

第五章 存储器与存储器系统扩展

- *本章内容提要
- ▶8086CPU对存储器的读/写操作过程;
- >半导体存储器的结构、工作原理与操作时序;
- >常用存储芯片与存储体容量扩展;
- ▶微机系统中的存储器设计。



本章及后续章节信号表示法的说明

- ➤低电平有效信号用信号名称加符号#表示,如: WR用WR#表示。
- ▶为表达方便,后续授课过程中,这两种方式混合使用。



第一节 8086引脚信号及其存储器操作时序

- 1. 引脚信号的复习;
- 2. 8086CPU的存储器操作时序
- 3. 8086CPU的最小与最大系统

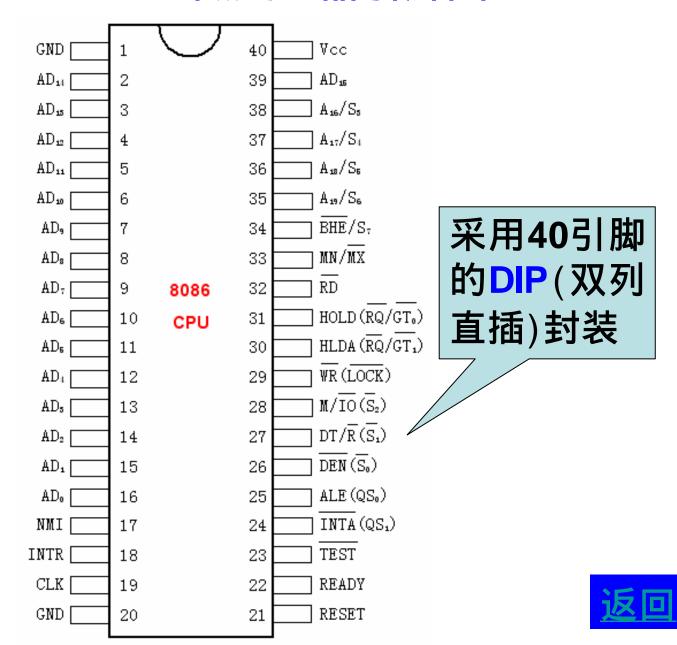
8086CPU引脚功能



- ❖主要引脚功能回顾(查看引脚图)
- ➤ AD₁₅ ~ AD₀—地址/数据分时复用总线;
- ➤ A₁₉ ~ A₁₆/S₆ ~ S₃—地址/状态分时复用总线;
- ➤ BHE#/S₇—高8位数据允许/状态分时复用引脚; 查看BHE#和A₀的代码组合对应的操作
- ➤ RD# 读控制信号引脚;
- ➤ WR# 写控制信号引脚;
- ► M/IO# 存储器/输入输出操作指示信号引脚;
- ▶ DT/R# 数据发送/接受信号引脚;
- ▶ DEN# 数据允许信号引脚;
- ➤ ALE 地址锁存允许信号;

NEXT

8086微处理器引脚图





BHE和Ao的代码组合对应的操作

BHE	A_0	操作	所用数据引脚
0	0	从偶地址单元开始读/写一个字	AD ₁₅ ~AD ₀
0	1	从奇地址单元或端口读/写一个字节	AD ₁₅ ~AD ₈
1	0	从偶地址单元或端口读/写一个字节	AD ₇ ~AD ₀
1	1	无效	
0	1	从奇地址开始读/写一个字(在第一个 总 线 周 期 将 低 8 位 数 据 送 到	
1	0	AD15~AD8,下一个周期将高8位数据 送到AD ₇ ~AD ₀)	



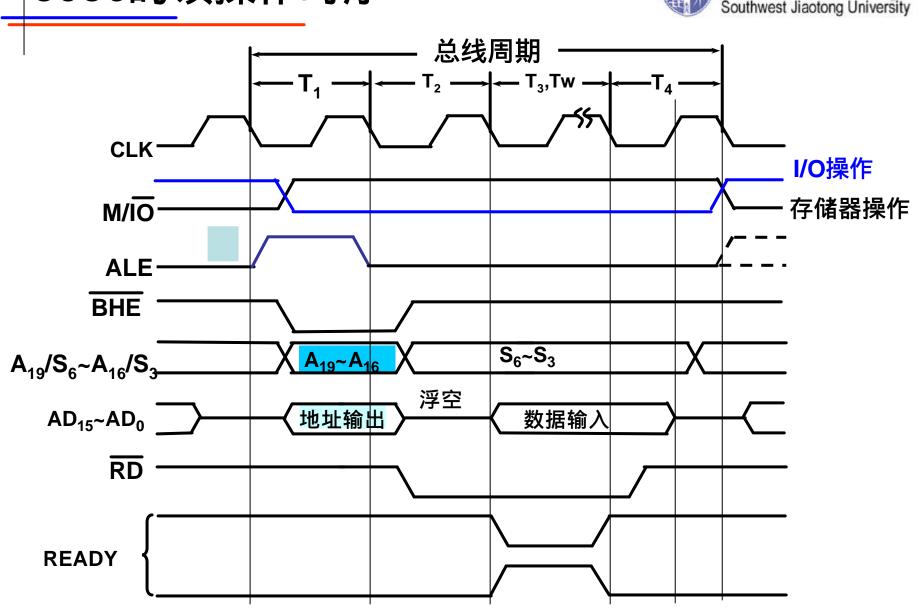
8086的读/写操作时序



- * 几个术语
- > 时钟周期(T状态):指主时钟脉冲周期。
- ➤ 总线周期:CPU从内存或I/O端口存取一个字节 或字所需要的时间。
- > 指令周期: CPU执行一条指令所需时间。
- 一个基本的8086总线周期由4个时钟周期组成。 为满足低速外设或内存的时序要求,可插入等待 周期。

