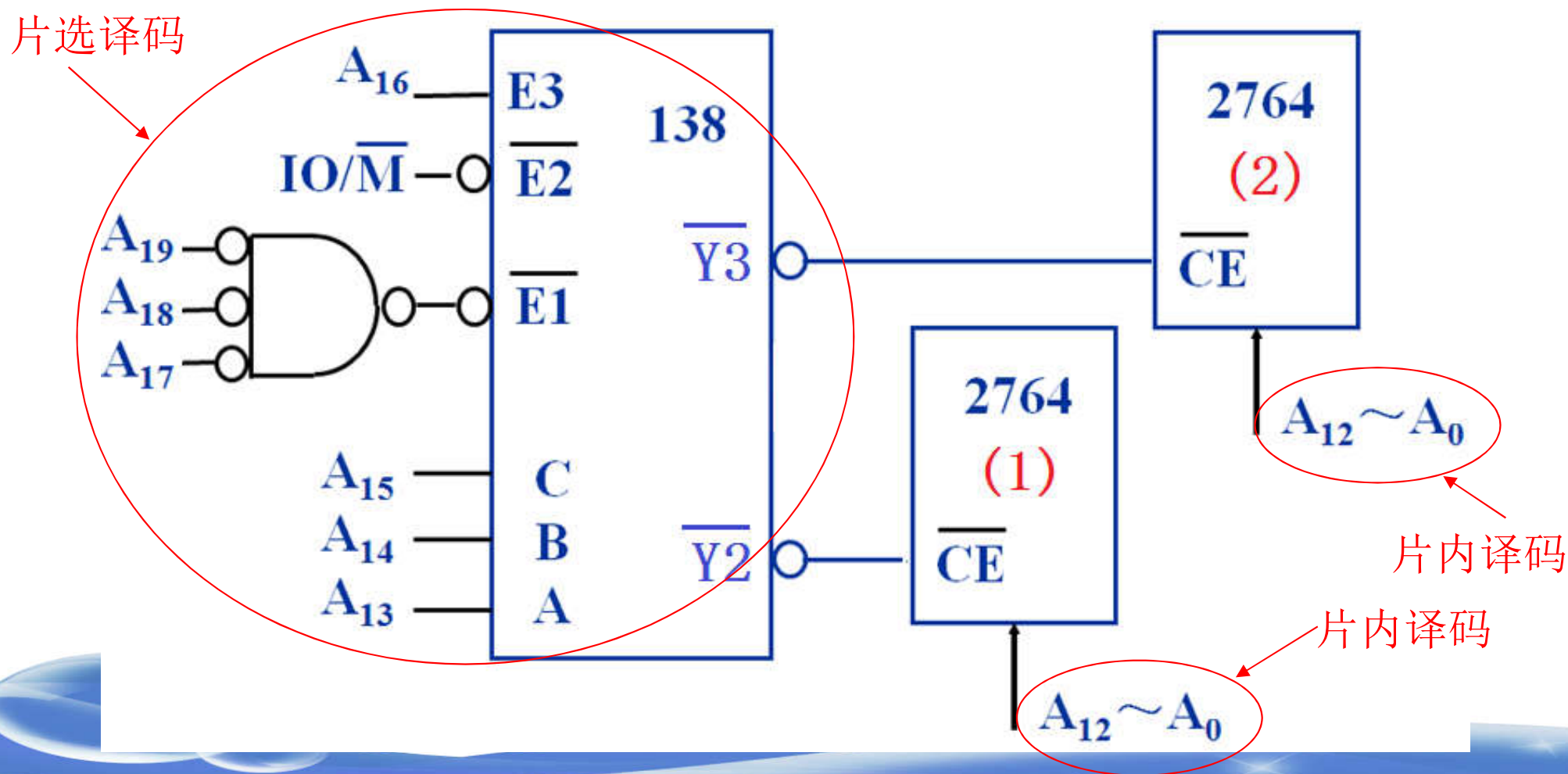
- 适用于存储容量较小的简单

