

计算机组成原理与系统结构

第五章 存储体系

<http://jpkc.hdu.edu.cn/computer/zcyl/dzkjdx/>





5.3 主存储器与 CPU 的连接



背景知识——存储芯片简

介

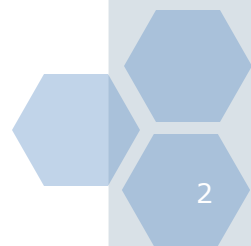


存储器容量扩展的三种方

法



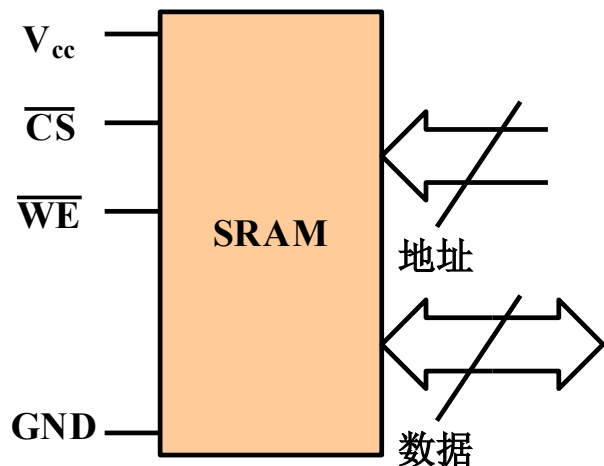
