· · · 微机原理及接口技术 串讲

刻面 2018.12

●●■课程内容

o 微机原理(1、2章)

运算基础、微机组成及基本原理 8086CPU内部结构、8086微机系统组成

o 软件应用(3、4章)

指令系统、伪指令 汇编语言程序设计及调试

○ 硬件应用(5、6、7、8章) (书6、7、8、9) 接口技术(8255A、8253、AD0809、DA0832: 应用)中断技术(概念)

第一章 绪论

• • 第一章 基本要求

- o 微型计算机运算基础
 - 不同计数制数及其相互转换
 - 数值数据和字符数据在计算机的编码和存放格式
 - 一进制数的算术和逻辑运算规则

* 常用计数制

十进制	二进制	八进制	十六进制
0D	0B	0 Q	0H
1	1B	1Q	1H
2	10 B	2Q	2H
3	11B	3 Q	3Н
4	100B	4Q	4H
5	101B	5Q	5H
6	110B	6Q	6H
7	111B	7Q	7H

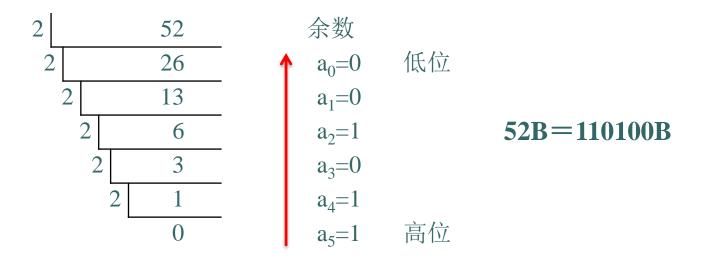
- 十进制
- 二进制
- 八进制
- 十六进制
- BCD码(压缩型/非压缩型)

十进制	二进制	八进制	十六进制
8	1000B	10Q	8H
9	1001B	11Q	9H
10	1010B	12Q	AH
11	1011B	13Q	BH
12	1100B	14Q	СН
13	1101B	15Q	DH
14	1110B	16Q	EH
15	1111B	17Q	FH
16	10000B	20Q	10H

任意进制数←→十进制数

○ 任意进制→十进制: 按"权"展开求和 $1101B = 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 13$

十进制→任意进制:除基取余法(商为0,余数从低往高)



● ● 十六进制/BCD数←→二进制数

十六进制←→二进制数4位二进制数位 ←→ 1个十六进制位

o 压缩**BCD**码: 23=0010 0011 BCD

o 非压缩BCD码: 23=00000010 00000011 BCD

二进制←→ BCD码
 二进制←→十进制数←→ BCD码

- - 二进制数运算规则

算术运算加法、减法、乘法、除法

逻辑运算与、或、非、异或

数据表示方法

o数值数据

无符号数(所有位均表示数值;数值范围) 有符号数(原码、反码、补码)

o符号数据: ASCⅡ码(美国信息交换标准码)

 $0\sim9$ \rightarrow $30H\sim39H$

 $A \sim F \rightarrow 41H \sim 46H$

 $a \sim f \rightarrow 61 H \sim 66 H$

回车 \rightarrow 0DH

换行 \rightarrow 0AH

• • 有符号数

o 原码

• 最高位表示符号(正数: 0; 负数: 1),数值保持不变

o补码

- 正数与原码相同;
- 负数补码求法: ①定义: $[X]_{i} = 2^n + X$

- 真值←→补码 [X]¾ = 1000 0010B, 求X, [2X]¾, [0.5X]¾
- 计算机内以补码方式对数据进行存储和运算
- o 溢出: 超出数的存储范围
 - 8位补码 [-128,+127]16位补码 [-32768,+32767]
 - 判断溢出: 直接看十进制运算结果; 进位的异或运算

● ● 第二章 8086 CPU

• • ● 第二章 基本要求

- o 8086 CPU的内部结构
- 。8086微机系统的组成

存贮器的分段结构,存贮器物理地址的形成,字节数据及字数据在存贮器中的存放形式及存取过程,堆栈的概念,8086 CPU引脚信号与8086最小系统,8086 CPU最小模式下的读/写总线时序。