- 不需要扩充内存空间的系统。

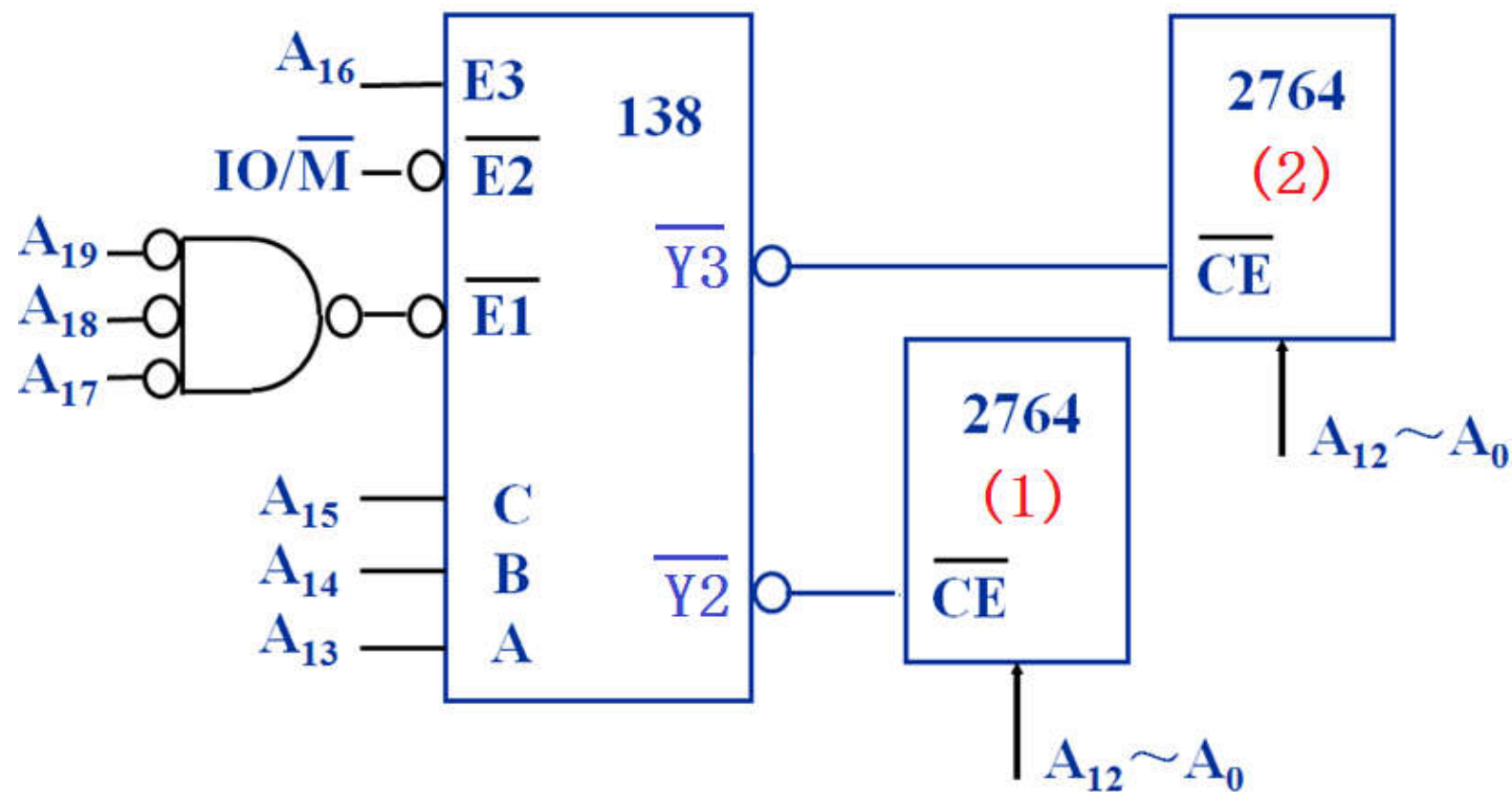
全译码法

● 方法:

