



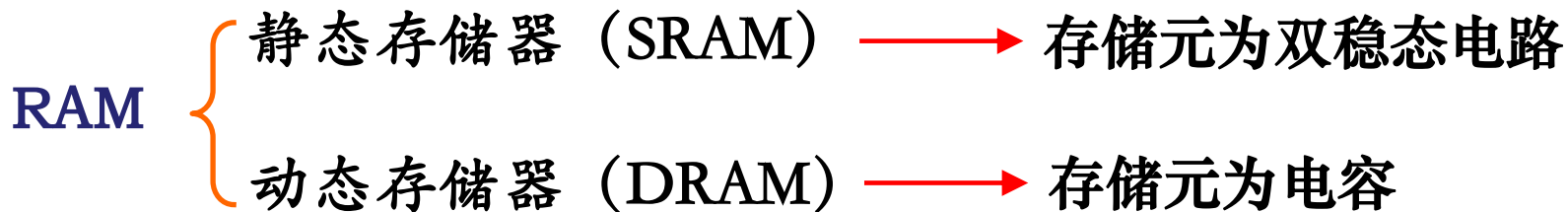
西安交通大学  
XI'AN JIAOTONG UNIVERSITY



# 随机存取存储器RAM



# 随机存取存储器 ( RAM )

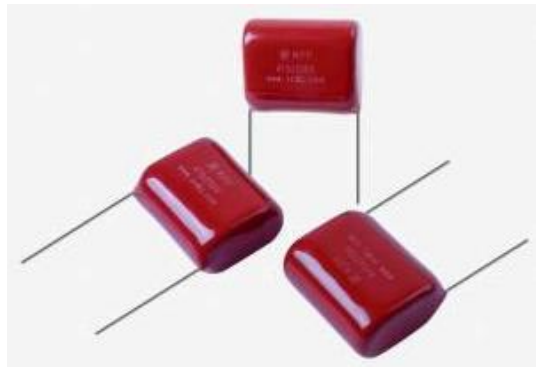


# DRAM的特点

- 存储元主要由电容构成；
- 主要特点：
  - ◆ 需要**定时刷新**



定时为存储元进行读或写  
操作（电容充电、放电）



# 静态随机存取存储器SRAM

# 静态随机存取存储器

存储元由双稳电路构成，存储信息稳定。

**示例芯片: SRAM 6262**

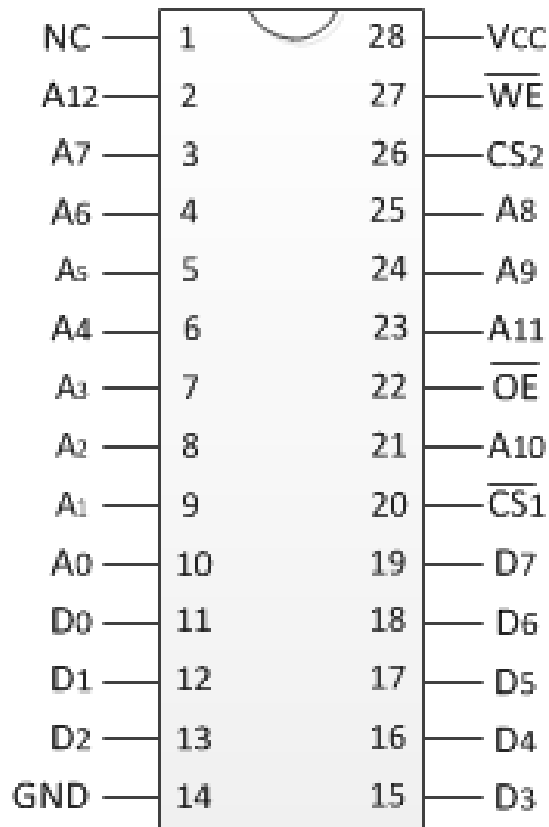
# 1. SRAM 6264

## ■ 容量:

- ◆  $8K \times 8b$

## ■ 主要引线:

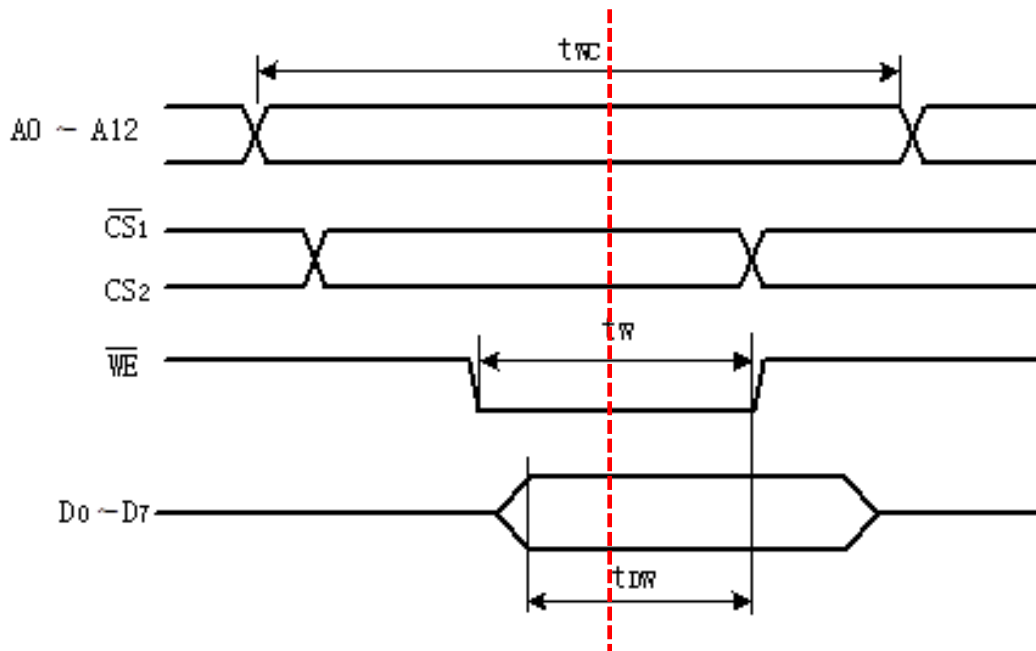
- ◆ 地址线:  $A_0$ ----- $A_{12}$ ;
- ◆ 数据线:  $D_0$ ----- $D_7$ ;
- ◆ 输出允许信号:  $\#OE$ ;
- ◆ 写允许信号:  $\#WE$ ;
- ◆ 选片信号:  $\#CS_1$ ,  $CS_2$ 。