8086 CPU内部结构

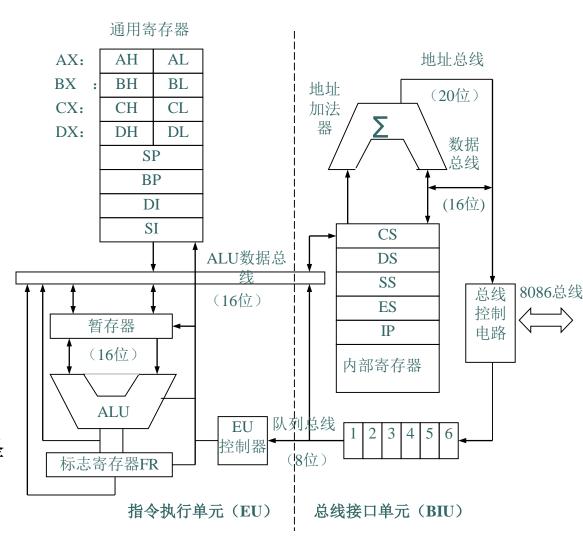
- 组成
- EU和BIU
- 并行操作

o BIU

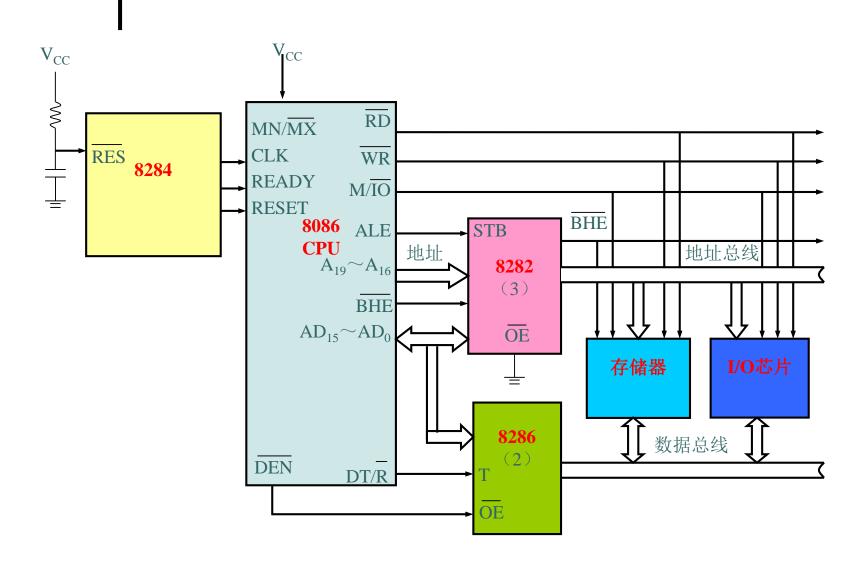
- 取指、读数、送结果
- 指令队列,自动取指

o EU

- 从指令队列中取指令译码执行(不必访问M)
- 节省读指时间



8086 CPU引脚及最小系统配置



●●■最小模式读写时序

o 总线周期

- CPU通过数据总线对存储器或IO端口进行一次读或者写操作所需要的时间(M/IO读/写总线周期)
- 由几个时钟周期组成(标准的4个时钟周期,可插入等待周期Tw)

• 指令周期

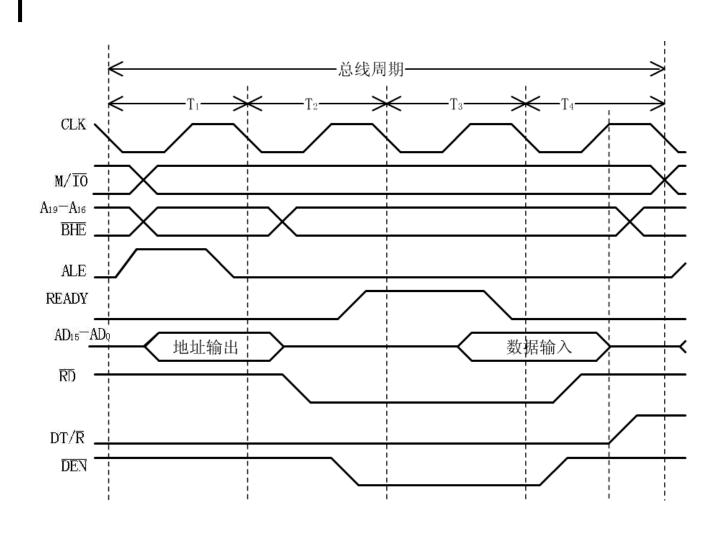
- 读取、译码和执行一条指令所需要的时间
- 由几个总线周期组成

o时序

- 不同操作下CPU引脚信号随时间变化情况
- 时序图

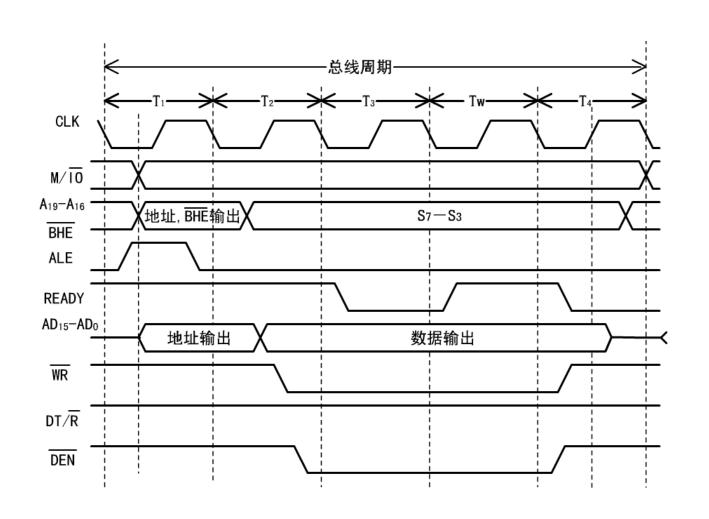
读总线周期

读总线周期: CPU通过DB对M或 I/O进行一次读操作所需时间。



写总线周期

写总线周期: CPU通过DB对M或 I/O进行一次写操作所需时间。



· 第三章 8086寻址方式和指令系统

• • ● 第三章 基本要求

- 指令代码的构成
- o寻址方式
- o 常用指令

数据传送指令,算术运算指令,逻辑运算和移位指令, 控制转移指令,串操作指令

• • 8086指令格式及寻址方式

○ 8086 指令格式

[标号:]指令助记符 操作数[;注释]

