



中山大學  
SUN YAT-SEN UNIVERSITY

2025-2026学年秋季学期

# 微机原理

Principle of Microcomputer

肖山林

xiaoshlin@mail.sysu.edu.cn

微电子科学与技术学院

# 目录

- 第1章 绪论 (6学时)
- 第2章 微处理器结构 (6学时)
- 第3章 80x86指令系统 (10学时)
- 第4章 汇编语言程序设计 (4学时)
- 第5章 微机总线技术 (6学时)
- 第6章 存储系统 (4学时)
- 第7章 输入/输出接口 (8学时)
- 第8章 常用接口技术 (8学时)

# 内容

1 存储器分类及主要指标

2 常用存储器芯片介绍

3 扩展存储器设计

4 存储器地址译码电路设计

5 存储器与CPU的连接

# 本章学习要点

- ❖ 存储器分类，组成及性能指标。
- ❖ 熟悉几种常用芯片的规格和封装引脚
- ❖ 重点掌握位数扩充和字节扩充技术
- ❖ 存储器地址译码电路设计
- ❖ 熟练掌握存储器与CPU连接时应注意的问题

# 存储器的定义

- [广义]信息的载体。

结绳记事→甲骨→竹简→雕刻→纸张→U盘、硬盘、软盘、光盘。信息的存储是各种科学技术得以存在和发展的基础。

五大类  
信息存  
储技术

{ 印刷存储技术（中国四大发明）  
缩微胶片存储技术（1839England）  
磁存储技术（1898Denmark）  
半导体存储技术（20世纪60年代）  
光存储技术（20世纪70年代）

- [狭义]计算机系统中专门用于存放一定量数字信息的器件。

# 存储器分类及主要指标

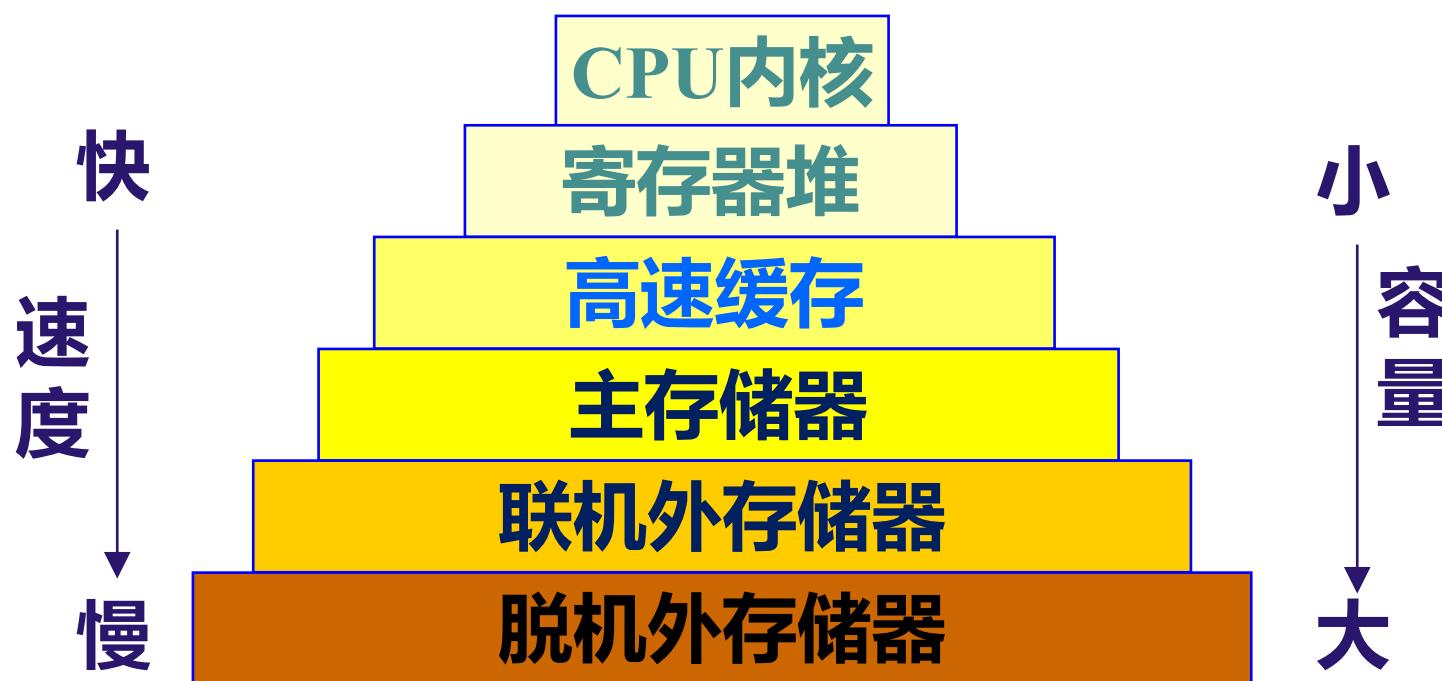
- ❖ 存储器是微型机重要的组成部分
- ❖ 存储计算机所用的工作信息—程序、数据等
- ❖ 没有存储器的计算机如同失掉记忆的人一样。





# 存储器分类及主要指标

- ❖ 微机拥有不同类型的存储部件
- ❖ 由上至下容量越来越大，但速度越来越慢



# 存储器分类及主要指标

## ❖ 内存——存放当前运行的程序和数据。

- 特点：快，容量小，随机存取，CPU可直接通过系统总线进行访问。
- 通常由半导体存储器构成。
- RAM、ROM

## ❖ 外存——存放非当前使用的程序和数据。

- 特点：慢，容量大，顺序存取/块存取，需通过I/O接口电路调入内存后CPU才能访问（CPU不能直接访问）。
- 例如，硬盘，磁盘，磁带，CD-ROM、DVD-ROM、U盘，移动硬盘。



# 存储器分类及主要指标

硬盘



内存



# 内存(主存)的分类



# RAM类型的特点

## ❖ 静态SRAM ( Static RAM )

利用半导体双稳态触发器的两种稳定状态来表示逻辑“1”和“0”。不需要刷新电路即能保存它内部存储的数据。只要不掉电，保存的信息就不会丢失

- 相对集成度低，外围控制电路简单
- 多用于小容量存储，如单片机、微机的缓存

## ❖ 动态DRAM (Dynamic RAM)

利用MOS管的栅极对其衬底间的分布电容来保存信息，每隔一段时间，要刷新充电一次，否则内部的数据即会消失

- 相对集成度高，外围控制电路复杂
- 多用于大容量存储，如微机中的内存

# 半导体存储器的性能指标

存储容量 = N×M

字数×字长

SRAM

型号	6264	62128	62256	62512
容量	8KB	16KB	32KB	64KB
	8K×8	16K×8	32K×8	64K×8
型号	2114	6116		
容量	1K×4	2K×8		

EPROM

型号	2716	2764	27256	27512
容量	2KB	8KB	32KB	64KB
	2K×8	8K×8	32K×8	64K×8

# 半导体存储器的性能指标

## ❖ 存储容量的表示

- bit —— 用二进制位定义存储容量
- Byte —— 用二进制字节定义存储容量

## ❖ 存储容量的常用单位

- 字节 —— B ( Byte )
- 千字节 —— KB ( Kilo Byte )
- 兆字节 —— MB ( Mega Byte )
- 吉字节 —— GB ( Giga Byte )

## ❖ 常用的单位换算

- $1KB = 1024 B$
- $1MB = 1024 KB$
- $1GB = 1024 MB$

# 半导体存储器的性能指标

❖  $1KB = 1 * 1024 = 2^0 * 2^{10} = 2^{10}$

$$2^{10} = 100\ 0000\ 0000B = 400H$$

$$2KB = 2 * 1024 = 2^1 * 2^{10} = 2^{11}$$

$$2^{11} = 1000\ 0000\ 0000B = 800H$$

$$32KB = 32 * 1024 = 2^5 * 2^{10} = 2^{15}$$

$$2^{15} = 1000\ 0000\ 0000\ 0000B = 8000H$$

$$64KB = 64 * 1024 = 2^6 * 2^{10} = 2^{16}$$

$$2^{16} = 1\ 0000\ 0000\ 0000\ 0000B$$

$$= 10000H$$

# 存储器地址范围与容量关系计算

## ❖ 1. 存储器地址范围分析

一个32KB的存储器首地址为4000H，求其末尾地址。

## ❖ 2. 存储器容量计算

- 1 ) 一个RAM，能存储1024个字，求容量
- 2 ) 一个RAM，首地址为9000H，末尾地址为0FFFFH，求其容量

# 半导体存储器的性能指标

## ❖ 存取时间（读写速度）

- 半导体存储器芯片的速度一般用存取时间和存储周期两个指标来衡量。
- 在存储器芯片的手册中可以查得  
最小读出周期 $t_{cyc}(R)$ (Read Cycle Time)  
最小写周期 $t_{cyc}(W)$ (Write Cycle Time)。

- ❖ 非易失性
- ❖ 可靠性
- ❖ 功耗
- ❖ 价格

# 内容

1 存储器分类及主要指标

2 常用存储器芯片介绍

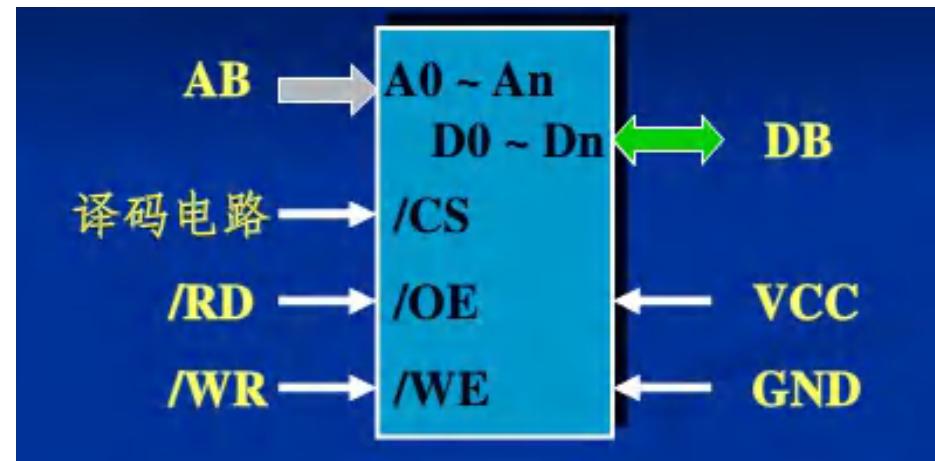
3 扩展存储器设计

4 存储器地址译码电路设计

5 存储器与CPU的连接

# 常用存储器芯片介绍

## ❖ SRAM芯片的引脚特点



地址线 A0~An 接CPU的**地址总线 AB** ( 个数取决于字数N )

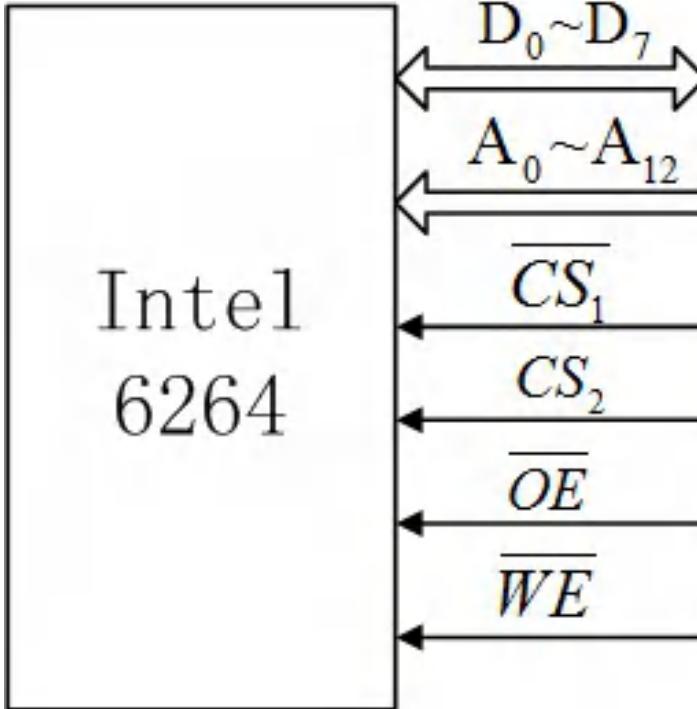
数据线D0~Dn接CPU的**数据总线 DB** ( 个数取决于字长M )

片选线 **CS** 由地址译码电路产生

读写线 **OE**、**WE** 由CPU的 **RD** 和 **WR** 控制

# 常用存储器芯片介绍

## ❖ SRAM芯片 Intel 6264



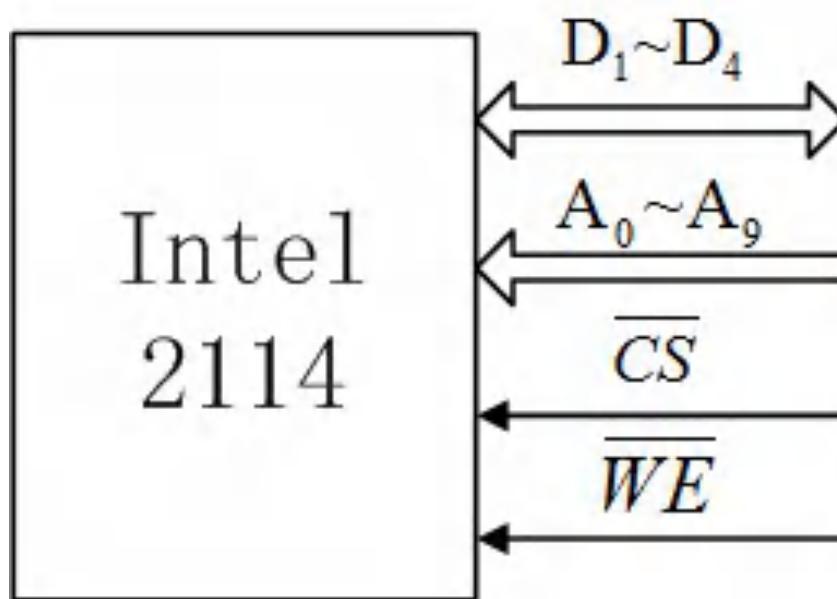
提供两个片选信号  
是为了提供片选方式的多样性

A0~A12 用作 存储器  
片内寻址

$\overline{CS}_1$ ,  $CS_2$  用作CPU  
片选寻址

# 常用存储器芯片介绍

## ❖ SRAM芯片 Intel 2114



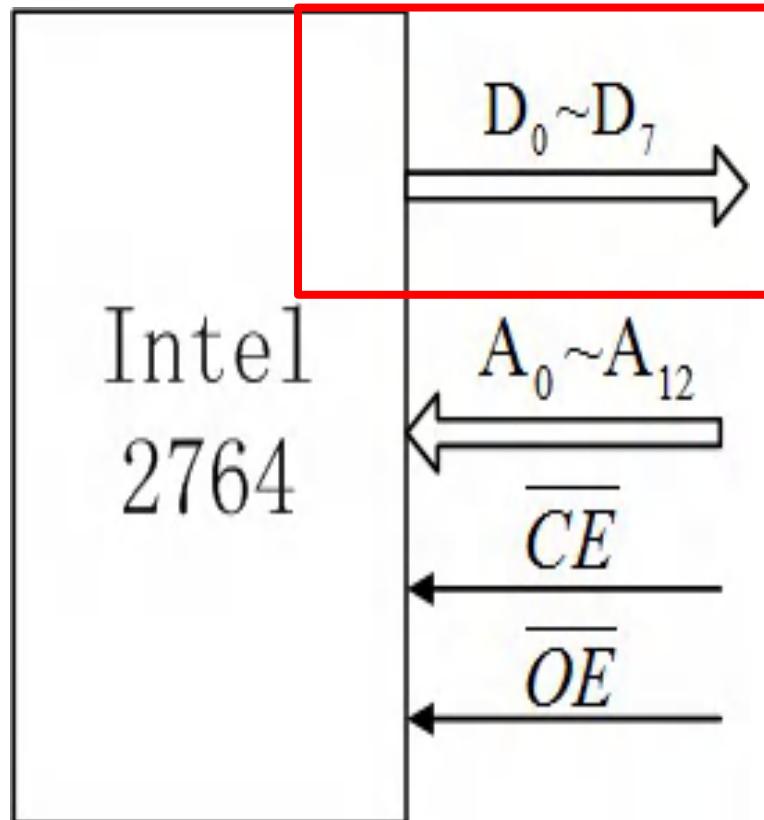
CS	WE	D1~D4
0	0	写入数据
0	1	读出数据
1	X	高阻

$$\text{容量} = 1K \times 4 = 0.5\text{KB}$$

**地址线** —— A<sub>0</sub>~A<sub>9</sub>  
**数据线** —— D<sub>1</sub>~D<sub>4</sub>  
**控制线** —— WE、CS

# 常用存储器芯片介绍

## ❖ EPROM芯片 Intel 2764



容量 =  $8K \times 8 = 8KB$

地址线 ——  $A_0 \sim A_{12}$

数据线 ——  $D_0 \sim D_7$

控制线 ——  $\overline{CE}$ 、 $\overline{OE}$ 、 $\overline{PGM}$

$\overline{CE}$  : 片选信号

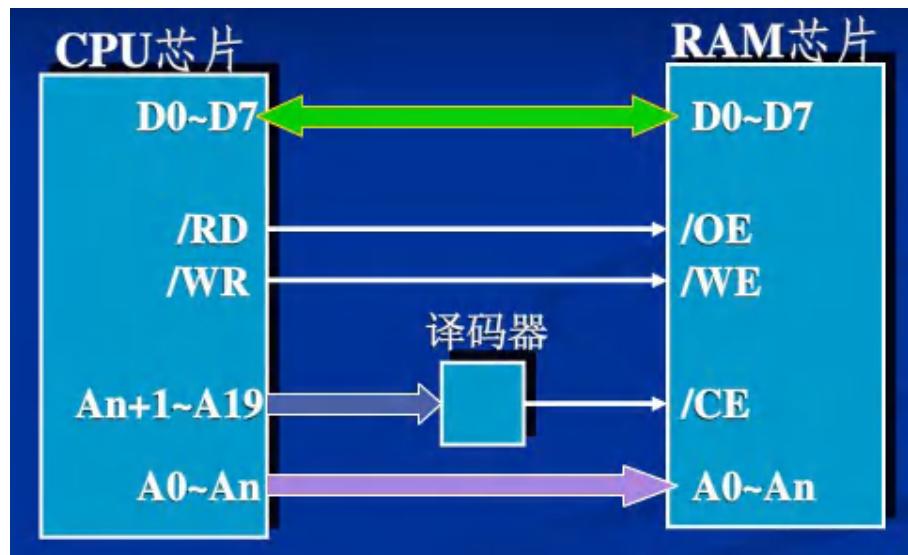
$\overline{OE}$  : 输出允许信号

$\overline{PGM}$  : 编程脉冲输入端

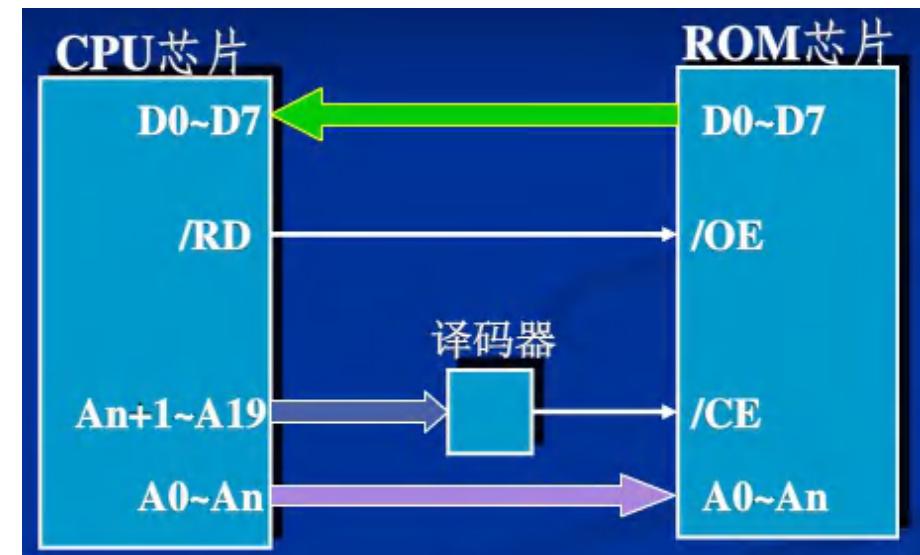
对EPROM编程时，在该端加上50ms左右的**负脉冲**，读操作时 $PGM=1$