主存储器与 CPU 的连接



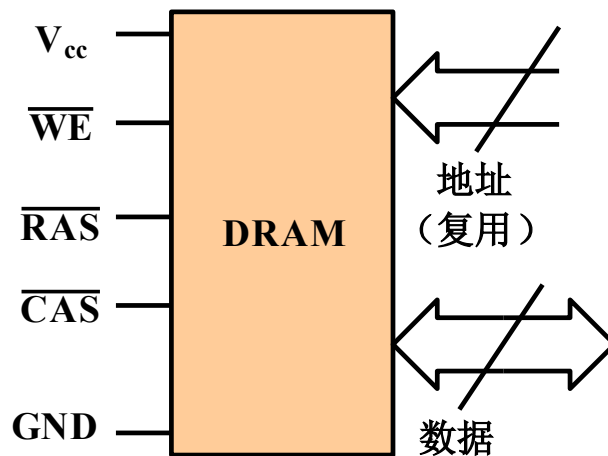


一、背景知识——存储芯片简介

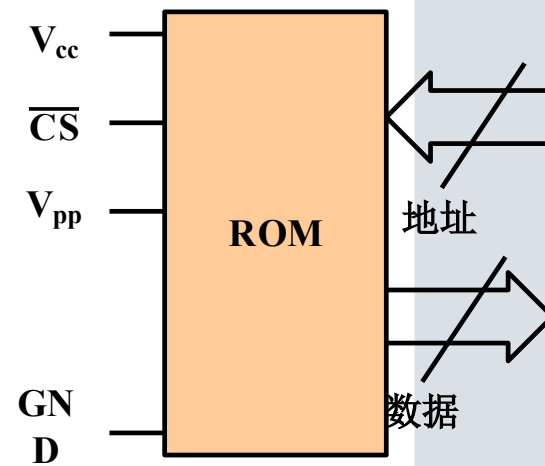
❖ 存储芯片的引脚封装



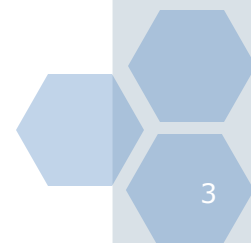
(A)SRAM芯片引脚



(B)DRAM芯片引脚