# 6264的工作过程

- 读操作
- 写操作

写操作时序



## 2. 6264芯片与系统的连接

### ■ 存储器芯片与系统的连接分为两部分：

#### ◆ 确定要访问的存储芯片

- 系统中可能存在多片存储器芯片，要访问的单元只能存在于某一片芯片上。

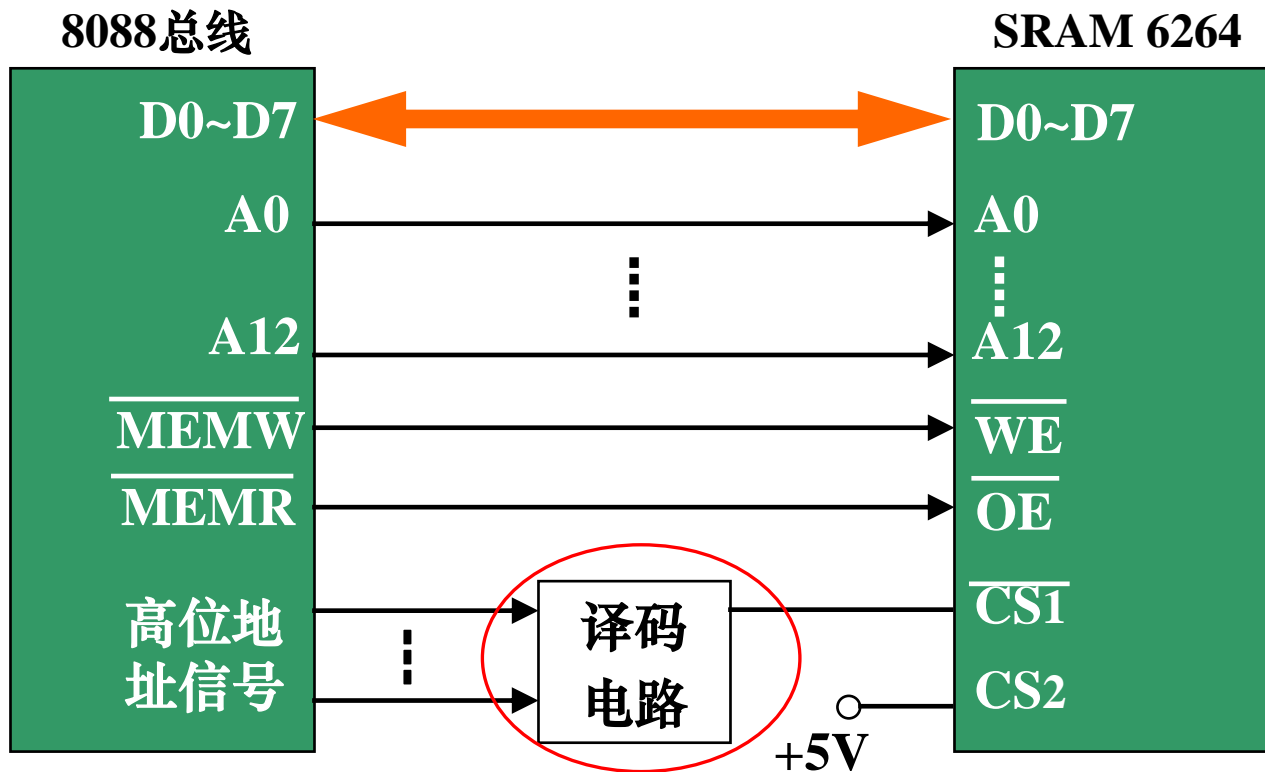
#### ◆ 找到芯片后，寻找该芯片上要访问的单元。

- 6264芯片上有8K个单元，每个单元在该芯片上有惟一的13位地址码
- 每片6264芯片上第一个单元在该芯片上的地址：0
- 每片6264芯片上最后一个单元在该芯片上的地址：8191