# 存储器芯片与CPU的连接

RAM芯片与CPU芯片的连接



ROM芯片与CPU芯片的连接



注意：低位地址线用于片内寻址，  
高位地址线用于片选寻址

# 内容

1 存储器分类及主要指标

2 常用存储器芯片介绍

3 扩展存储器设计

4 存储器地址译码电路设计

5 存储器与CPU的连接

# 扩展存储器设计

- ❖ 单片存储器的容量有限难以满足实际存储容量的需求。
- ❖ 因此需要将若干片芯片连接在一起进行扩展，通常有三种扩展方式：



# 扩展存储器设计

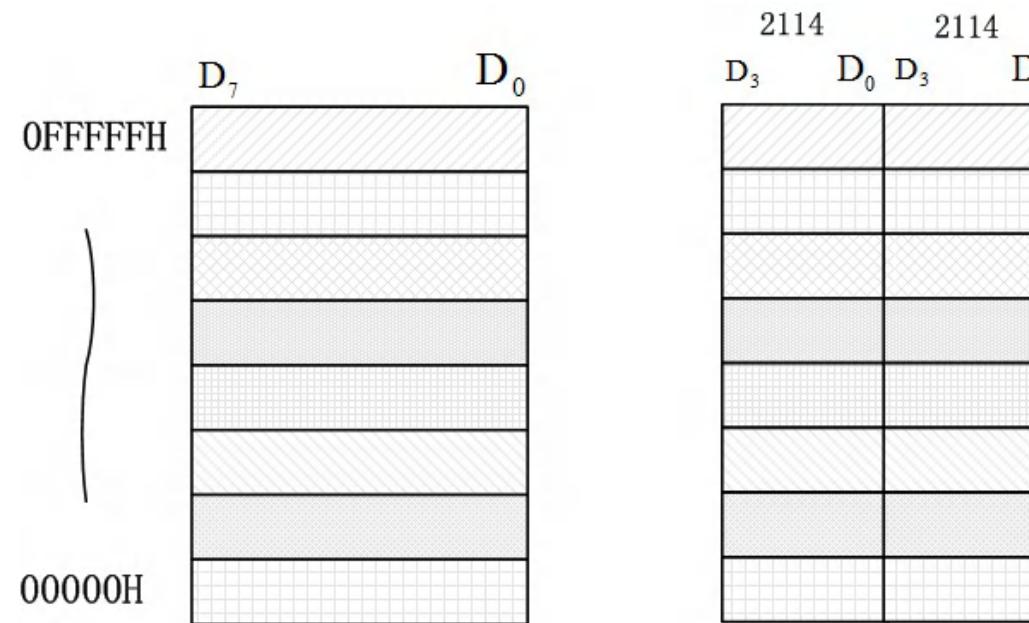
## ❖ 1. 位扩展

- 所选择的存储器芯片的字长达不到设计要求的字长时，用这样的存储器芯片构成系统所需的存储器子系统电路，就必须进行位扩展
- 用几片存储器芯片并起来，以增加存储字长

**EG : 2114 1K×4**

# 扩展存储器设计

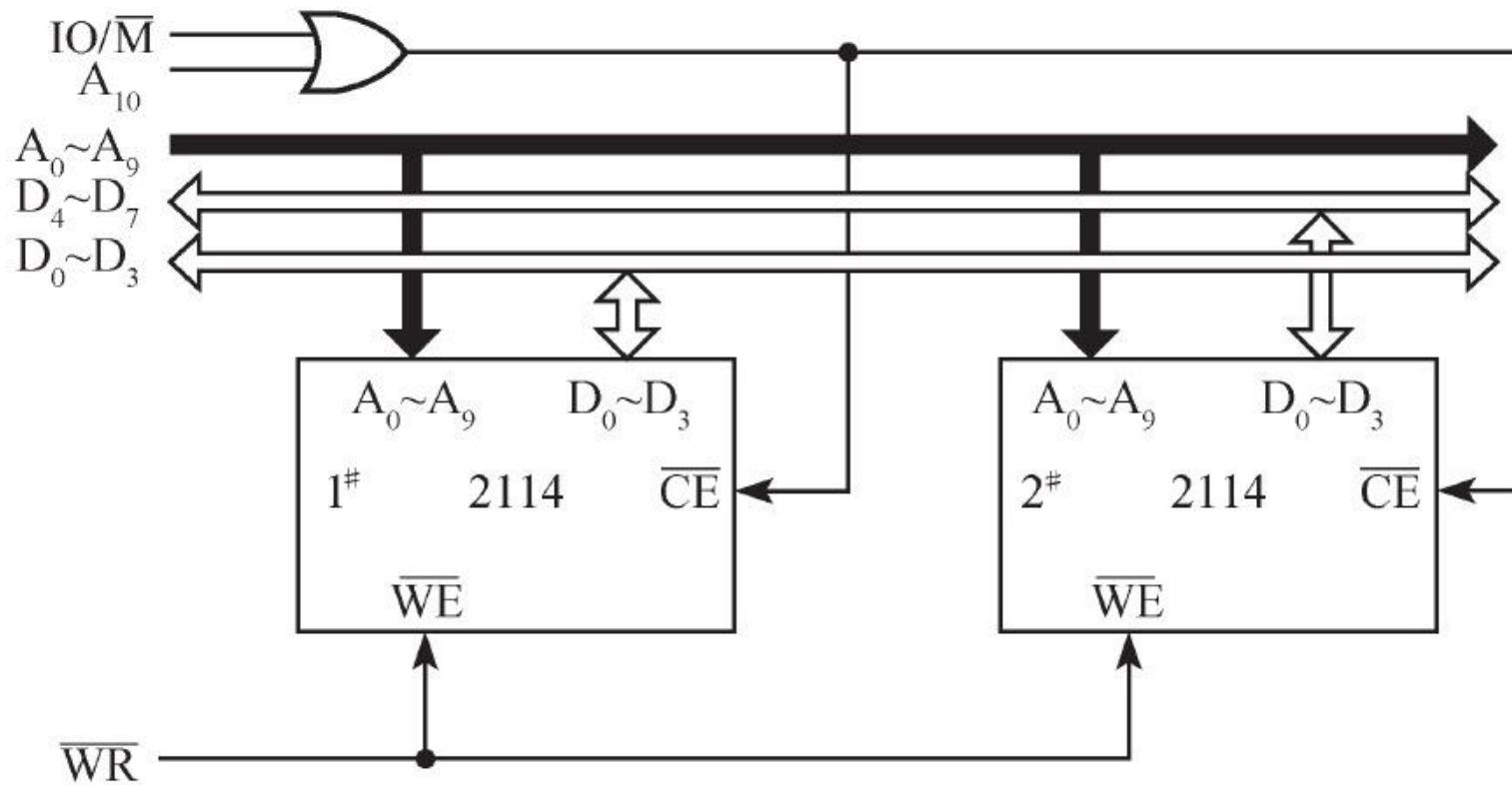
- ◆ 位扩展的特点：两个片子用同一地址进行片选，但数据线挂接不同，选定一个字节地址的时候两个片子同时被选中
- ◆ 位扩展 → 芯片的并联



# 扩展存储器设计

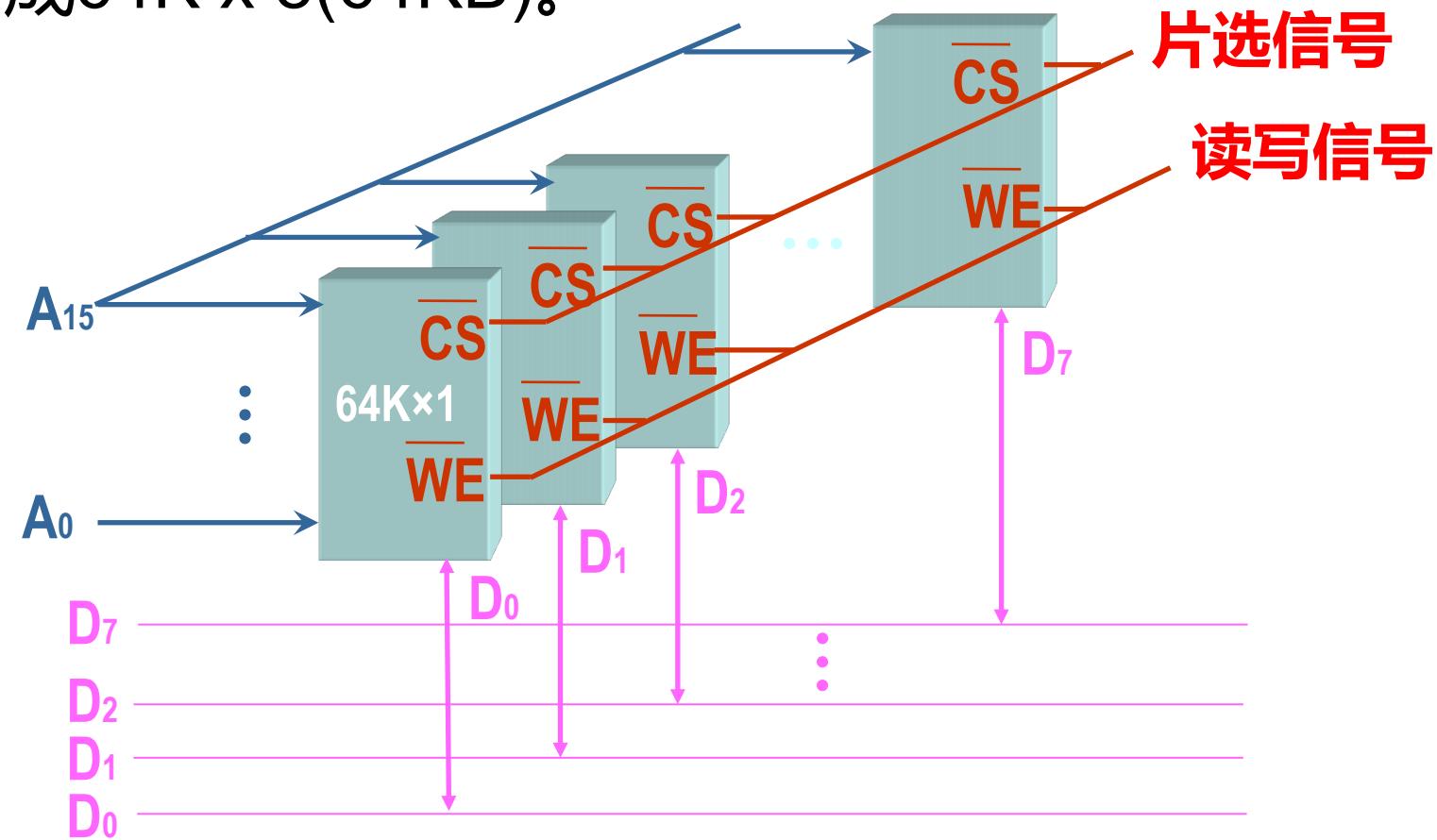
## EG1：位扩展设计实例

在8088最小方式系统下 用Intel 2114 扩充设计一块 1KB 的存储空间



# 扩展存储器设计

EG2：用  $64K \times 1$  SRAM芯片构成  $64KB$  存储器，  
需8片构成 $64K \times 8(64KB)$ 。



# 扩展存储器设计

## ❖ 位扩展的方法总结：

- 每片的地址线并联；
- 数据线分别引出；
- 控制线并联。

## ❖ 位扩展特点：

- 存储器的单元数不变；
- 位数增加；
- 位扩展的所有芯片在访问时同时选中。

# 扩展存储器设计

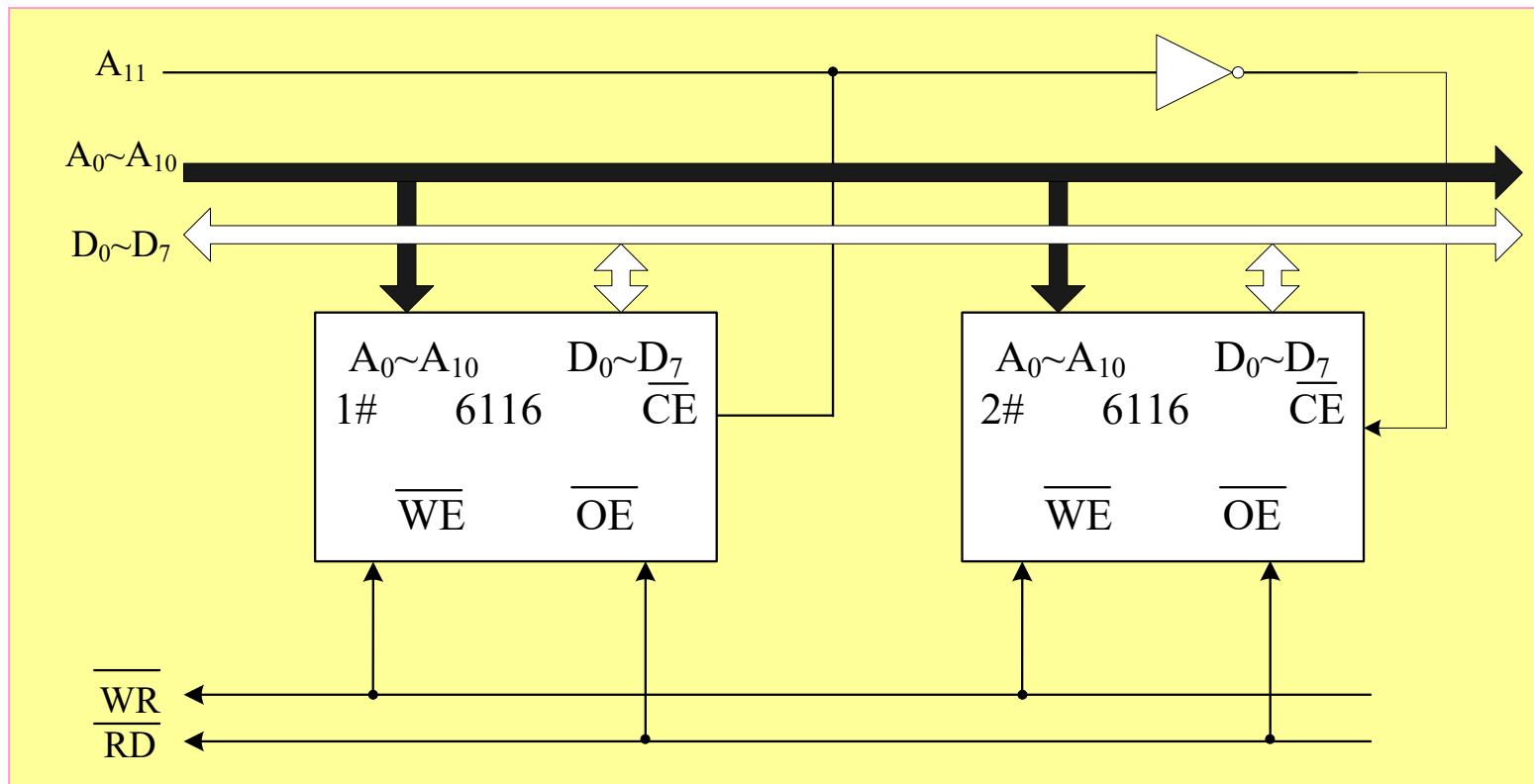
## 2 . 字节扩展

- ❖ **字节扩展是指芯片字长不变的情况下增加存储器字节的数量（容量）**
- ❖ 在字节扩展时，每个芯片的地址线、数据线、控制线并联，仅片选信号分别引出，使每个芯片占据不同的地址范围。通过下面的例题加以理解。



# 扩展存储器设计

❖ EG3：8088最小系统下，用2片 $2K \times 8b$ 的SRAM芯片6116，组成一段连续 $4K \times 8b$ 的存储器，字节扩展设计如下图所示





中山大學  
SUN YAT-SEN UNIVERSITY

2025-2026学年秋季学期

# 微机原理

Principle of Microcomputer

肖山林

xiaoshlin@mail.sysu.edu.cn

微电子科学与技术学院

# 回顾：内存的分类

读写存储器 ( RAM )

( Random Access Memory )

随机读写，易失，  
掉电数据丢失

只读存储器 ( ROM )

( Read Only Memory )

只读，非易失，  
掉电数据保持

静态RAM ( SRAM )

动态RAM ( DRAM )

掩膜ROM ( MROM )

可编程ROM ( PROM )

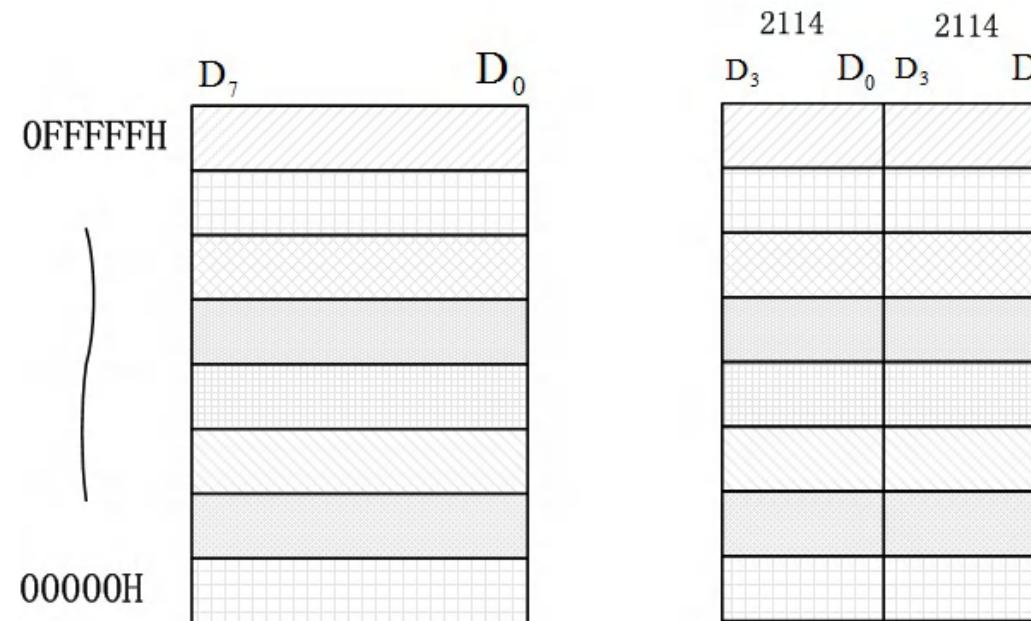
紫外线光可擦除ROM  
( EPROM )

电可擦除ROM  
( E<sup>2</sup>PROM )