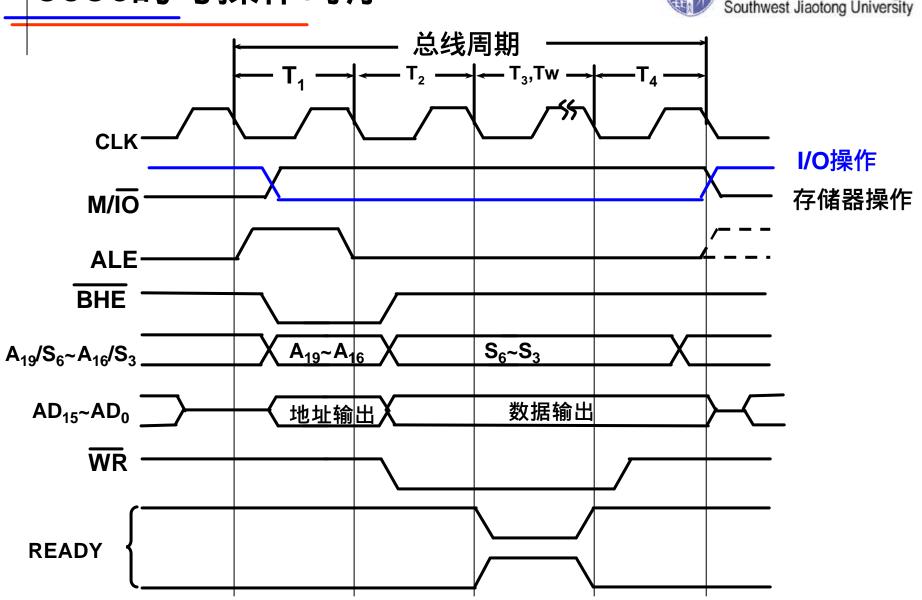
8086的读操作时序





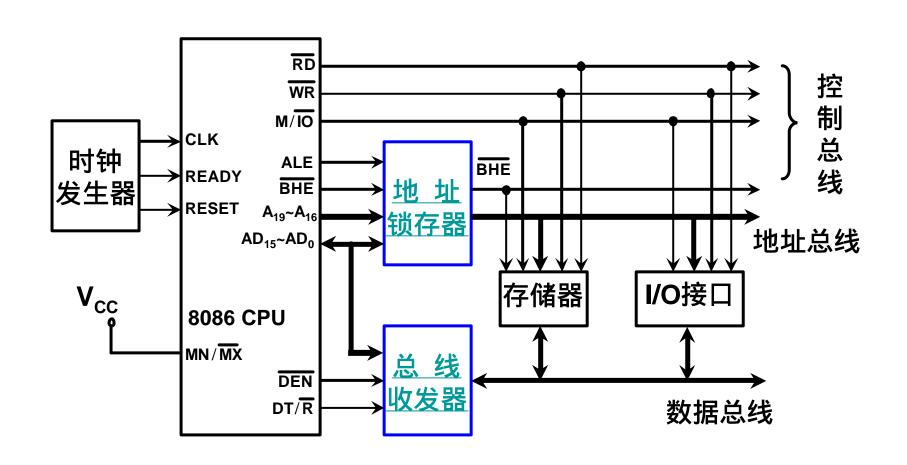
8086的写操作时序





最小方式基本配置



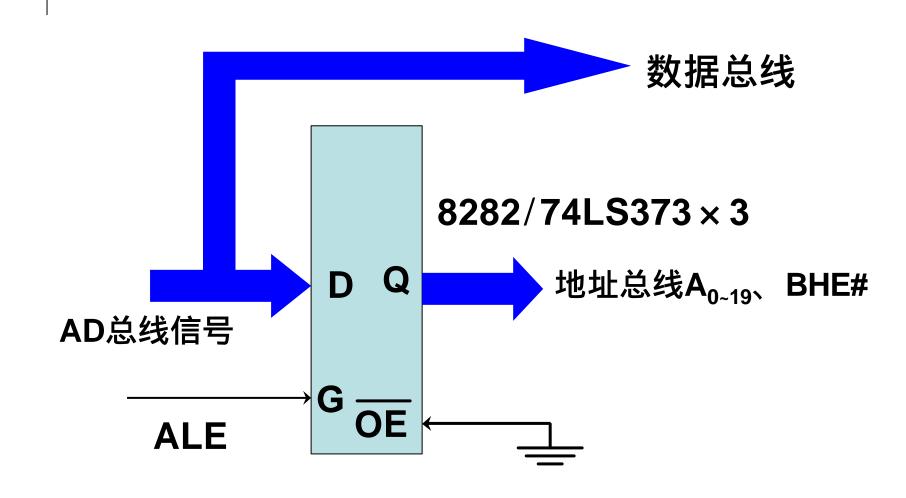


8086最小方式下的基本配置



地址锁存器电路





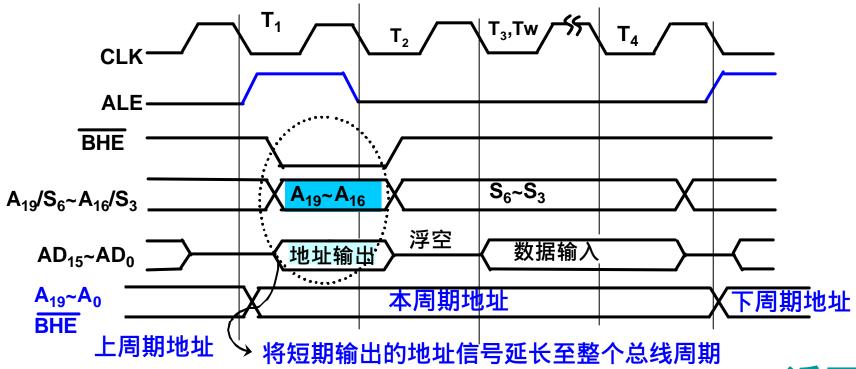
地址锁存接线示意图

地址锁存器电路



- ❖8282、74LS373的 特性真值表见右表;
- ❖地址锁存电路工作 波形如下图。

OE	G	D	输出Q
1	×	×	高阻态
0	1	0	0
0	1	1	1
0	0	×	维持原状Q。

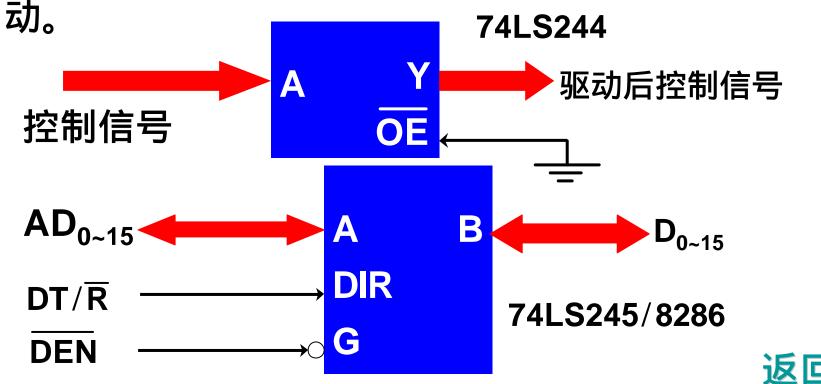


总线收发器电路



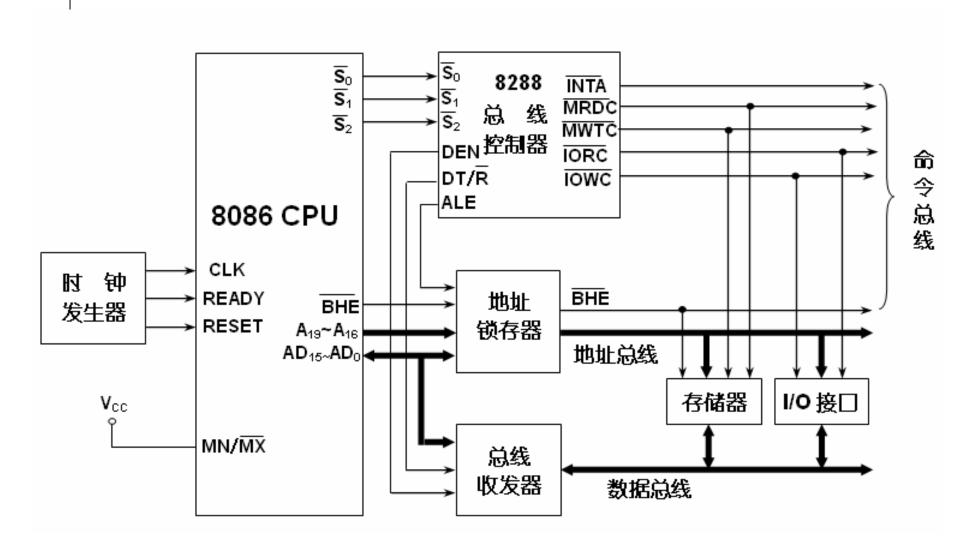
❖任一总线负载过多时,都将影响微机系统的工作,此 时可在总线上串接缓冲器,以增加总线的驱动能 力。