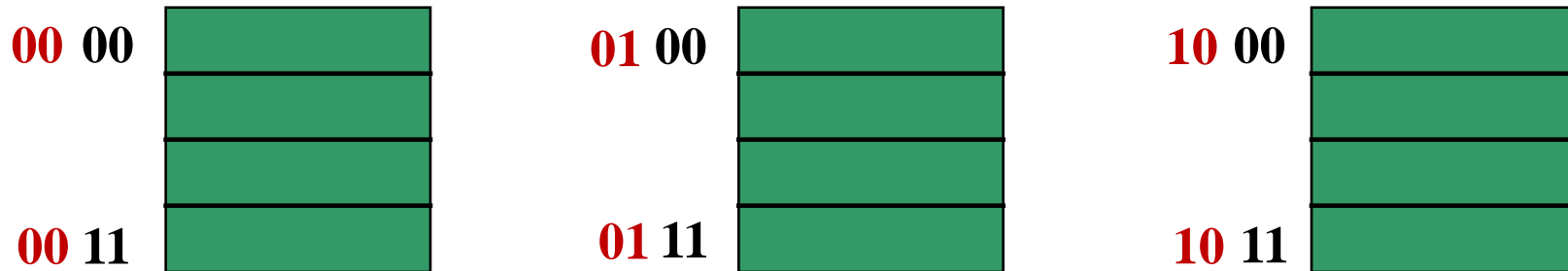
用芯片的13位地址码A0-A12寻址片内的每个单元



# 6264与系统的连接框架图



# 存储器编址



# 6264芯片的编址



芯片上所有的单元具有相同的高位地址（片选地址）

从片首地址到片尾地址，构成芯片在内存空间中占有的地址范围

# 3. 译码电路设计

## ■ 译码：

- ◆ 将输入的一组高位地址信号通过变换，产生一个有效的输出信号，用于选中某一个存储器芯片，从而确定了该存储器芯片在内存中的地址范围。

**将输入的一组二进制编码变换为一个特定的输出信号**

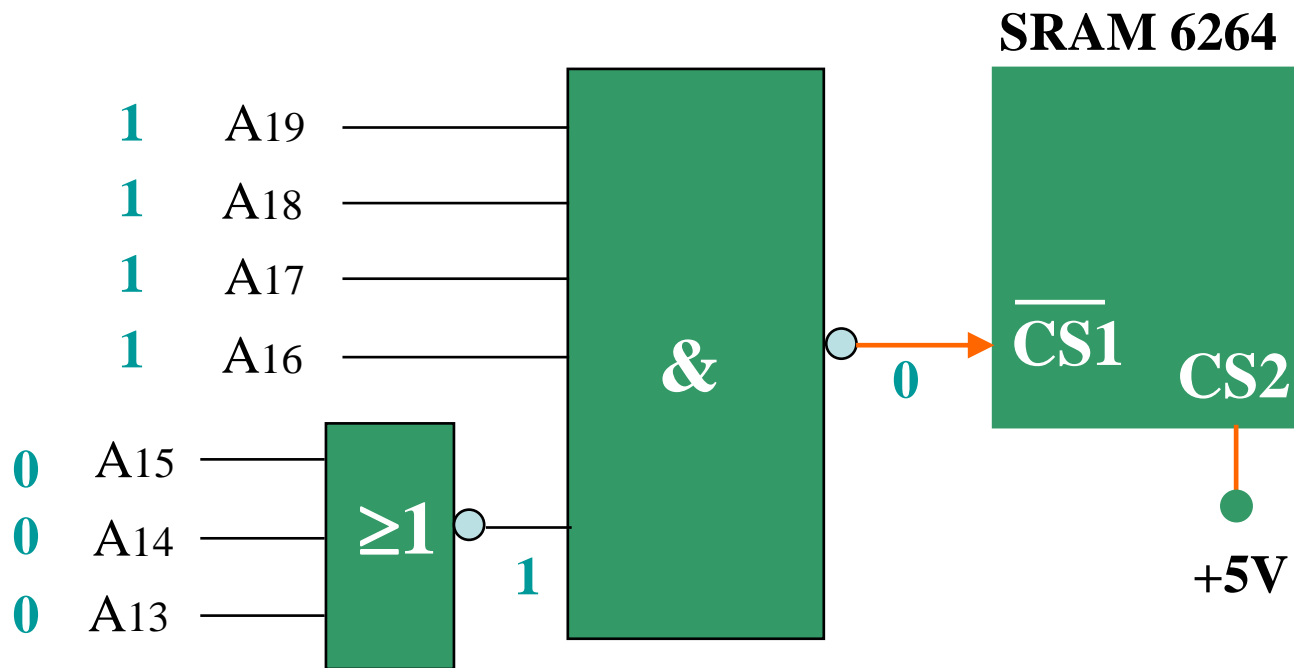
