

# 复习课（4-6章）

第4部分：半导体存储器

第5部分：数字量输入输出

总线、接口概念

\*接口电路（芯片）、端口地址

\* 数据传送方式

\* 中断电路及其处理

\*定时/计数器电路与应用

\* 并行接口电路与应用

\* 串行接口电路与应用

**DMA**电路与应用

第6部分：模拟量输入输出

所有控制字格式不需要死记硬背，提供参考查阅!

# \*四种I/O方式

问题的提出：**CPU与外设的工作速度不一致，如何解决效率和可靠性？**

- 无条件传送
- 程序I/O（查询）--三种端口数据/状态/控制及作用，
- 中断，效率较高，CPU响应较快，但传送由中断软件完成
- DMA，可以实现外设和存储器之间的数据高速传送，不需要处理器。效率较高，硬件完成，但电路复杂

# 总线、接口概念

## CPU 片内总线

——CPU内部，各部件之间的连接总线

## CPU总线

——计算机内部，CPU与各芯片的连接总线

## 系统总线（PC总线）

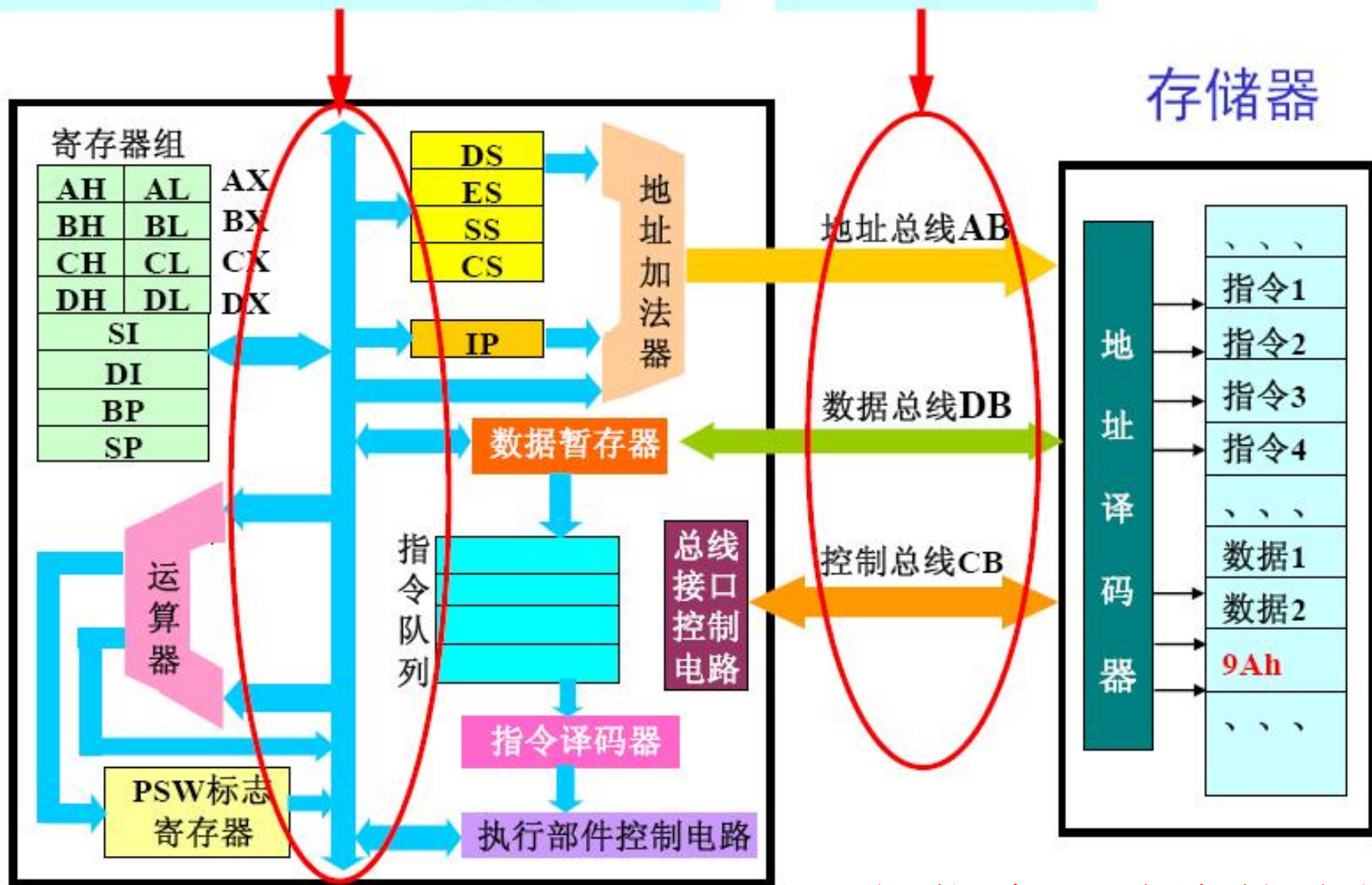
——计算机内部，插卡与插卡间的连接总线  
使计算机成为开放体系，实现技术的兼容与共享；