❖ 经分离后的地址总线驱动能力较强, 一般无需再驱



最大方式基本配置







第二节 半导体存储器及其工作原理

- 1. 半导体存储器的分类与工作原理;
- 2. 半导体存储器的读/写操作时序。

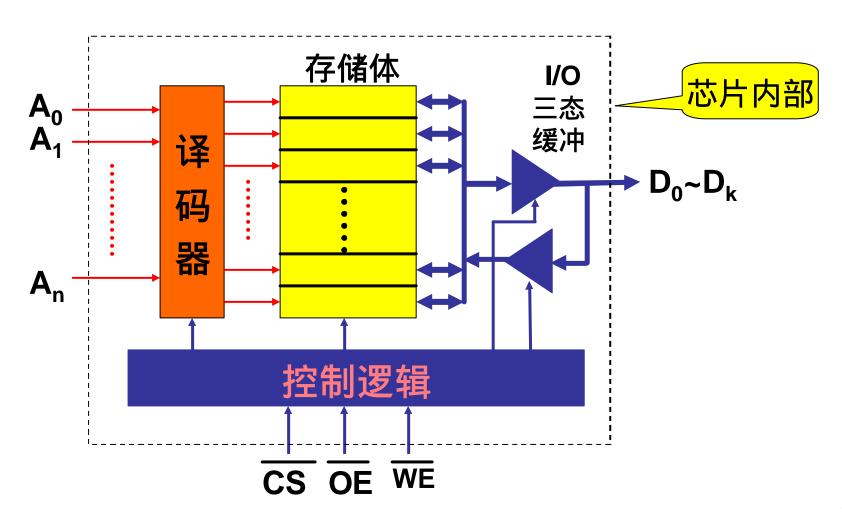
存储器的分类



- ❖存储器按用途分类
- ▶ 内部存储器:主要由半导体存储器构成, CPU可直接访问, 速度快。
- ➤ 外部存储器:也称辅助存储器,物理存储设备包括硬盘、软盘、光盘、'U盘'等, CPU不能直接访问,速度慢,但容量大,也称为'海量存储器'。
- ❖半导体存储器分类
- ➤ RAM存储器: SRAM、DRAM、SDRAM、DDR 等;
- ➤ ROM 存储器:ROM、PROM、EPROM、E²PROM、FLASHROM等。

SRAM内部结构模型





SRAM结构说明



❖存储矩阵

多个基本存储单元电路构成存储矩阵,每个基本存储单元可由一位或多位组成。通常引脚信号中的 $D_0 \sim D_k$ 表示了每个基本存储单元的构成位数,如:k=0、3、7时,每个存储单元由1、4、8位构成。

❖地址译码器

地址译码器译码选择要访问的片内存储单元。 地址信号由外部访问者提供,地址引脚信号常由 $A_0 \sim A_n$ 构成, n与存储单元数目有关,存储单元数 = 2^{n+1} 。

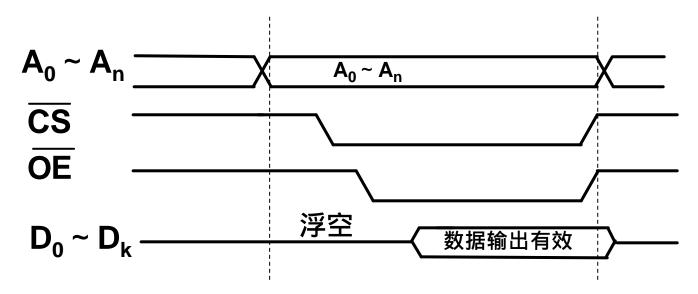
SRAM结构说明



- ❖ 控制逻辑与三态数据缓冲器 控制逻辑根据存储器输入控制信号,控制存储器 内部单元内容的读出或写入。
- ❖SRAM芯片的引脚信号说明
- ▶ 地址输入总引脚A₀ ~ A_n;
- \rightarrow 数据输入/输出引脚 $D_0 \sim D_k$;
- ▶ 片选CS#, 当其有效时, 存储器芯片被选中, 允许对 其进行读/写操作, 否则芯片不能被访问;
- ightharpoonup 输出允许OE#, 当其有效时, 允许被选中单元内容输出到数据引脚 $D_0 \sim D_k$;
- ightharpoonup 写允许WR#, 当其有效时, 允许将数据引脚 $D_0 \sim D_k$ 上的数据写入被选中单元。

SRAM的输出操作时序

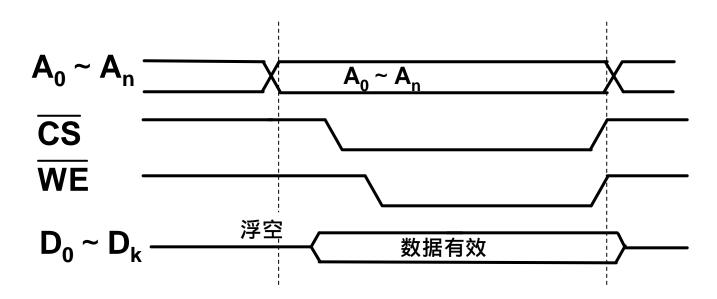




- ➤SRAM 输出时CPU 应该执行读总线操作
- ▶输出操作步骤:(1)CPU发出地址A₀~A_n以便选定操作单元;(2)系统控制逻辑产生片选CS#;(3)CPU给出输出允许信号OE#;(4)CS#及OE#都有效后,被选中单元内容输出到数据引脚D₀~D_k。

SRAM的输入操作时序





▶SRAM输入时CPU应该执行写总线操作

▶写操作步骤:(1) CPU发出地址A₀ ~ A_n以便选定操作单元;(2) 系统控制逻辑产生片选CS#;(3) CPU给出写允许信号WR#;(4) CS#及WR#都有效后,被选中单元的内容被数据引脚D₀ ~ D_k上的内容所替代。

典型SRAM芯片



- ❖常用型号:6116、6264、62256、628128, 其容量分别为2K×8、8K×8、32K×8、 128K×8;
- ❖典型SRAM芯片命名与容量之间有一定关系。
- ❖典型SRAM芯片引脚图

典型SRAM芯片引脚



62256 A ₁₄ A ₁₂ A ₇ A ₆ A ₂ A ₁ A ₀ D ₁ D ₂ GND	6264 NC A ₁₂ A ₇ A ₆ A ₅ A ₄ A ₂ A ₁ A ₀ D ₁ D ₂ GND	6232 NC NC A ₇ A ₆ A ₂ A ₁ A ₀ D ₁ D ₂ GND	1 28 2 27 3 26 4 25 5 24 6 23 7 22 8 21 9 20 10 19 11 18 12 17 13 16 14 15	6232 Vcc WE NC A ₉ A ₁₁ O A ₁₀ CE D ₇ D ₆ D ₅ D ₄ D ₃	6264 Vcc WE NC A ₉ A ₁₁ O A ₁₀ O D ₅ D ₄ D ₃	62256 VCC WE A ₁₃ A ₈ A ₉ A ₁₀ CE D ₇ D ₆ D ₅ D ₄ D ₃
--	--	---	---	---	---	--

SRAM芯片操作逻辑



❖6264芯片的操作逻辑真值表

引脚 操作方式	CE1	CE2	ŌĒ	WE	D ₀ ~D ₇
未选中	1	×	×	×	高阻
未选中	×	0	×	×	高阻
输出禁止	0	1	1	1	高阻
读	0	1	0	1	输出
写	0	1	1	0	输入
写	0	1	0	0	输入

其中的×表示任意逻辑值

动态随机存取存储器(DRAM)