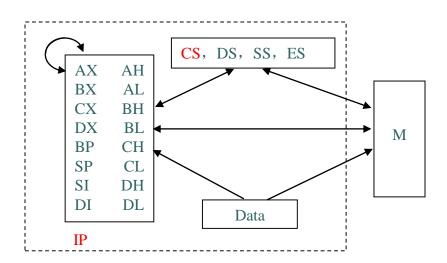
- o寻址方式
- 立即数: 立即寻址方式 (指令队列)
- 寄存器: 寄存器寻址方式(速度最快)
- 存储器
 - 直接寻址方式
 - 寄存器间接寻址方式
 - 寄存器相对寻址方式
 - 基址变址寻址方式
 - 相对基址加变址寻址方式

- 省令系统

- o数据传送类指令
- o算术运算类指令
- o逻辑运算和移位指令
- o字符串处理类指令
- o控制转移类指令

数据传送类指令

- MOV
 - 允许数据传送的途径
- PUSH / POP
- XCHG
- XLAT
- IN / OUT
- o LEA



例: 判断对错,说明原因并改正。

- 1. MOV BL, AX
- 2. MOV 1000H, AX
- 3. MOV DS, 2000H
- 4. MOV ES, DS
- 5. MOV CS, 3000H
- 6. MOV [BX], [1000H]
- 7. MOV ES: AX, 2000H
- 8. MOV VAR[SI][DI], AX
- 9. MOV AL, F0H
- 10. PUSH AL
- 11. IN AL, 100H

• ● 算术运算指令

- o加法
- ADD
- ADC
- INC
- DAA
- AAA

- o减法
- SUB
- SBB
- DEC
- NEG
- CMP

• • 位处理指令

- o逻辑运算指令
 - NOT
 - AND / OR / XOR
 - TEST
- o 算术/逻辑移位指令
 - SHL / SAL: 逻辑左移(无符号数乘2)/算术左移
 - SHR:逻辑右移,无符号数除2
 - SAR: 算术右移(保留符号位),有符号数除2
- 循环移位
 - ROL/ROR
 - RCL/RCR (CL)

控制转移指令

- o 无条件转移指令JMP
- ※ 段内转移:修改IP 短转移(2字节指令) 近转移(3字节指令) 段间转移:修改CS: IP
- 条件转移指令段内短转移跳转距离-128-+127
- o 过程调用指令 CALL 过程返回指令 RET

返回地址:入栈/出栈

修改IP / CS: IP

指令助记符	测试条件	指令功能	
JC	CF=1	有进位	转移
JNC	CF=0	无进位	转移
JZ/JE	ZF=1	结果为 0/相等	转移
JNZ/JNE	ZF=0	不为0、不相等	转移
JS	SF=1	符号为负	转移
JNS	SF=0	符号为正	转移
JO	OF=1	溢出	转移
JNO	OF=0	无溢出	转移
JP/JPE	PF=1	奇偶位为 1/为偶	转移
JNP/JPO	PF=0	奇偶位为 0/为奇	转移

类别	指令助记符	测试条件	指令功能	
无符号数	JA/JNBE	CF∨ZF=0	高于/不低于等于	转移
九刊 与剱	JAE/JNB	CF=0	高等于于/不低于	转移
比较测试	JB/JNAE	CF=1	低于/不高于等于	转移
	JBE/JNA	CF∨ZF=1	低于等于/不高于	转移
带符号数	JG/JNLE	(SF∀OF) ∨ZF=0	大于/不小于等于	转移
市何与剱	JGE/JNL	SF∀OF=0	大于等于/不小于	转移
比较测试	JL/JNGE	SF∀OF=1	小于/不大于等于	转移
	JLE/JNG	(SF∀OF) ∨ZF=1	小于等于/不大于	转移

●●● 串操作指令

- · 字符串操作指令的特点:
 - 源串: DS SI 目的串: ES DI
 - DF: 控制处理方向(CLD 增加; STD 减小)
 - SI、DI自动修改
 - 与重复指令REP配合使用(重复次数: CX)

指令名称	字节/字操作	字节操作	字操作
字符串传送	MOVS 目的串,源串	MOVSB	MOVSW
字符串比较	CMPS 目的串,源串	CMPSB	CMPSW
字符串扫描	SCAS 目的串	SCASB	SCASW
字符串装入	LODS 源串	LODSB	LODSW
字符串存储	STOS 目的串	STOSB	STOSW

第四章 汇编语言程序设计

• • ● 第四章 基本要求

- 常用伪指令:数据定义,符号定义,指定存贮单元类型, 段定义,段寄存器说明,过程(子程序)定义,源程序结束。
- 汇编语言程序的执行过程:编辑、汇编与连接、运行及调试、汇编语言与PC-DOS的接口。
- o 常用DOS子程序: 1、2、6、8、9、A号功能调用
- **程序结构:** 顺序程序, 分支程序, 循环程序, 子程序调用
- o 常用汇编语言程序: 数据显示、排序、传送、比较、查询。

● 常用伪指令 [名字] 助记符 [参数] [;注释]

o 运算符 TYPE, LENGTH, SEG, OFFSET, SIZE

o 数据定义 DB, DW, DD

o 表达式赋值 符号名 EQU 值

• 段定义 段名 SEGMENT [定位类型] [组合类型] [分类名]