# 译码方式

- 全地址译码
- 部分地址译码

# 全地址译码

用全部的高位地址信号作为译码信号，使得存储器芯片的每一个单元都占据一个唯一的内存地址。

# 全地址译码例



## 6264芯片全地址译码例



该6264芯片的地址范围 = F0000H~F1FFFH



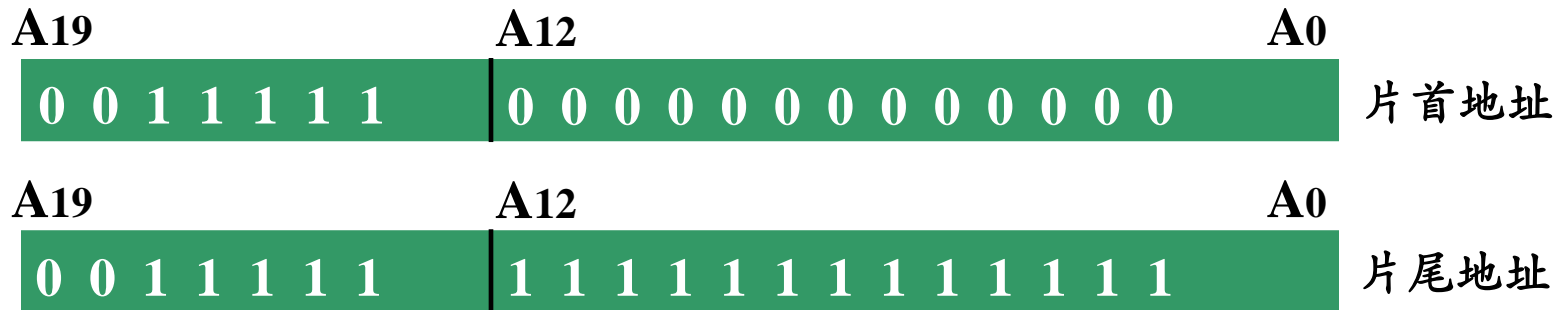
## 全地址译码例

■ 例：

- ◆ 已知某SRAM 6264芯片在内存中的地址为：3E000H~3FFFFH。试画出将该芯片连接到系统的译码电路。

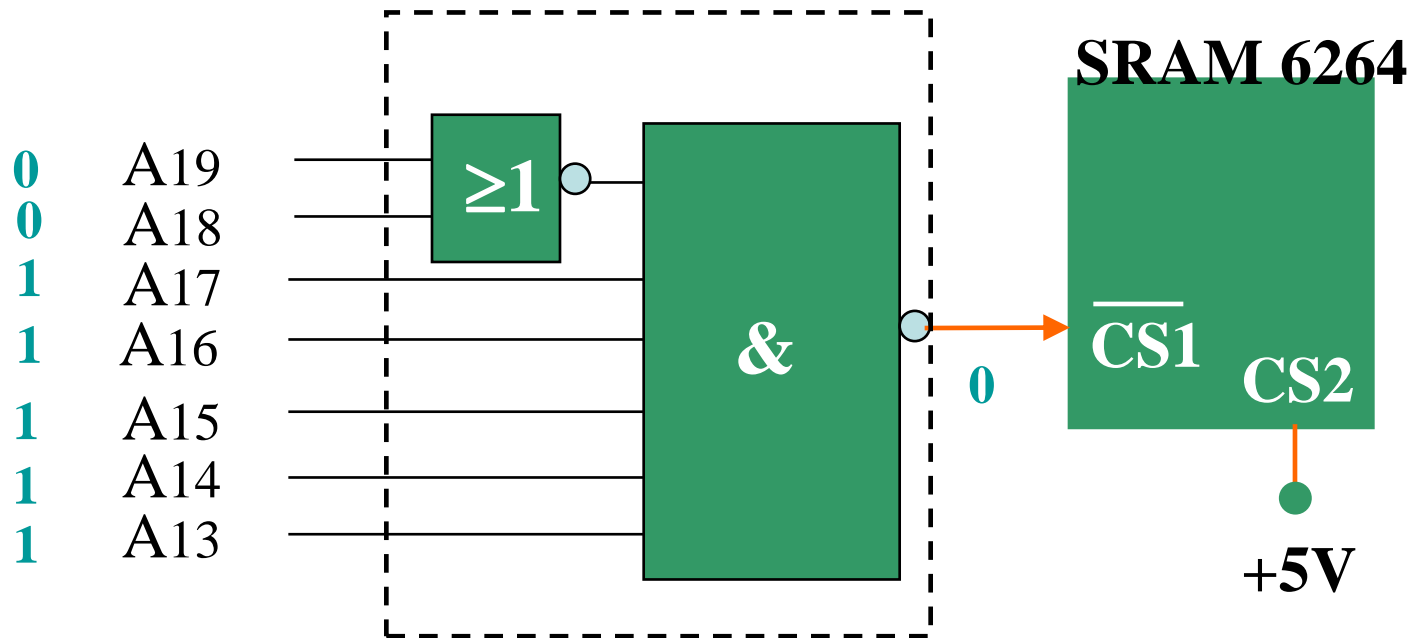