- ❖ DRAM基本结构与SRAM大致相同,由存储体、 地址译码器、控制逻辑和数据I/O缓冲器构成;
- ❖ DRAM与SRAM的信息存储原理不同;
- ❖ DRAM存储内容需要定期刷新;
- ❖ DRAM芯片为了减少封装引脚,地址输入采用行、列两路复用锁存方式(相应增加行与列选通信号);
- ❖IRAM是近年来出现的一种新型DRAM芯片,它克服了DRAM需要外加刷新电路的缺点,内部含有动态刷新电路,使之兼有SRAM和DRAM的优点。

内存条



- ❖ SIMM(Single In-line Memory Module,单列直插式存储器模块)内存条的接口标准有30线(9位数据)、72线(36位数据)和100线(72位数据);
- ❖ DIMM(Dual In-line Memory Module, 双列直插式存储器模块)内存条的接口标准有168线、184线和200线;
- ❖ FPM DRAM(Fast Page Mode DRAM,快速页面模式内存)是把连续的内存块以页的形式来处理;
- ❖ EDO DRAM(Extended Data Out DRAM,扩展数据输出内存);
- ❖BEDO DRAM(Burst EDO DRAM, 突发式EDO内存);
- ❖SDRAM(Synchronous DRAM,同步DRAM);
- *Rambus DRAM, DDR DRAM。

只读存储器ROM



- ❖在微机系统运行过程中只能对其进行读操作,而 不能进行写操作的一类存储器称为只读存储器。
- ❖只读存储器分类:
- ➤ 掩模ROM;
- ➤ 可编程ROM(PROM);
- ➤ 可擦除可编程ROM(EPROM);
- ▶ 电可擦除可编程ROM(EEPROM);
- ➤ 闪速存储器(Flash ROM)。

EPROM存储器



- ❖ EPROM内部结构与SRAM非常相似,仍然由存储矩阵、地址译码器、控制逻辑与三态数据缓冲器等组成。
- 常用EPROM与常用SRAM相比较,在容量相同的情况下,其引脚中没有写允许信号。
- ❖ 计算机应用系统设计时,往往不考虑EPROM的编程电路,EPROM由专用编程器编程。
- ❖EPROM输出过程与SRAM输出过程完全相同, 写时序实际上是指编程时序。
- ❖典型EPROM芯片<u>引脚</u>及其<u>操作逻辑</u>。
- ❖其它ROM类芯片(E²PROM、Flash Memory)工作原理与EPROM基本相同。

典型EPROM引脚图



27512	27256	27128	2764	2732	2716					2716	2732	2764	27128	27256	27512
A 15	Vpp	Vpp	Vpp			1		フ	28			<u>Vсс</u>	<u>Vcc</u>	Vcc	Vcc
A_{12}	A_{12}	A 12	A 12			2			27			PG	PGM	A 14	A_{14}
ΑŢ	ΑŢ	A_7	A_7	Αz	A_7	3	(1)	(24)	26	Vcc	Vcc	M	A 15	A 15	A 15
Ae	A_6	A_6	A_6	Ae	A_6	4	(2)	(23)	25	A_s	A_s	NC	$\mathbb{A}_{\mathbf{s}}$	A_8	Å₅
A_{ϵ}	A_{ϵ}	$A_{\rm s}$	A_{ϵ}	A_{ϵ}	A_{s}	5	(3)	(22)	24	A,	A,	$\mathbb{A}_{\mathbf{s}}$	A,	A۹	Аs
A_4	A_4	\mathbb{A}_4	\mathbb{A}_4	\mathbb{A}_4	\mathbb{A}_4	6	(4)	(21)	24	A	A11	\mathbb{A}_9	A	A	À 11
A_5	A_5	A_5	A_5	A ₅	A_5	7	(5)	(20)	22	ŌE	OE/Vpj	A 11	OE	OE	OE/Vpp
\mathbb{A}_2	\mathbb{A}_2	\mathbb{A}_2	\mathbb{A}_2	\mathbb{A}_2	\mathbb{A}_2	8	(6)	(19)	21	A 10	Αn	OE	A 10	À 10	À 10
Aı	Aı	A,	Aı	A	Aı	9	(7)	(18)	20	CE/PGM	$\overline{\mathtt{CE}}$	A 10	CE	CE	CE
A_o	A_o	A_o	A_0	A,	A_0	10	(8)	(17)	19	D_{τ}	D_7	CE	D_7	D_{τ}	D_7
D_{o}	D_{o}	D_0	D_{o}	D_{o}	D_{o}	11	(9)	(16)	18	D^{e}	D^{e}	D_7	D_{e}	D^{e}	D^{e}
D_{i}	D_{1}	D_1	D_{1}	D_{i}	D_1	12	(10)	(15)	17	D_{ϵ}	D_{s}	D_6	D_{5}	D_{ϵ}	D_{5}
D_2	D_2	D_2	D_2	D_2	D_2	13	(11)	(14)	16	D_4	D_4	D_{ϵ}	D_4	D_4	D_4
GND	GND	GND	GND	GND	GND	14	(12)	(13)	15	D_5	D_{5}	D_4	D_{s}	D_{5}	D_s

- ❖引脚信号仍然包括由地址输入、数据输入/输出 与控制等构成;
- ❖与SRAM相比, EPROM增加了编程电压输入引脚。



EPROM芯片2764的操作



❖2764的操作逻辑真值表

T 1/m + - + - + - + - + - + - + - + -	引脚名称								
工作方式 	CE	CE OE PGM VPP		VCC	D ₀ ~D ₇				
读出	0	0	1	+5V	+5V	输出			
未选中	1	×	×	+5V	+5V	高阻			
编程	0	1	负脉冲	+21V	+5V	输入			
编程校验	0	0	1	+21V	+5V	输出			
编程禁止	1	×	×	+21V	+5V	高阻			
编程禁止	×	×	1	+21V	+5V	高阻			



第三节 存储器系统的结构

本章内容

- ❖ 容量扩展: 位数扩展与字数扩展;
- ❖ 多存储体主存结构。

位数的扩展

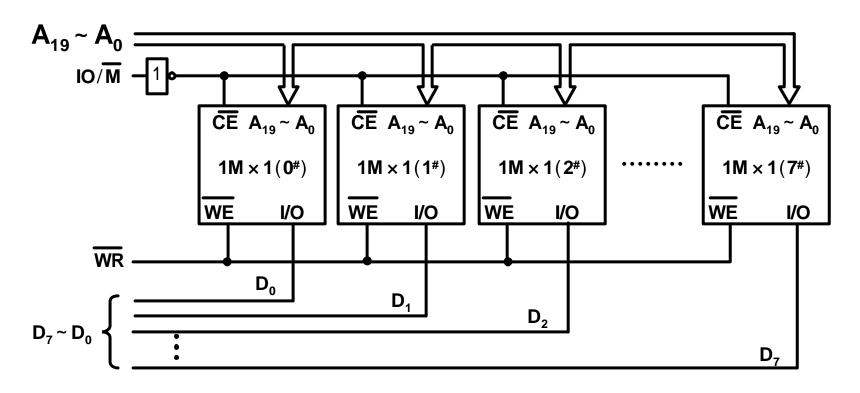


- ❖问题的提出:
- ➤ CPU存储空间结构(容量×单元位数)要求;
- > 半导体存储器结构(芯片单元数×每单元位数)
- ❖位数的扩展方法 —— 采用芯片并联法