逻辑段内容

段名 ENDS

o 段分配语句 ASSUME CS:段名, DS: 段名, SS: 段名, ES: 段名

o 过程定义 过程名 PROC NEAR或FAR

..... ; 指令语句

RET

过程名 ENDP

o 程序结束 END 表达式

□ 汇编语言程序结构

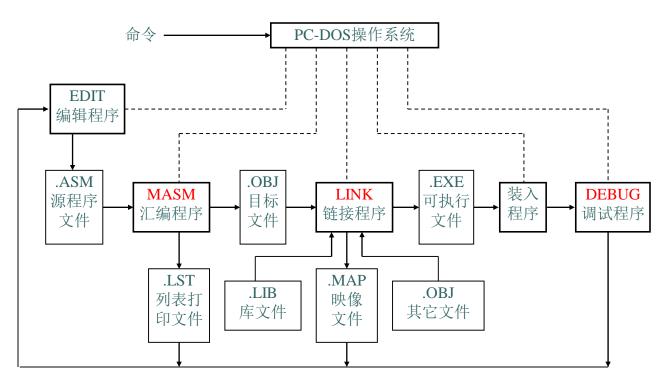
END

MAIN

SEGMENT STACK SSEG ; 定义堆栈段 SSEG ENDS DSEG SEGMENT ; 定义数据段 **ENDS** DSEG **ESEG SEGMENT** ; 定义附加段 **ESEG ENDS** CSEG **SEGMENT** ASSUME CS: CSEG, DS: DSEG, ES: ESEG, SS: SSEG MAIN PROC FAR ; 定义代码段 RET MAIN **ENDP CSEG ENDS**

: 源程序模块结束。

汇编语言程序的运行过程



o Debug调试命令

- ✓显示内存单元内容D
- ✓修改存贮单元内容E
- ✓检查/修改寄存器内容R
- ✓运行命令G
- ✓单步命令T, P
- ✓反汇编命令U
- ✓退出命令Q

• • DOS系统功能调用

o 1号功能: 字符键入并显示,ASCⅡ码存入AL中,等待输入

o 2号功能: 屏幕显示1个字符,ASCII码送入DL寄存器

o 6号功能: 字符输入输出

DL= OFFH,输入。ZF=1无键按下,否则AL存键值。

o 8号功能: 类似1号调用,键入的字符不显示在屏幕上

o 9号功能: 显示字符串

DS:DX(字符串首地址),'\$'字符结束

o OAH号功能:输入字符串

开辟缓冲区,DS:DX指向缓冲区

o 4CH号功能:返回DOS

• • • 程序设计典型应用

- · 字符(数据)排序
- · 字符串移动、比较以及查找
- 判断字符(数据)正负、奇偶及大小
- · 字符的输入与显示(不同数制及字符)

第五章1/0接口和并行接口芯片8255A

• • ● 第五章 基本要求

- 数据输入/输出的概念及指令、接口的概念、接口的基本功能、外设的编址方法
- 数据输入/输出方法:程序控制(无条件传送、查询)方式,程序中断方式,直接存贮器存取方式(DMA)
- 。 8255A的内部结构及工作方式(方式0)
- 8255A应用:通过8255A读入开关状态及控制LED显示,数码管的静态扫描和动态扫描,键盘阵列的识别(行扫描法,反转法)