Flash Memory

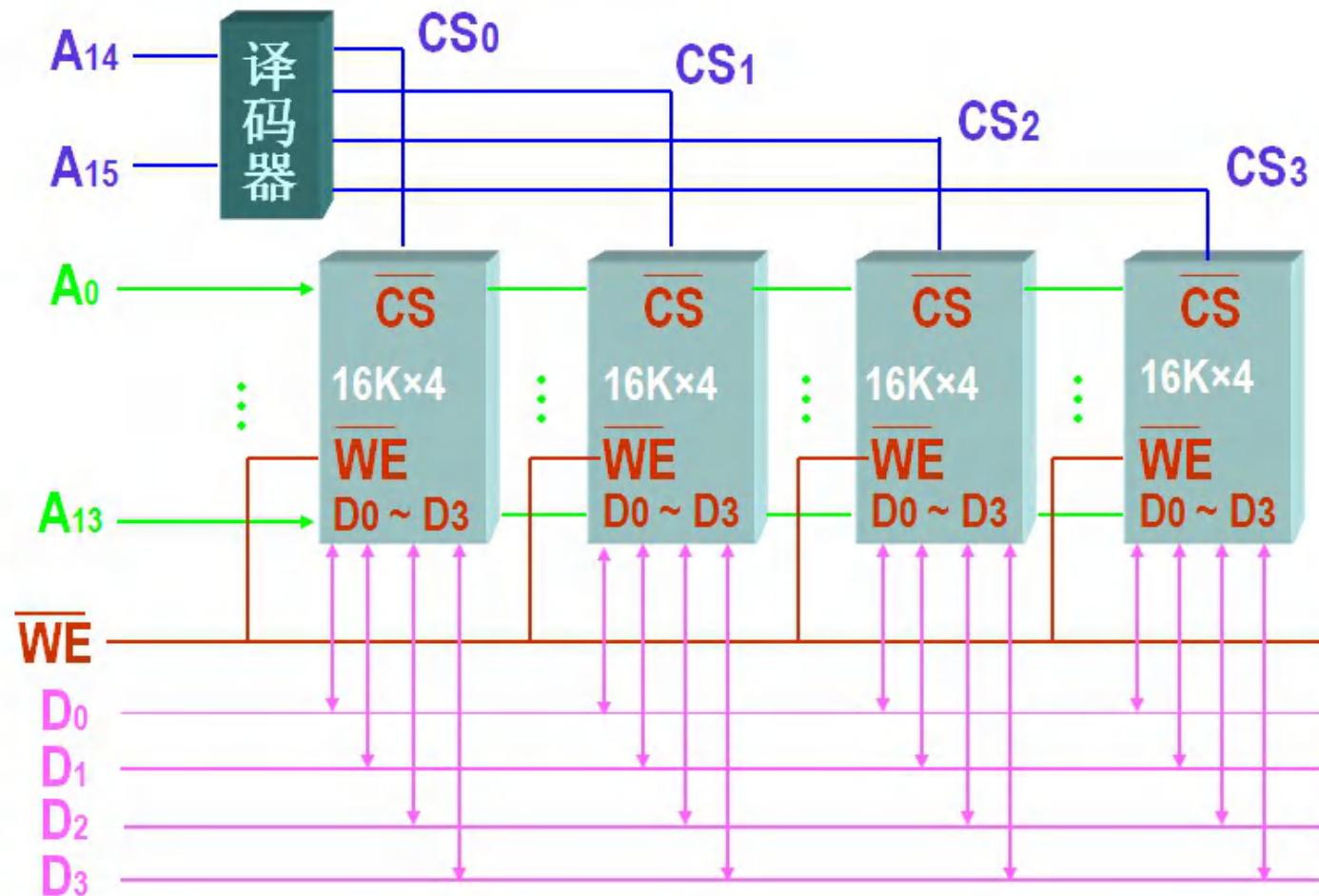
# 回顾：扩展存储器设计

- ◆ 位扩展的特点：两个片子用同一地址进行片选，但数据线挂接不同，选定一个字节地址的时候两个片子同时被选中
- ◆ 位扩展 → 芯片的并联



# 扩展存储器设计

EG4：用 $16K \times 4$ bit芯片组成 $64K \times 4$ bit的存储器。



# 扩展存储器设计

## 3 . 字节和位扩展

在实践中，会有这样的情况，存储芯片的位数和字节数都不满足存储器的要求，为此需要同时进行字节和位的扩展。

### 方法如下：

- 根据主存容量及芯片容量确定所需**存储芯片数**。若现有芯片的容量为 $L \times K$ ，要构成容量为 $M \times N$ 的存储器，需要的芯片数为 $= (M / L) \times (N / K)$
- 先进行位扩展以满足**字长要求**，构成芯片组；
- 再进行字节扩展以满足**容量要求**(可利用芯片组)。

# 扩展存储器设计

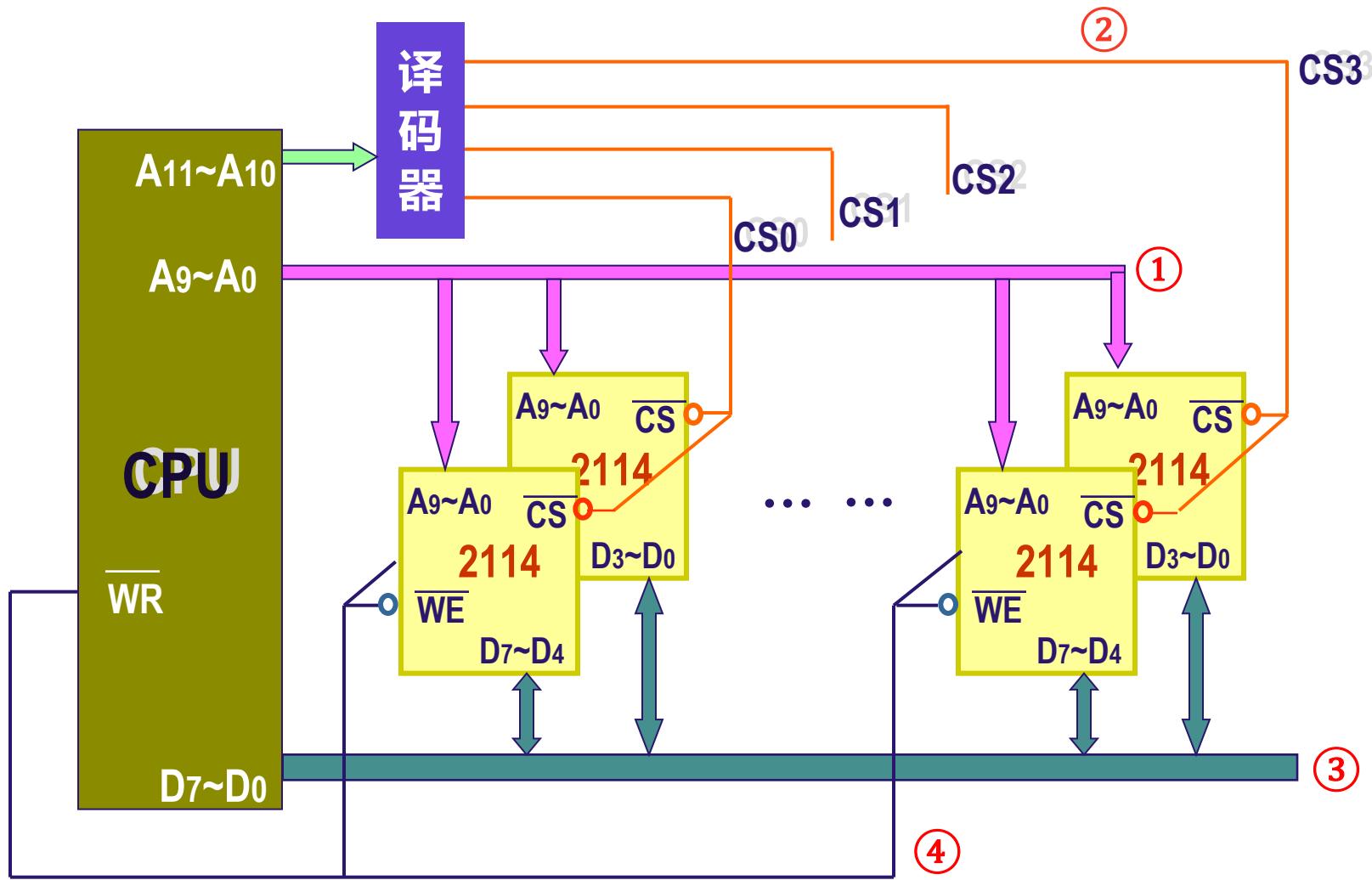
**EG5：8088最小系统下  
用2114(1K×4)芯片组成4KB的存储器**

**分析：** 先扩展成1KB —— 2片  
再扩展成4KB ——  $4 \times 2 = 8$  片

4KB存储器需要12位地址线( $A_{11}-A_0$ )，  
其中低10位地址( $A_9-A_0$ )用于片内寻址，  
高2位地址线( $A_{11}-A_{10}$ )用于片选译码，以选择不同芯片组。



# 扩展存储器设计



# 扩展存储器设计

构成存储器时，一般可按下列步骤进行：

- ◆ 根据可用芯片，计算容量和所需芯片数
- ◆ 地址分配(完成扩展)
- ◆ 片选逻辑设计
- ◆ 画出逻辑连接图
  - 连接地址线
  - 连接数据线
  - 连接控制线（片选、读写控制）

# 内容

1 存储器分类及主要指标

2 常用存储器芯片介绍

3 扩展存储器设计

4 存储器地址译码电路设计

5 存储器与CPU的连接

# 存储器地址译码电路设计



存储芯片的数据线



存储芯片的地址线



存储芯片的片选端



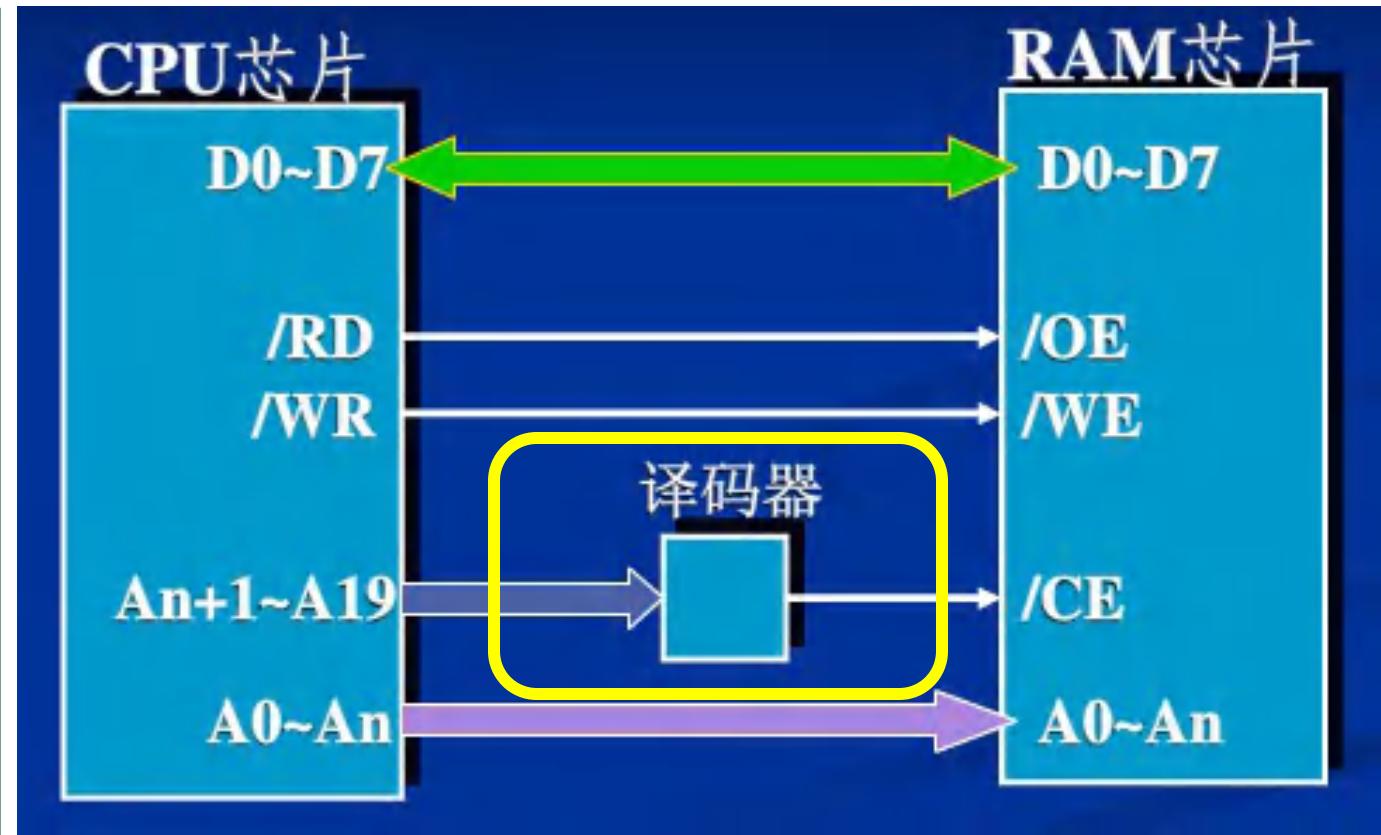
存储芯片的读写控制线

# 存储器地址译码电路设计

2:4译码器：  
**74LS139**

3:8译码器：  
**74LS138**

4:16译码器：  
**74LS154**



- ◆ 译码电路可以采用逻辑门电路，也可以采用各种译码器。



# 存储器地址译码电路设计

- 存储器芯片所需的片内地址线和片选地址线
    - 设CPU芯片的地址线为 A0 ~ A15
    - 设存储器芯片的地址线为 A0 ~ A9
    - CPU供给存储器的片内地址线为 A0 ~ A9
    - CPU供给存储器的片选地址线为 A10 ~ A15

## 地址图

# 存储器地址译码电路设计

**全译码**

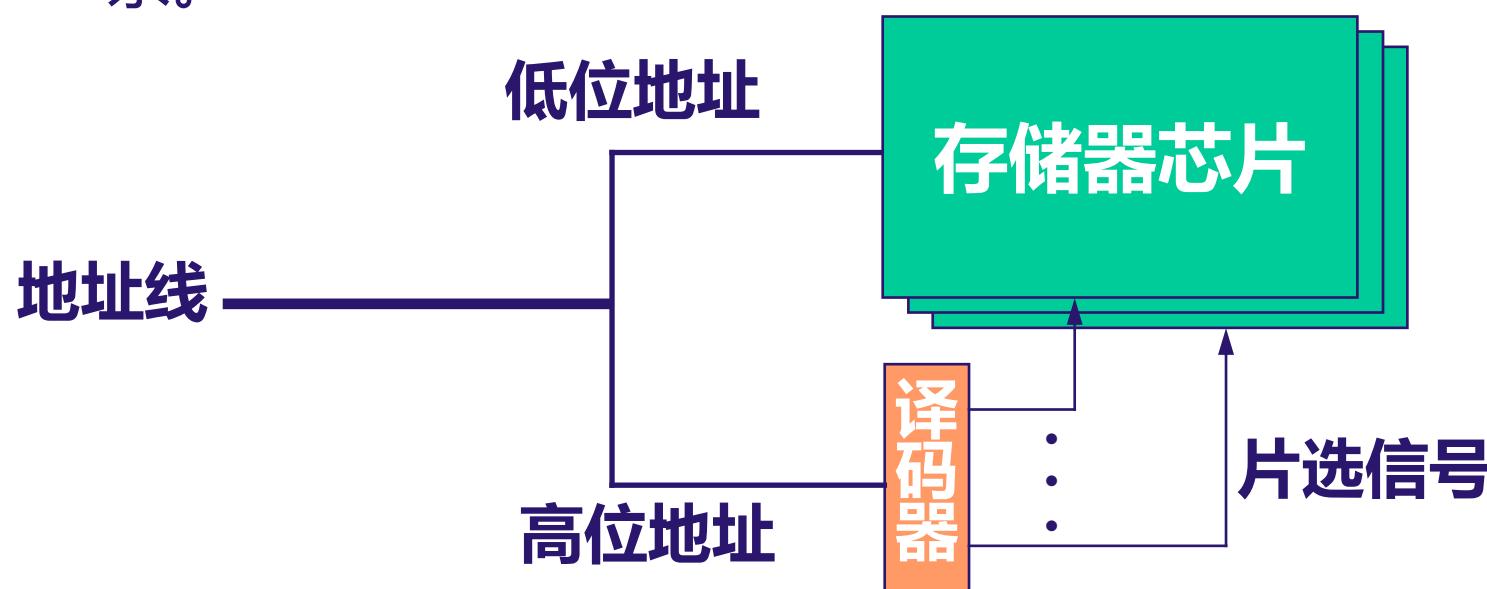
- ※每组存储芯片地址和CPU低地址连接。
- ※利用剩余所有高位地址译码后连接不同芯片组的片选，当前只选择一组存储芯片工作。
- ※其优点是能保证每组存储芯片的地址不重叠（即每个地址单元只能使用一个唯一的地址来访问）而且每组存储芯片所占地址连续。

**部分译码**

- ※每组存储芯片地址和CPU低地址连接。
- ※只利用部分高位地址（或不用）译码连接不同芯片组的片选。
- ※其优点是译码电路简单，但会导致地址重叠（即每个地址单元可以用不同的地址来访问）

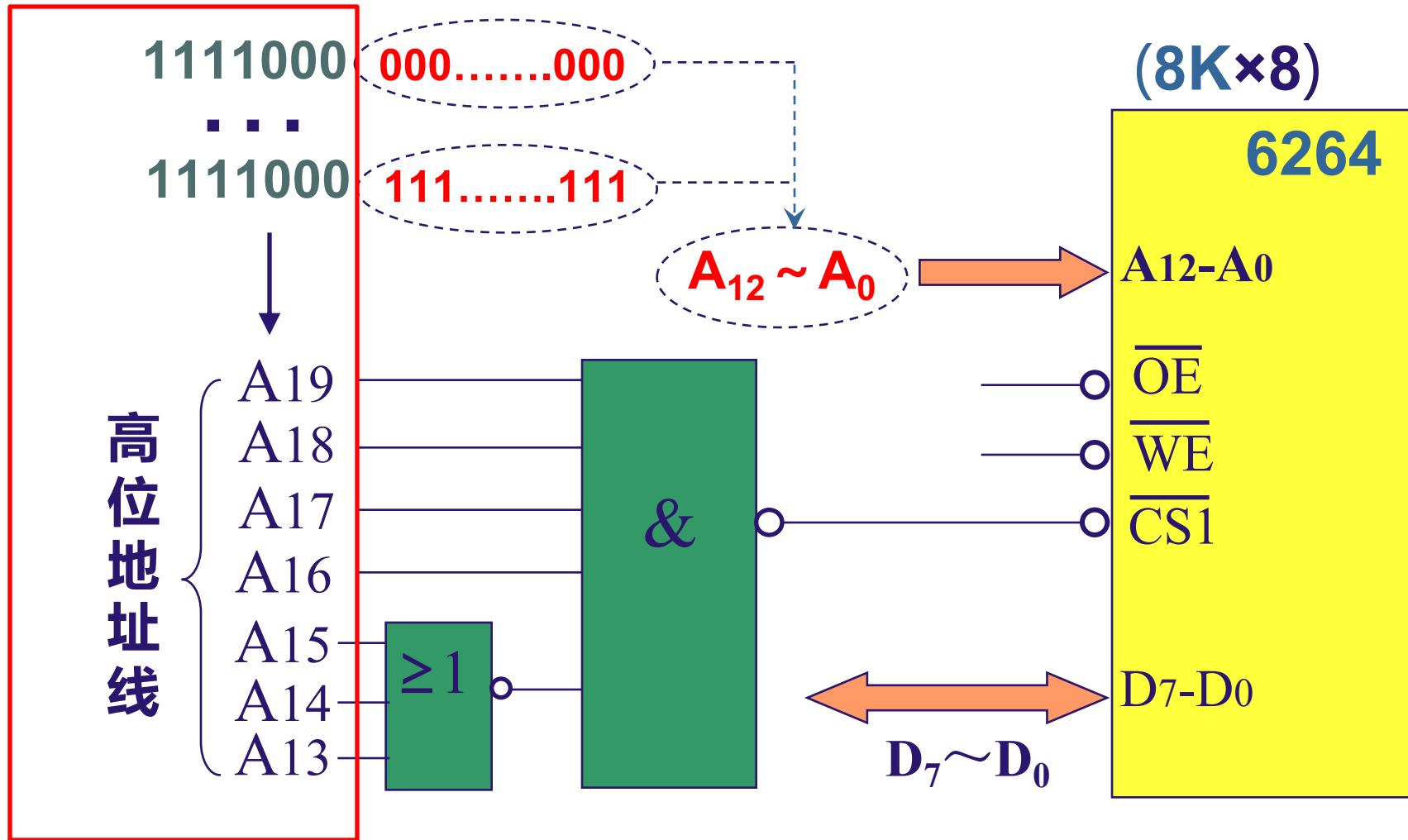
# 存储器地址译码电路设计

- ❖ 全地址译码
- ❖ 除片内寻址外，用剩余的高位地址信号全部作为译码输入信号，译码器的输出作为各芯片的片选信号。存储器的每一个单元都占据一个唯一的内存地址，不会产生地址重叠现象，但译码比较复杂。