### ■ 设计步骤:

- ◆ 写出地址范围的二进制表示;
- ◆ 确定各高位地址状态;
- ◆ 设计译码器。



# 全地址译码例

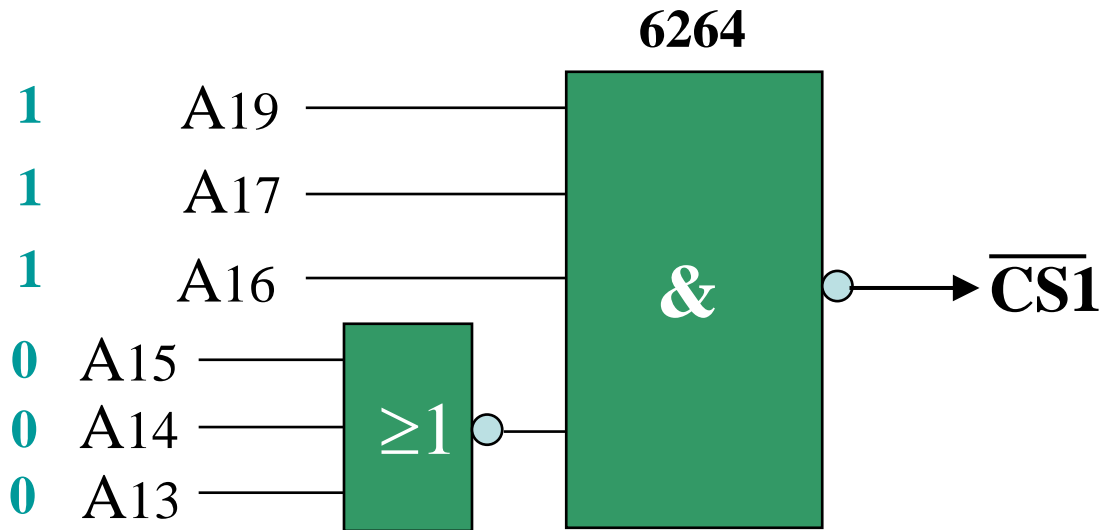
高位地址：0011111



# 部分地址译码

用部分高位地址信号（而不是全部）作为译码信号，使得被选中存储器芯片占有几组不同的地址范围。

# 部分地址译码例



高位地址:  $1 \times 11000$  ———  $1011000$ ,  $1111000$

两组地址: **F0000H ——— F1FFFH**

**B0000H ——— B1FFFH**