I/O接口概念

o I/O接口功能

- 输入缓冲
- 输出锁存
- 外设编址控制
- 电平转换
- 转换信息格式: A/D、D/A
- 时序控制: 同步

• 输入/输出方法

- 程序控制方式
- ✓ 无条件传送
- ✓ 查询方式
- 中断控制方式
- 直接存贮器存取(DMA)方式

- 输入/输出指令
- ✓ 直接寻址

IN AL, n

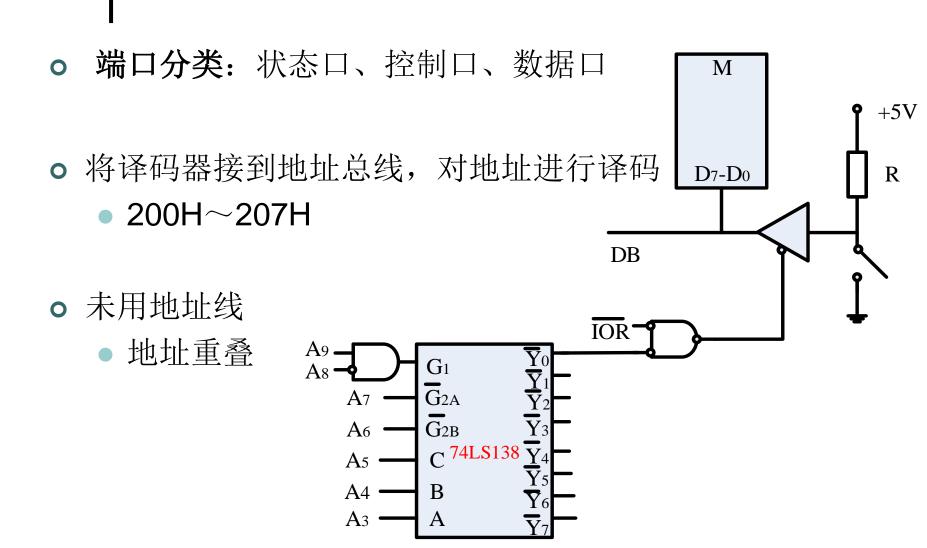
OUT n, AL

✓ 间接寻址

IN AL, DX

OUT DX, AL

I/O端口地址译码方法



8255A

o 8255A的工作方式

方式0——基本输入/输出方式

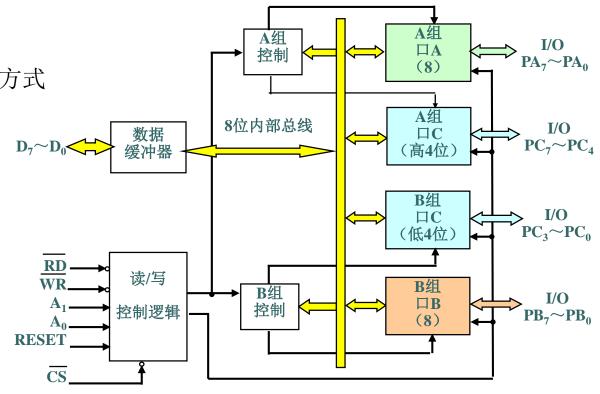
o 8255A的应用

开关输入

LED输出显示

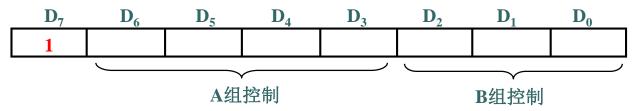
七段码显示

键盘识别



• 8255A控制字

工作方式控制字



$$D4=$$
$$\begin{cases} 0 \text{ PA口为输出} \\ 1 \text{ PA口为输入} \end{cases}$$

$$D2=$$
 $\begin{cases} 0 B组工作在方式0 \\ 1 B组工作在方式1 \end{cases}$

$$D1 = \begin{cases} 0 & PB \square 为输出 \\ 1 & PB \square 为输入 \end{cases}$$

$$D1=$$
 $\begin{cases} 0 & PB \Box 为输出 \\ 1 & PB \Box 为输入 \end{cases}$ $D0=$ $\begin{cases} 0 & PC \Box ((4 \% \Delta)) \end{pmatrix}$ $D0=$ $\begin{cases} 0 & PC \Box ((4 \% \Delta)) \end{pmatrix}$

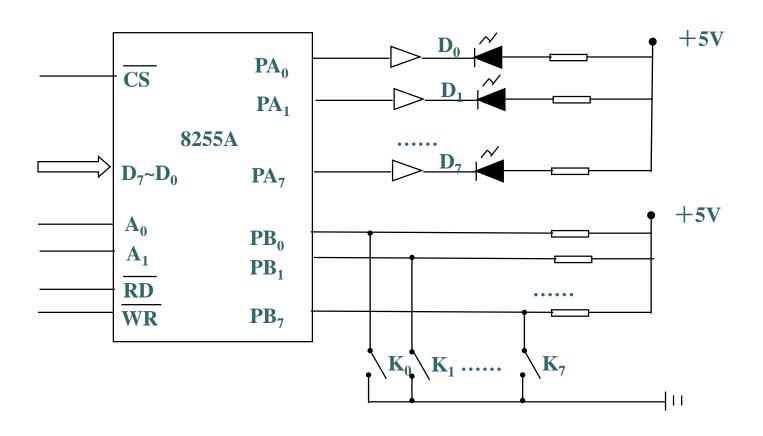
置位/复位控制字

	\mathbf{D}_7	\mathbf{D}_{6}	\mathbf{D}_{5}	$\mathbf{D_4}$	\mathbf{D}_3	$\mathbf{D_2}$	\mathbf{D}_1	$\mathbf{D_0}$
	0	×	×	×				

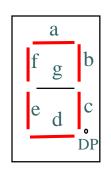
位选择: PC7~PC0的编码

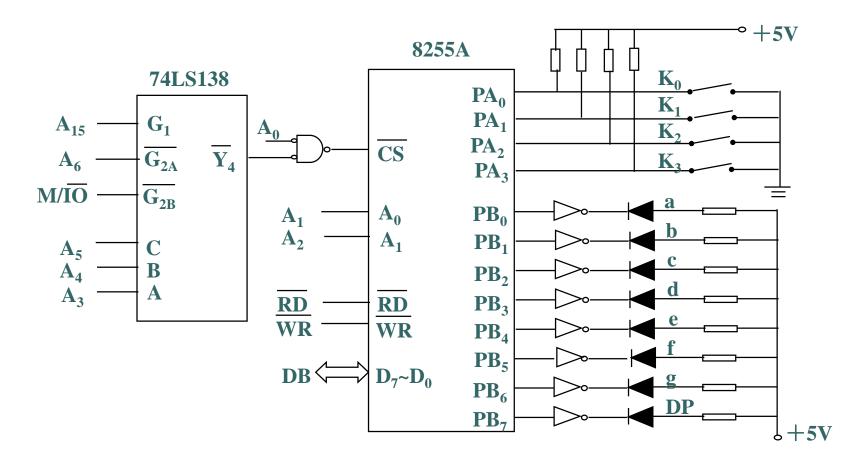
$$D_3D_2D_1= \left\{ egin{array}{lll} 000 & 选择D0位 \\ 001 & 选择D1位 \\ 010 & 选择D2位 \\ 011 & 选择D3位 \\ 100 & 选择D4位 \\ 101 & 选择D5位 \\ 110 & 选择D6位 \\ 111 & 选择D7位 \end{array}
ight.$$