# 存储器地址译码电路设计

❖ EG6 : 6264芯片的地址范围 F0000H~F1FFFH

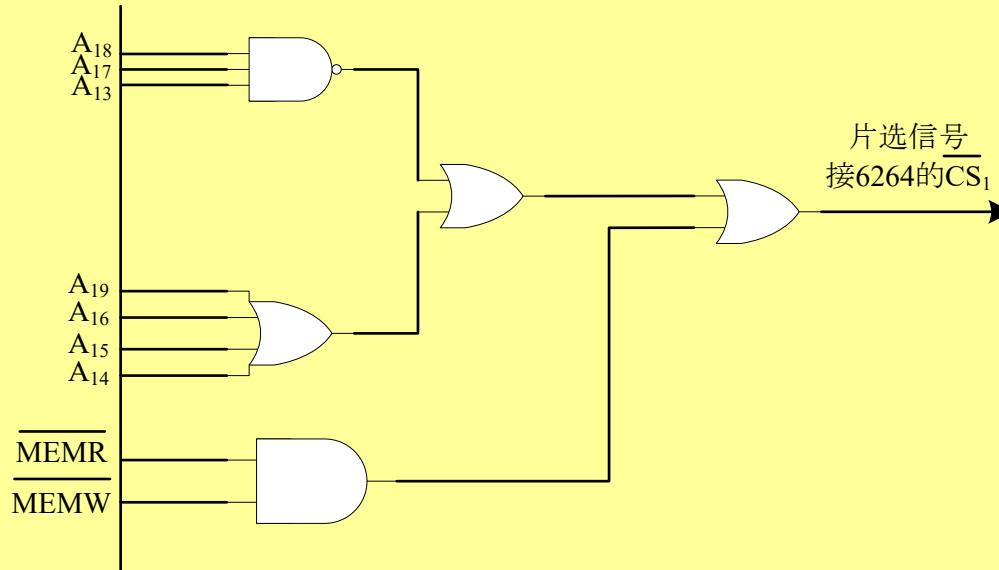


# 存储器地址译码电路设计

**EG7：**在8088 CPU工作在**最大方式**组成的微机应用系统中，扩充设计8KB的SRAM电路，SRAM芯片用Intel 6264。若分配给该SRAM的起始地址为62000H

A <sub>19</sub>	A <sub>18</sub>	A <sub>17</sub>	A <sub>16</sub>	A <sub>15</sub>	A <sub>14</sub>	A <sub>13</sub>	A <sub>12</sub>	A <sub>11</sub>	A <sub>10</sub>	A <sub>9</sub>	A <sub>8</sub>	A <sub>97</sub>	A <sub>6</sub>	A <sub>5</sub>	A <sub>4</sub>	A <sub>3</sub>	A <sub>2</sub>	A <sub>1</sub>	A <sub>0</sub>
0	1	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	1	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1

8088最大方式BUS



**地址信号译码电路：**  
所有为1的地址线通过与非门，所有为0的地址线通过或门，两个门输出的信号再通过或门汇总，得到选通信号

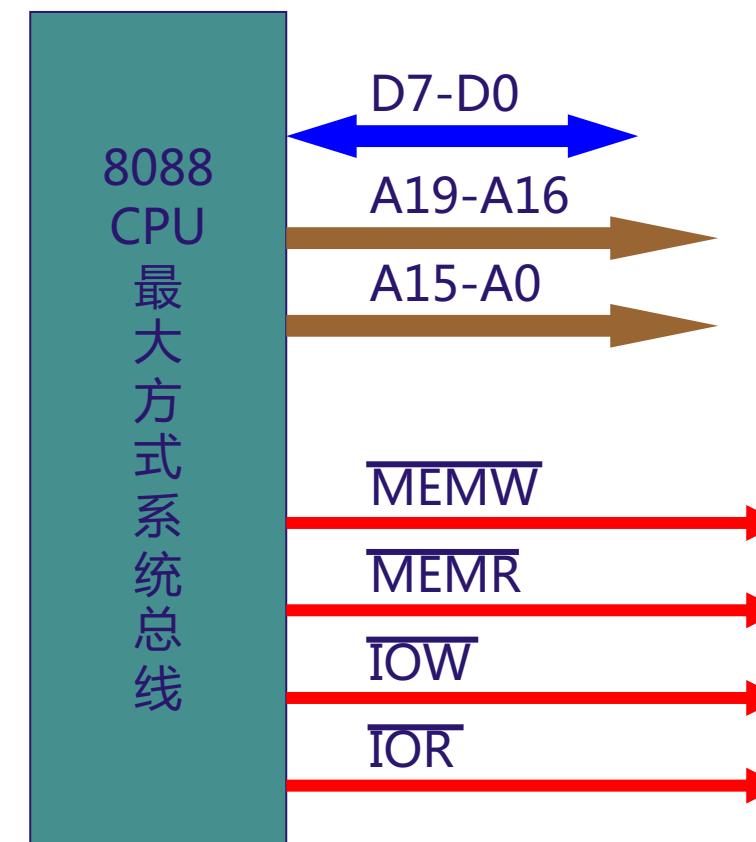
# 存储器地址译码电路扩展举例

**例1.** 在8088最大方式系统总线上扩充设计4K字节的SRAM存储器电路。SRAM芯片选用Intel 2114，起始地址从00000H开始。试画出此存储器电路与系统总线的连接图。



# 存储器地址译码电路扩展举例

## ①确定总线及总线信号



# 存储器地址译码电路扩展举例

## ②确定存储器芯片数

- ❖ 2114 :  $1K \times 4 \longrightarrow$ 生成4K空间需要8片
- ❖ 两片2114组合进行扩展构成1KB的空间，我们称为一个模块

## ③地址分析

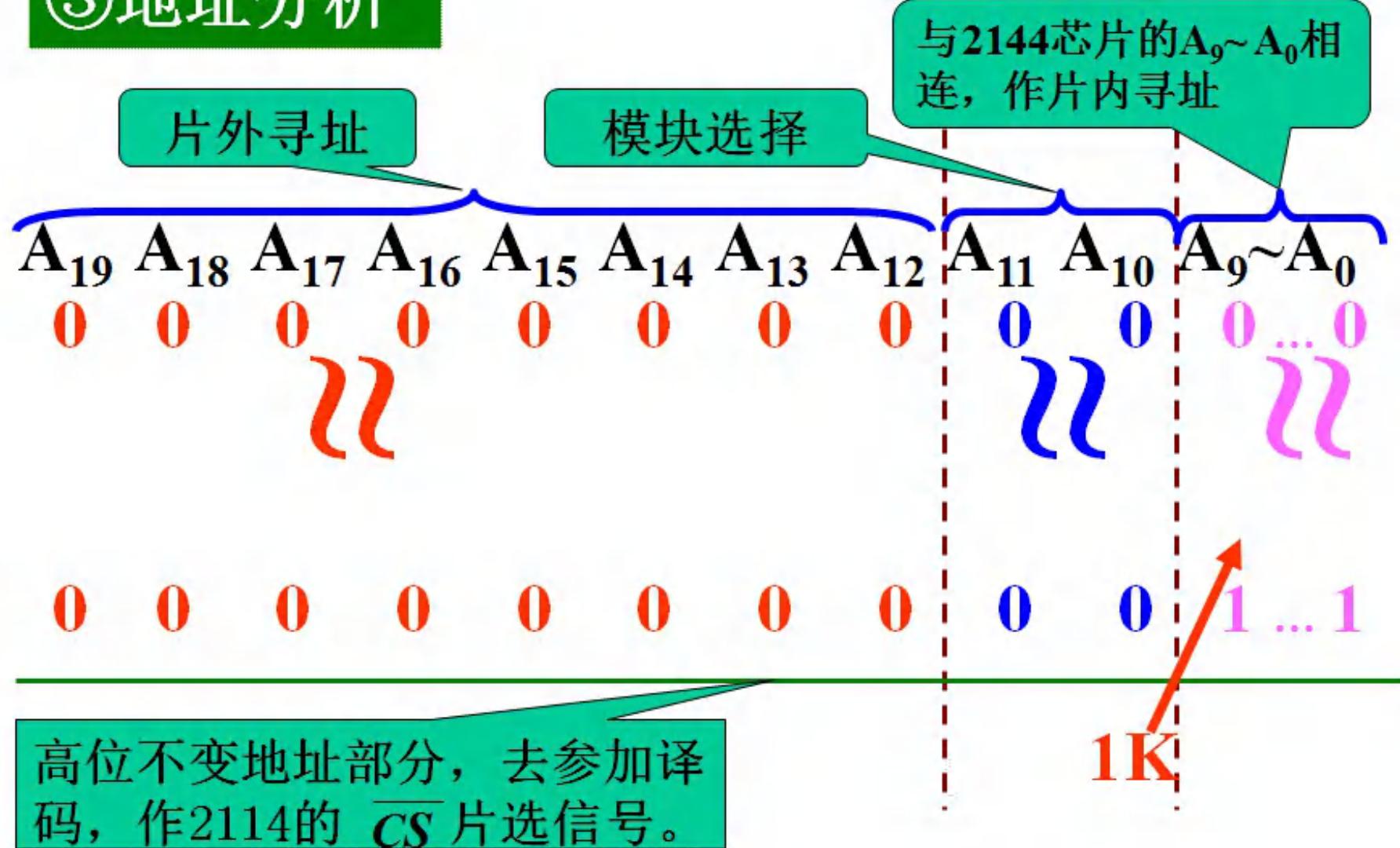
- ❖ 4K，首地址为00000H  $\longrightarrow$ 末地址0FFFFH

$A_{19} \sim A_{12}$	$A_{11} \ A_{10}$	$A_9 \sim A_0$
0 0 0 0 0 0 0 0	0 0	0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0	1 1	1 1 1 1 1 1 1 1 1