位数扩展举例



❖例1、1M×1位芯片构成1M×8位的存储器模块。扩展方法如下图所示(可用于8088系统)。

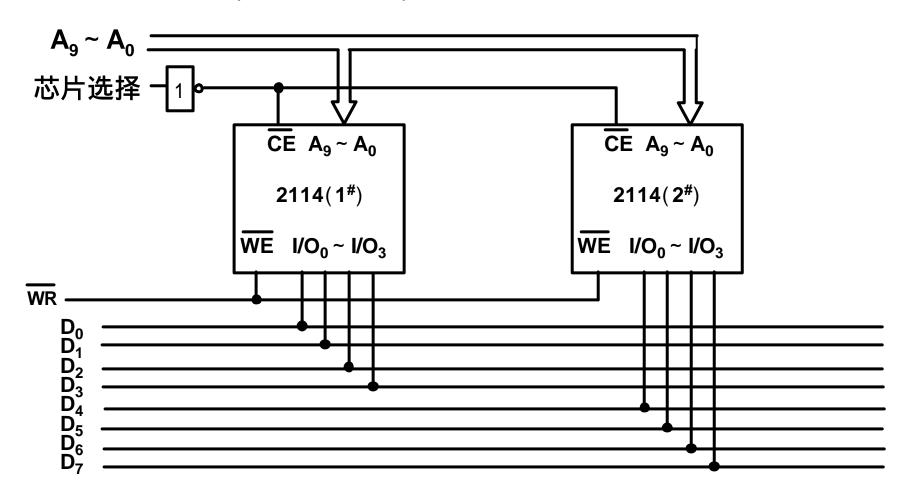


1 M×1位构成1M×8位的存储器模块结构图

位数扩展举例



❖例2、2114(1 K×4位)扩展1 K×8位存储器。



位扩展连接方法总结



- 芯片的地址线全部并联且与地址总线相应的 地址线连接。
- 2. 片选信号线并联,可以接控制总线中的存储器选择信号,也可以接地址线高位,或接地址译码器的输出端。
- 3. 读写控制信号并联接到控制总线中的读写控制线上。
- 4. 数据线分高低部分分别与数据总线相应位连接。

字数(存储容量)的扩展



- ❖ 问题的提出
- ➤ CPU的存储空间往往比单个存储芯片容量要大, 因此常常采用多个存储芯片组成计算机系统的 主存储器模块。
- ❖字数扩展要解决的问题
- ➤ 按照系统设计要求确定存储器物理地址空间的映射关系。即确定每片存储芯片映射(定位)到 CPU存储空间的哪一段(如 PC/XT 机器中ROM(存放BIOS)芯片地址安排在存储空间最高端,RAM芯片地址安排在存储空间最低端);

字数扩展

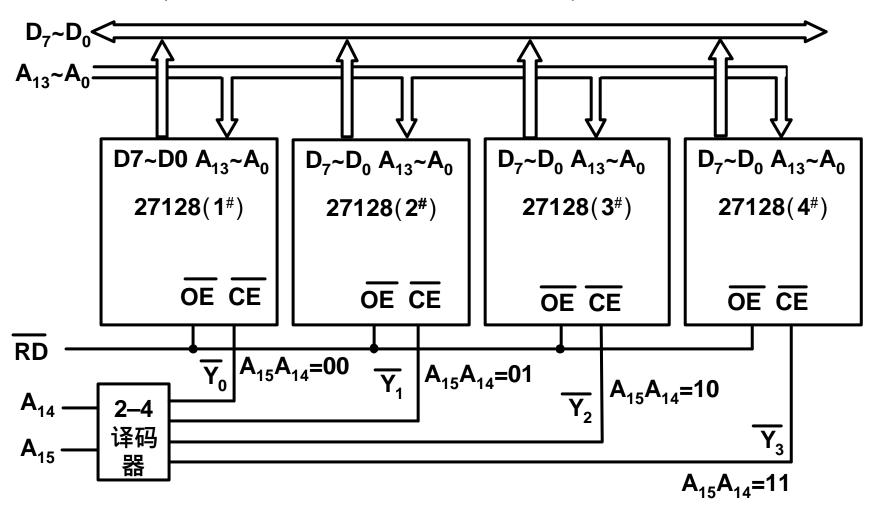


- 》多个存储芯片与数据总线的连接。每个芯片都通过数据总线与CPU交换数据。一般在某个特定时刻,CPU只能与被其选中(由输出地址编码确定)单元进行数据交换,其它单元应与总线隔离。片内单元由片内译码器选择,而芯片选择是存储控制逻辑要解决的问题(实质上就是两级译码选择某一单元的问题)。
- > 需要设计译码电路,原因有两个。其一,使得系统物理存储器映射到CPU存储空间的某一固定的区域;其二,由于单个存储芯片容量往往小于CPU寻址空间,其地址引脚数目不同,译码电路可以修正两者之间的不匹配,充分利用CPU的所有存储空间。

字数扩展举例



例、用四片27128(16K×8)组成64K×8的系统主存。(设系统存储空间为64KB)



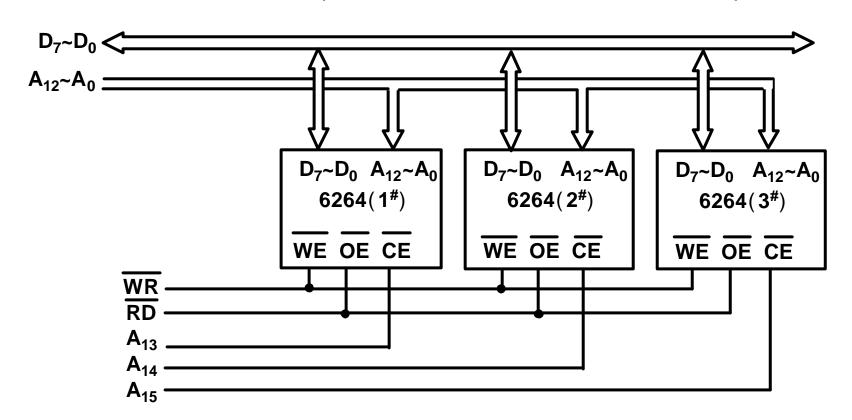
字数扩展举例



译码电路设计



❖线选法译码电路:用高端地址线作为芯片片选控制线。如下例(设计CPU寻址空间64KB)。



线选法译码电路设计



❖各芯片地址空间分布

	A ₁₅	A ₁₄	A ₁₃	A ₁₂ ~ A ₀	地址范围
1# 6264	1	1	0	0 ~ 0	C000H(首地址)
	1	1	0	1 ~ 1	DFFFH(末地址)
2# 6264	1	0	1	0 ~ 0	A000H(首地址)
	1	0	1	1 ~ 1	BFFFH(末地址)
3# 6264	0	1	1	0 ~ 0	6000H(首地址)
	0	1	1	1 ~ 1	7FFFH(末地址)

线选法译码电路特点



- 1. 各芯片间地址不连续;
- 2. 有相当数量的地址不能使用,否则造成片选混 乱;
- 3. 若高位地址没有使用,将造成地址重叠区;
- 4. 线选法影响地址空间的有效使用,限制了芯片的进一步扩展;
- 5. 译码器简单。