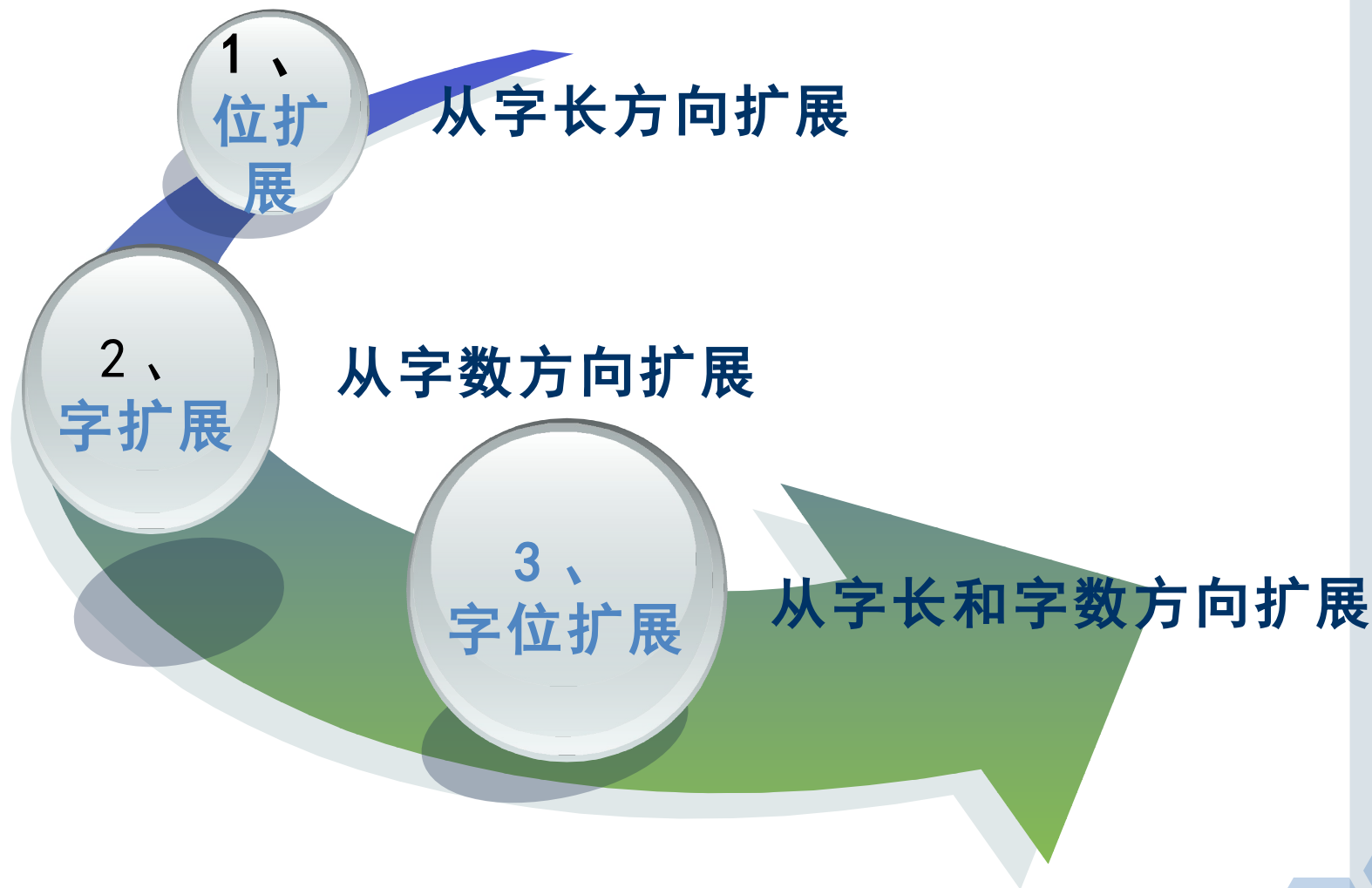


(C)ROM芯片引脚



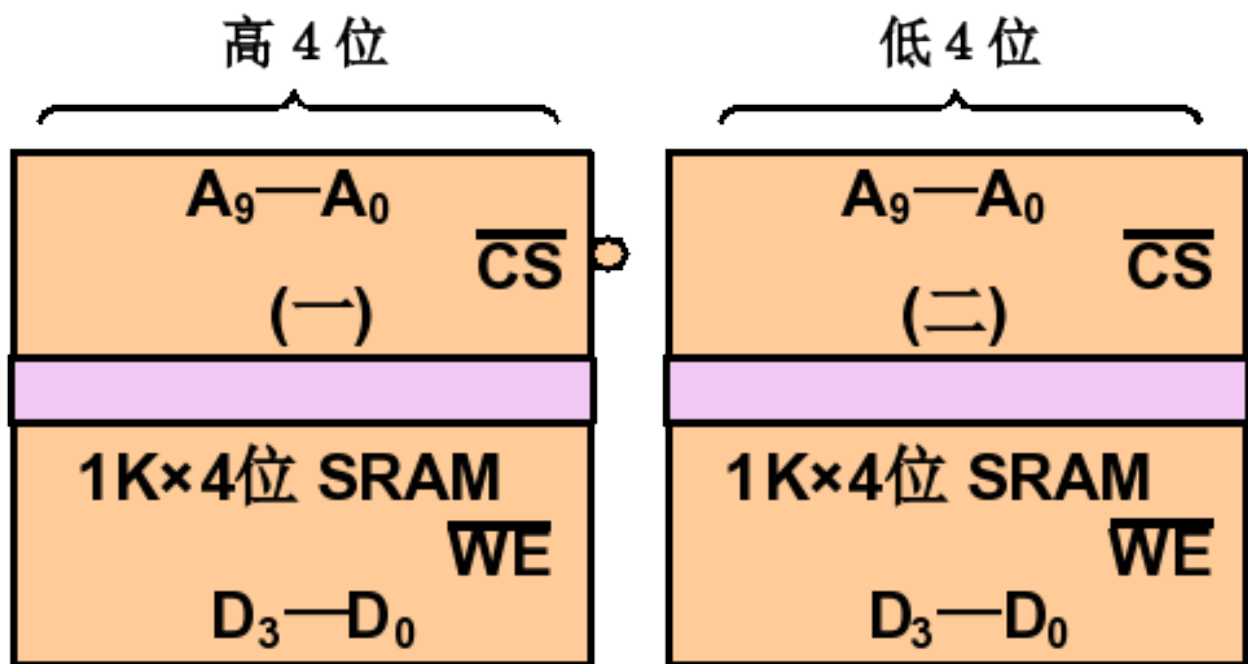


二、存储器容量扩展的三种方法



1、位扩展

❖ 要求：用 $1\text{K} \times 4$ 位的 SRAM 芯片 \rightarrow $1\text{K} \times 8$ 位的 SRAM 存储器