# 存储器地址译码电路扩展举例

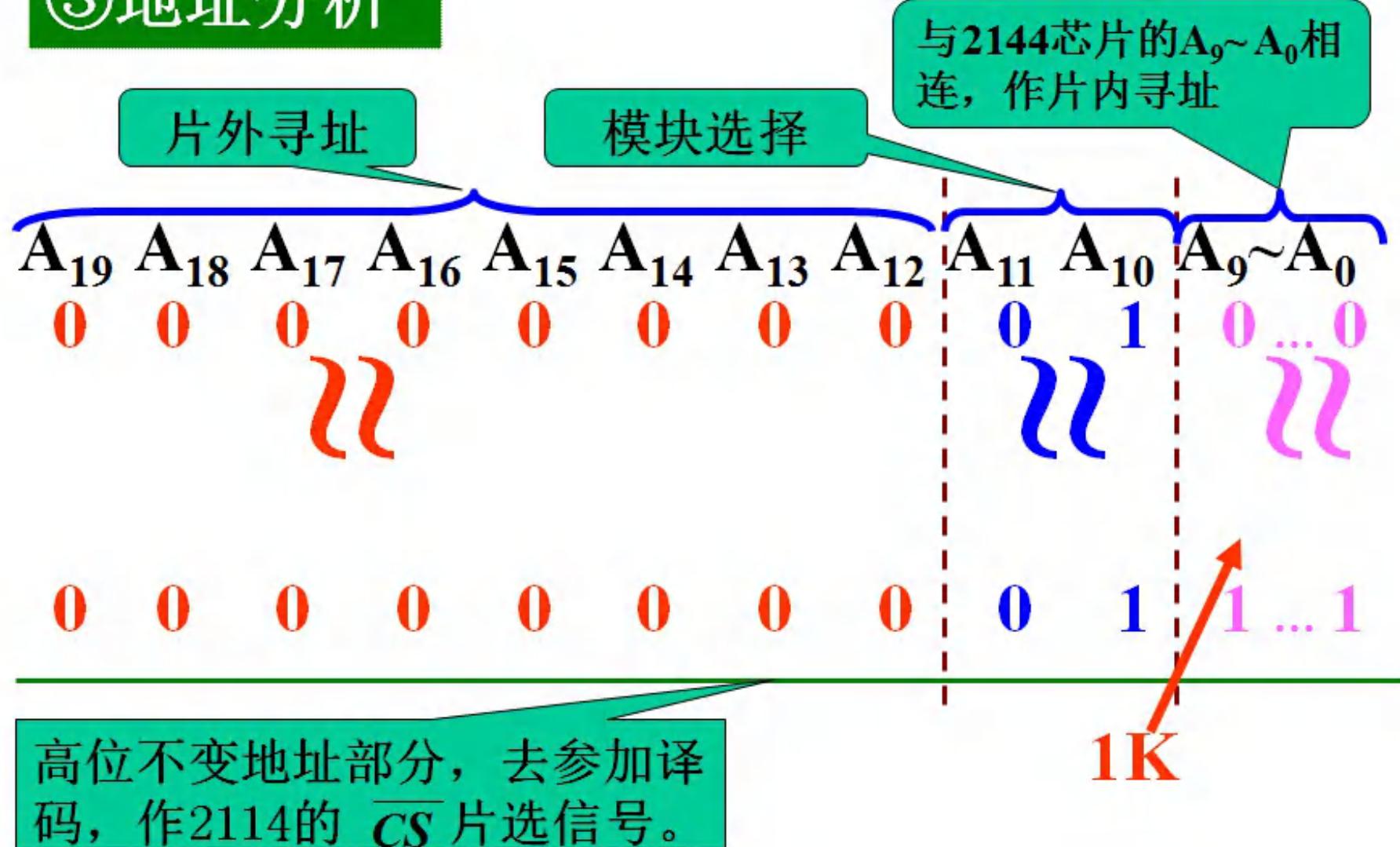
## ③ 地址分析





# 存储器地址译码电路扩展举例

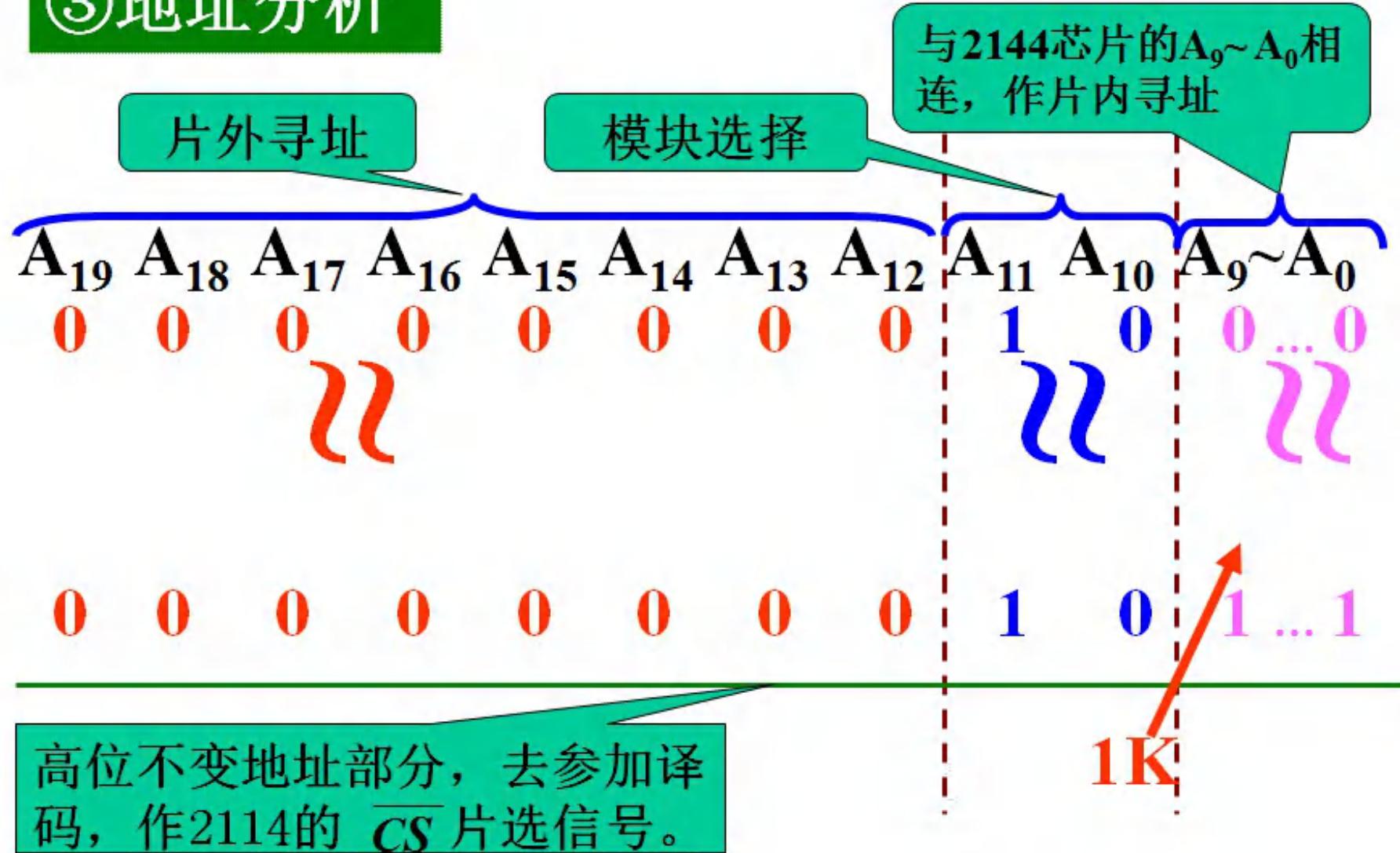
## ③ 地址分析





# 存储器地址译码电路扩展举例

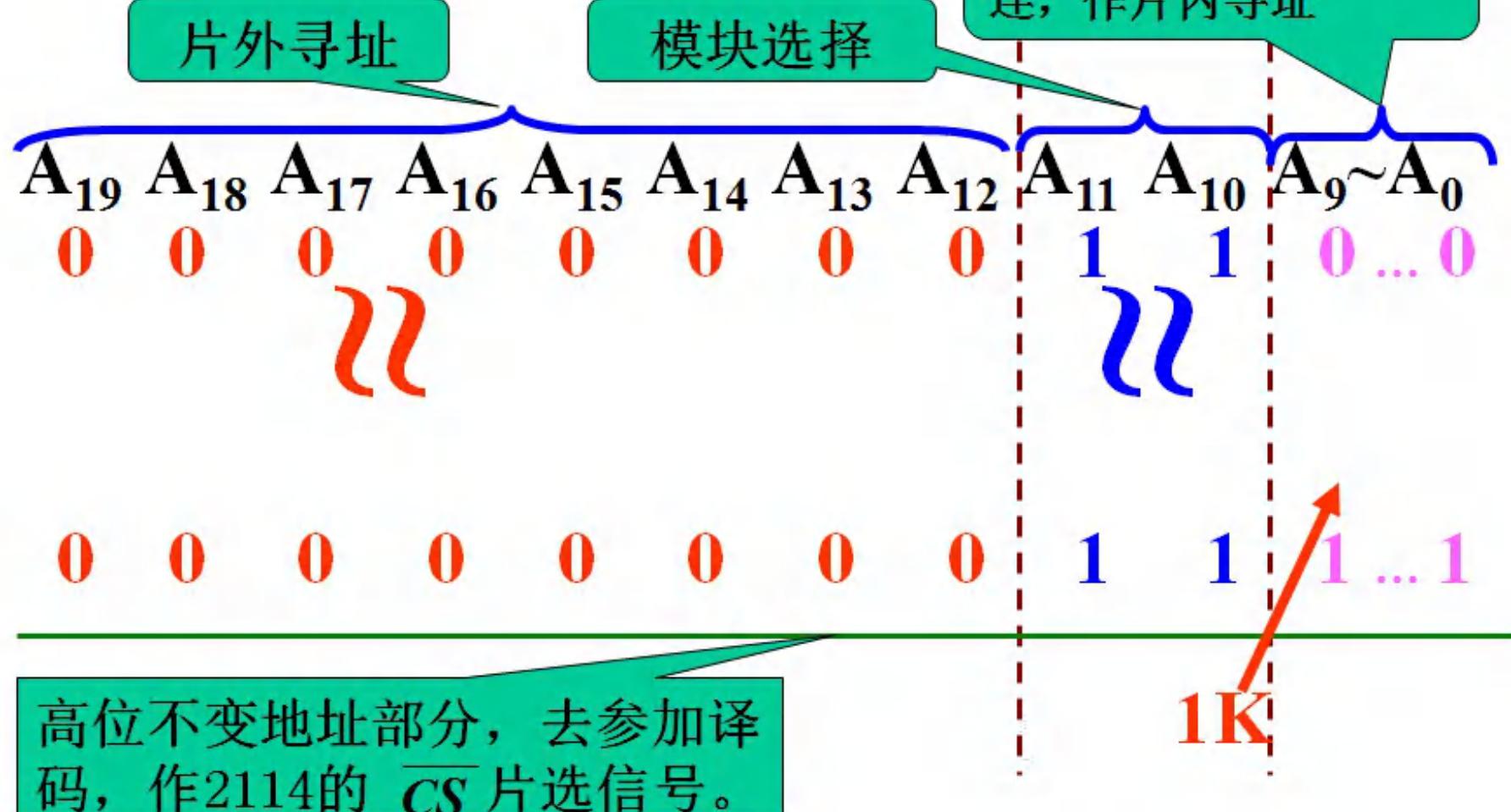
## ③ 地址分析



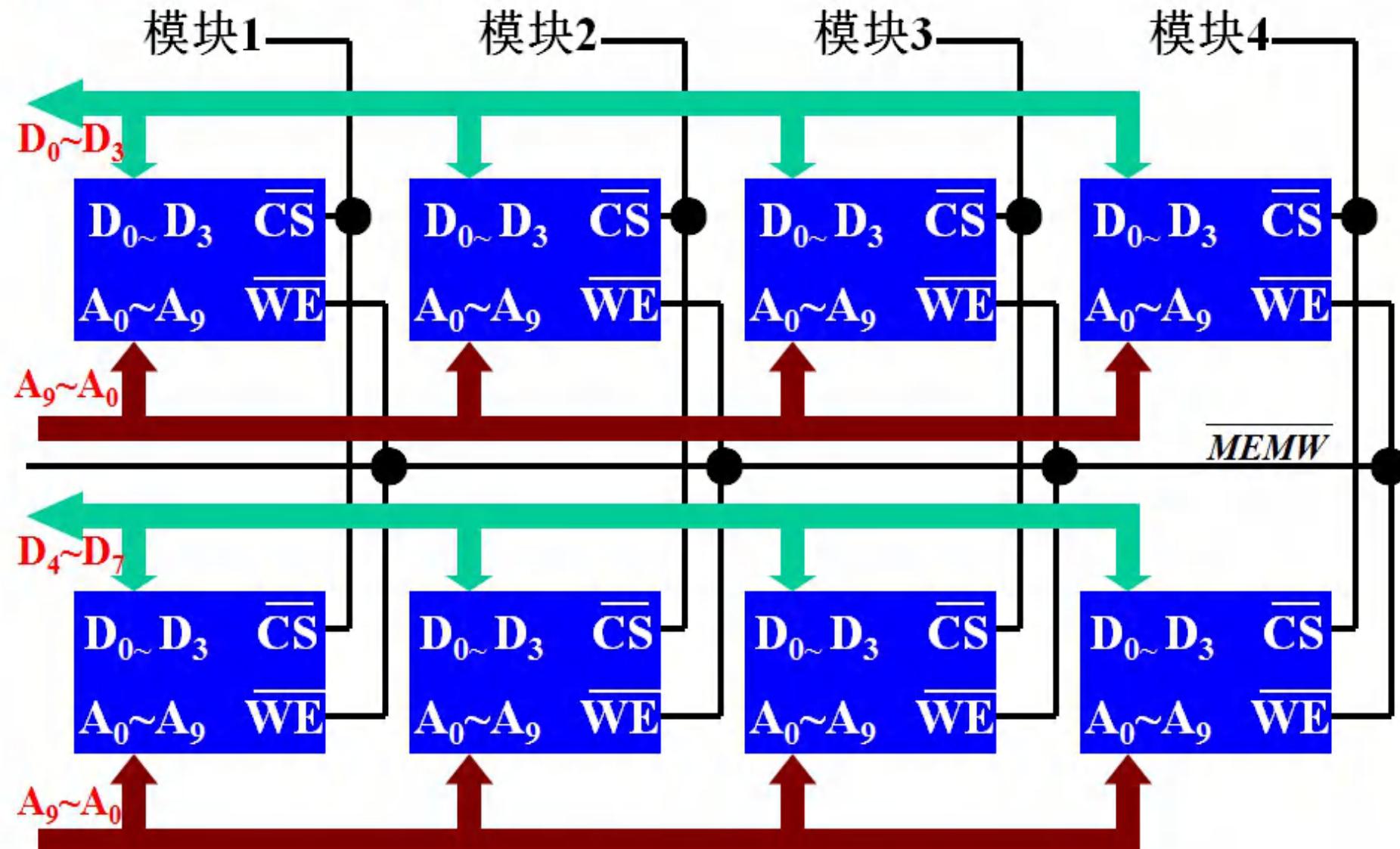


# 存储器地址译码电路扩展举例

## ③ 地址分析



## ④ 电路连接



由 $1K \times 4$  SRAM构成的 $4K \times 8$ 存储器模块

## ⑤存储器地址译码电路设计

4KB

00000H  
*u*

00FFFH

模块1

00000H~003FFH

模块2

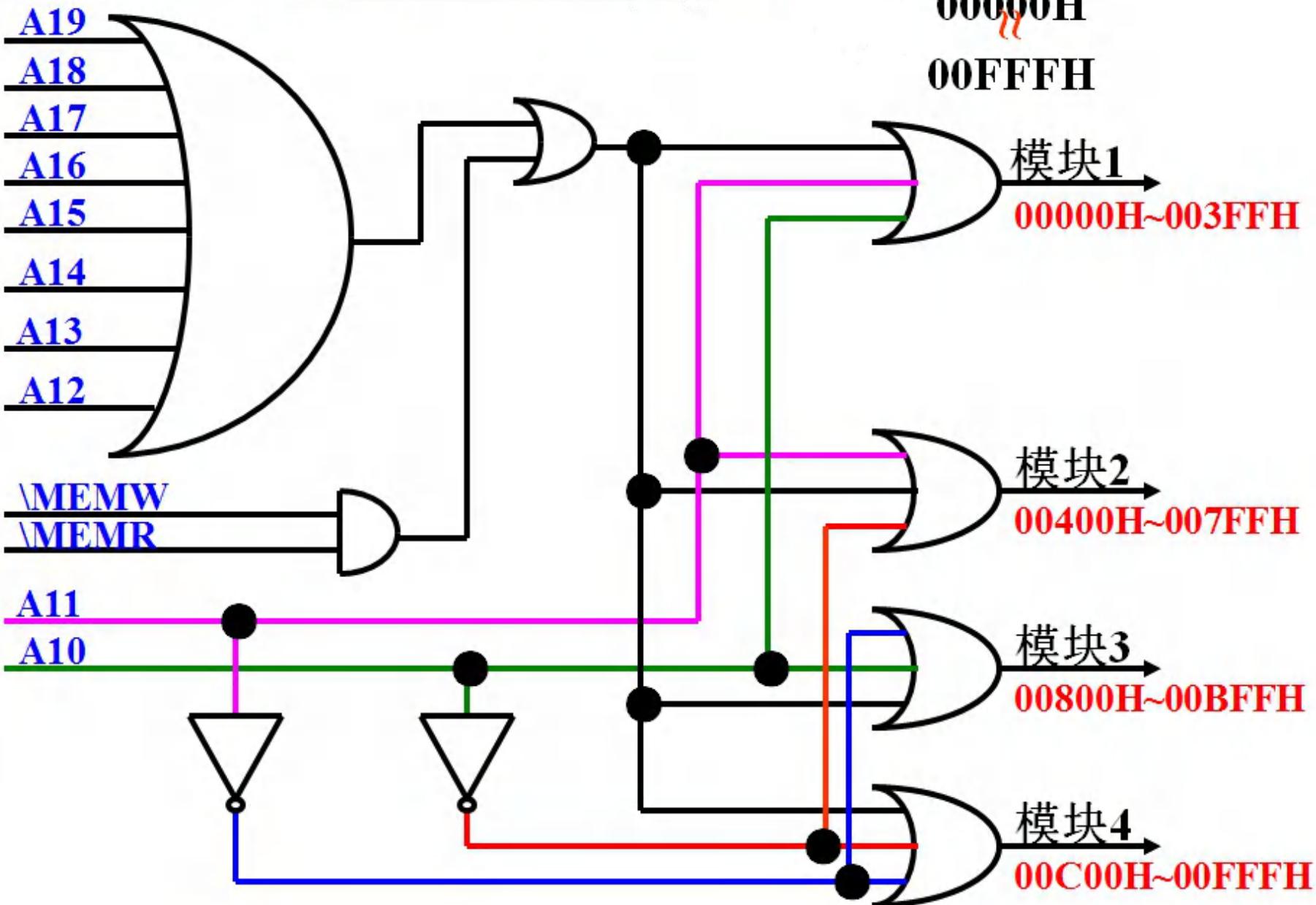
00400H~007FFH

模块3

00800H~00BFFH

模块4

00C00H~00FFFH



# 存储器地址译码电路设计

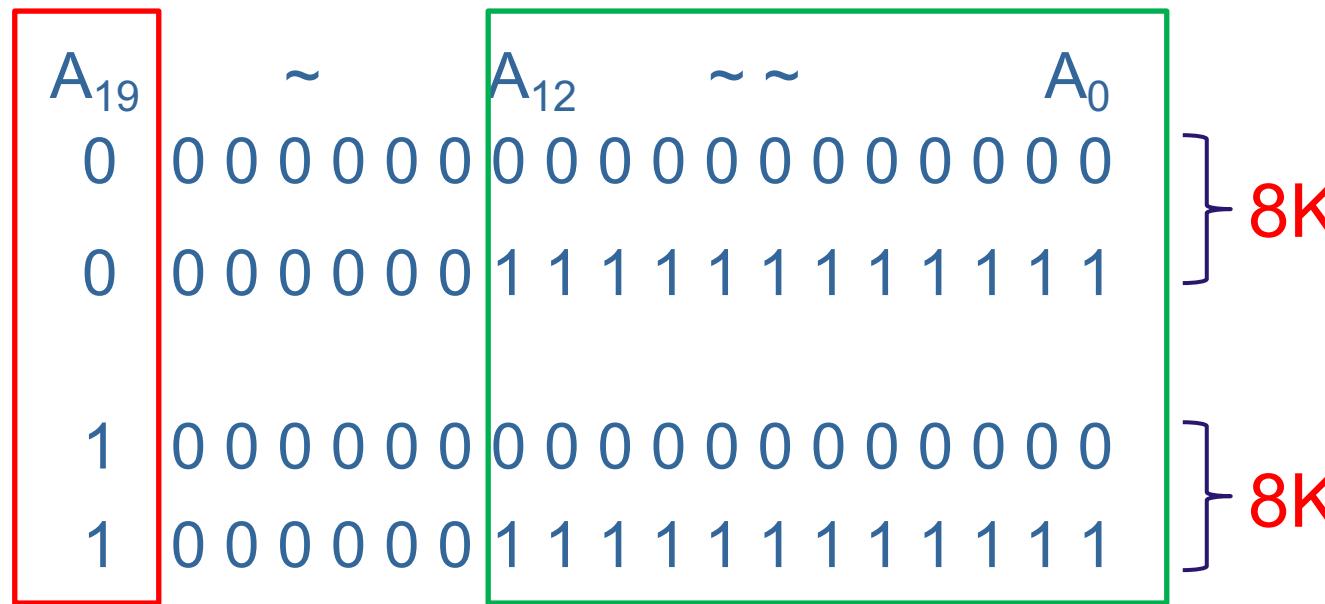
## ❖ 部分地址译码

- 只有部分（高位）地址线参与对存储芯片的译码；
- 每个存储单元将对应多个地址（地址重复），需要选取一个可用地址；
- **优点**：可简化译码电路的设计；
- **缺点**：但系统的部分地址空间将被浪费。



# 存储器地址译码电路设计

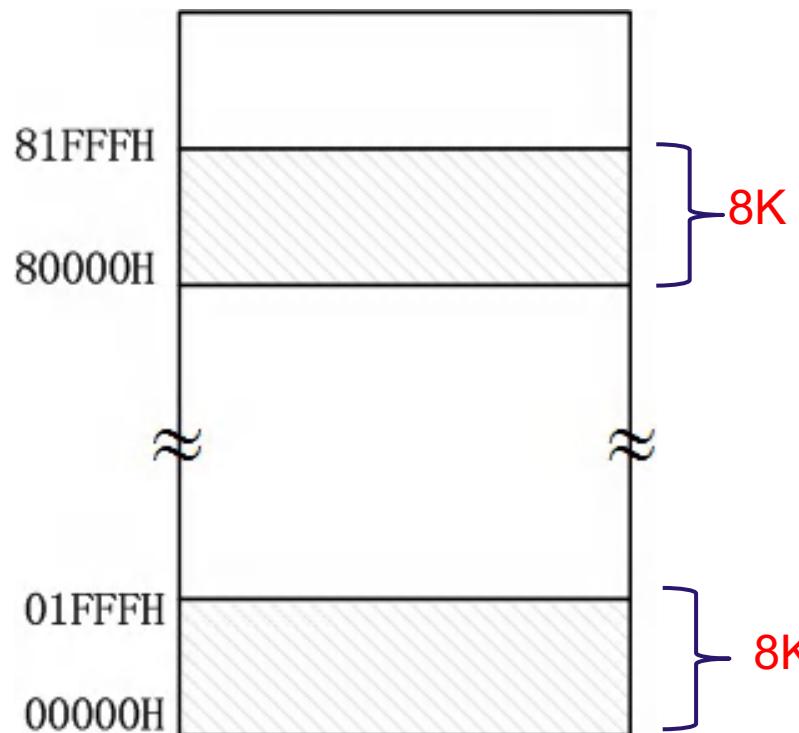
## ❖ 部分地址译码 8K存储空间



- ❖  $A_{19}$ 不参加地址译码：不管 $A_{19}$ 等于0还是等于1，只要 $A_{18}\sim A_{13}$ 满足条件，该芯片就会被选中 → 地址重叠

# 存储器地址译码电路设计

- ❖ 地址重叠：由于部分地址译码，一个芯片占用两段或多段地址空间



采用部分地址译码，只允许在一段地址空间内挂接存储器芯片。

EG，6264的地址范围为：  
00000H~01FFFFH

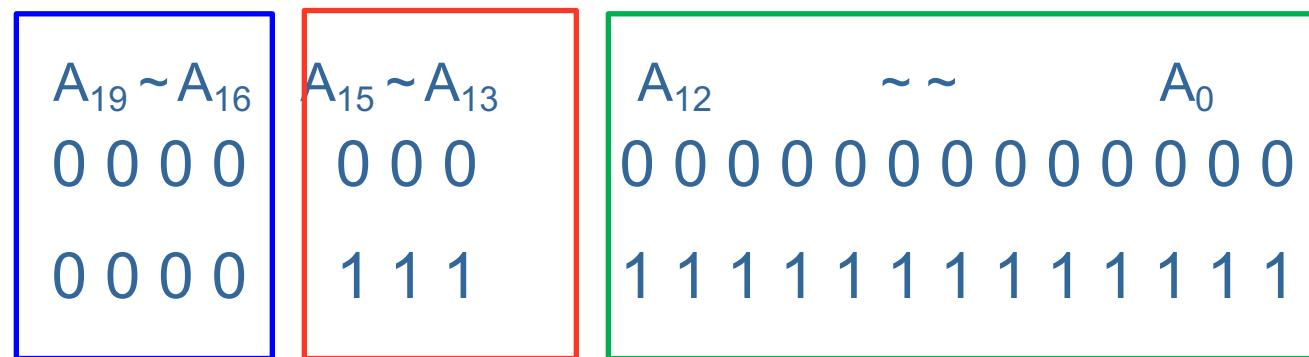
采用部分地址译码（不用A19），  
则80000H~81FFFFH这段存储空  
间被浪费，不能再挂接芯片