全地址译码法

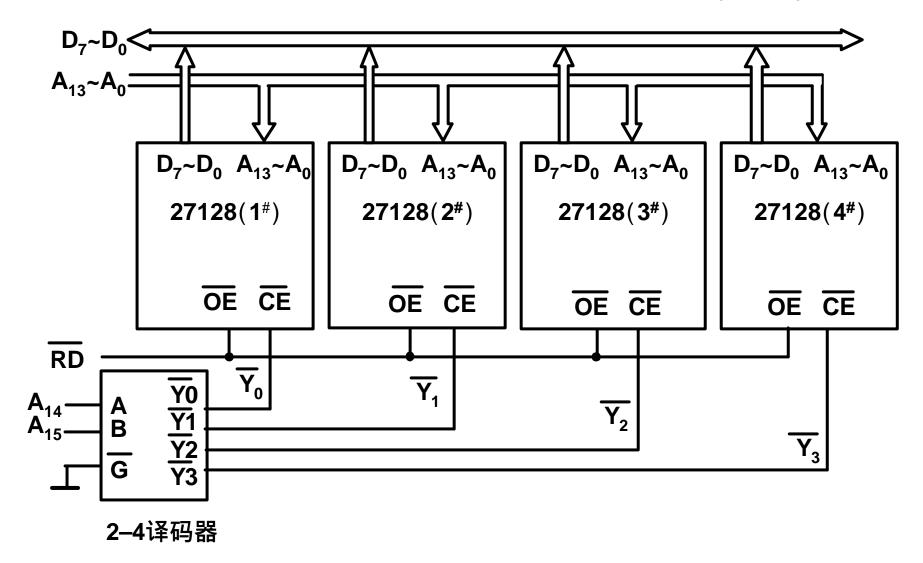


- ❖ 全译码将片内寻址线外的所有高位地址线全部 译码,译码输出作为芯片的片选信号。
- ❖ 特点:
- 1. 每块芯片的地址范围是唯一的;
- 2. 各芯片间的地址是连续的,寻址空间得到充分 利用;
- 3. 需要专门的译码电路。

全地址译码法举例

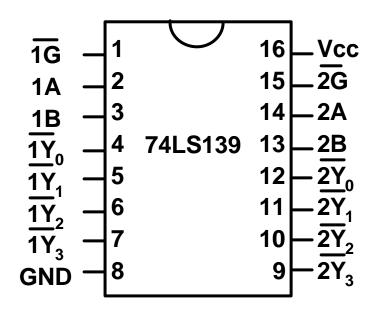


❖举例:设系统CPU寻址空间为64KB(A₀~A₁₅)



译码电路分析





2-4译码器

输入端			松山地			
允许	选	择	输出端			
G	В	Α	Y ₀	Y_1	Y ₂	Y_3
1	×	×	1	1	1	1
0	0	0	0	1	1	1
0	0	1	1	0	1	1
0	1	0	1	1	0	1
0	1	1	1	1	1	0

逻辑真值表

译码结果



选中芯片	A ₁₅	A ₁₄	A ₁₃ ~ A ₀	地址范围
	0	0	0 ~ 0	0000H(首地址)
1 [#] 27128	0	0		
	0	0	1 ~ 1	3FFFH(末地址)
	0	1	0~0	4000H(首地址)
2 [#] 27128	0	1		
	0	1	1~1	7FFFH(末地址)
	1	0	0 ~ 0	8000H(首地址)
3 [#] 27128	1	0		
	1	0	1~1	0BFFFH(末地址)
4 [#] 27128	1	1	0~0	0C000H (首地址)
	1	1		
	1	1	1~1	0FFFFH (末地址)

部分译码法



❖部分译码是将片内寻址地址线以外的高位地址线的一部分接译码器的输入端,译码器的输出作为片选。该方法兼有线译码法与全译码法的特点。

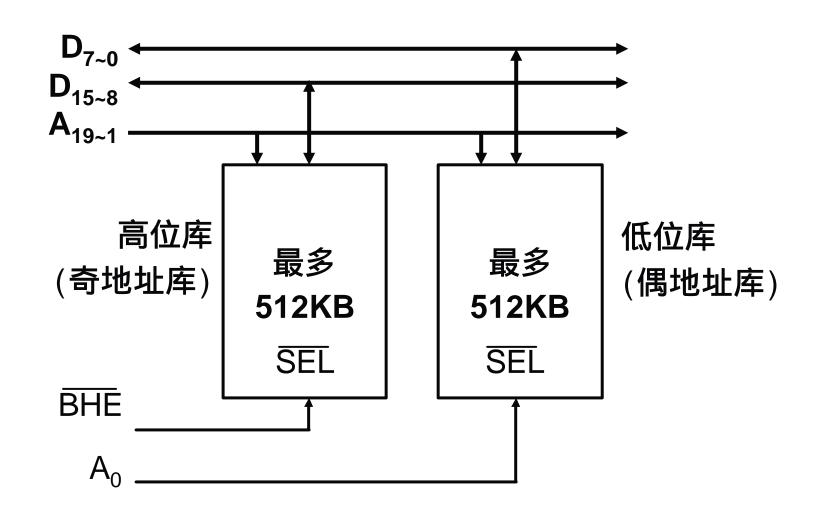
多存储体主存结构



- ❖CPU的外部数据总线为8位时,其存储器只需用单体结构。
- ❖对于80X86CPU,外部数据总线为16位(或32位)时,一般需用多个8位的存储体。8086CPU称为称奇数库(或高位库)和偶数库(或低位库),80386以上的CPU有多个存储体。
- ❖ 多存储体结构下各存储体的体选择信号由CPU 提供,如8086的BHE#和A₀,80386的BE₀# ~ BE₃#等。