- 将地址总线全部高位线接到片选地址译码器的输入端参加译码，输出信号连接各芯片的片选信号。



全译码法



$A_{19}A_{18}A_{17}A_{16}$ $A_{15}A_{14}A_{13}$ X XXXX XXXX XXXX

芯片1地址:	0001	0100	0000	0000	0000
	0001	0100	1111	1111	1111
芯片2地址:	0001	0110	0000	0000	0000
	0001	0110	1111	1111	1111

- 优点:

- 存储单元地址唯一

- 缺点:

- 对译码电路要求较高。

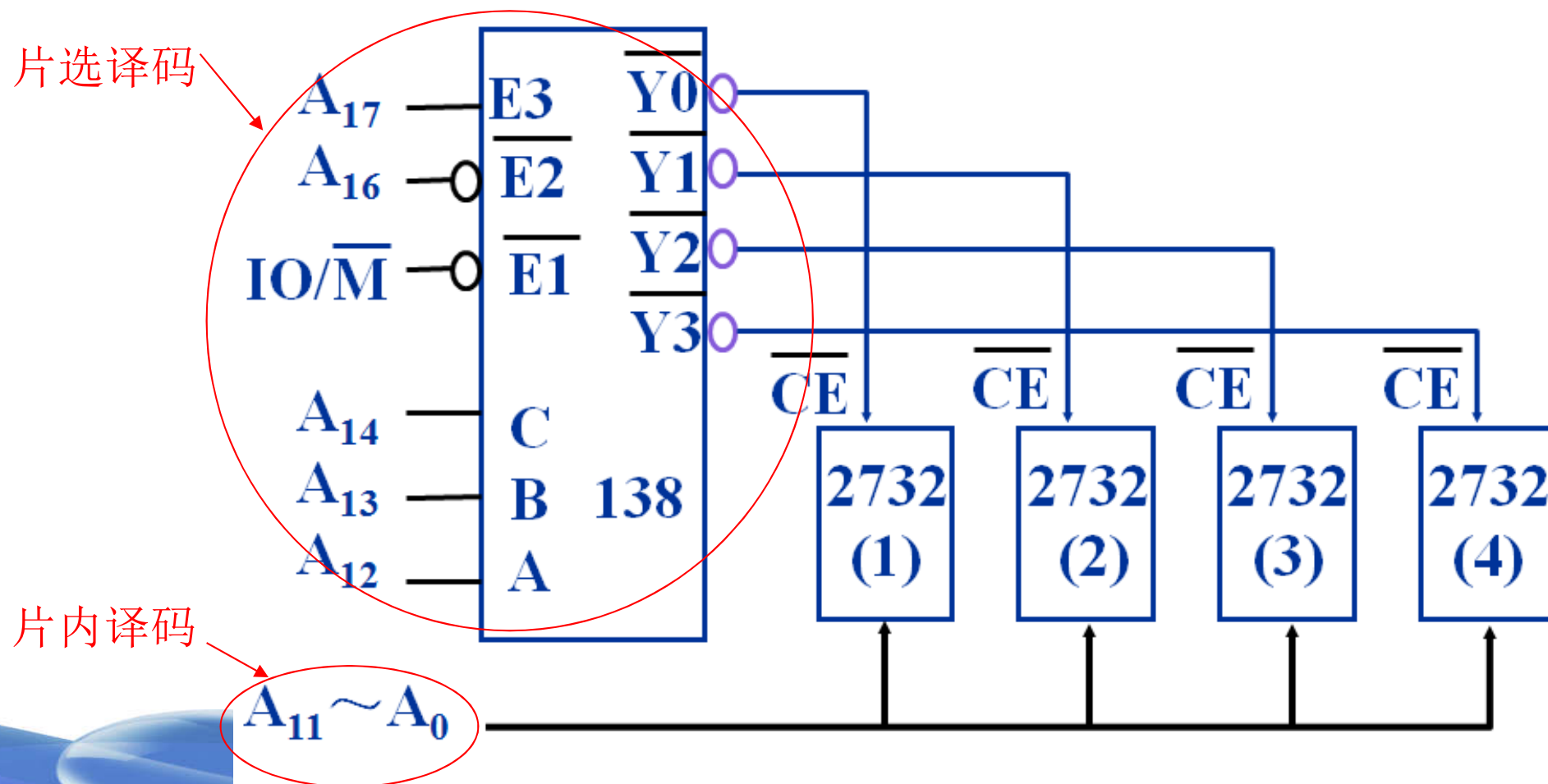
- 适用场合

- 适用于存储器芯片较多的系统。

部分译码法

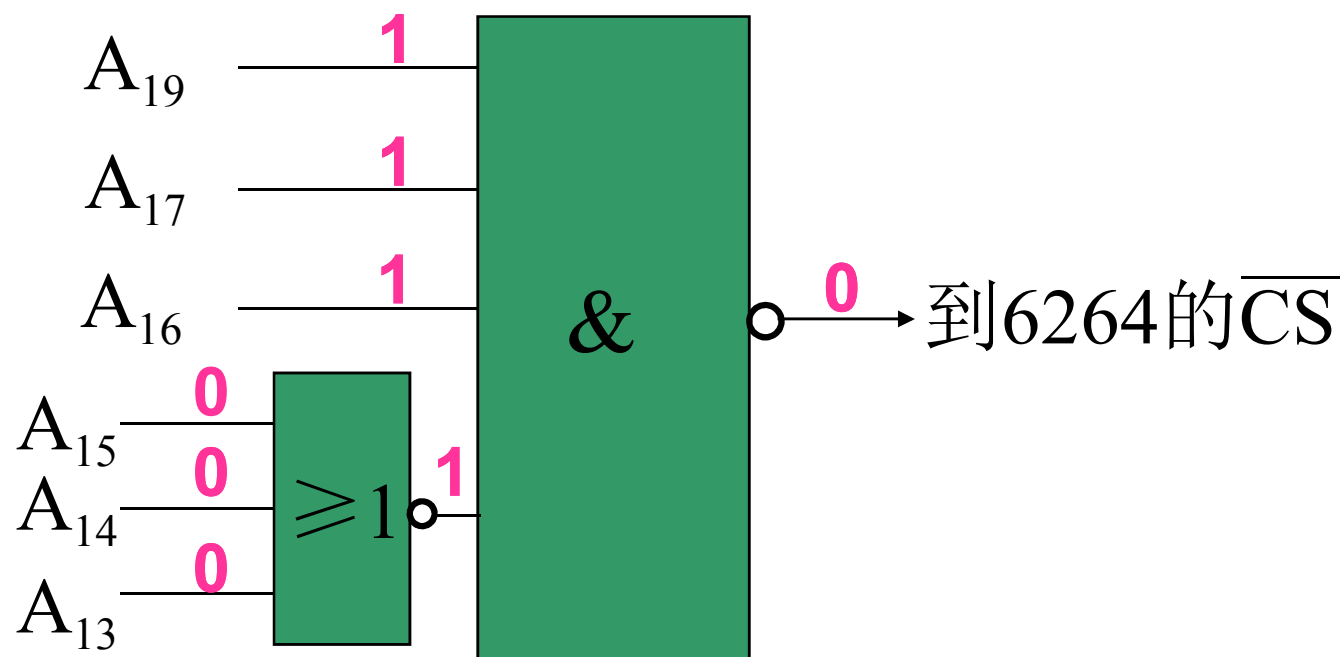
- 方法:

- 高位地址线中部分线(**不是全部**)输入到片选地址译码器产生片选信号。线选和全译码混合。存在地址重叠



部分译码的例子

- A_{18} 不参与译码



- 存储芯片的地址范围

- F0000H~F1FFFH 和 B0000H~B1FFFH

- 被选中芯片的每个单元都占有两个地址。