部分地址译码中，被省略的高位地  
址线不参加译码，只需要其它地址  
线参加译码

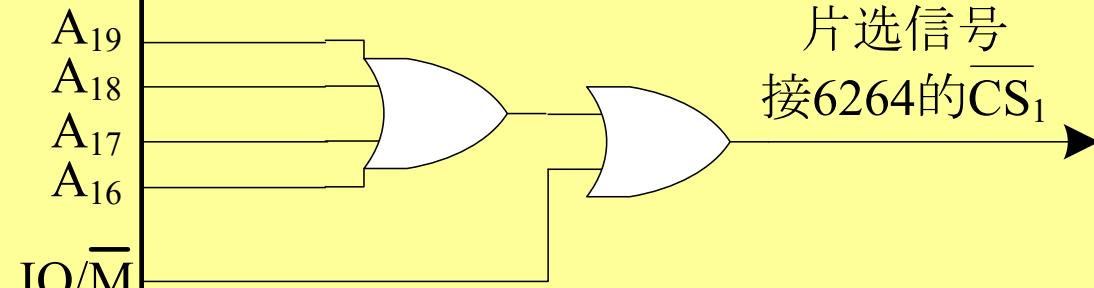
重叠空间的个数=  $2^n$   
 $n$  为被省略的地址线个数

# 部分地址译码例题

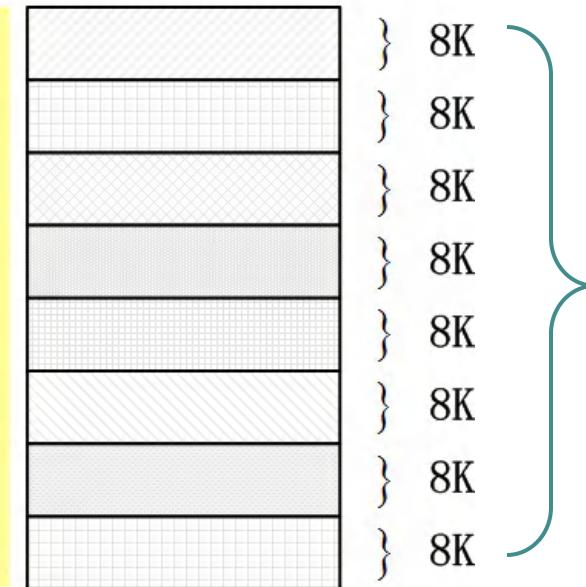
**EG8**：在8088CPU工作在最小方式组成的微机应用系统中，扩充设计**8KB**的SRAM电路，SRAM芯片用Intel 6264。若分配给该SRAM的地址范围 **00000H ~ 0FFFFH**，片选信号为低电平有效。请用部分地址译码方法设计该SRAM存储器的片选信号形成电路。



## 8088最小方式BUS



$A_{15}$	$A_{14}$	$A_{13}$	地址范围
0	0	0	00000H ~ 01FFFFH
0	0	1	02000H ~ 03FFFFH
0	1	0	04000H ~ 05FFFFH
0	1	1	06000H ~ 07FFFFH
1	0	0	08000H ~ 09FFFFH
1	0	1	0A000H~0BFFFFH
1	1	0	0C000H~0DFFFFH
1	1	1	0E000H ~ 0FFFFH



64K  
00000H  
OFFFH

# 存储器地址译码电路设计

## ❖ 线选译码

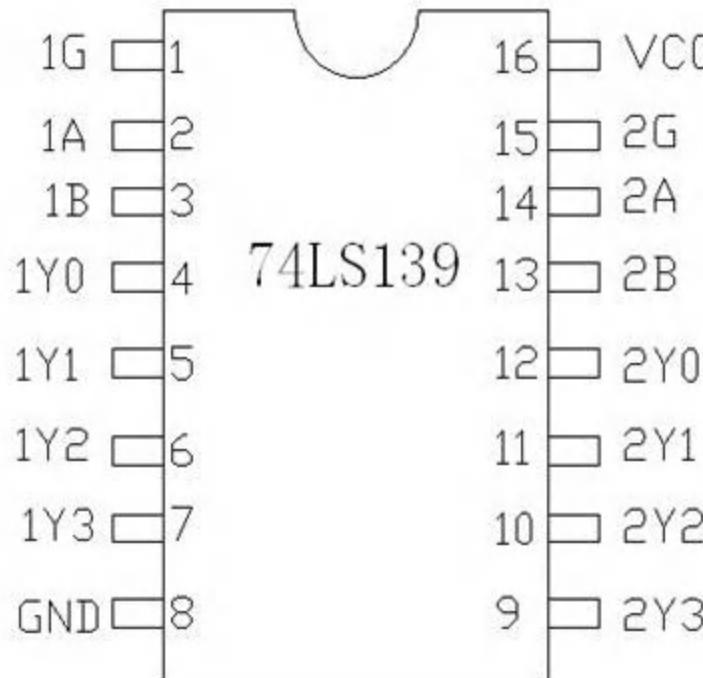
- 只用少数几根高位地址线进行芯片的译码，且每根负责选中一个芯片（组）；
- 虽构成简单，但地址空间严重浪费；
- 必然会出现地址重复；
- 一个存储地址会对应多个存储单元；
- 多个存储单元共用的存储地址不应使用。

# 存储器地址译码电路设计

用译码器生成译码电路

# 存储器地址译码电路设计

## ◆译码器 74LS139



输入			输出			
G	B	A	Y0	Y1	Y2	Y3
H	X	X	H	H	H	H
L	L	L	L	H	H	H
L	L	H	H	L	H	H
L	H	L	H	H	L	H
L	H	H	H	H	H	L

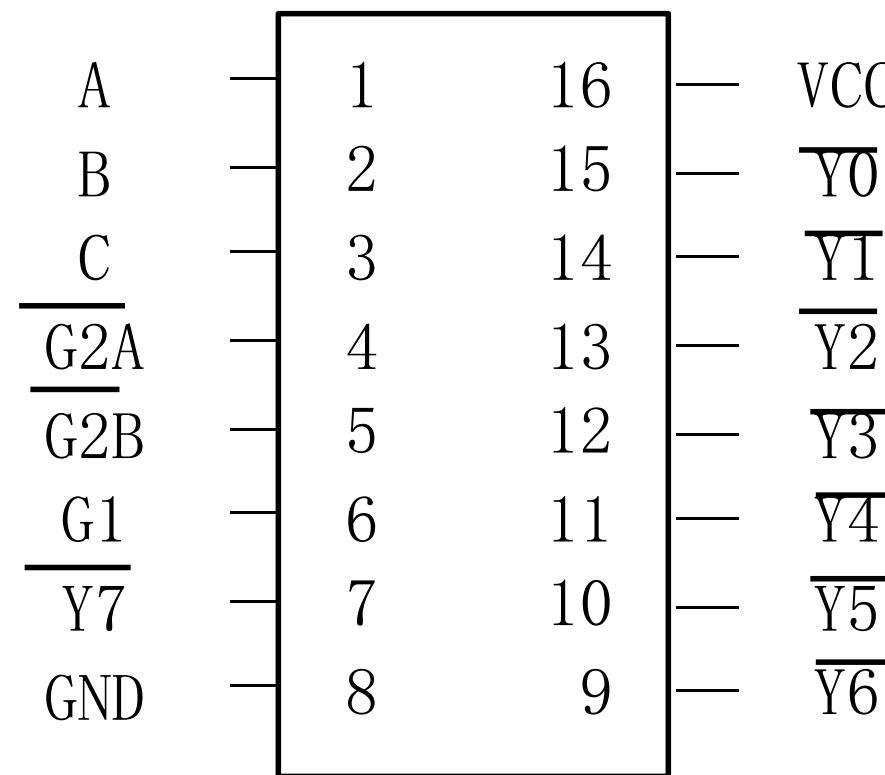
H=高电平 L=低电平 X=任意



A、B 译码地址输入端  
G1、G2 选通端（低电平有效）  
Y0 ~ Y3 译码输出端（低电平有效）

# 存储器地址译码电路设计

## ❖ 译码器 74LS138





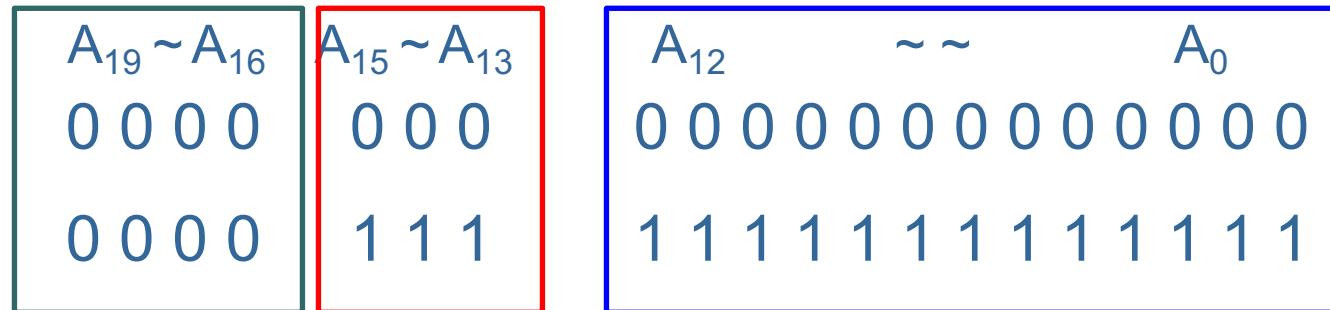
# 存储器地址译码电路设计

74LS138的真值

G <sub>2B</sub>	G <sub>2A</sub>	G <sub>1</sub>	C B A	Y <sub>7</sub> ~ Y <sub>0</sub>	有效输出
0	0	1	0 0 0	1 1 1 1 1 1 0	Y <sub>0</sub>
0	0	1	0 0 1	1 1 1 1 1 1 0 1	Y <sub>1</sub>
0	0	1	0 1 0	1 1 1 1 1 0 1 1	Y <sub>2</sub>
0	0	1	0 1 1	1 1 1 1 0 1 1 1	Y <sub>3</sub>
0	0	1	1 0 0	1 1 1 0 1 1 1 1	Y <sub>4</sub>
0	0	1	1 0 1	1 1 0 1 1 1 1 1	Y <sub>5</sub>
0	0	1	1 1 0	1 0 1 1 1 1 1 1	Y <sub>6</sub>
0	0	1	1 1 1	0 1 1 1 1 1 1 1	Y <sub>7</sub>
其他值		x x x		1 1 1 1 1 1 1 1	无效

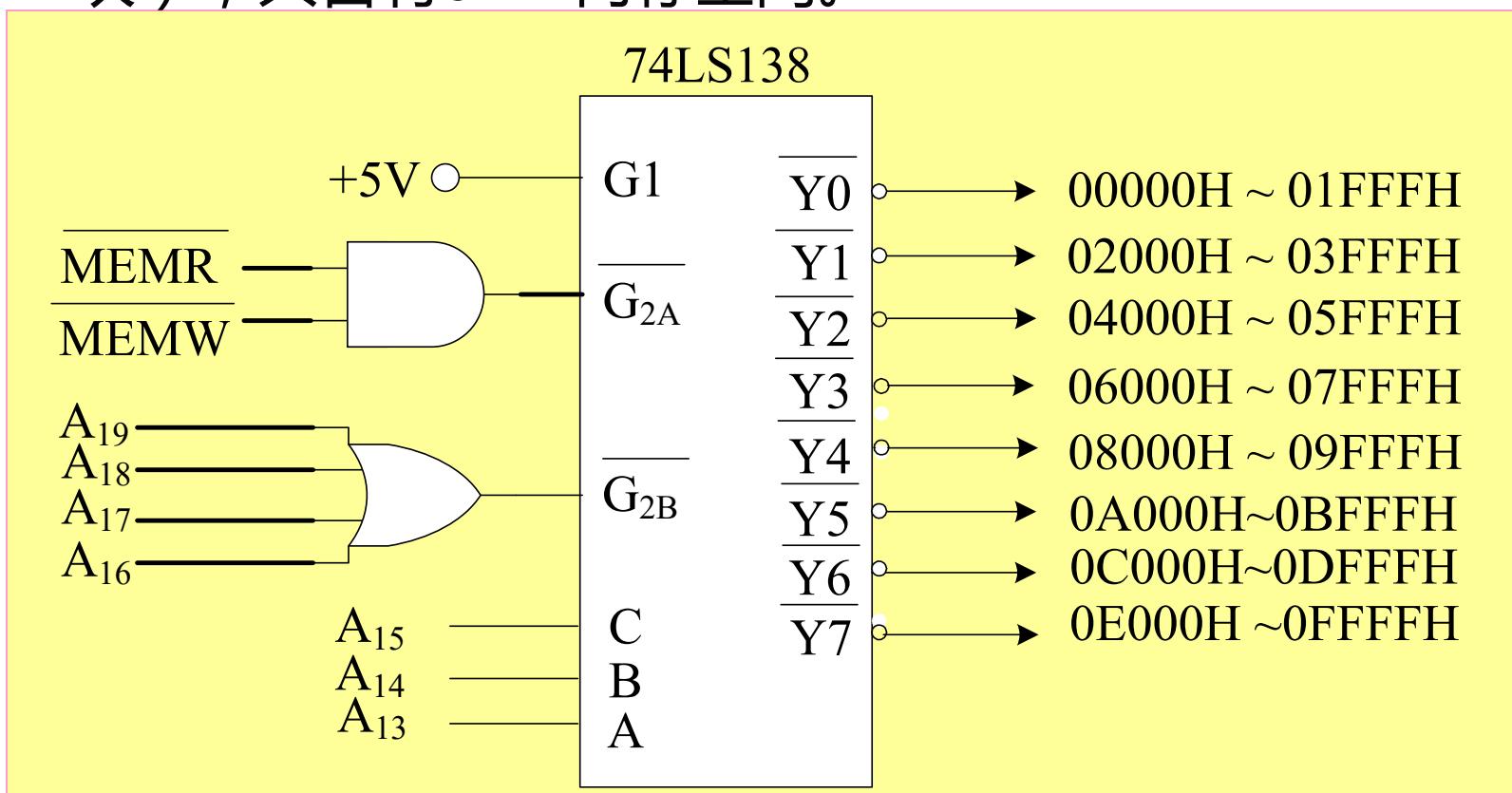
# 存储器地址译码电路设计

**EG9：** 在某8088微处理器系统中，需要用6264构成一个64KB的存储器。其地址分配在00000H~0FFFFH内存空间，地址译码采用全译码方式，用74LS138作译码器，请画出存储器译码电路。



# 存储器地址译码电路设计

**解：**根据题目已知条件和74LS138译码器的功能，设计的存储器译码电路如下图所示。图中74LS138的每一个输出端均与一块6264芯片的片选端相连，8个输出端分别选通1个8kB的存储空间（即1个6264模块），共占有64kB内存空间。





中山大學  
SUN YAT-SEN UNIVERSITY

2025-2026学年秋季学期

# 微机原理

Principle of Microcomputer

肖山林

xiaoshlin@mail.sysu.edu.cn

微电子科学与技术学院

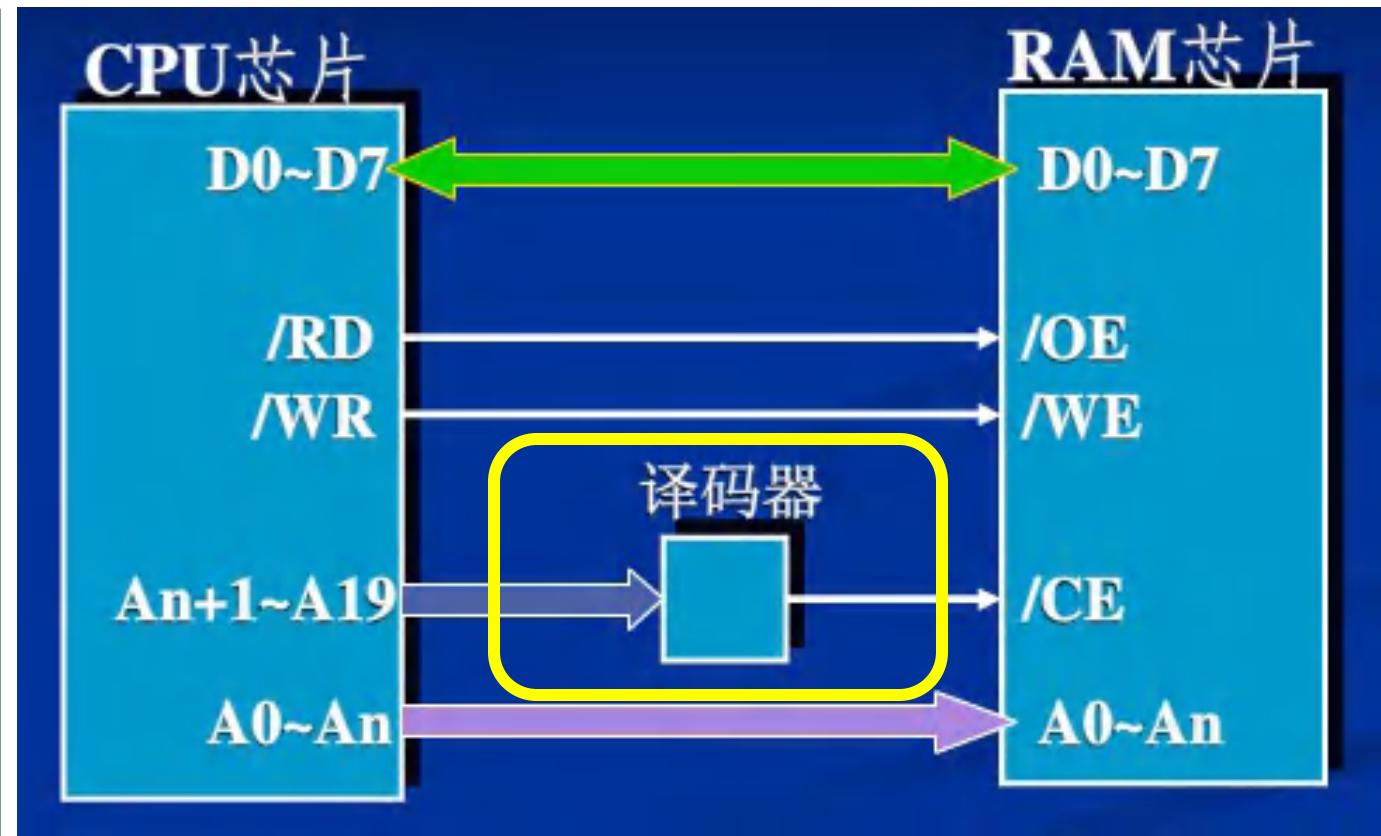


# 回顾：存储器地址译码电路设计

2:4译码器：  
74LS139

3:8译码器：  
74LS138

4:16译码器：  
74LS154



- ◆ 译码电路可以采用逻辑门电路，也可以采用各种译码器。

# 内容

1 存储器分类及主要指标

2 常用存储器芯片介绍

3 扩展存储器设计

4 存储器地址译码电路设计

5 存储器与CPU的连接

# 存储器与CPU的连接

## ◆ 8088系统中存储器的组成

8088 CPU的地址总线有20条，它的存储器是以字节为存储单元组成的，每个字节对应一个唯一的地址码，所以具有1MB的寻址能力。但**8088 CPU 只有8条数据线**，是准16位微处理器，所以存储器的组成与一般8位微机系统中存储器接口电路的设计方法是相同的。