• • 8255A应用举例

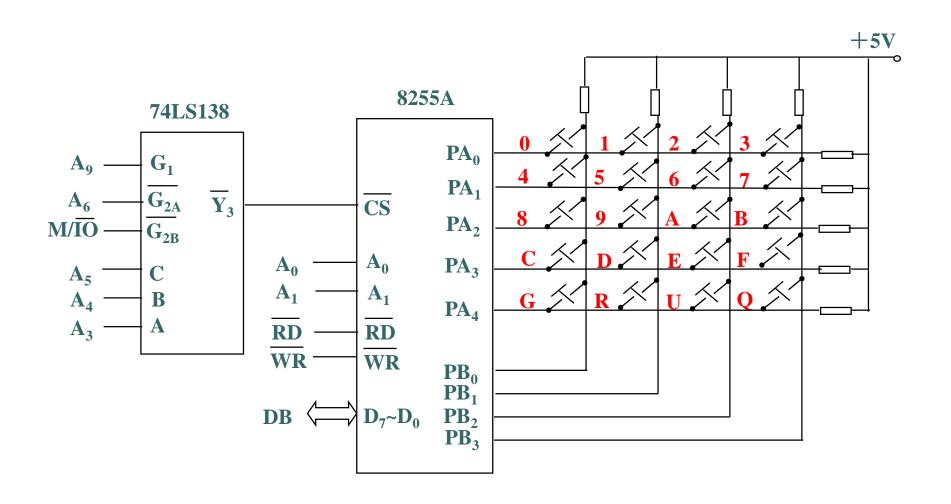


8255A应用举例





8255A应用举例



第二章可编程计数器/定时器8253及其应用

• • • 第六章 基本要求

- o 可编程计数/定时器8253的内部结构
- o 8253的工作方式(方式0、2、3)
- o 8253应用举例:计数器和定时器,初始化编程、波形

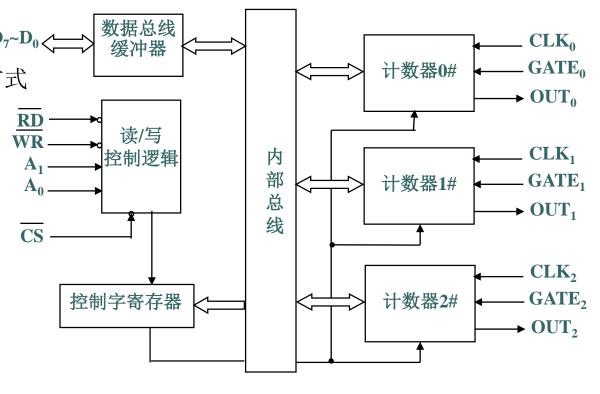
8253内部结构

- 工作方式
- ✓ 方式0——计数结束中断方式
- ✓ 方式2——分频方式
- ✓ 方式3——方波方式
- 初始化编程
- ✓ 写控制字
- ✓ 写计数初值

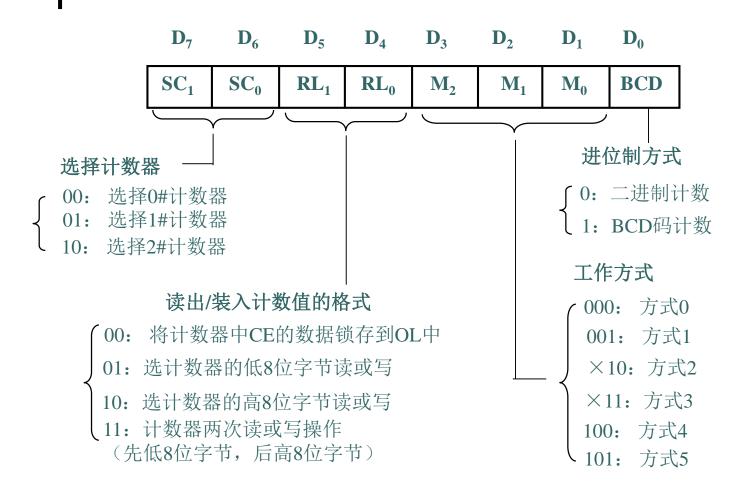
8位(高8位或低8位)

16位(先低后高)

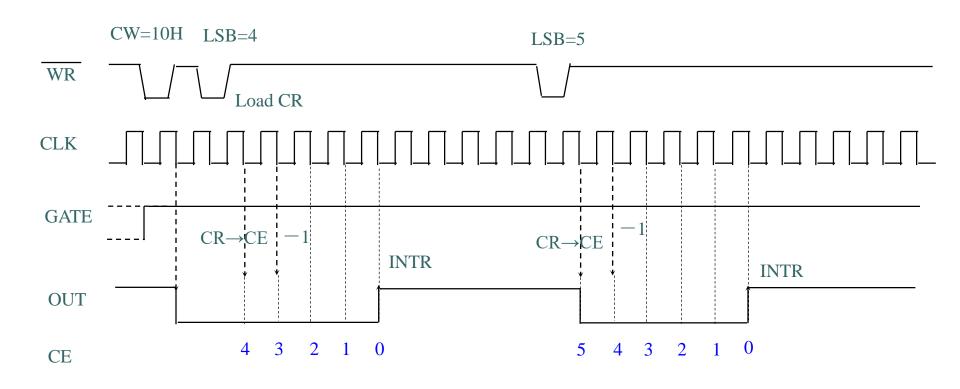
※ 数据进制方式(BCD码——H)