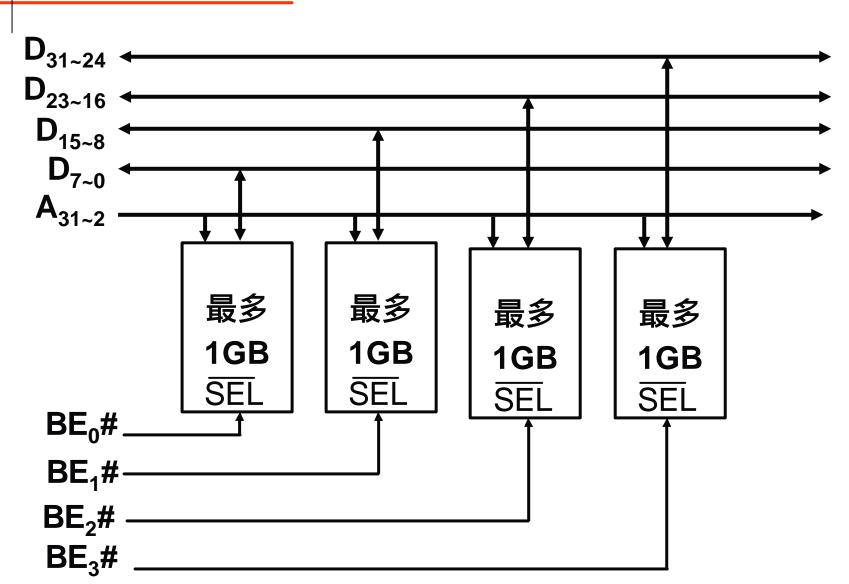
8086存储器的奇偶分体





80386的4个存储分体

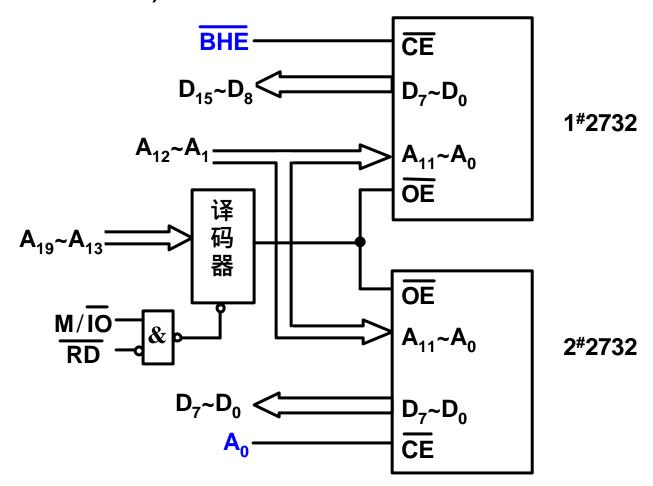




8086两存储体结构的主存

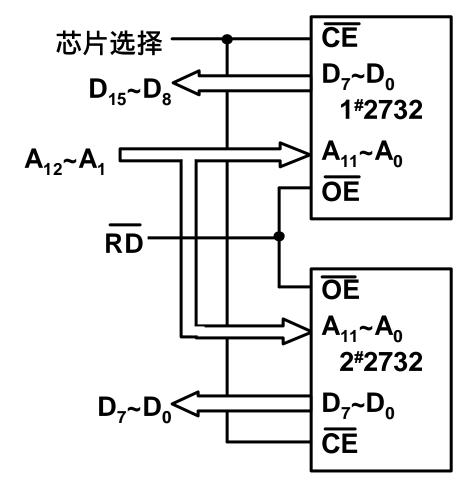


❖可分别按字节或字操作的主存结构(以2732芯片 为例介绍)。





❖只可按字操作的主存结构(以2732芯片为例介绍)。





第四节 8086/8088存储器系统设计

- 1. 8088系统中存储器与CPU的连接;
- 2. 8086系统中存储器与CPU的连接。

主存储器与CPU连接的一般问题 🛗 🐠 🐔



- ❖CPU总线的负载能力。
- > 数据总线、地址总线的驱动;
- > 控制信号的驱动;
- ❖信号线的配合与连接。
- ➤ 读/写控制信号的连接, CPU读入数据或指令时存储器输出数据, CPU写数据时存储器输入数据。
- ❖时序配合问题。

主存与8088CPU的连接



- ❖ 例: 8088CPU组成的最小系统需要32 KB的ROM, 地 范 围 安 排 在 0F8000H~0FFFFH; 需 要 64 KB 的 RAM, 其 地 址 范 围 安 排 为 00000H~0FFFFH。 ROM 选 用 芯 片 EPROM 27256, RAM选用芯片SRAM 62256, 画出系统连接图。
- ❖分析:
- ➤ SRAM62256 与 EPROM27256 的 容 量 都 是 32 K×8, 需 要 片 内 地 址 线 = log₂(32 K) = log₂(2¹⁵) = 15条(A₁₄~A₀);
- > 27256需要一片, 62256需要二片;



- ❖译码器的设计:
- ▶ 8088地址总线宽度为20位,其低15位用于芯片内 译码,直接与存储芯片连接;
- ▶ 用于片选的地址线共有5条(A₁₉~A₁₅),考察最小系统要求的物理地址范围,其对应的二进制编码如下表;
- ▶ 分析地址范围的二进制编码,设计译码控制逻辑 如下图;
- ❖注意控制总线信号(读/写控制)的连接,存储器扩展电路原理图如下。

地址编码表



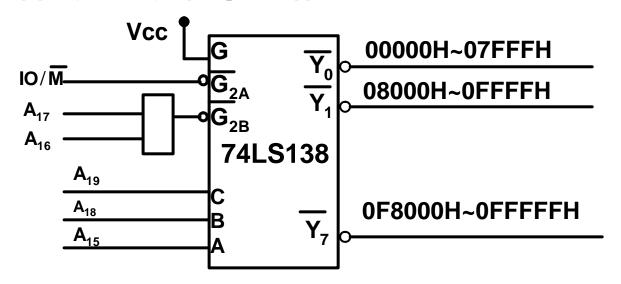
选中芯片	A ₁₉	A ₁₈	A ₁₇	A ₁₆	A ₁₅	$A_{14} \sim A_0$	地址范围
	1	1	1	1	1	0 ~ 0	F8000H(首地址)
27256							
	1	1	1	1	1	1~1	FFFFFH(末地址)
	0	0	0	0	0	0 ~ 0	00000H(首地址)
62256 #1							
	0	0	0	0	0	1~1	07FFFH(末地址)
	0	0	0	0	1	0 ~ 0	08000H(首地址)
62256 #2							
	0	0	0	0	1	1~1	0FFFFH(末地址)

地址范围由系统 设计要求确定

返回

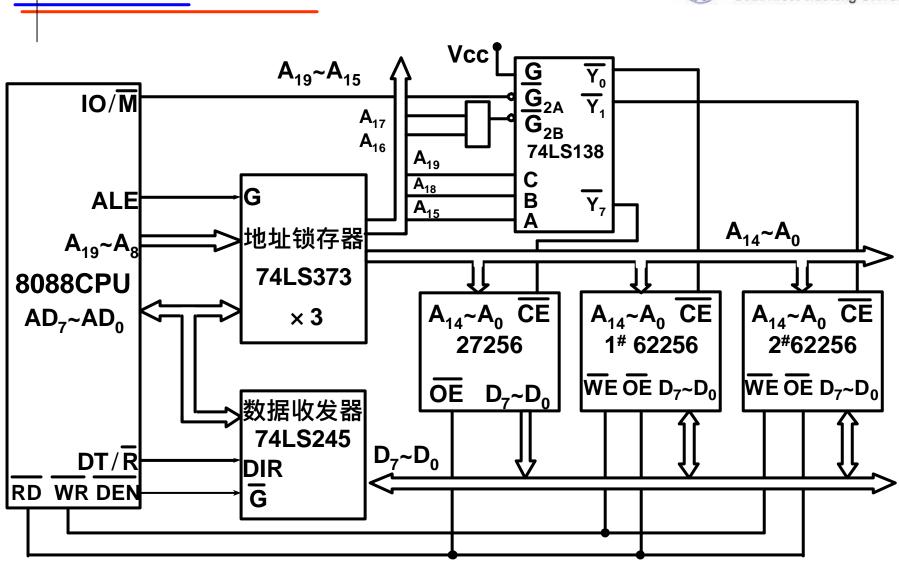


◆一种可用的译码电路



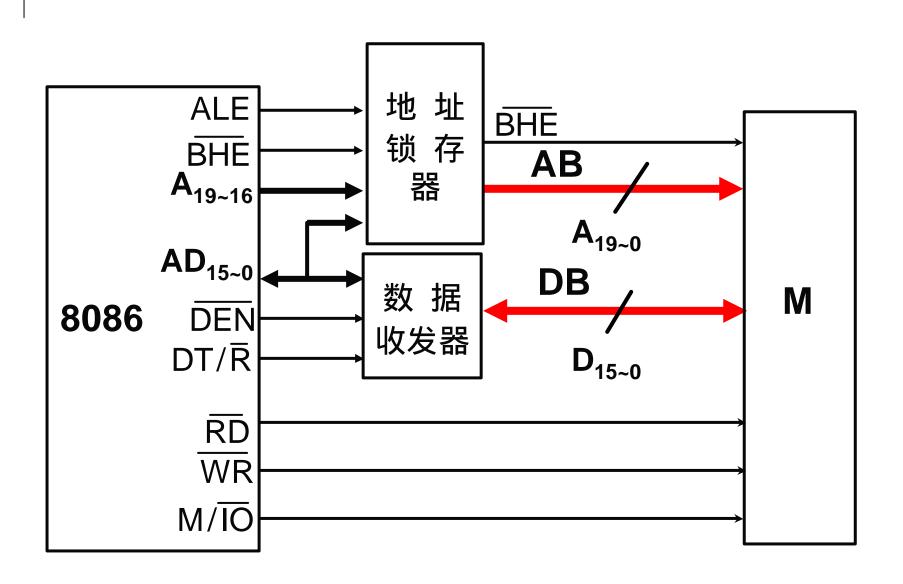
❖该译码电路仍然有重叠地址,请同学们指出其重叠地址。如果要求译码电路无重叠地址,译码电路应如何修改?