# 存储器与CPU的连接

**EG10.** 在8088最大系统总线上扩充设计

8K字节的SRAM存储器电路。SRAM芯片选用Intel6264，起始地址从04000H开始，译码电路采用74LS138。

- (1)计算此RAM存储区的最高地址为多少？
- (2)画出此存储器电路与系统总线的连接图。

# 存储器与CPU的连接

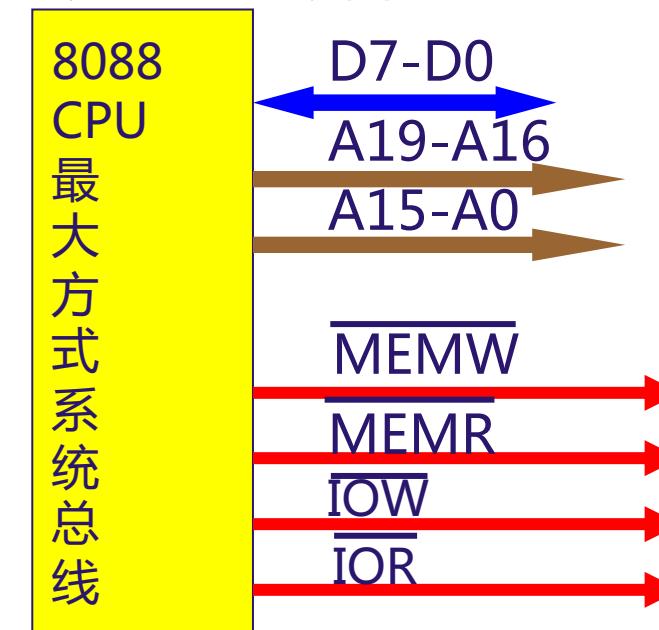
(1) 计算此RAM存储区的最高地址为多少？

因为Intel 6264存储容量为 $8K \times 8$ （字节），所以设计此存储电路共需1片6264芯片。因此最高地址为：

$$04000H + 01FFFH = 05FFFH$$

(2) 画出此存储器电路与系统总线的连接图

确定总线及总线信号  

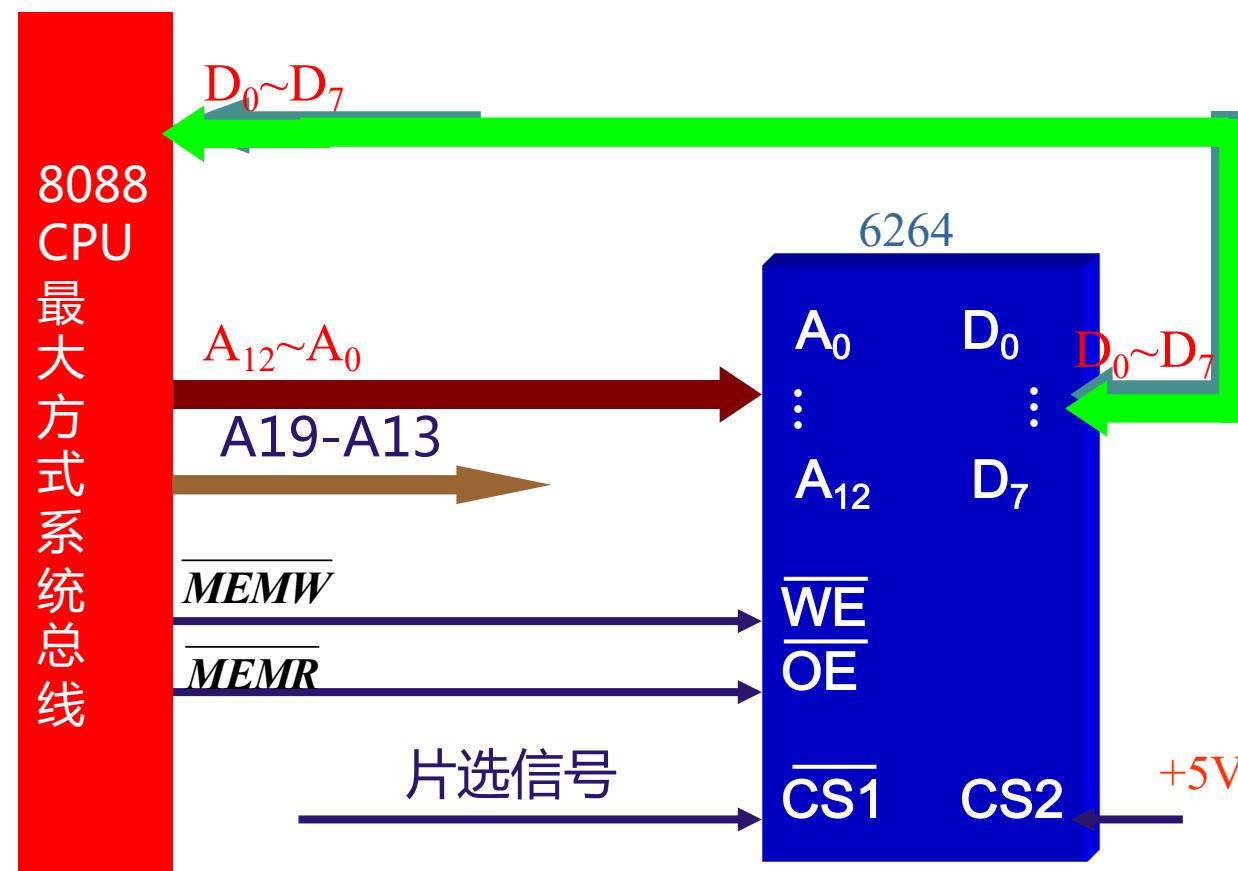





# 存储器与CPU的连接



电路连接

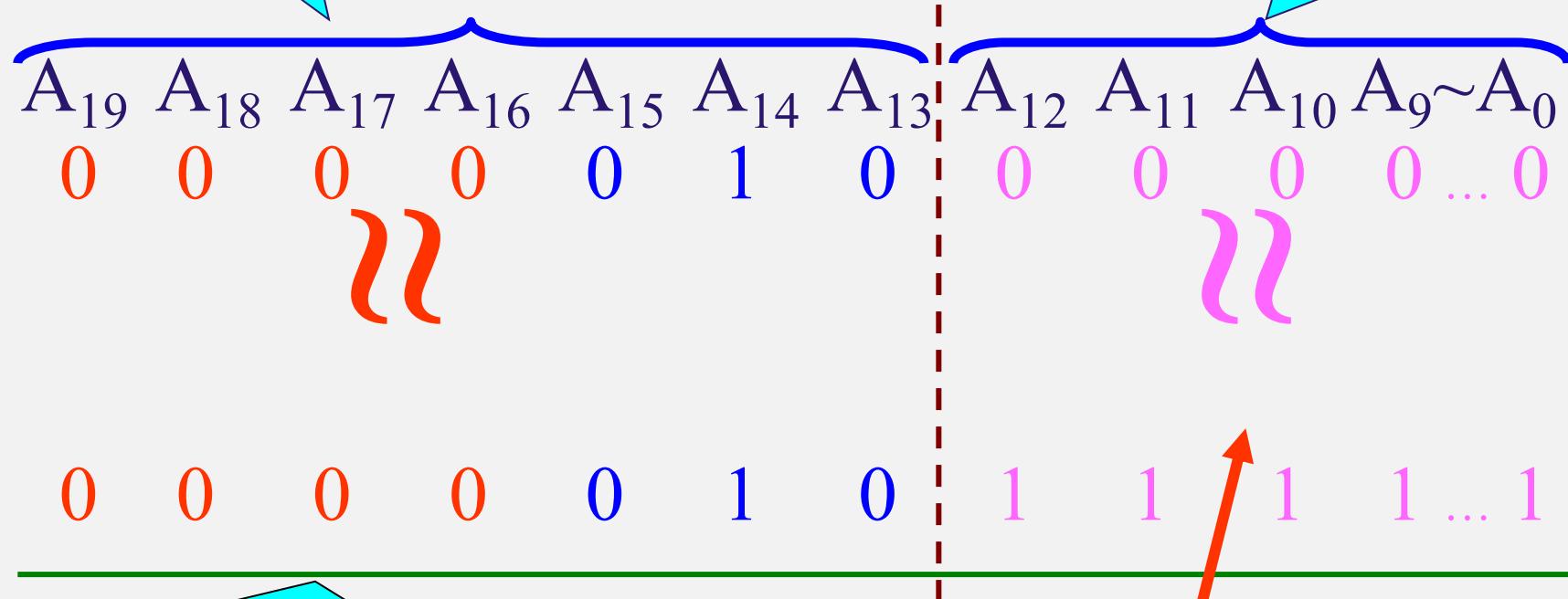


# 存储器与CPU的连接

## 地址分析

片外寻址

与6264芯片的A<sub>12</sub>~A<sub>0</sub>相连，作片内寻址

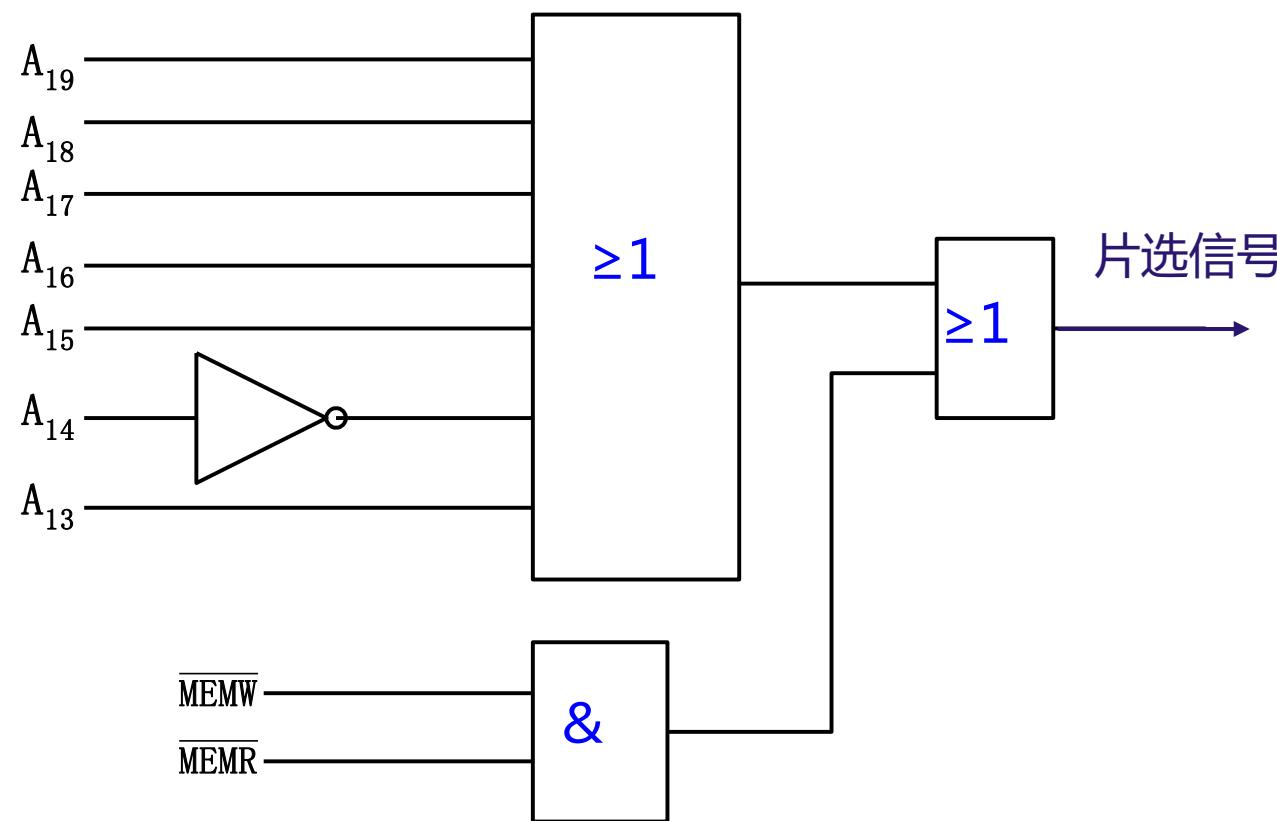


高位不变地址部分，去参加译码，作6264的  
 $\overline{CS1}$ 片选信号。

8K

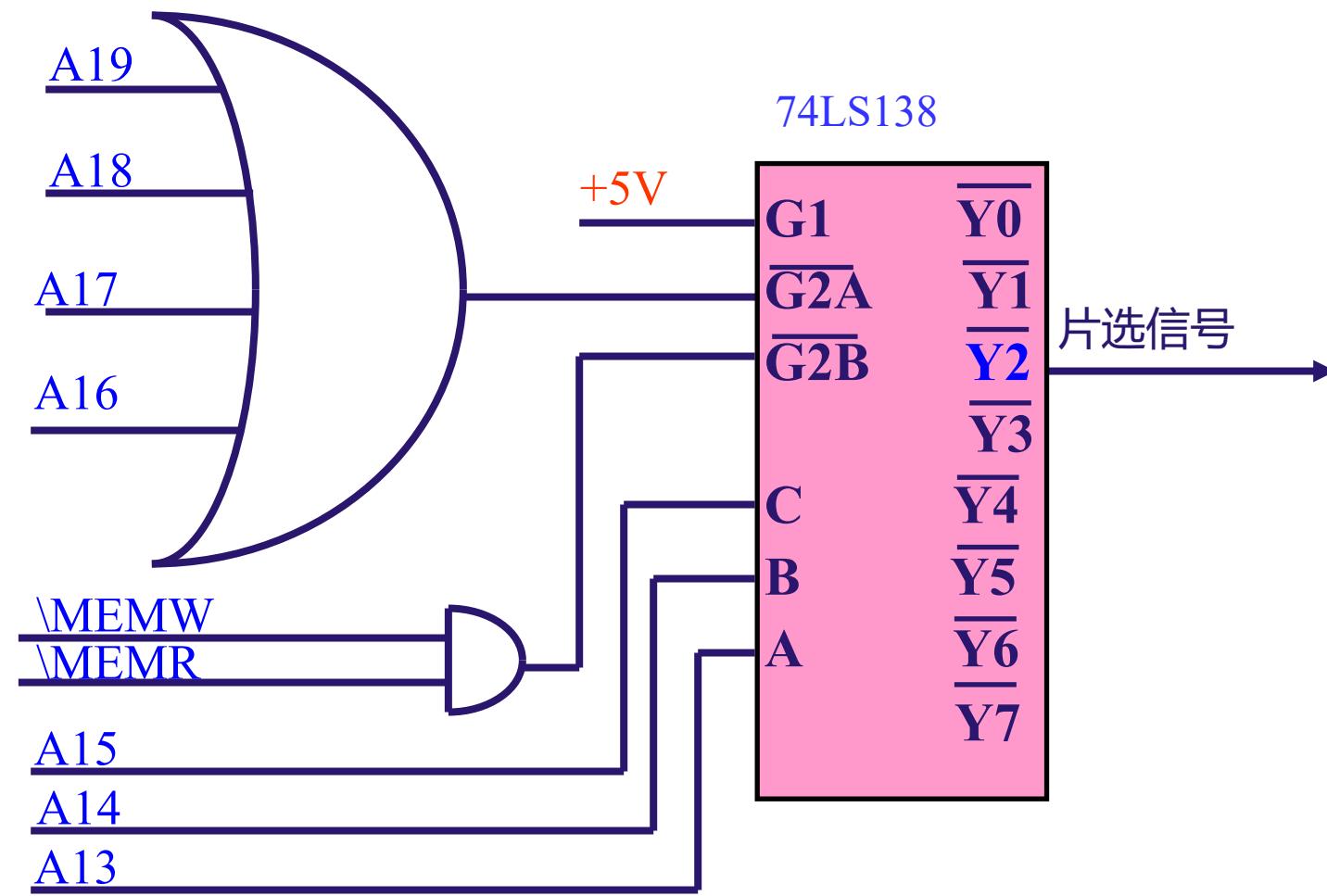
# 存储器与CPU的连接

## 译码电路设计方法1（用门电路译码）



# 存储器地址译码电路设计

译码电路设计方法2 (用74LS138专用译码器)



# 存储器与CPU的连接

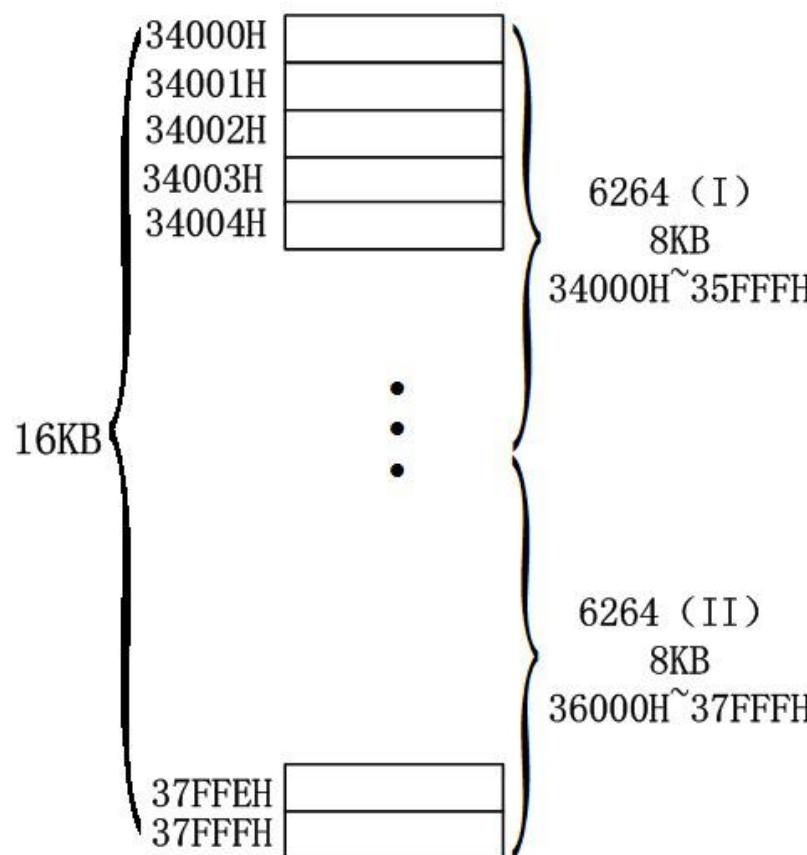
## ◆ 8086系统中存储器的组成

8086 CPU同8088 CPU一样，也有20条地址总线，其寻址能力达1MB。不同之处是8086 CPU 数据总线是16位的，与8086 CPU对应的1MB存储空间可分为两个512kB的存储体。**其中一个存储体由奇地址的存储单元(高字节)组成，另一个存储体由偶地址的存储单元(低字节)组成。**前者称为奇地址的存储体，后者称为偶地址的存储体。