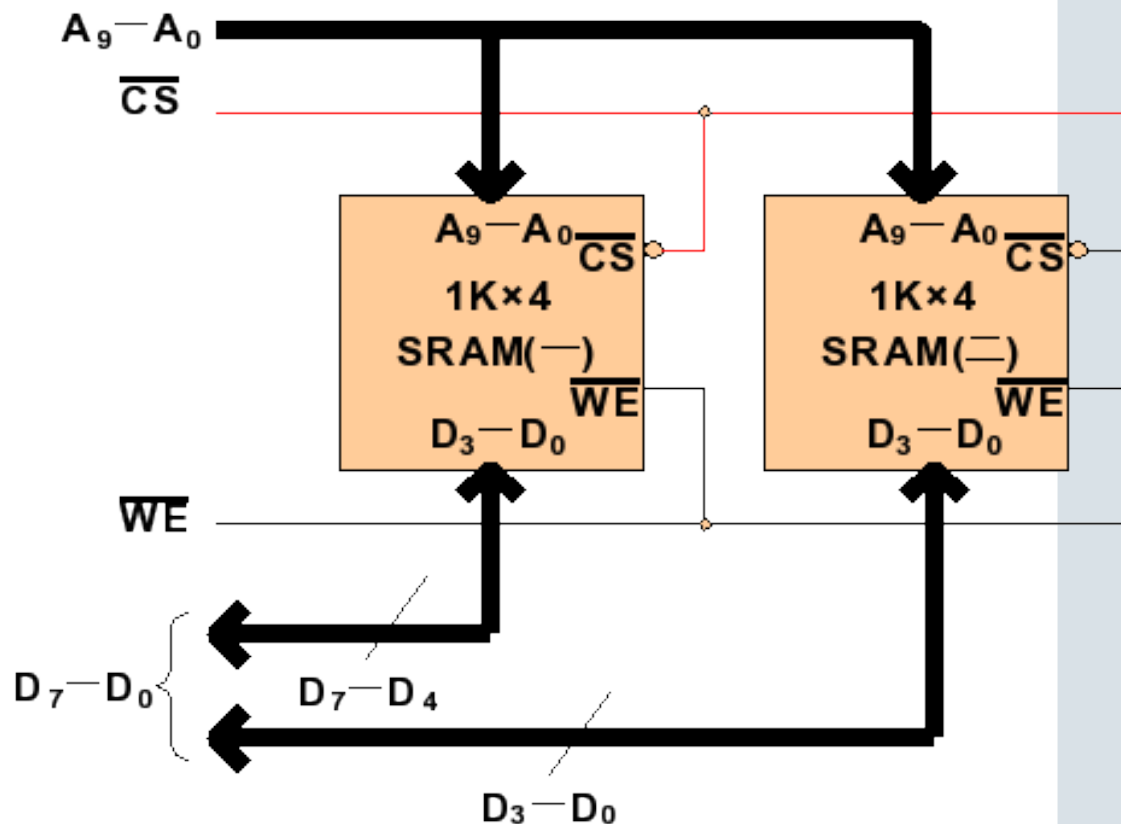


1、位扩展

❖ 容量 = $2^{10} \times 8$
位

❖ 举例验证：

读地址为 0
的存储单元的
内容



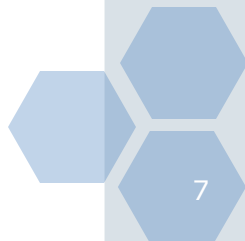


1、位扩展

❖ 要点：

❖ （1）芯片的地址线 A 、读写控制信号 WE# 、片选信号 CS# 分别连在一起；

❖ （2）芯片的数据线 D 分别对应于所搭建的存储器的高若干位和低若干位。

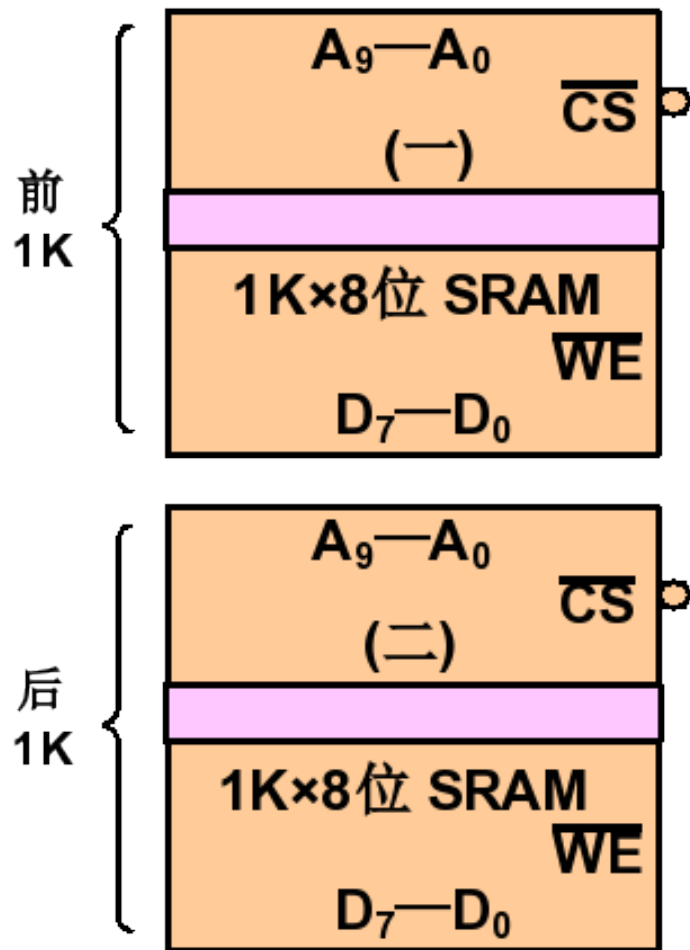




2、字扩展

❖要求：

用 $1\text{K} \times 8$ 位的 SRAM
芯片 \rightarrow $2\text{K} \times 8$ 位的
SRAM 存储器



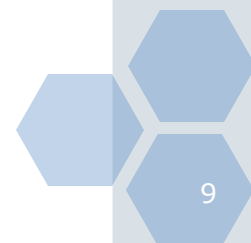


2、字扩展

❖ 分析地址：

- A_{10} 用于选择芯片
- $A_9 \sim A_0$ 用于选择芯片内的某一存储单元

A_{10}	A_9	\sim	A_0	
0	0	\sim	0	前 1K
\vdots		\vdots		
0	1	\sim	1	
1	0	\sim	0	后 1K
\vdots		\vdots		
1	1	\sim	1	