8086与主存的连接





8086与主存的连接



- ❖注意高位库与低位库的连接方法,采用位数扩展 方法将两片字节型的存储芯片扩展为字存储 器。由不同译码器产生高、低位库的片选信号, 用以区分奇数字节与偶数字节的操作。
- ❖注意片内地址总线的连接只从A₁开始考虑,A₀实际上已由CPU内部译码为BHE#与A₀,由不同引脚输出,用于选择高位库与低位库。

8086与主存的连接举例



❖例: 8086 CPU组成的最小模式系统需要16 K字的ROM, 其地址范围为F8000H~FFFFFH, 需要16 K字的RAM, 其地址范围为00000H~07FFFH, ROM选用芯片2764, RAM选用芯片6264, 画出系统连接图。

>解题分析:

2764与6264的容量都是8 K×8, 需要片内地址线 = $\log_2(8 \text{ K}) = \log_2(2^{13}) = 13 + (A_{12} - A_0)$;

16K字等同与32K字节,因此,2764需要四片,6264需要四片;

8086与主存的连接举例



译码器的设计

- ▶ 8086地址总线宽度为20位, A₁₃~A₁位用于芯片内部存储单元选择译码(8K字单元选择), 直接与存储芯片连接;
- ➤ 用于片选的地址线共有6条(A₁₉~A₁₄),同时还需考虑高、低位库的选择,设计译码控制逻辑的依据就是物理地址二进制编码(其实质即是将相应二进制编码转换为逻辑电路)。
- ➢分析地址范围的二进制编码,设计译码控制逻辑。
- ➤ 每两片2764与6264组成8K字的存储器,按设计要求占用地址空间及其编码如下。



(1)EPROM子系统译码电路

地址空间

$$A_{19} A_{18} A_{17} A_{16} A_{15} A_{14} A_{13} \sim A_{1}$$

0F8000H~0FBFFH 1 1 1 1 0 $\begin{cases} 0 & 0 \\ 4 & 3 \end{cases}$

0FC000H~0FFFFH 1 1 1 1 1 1 1

 A_0 =0时选择低位库 A_0 =1(BHE =0)时

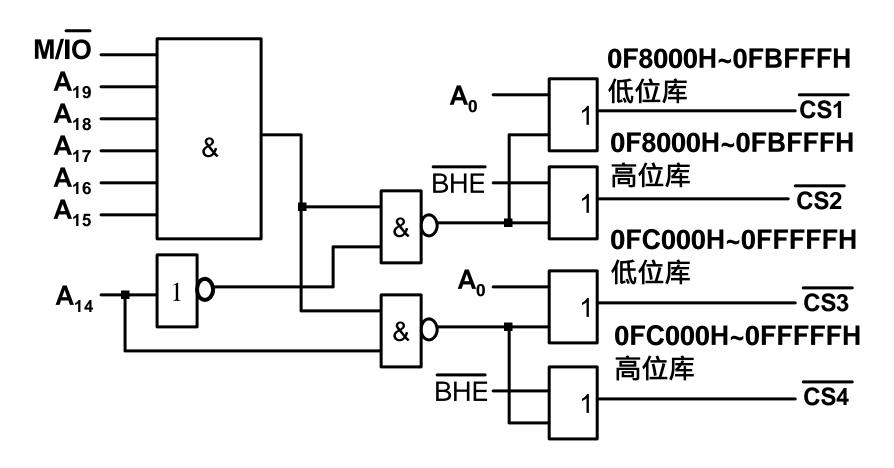
译码电 路设计

依据

¦选择高位库

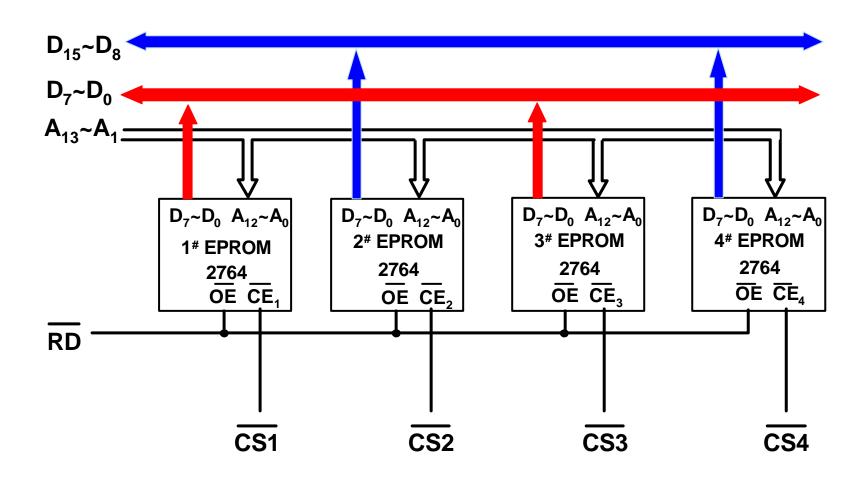


❖一种可能的译码电路设计如下图





❖EPROM子系统接线图





(2) RAM子系统译码电路

地址空间

$$A_{19} A_{18} A_{17} A_{16} A_{15} A_{14} A_{13} \sim A_{0}$$

00000H ~ 03FFFH $\begin{vmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & \sim & 1 \end{vmatrix}$

04000H ~ 07FFFH

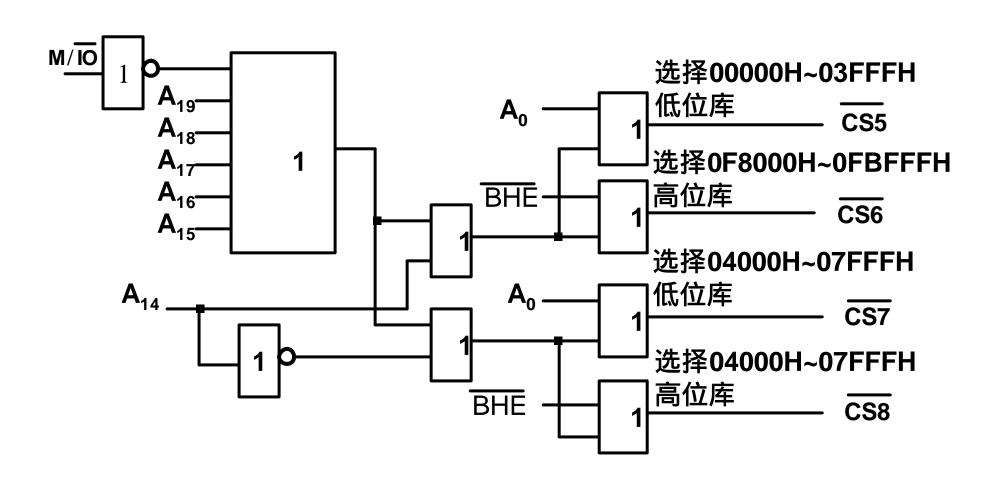
 $\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & \sim & 0 \\ 1 & \sim & 1 \end{bmatrix}$

译码电 路设计 依据

 $A_0=0$ 时选择低位库 A₀=1(BHE=0)时选 择高位库



❖一种可能的译码电路如下图。





❖SRAM子系统接线图