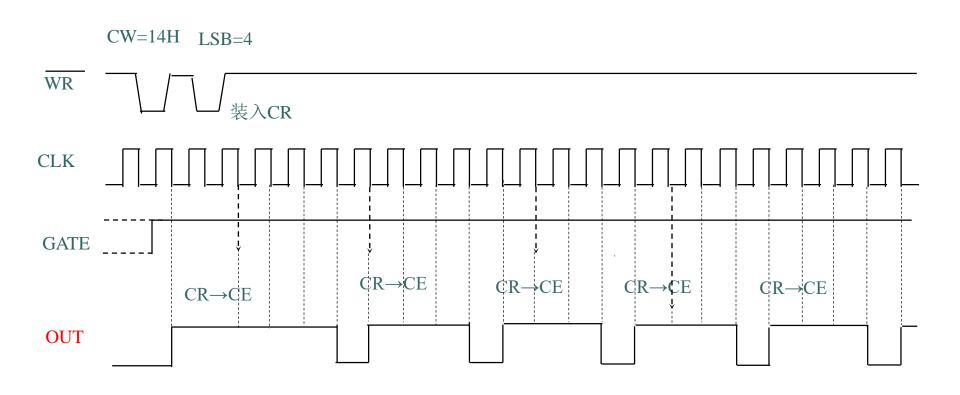
■ 8253控制字



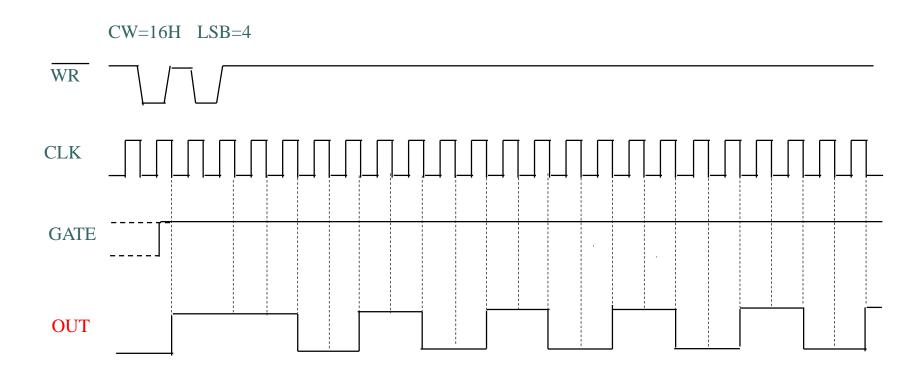
••• 方式0——计数结束中断方式



• • 方式2——分频工作方式



• • 方式3——方波方式



等七章中断和可编程中断控制器8259A

• • 第七章 基本要求

- 中断相关的概念、中断源的识别,8086系列微机的中断分类、中断优先级,中断向量表
- **中断处理过程**(可屏蔽中断、内部中断、非屏蔽中断)
- 中断类型码、中断向量表与中断服务程序的入口地 址三者的关系,中断向量的设置

中断概念

o中断

- CPU正在执行程序过程中,受到内部或外部事件的触发
- 暂停正在执行的程序,转到一段为触发事件而编写的程序去执行
- 该程序执行完,再返回到被打断处继续执行

o 中断优先级

- 内部中断(除法错、INTO、INT、INT n)
- 非屏蔽中断
- 可屏蔽中断
- 单步中断

o中断服务程序入口地址

- 中断服务程序的存放地址,段地址:偏移地址
- 中断向量

中断向量表

0型中断服务程序首地址的偏移地址

0型中断服务程序首地址的段地址

1型中断服务程序首地址的偏移地址

1型中断服务程序首地址的段地址

255型中断服务程序首地址的偏移地址

255型中断服务程序首地址的段地址

0型中断服务程序的指令码

中断程序入口地址表

存储256类中断的中断服务程序入口地址

占用1KB,位于内存00000~003FFH

中断类型号→中断向量

 $n \times 4$

 $4n, 4n+1\rightarrow IP$

4n+2, $4n+3 \rightarrow CS$

中断向量表建立方法

绝对地址置入法 (伪指令)

直接装入法 (MOV指令)

使用串指令装入法 (STOSW指令)

使用DOS调用法 (25H号)

DS: DX=中断服务程序入口地址

AL=中断类型号

****H

H00000

00001H

00002H

00003H

00004H

00005H

00006H

00007H

003FCH

003FDH

003FEH

003FFH

••• 中断执行过程——可屏蔽中断

o 中断请求

外设通过8259A向CPU申请中断。

• 中断响应

条件: ①无内部中断; ②无NMI; ③无总线请求; ④IF=1。 过程:

- CPU通过INTA 引脚发出两个总线周期的低电平信号:通知8259A 响应中断,读中断类型码;
- 保护断点(FR、当前CS及IP);
- 清IF及TF标志;
- 中断类型号n→中断向量地址(n*4)→中断处理程序首地址送IP及 CS(中断向量表),转入中断处理程序。
- 中断处理(保护现场、中断服务、恢复现场等)
- o 中断返回 (IRET)

第九章 和10和01年转换

• • | 第九章 基本要求

- o D/A转换器的主要技术参数
- 常用D/A转换器及其接口(DAC0832):单缓冲输入 电路接法,双缓冲输入电路接法,单极性电压输出电 路接法,双极性电压输出电路接法,CPU通过执行指 令选通DAC的输入控制引脚
- o A/D转换器的主要技术参数
- 常用A/D转换器及其接口(ADC0809):通道选择、 启动信号START产生、转换结束信号EOC的检测(延 时、查询、中断方式)、数字量的读出OE。