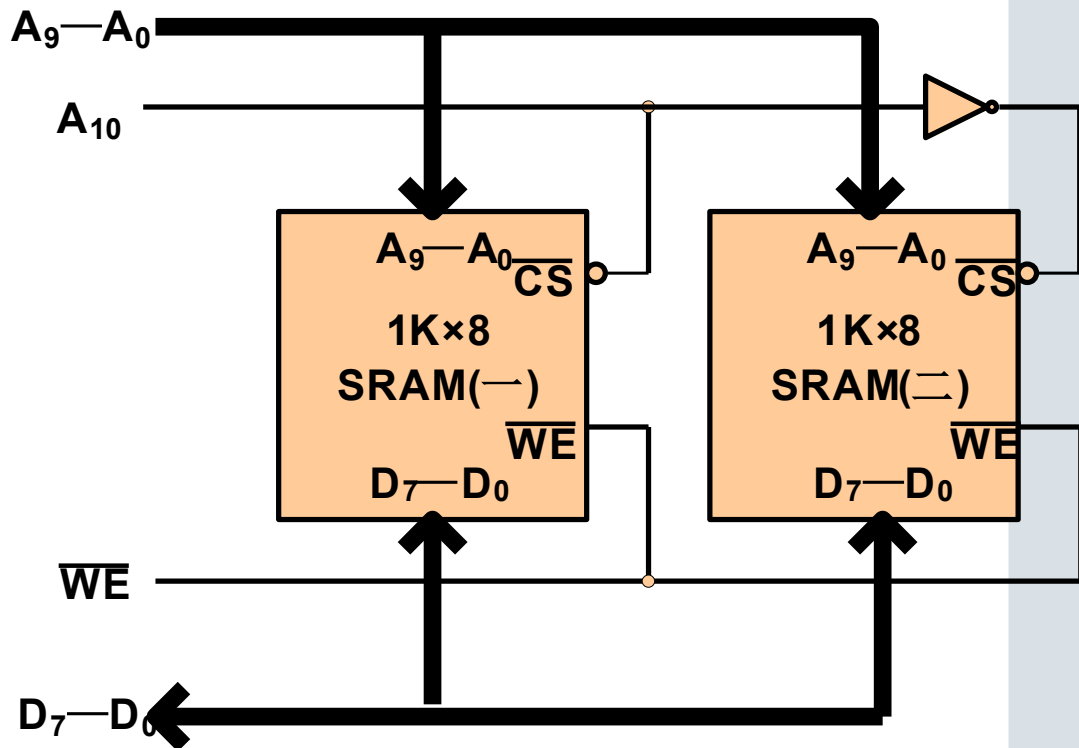


2、字扩展

❖ 容量 = $2^{11} \times 8$ 位

❖ 举例验证：

- 读地址为 0 的存储单元的内容
- 读地址为 10 ... 0 的存储单元的内容

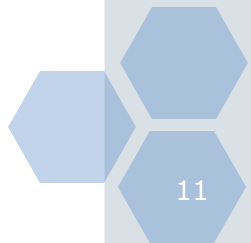




2、字扩展

❖ 要点：

- ❖ （1）芯片的数据线 D 、读写控制信号 WE# 分别连在一起；
- ❖ （2）存储器地址线 A 的低若干位连接各芯片的地址线；
- ❖ （3）存储器地址线 A 的高若干位作用于各芯片的片选信号 CS# 。





3、字位扩展

- ❖ 需扩展的存储器容量为 $M \times N$ 位，已有芯片的容量为 $L \times K$ 位 ($L < M, K < N$)

$$\frac{M \times N}{L \times K}$$

- ❖ 用 M/L 组芯片进行字扩展；
- ❖ 每组内有 N/K 个芯片进行位扩展。





三、主存储器与 CPU 的连接

- ❖ 1、根据 CPU 芯片提供的地址线数目，确定 CPU 访存的地址范围，
并写出相应的二进制地址码；
- ❖ 2、根据地址范围的容量，确定各种类型存储器芯片的数目和扩展方法；
- ❖ 3、分配 CPU 地址线。**CPU 地址线的低位**（数量＝存储芯片的地址线数量）直接连接存储芯片的地址线；**CPU 高位地址线**皆参与形成存储芯片的片选信号；
- ❖ 4、连接数据线、R/W# 等其他信号线，MREQ# 信号一般可用作



例 5 — 1

- ❖ 例 5-1：设 CPU 有 16 根地址线，8 根数据线，并用 MREQ# 作访存控制信号（低电平有效），用 R/W# 作读 / 写控制信号（高电平为读，低电平为写）。现有下列存储芯片：1K*4 位 SRAM；4K*8 位 SRAM；8K*8 位 SRAM；2K*8 位 ROM；4K*8 位 ROM；8K*8 位 ROM；及 3：8 译码器和各种门电路。
- ❖ **要求：**主存的地址空间满足下述条件：最小 8K 地址为系统程序区（ROM 区），与其相邻的 16K 地址为用户程序区（RAM 区），最大 4K 地址空间为系统程序区（ROM 区）。
- ❖ 请画出存储芯片的片选逻辑，存储芯片的种类、片数。
- ❖ 画出 CPU 与存储器的连接图。



解：首先根据题目的地址范围写出相应的二进制地址

A ₁₅	A ₁₄	A ₁₃	A ₁₂	A ₁₁	A ₁₀	A ₉	A ₈	A ₇	A ₆	A ₅	A ₄	A ₃	A ₂	A ₁	A ₀	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	} 最小8K 系统区
...	...															
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	} 相邻 16K 用户程 序区
...	...															
0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
...	...															
0	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	
...	...															} 最大4K 系统区
1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	
...	...															
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	

❖ 第二步：选择芯片

- 最小 8K 系统程序区 \leftarrow 8K*8 位 ROM , 1 片
- 16K 用户程序区 \leftarrow 8K*8 位 SRAM , 2 片;
- 4K 系统程序工作区 \leftarrow 4K*8 位 SRAM , 1 片。

❖ 第三步，分配 CPU 地址线。

- CPU 的低 13 位地址线 $A_{12} \sim A_0$ 与 1 片 8K*8 位 ROM 和两片 8K*8 位 SRAM 芯片提供的地址线相连；将 CPU 的低 12 位地址线 $A_{11} \sim A_0$ 与 1 片 4K*8 位 SRAM 芯片提供的地址线相连。

❖ 第四步，译码产生片选信号。