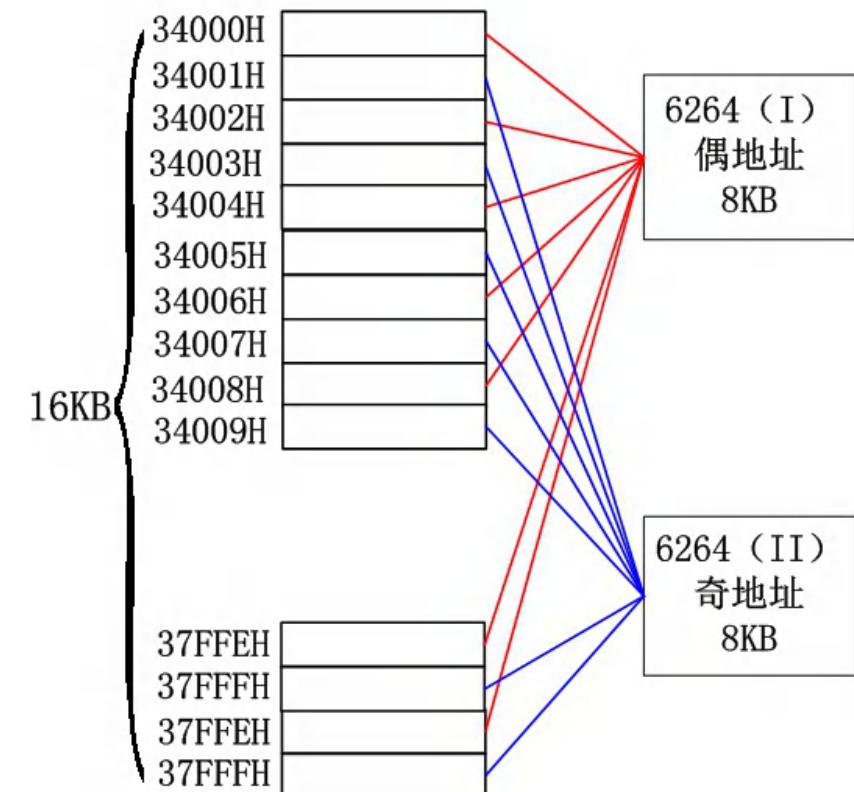


# 存储器与CPU的连接

8088存储器芯片设计

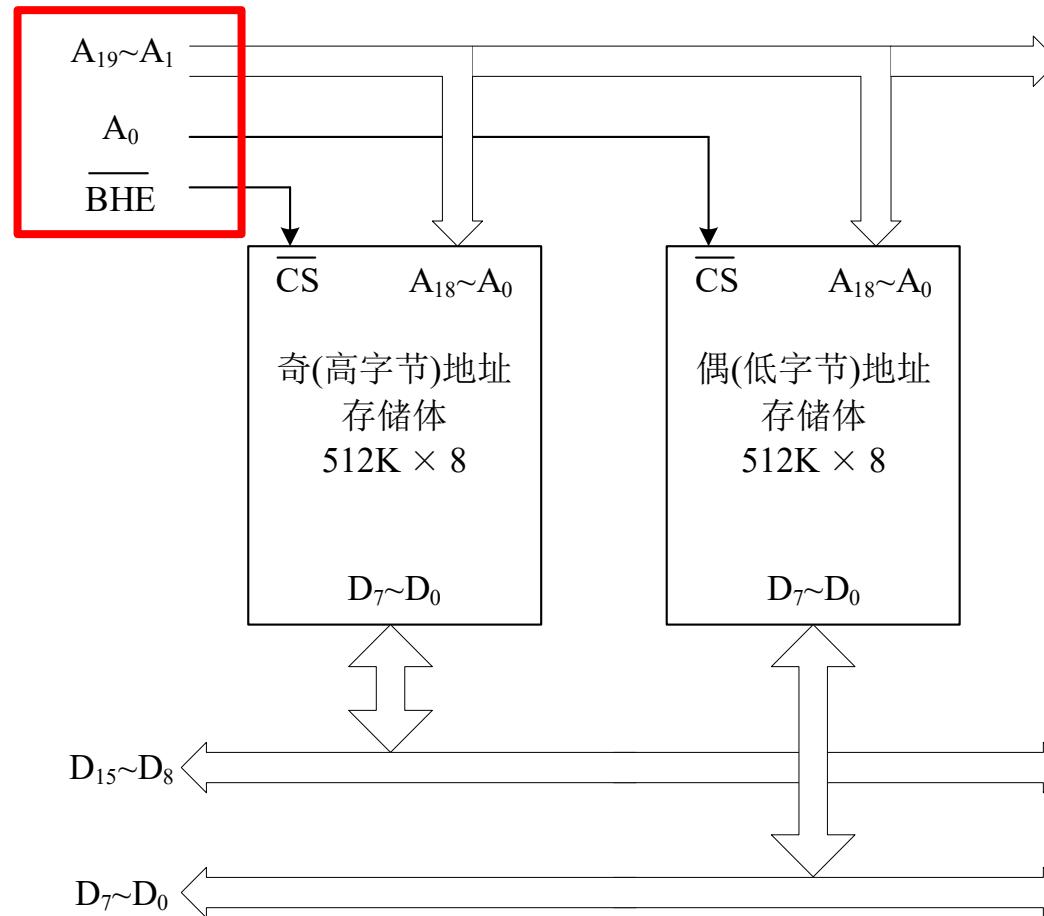


8086存储器芯片设计



# 存储器与CPU的连接

## ◆ 8086系统中存储器的组成



# 回顾： $\overline{\text{BHE}}$ 引脚

操作	$\overline{\text{BHE}}$	$A_0$	使用的数据引脚
读或写偶地址的一个字	0	0	$AD_{15} \sim AD_0$
读或写偶地址的一个字节	1	0	$AD_7 \sim AD_0$
读或写奇地址的一个字节	0	1	$AD_{15} \sim AD_8$
读或写奇地址的一个字	0	1	$AD_{15} \sim AD_8$ ( 第一个总线周期放低位数据字节 )
	1	0	$AD_7 \sim AD_0$ ( 第二个总线周期放高位数据字节 )

# 存储器与CPU的连接

**EG11** : 在8086最小方式系统中，利用2片Intel 6264构成连续的RAM存储区域，起始地址为00000H，采用全地址译码。

(1)可用的最高RAM地址；

(2)利用74LS155设计译码电路，画出此RAM电路与8086最小方式系统的连接图。

# 存储器与CPU的连接

解：Intel 6264的存储容量为 $8K \times 8$ ，因此由2片Intel 6264构成连续的RAM存储区域的总容量为 $2 \times 8KB = 16 KB = 04000H$ ，其可用的最高RAM地址为：

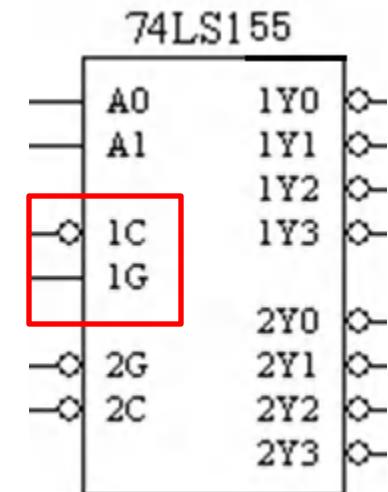
$$00000H + 04000H - 1 = 03FFFH$$

由于8086系统有16位数据总线，因此应将存储器模块分成两组：**奇片和偶片**，然后通过译码电路产生片选信号。



# 存储器与CPU的连接

Inputs			Outputs				
Select	Strobe	Data					
B	A	G1	C1	1Y0	1Y1	1Y2	1Y3
X	X	H	X	H	H	H	H
L	L	L	H	L	H	H	H
L	H	L	H	H	L	H	H
H	L	L	H	H	H	L	H
H	H	L	H	H	H	H	L
X	X	X	L	H	H	H	H

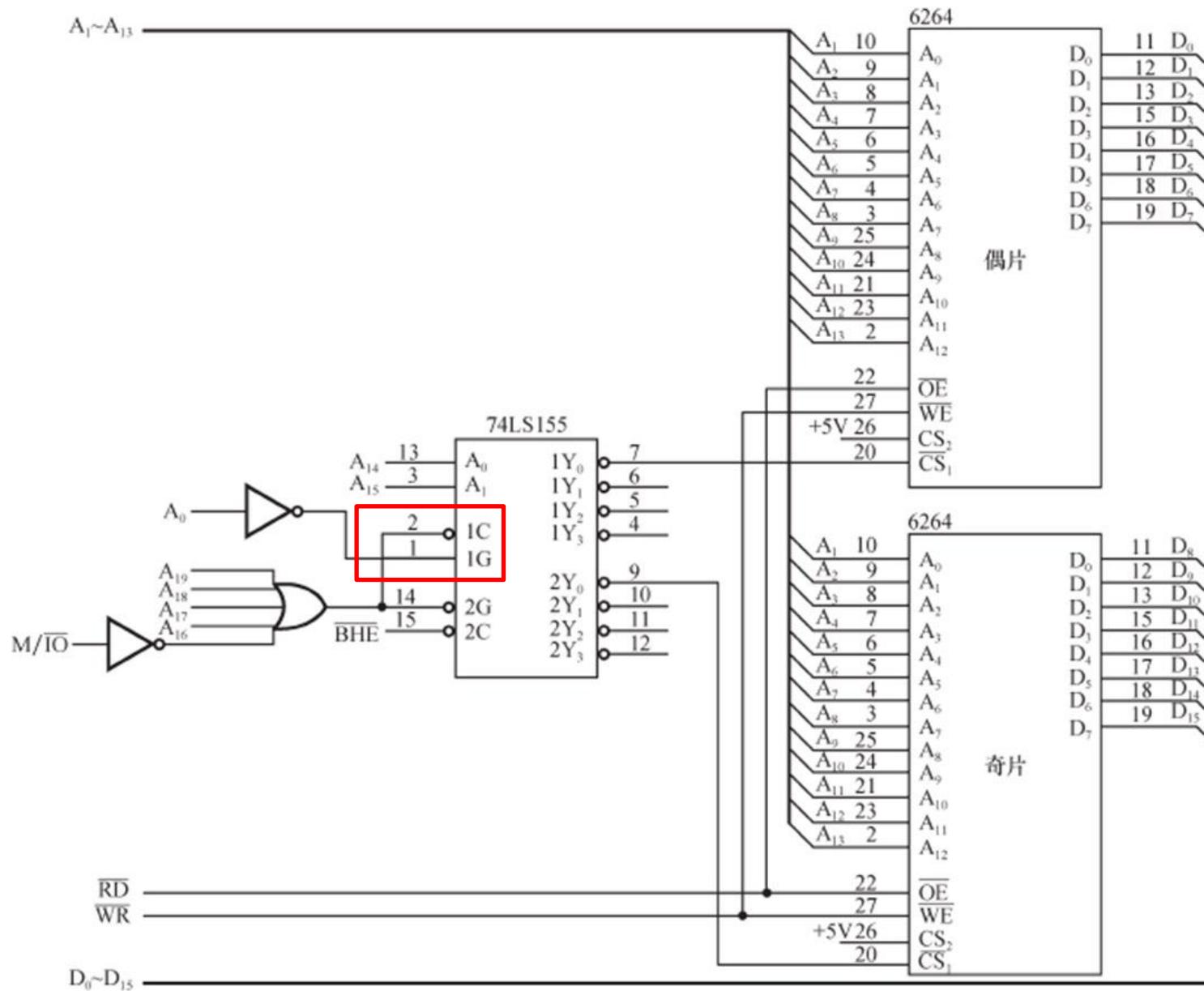


Inputs			Outputs				
Select	Strobe	Data					
B	A	G2	C2	2Y0	2Y1	2Y2	2Y3
X	X	H	X	H	H	H	H
L	L	L	L	L	H	H	H
L	H	L	L	H	L	H	H
H	L	L	L	H	H	L	H
H	H	L	L	H	H	H	L
X	X	X	H	H	H	H	H

C1 高电平有效  
G1 , G2 , C2 低电  
平有效



# 存储器与CPU的连接



# 存储器与CPU的连接

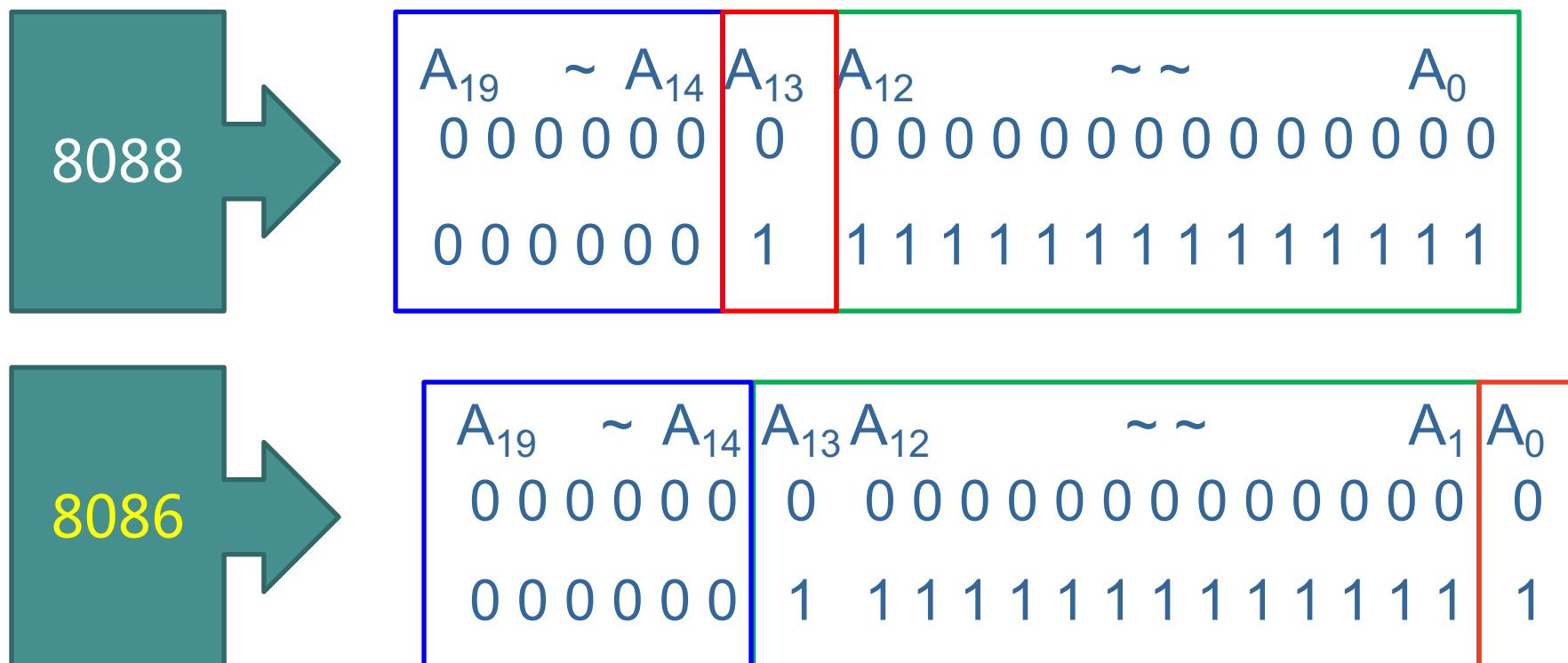


- ❖ 在存储器扩展设计时，一定要看清楚所挂接的CPU系统，是8086还是8088
- ❖ 8086：分奇偶片，奇地址和偶地址分别在两个存储器芯片上 A0作片选
- ❖ 8088：不分奇偶片，存储器芯片表示连续的地址空间

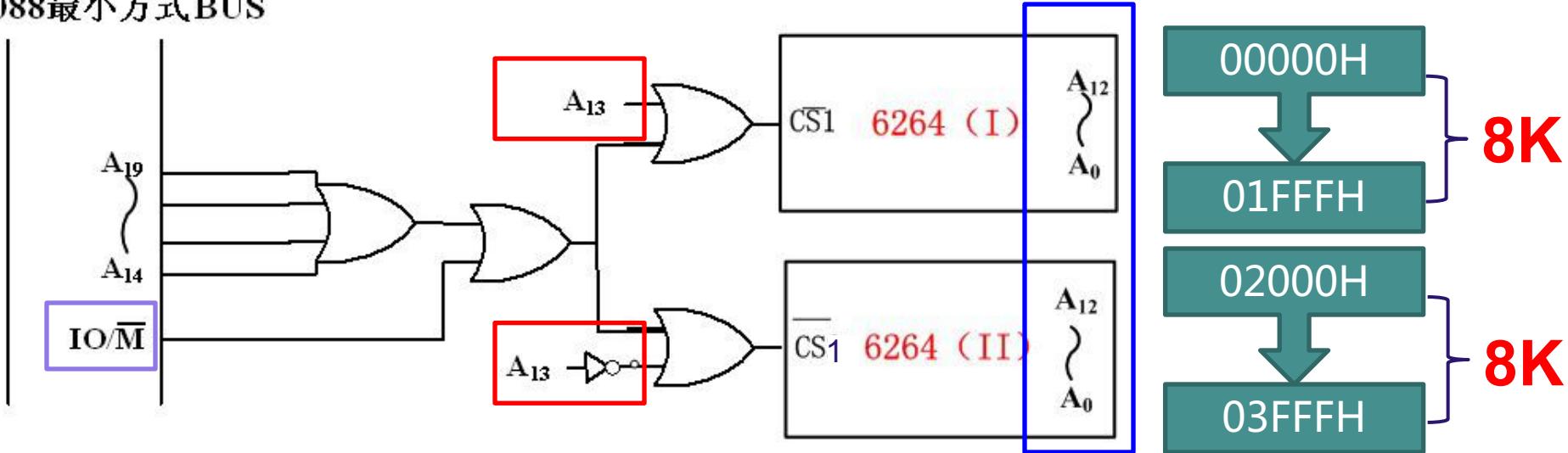


# 8086与8088存储器地址译码比较

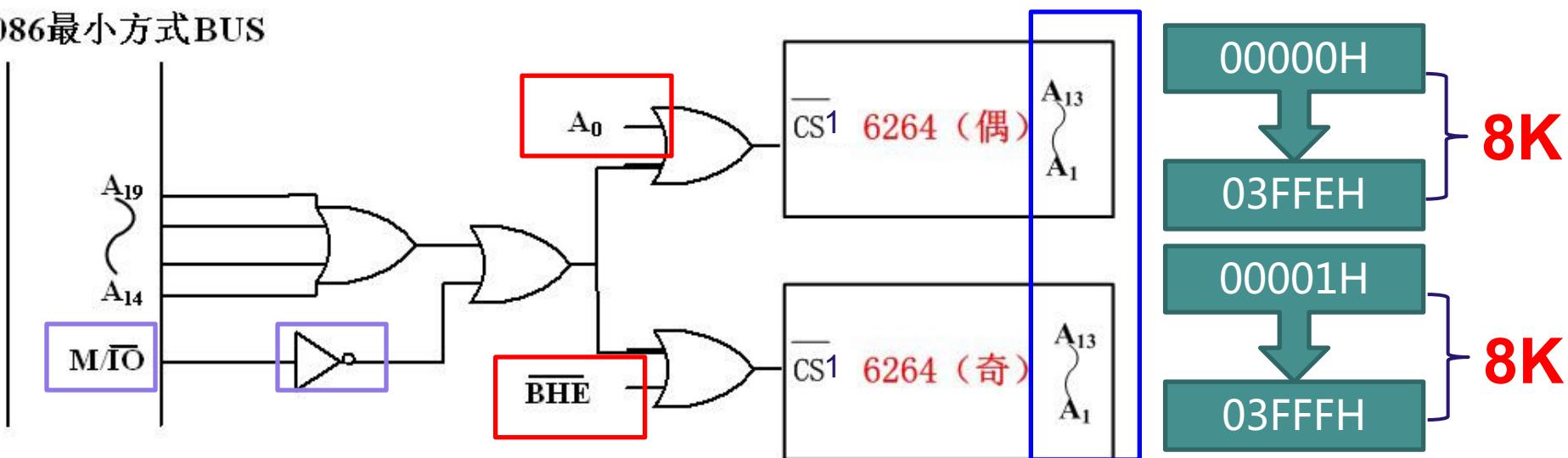
- 在8088最小系统下用6264扩展一段16KB的存储空间，地址为00000H~03FFFFH
- 在8086最小系统下用6264扩展一段16KB的存储空间，地址为00000H~03FFFFH



8088最小方式BUS



8086最小方式BUS



# 8086与8088存储器地址译码比较

- ❖ EG：在8088系统下用6264扩展一段16KB的存储空间，需要几片6264
- ❖ EG：在8086系统下用6264扩展一段16KB的存储空间，需要几片6264
- ❖ EG：在8088系统下用6264扩展一段8KB的存储空间，需要几片6264
- ❖ EG：在8086系统下用6264扩展一段8KB的存储空间，需要几片6264

# 本章作业

- ❖ 递交时间 : 2025.11.24 (一周后)
- ❖ 题目 : 6.2 ; 6.4 ; 6.5 ; 6.6 ; 6.8

# 第六章的知识点总结

## ❖ 1.概念性知识

- 存储器的概念，分类，性能指标，芯片

## ❖ 2.存储器扩展

- 位扩展
- 字节扩展
- 位与字节扩展

## ❖ 3.地址译码电路设计

- 全地址译码
- 部分地址译码
- 线地址译码

## ❖ 4.8086与8088存储器系统的异同