DAC0832

o DAC0832的工作方式

直通工作方式

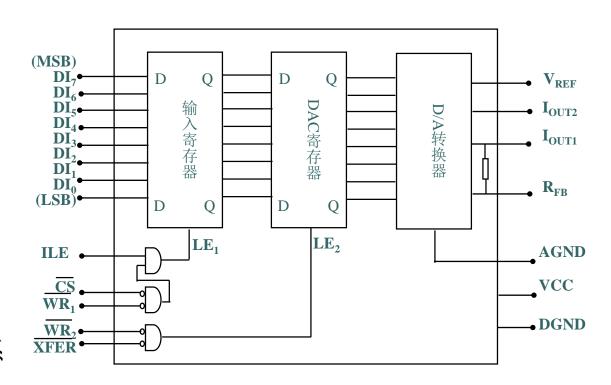
单缓冲工作方式

双缓冲工作方式

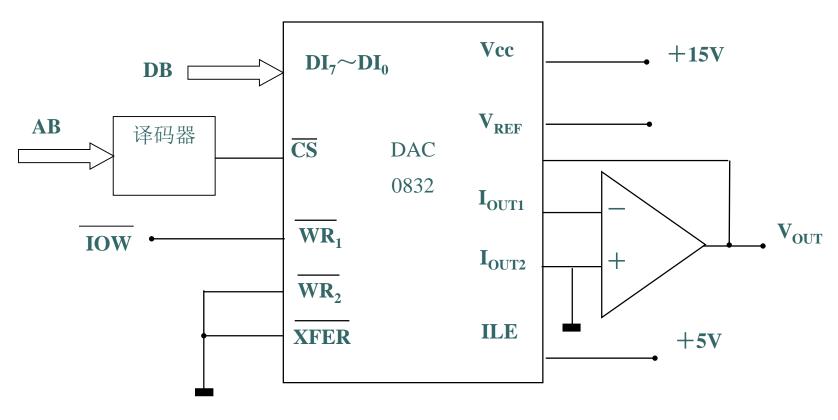
o DAC0832的应用

输出信号的极性

输出信号与数字量的关系

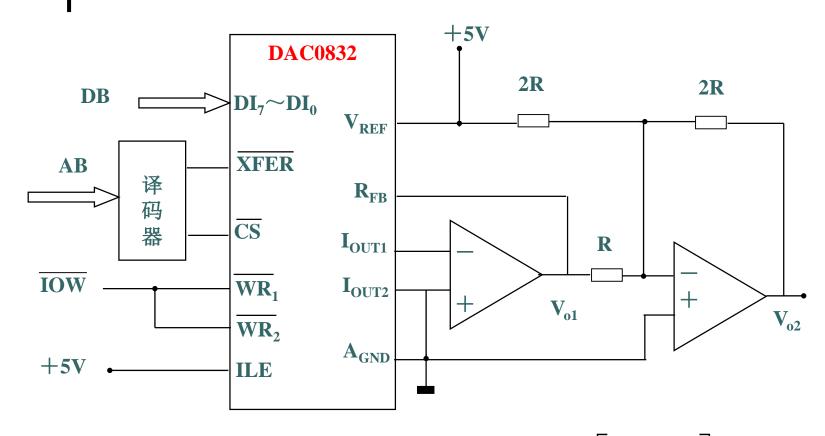


• ● 单缓冲方式,单极性输出



$$V_{\text{out}} = -\left(\frac{D}{256}\right) \cdot V_{\text{ref}}$$

• • 双缓冲方式,双极性输出



$$V_{o1} = -\left(\frac{D}{256}\right) \cdot V_{ref}$$

$$V_{o2} = \left[\frac{D-128}{128} \right] \cdot V_{ref}$$

ADC0809

8路

模拟

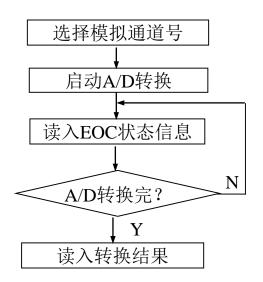
输入

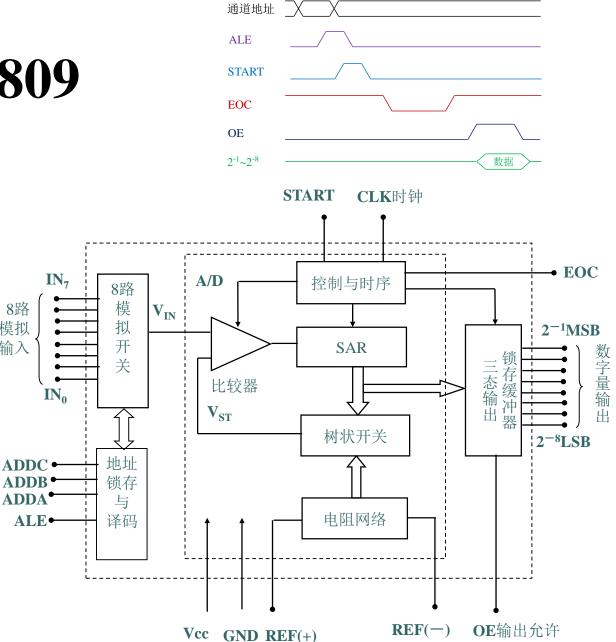
输出与输入的关系

$$D = \frac{V_{IN} - V_{REF(-)}}{V_{REF(+)} - V_{REF(-)}} \times 256$$

编程

信号接法及时序





应用举例