## 设备级总线

——计算机与外部设备之间的连接总线，  
进一步使计算机真正成为开放体系

# (8086 CPU) 片内总线

# CPU总线



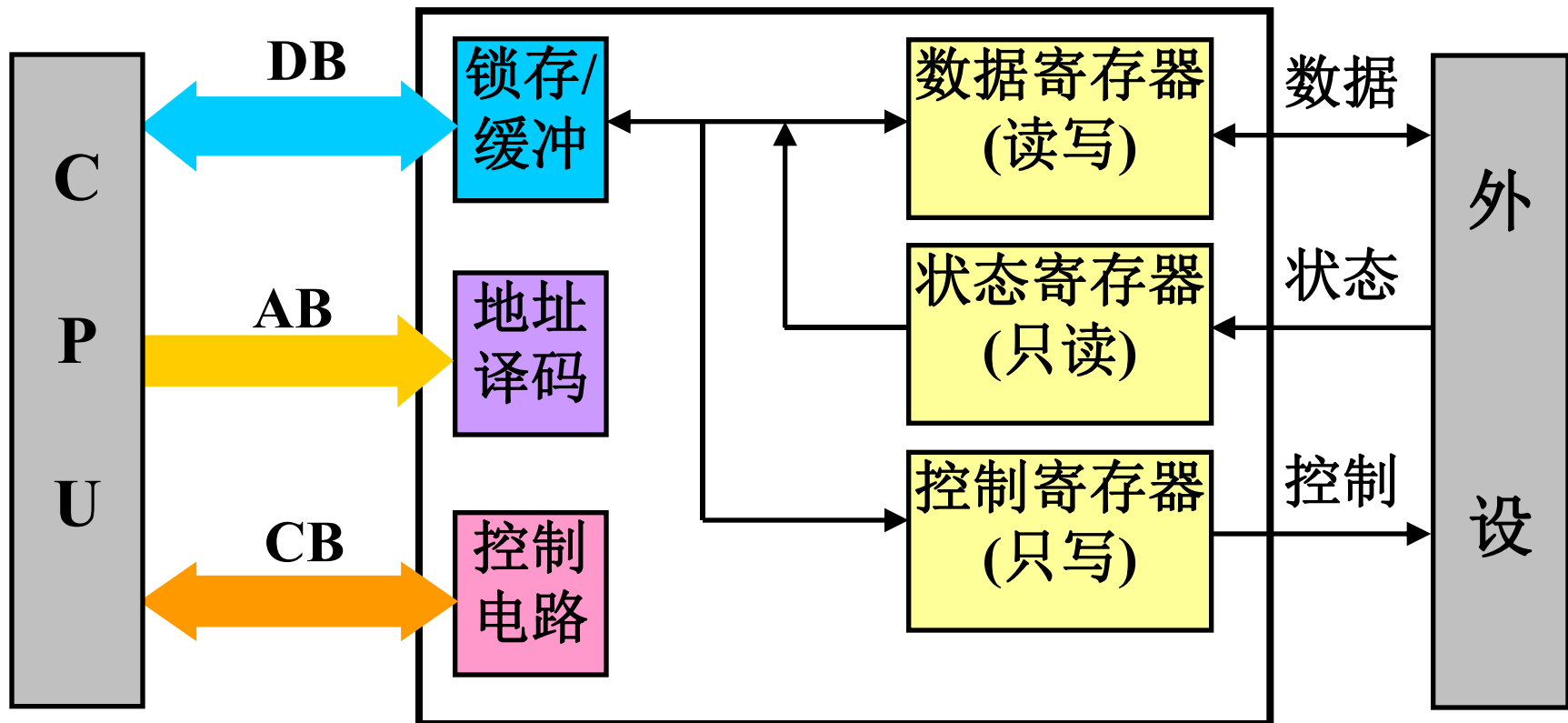
• 地址总线 (AB) —— 单向

数据总线 (DB) —— 双向

控制总线 (CB) —— 每根线单向

理解指令对总线的访问!

# 接口电路的组成结构



数据端口(读写), 同时有输入缓冲器和输出锁存器  
状态端口(只读), 有输入三态缓冲器  
控制端口(只写), 有输出锁存器

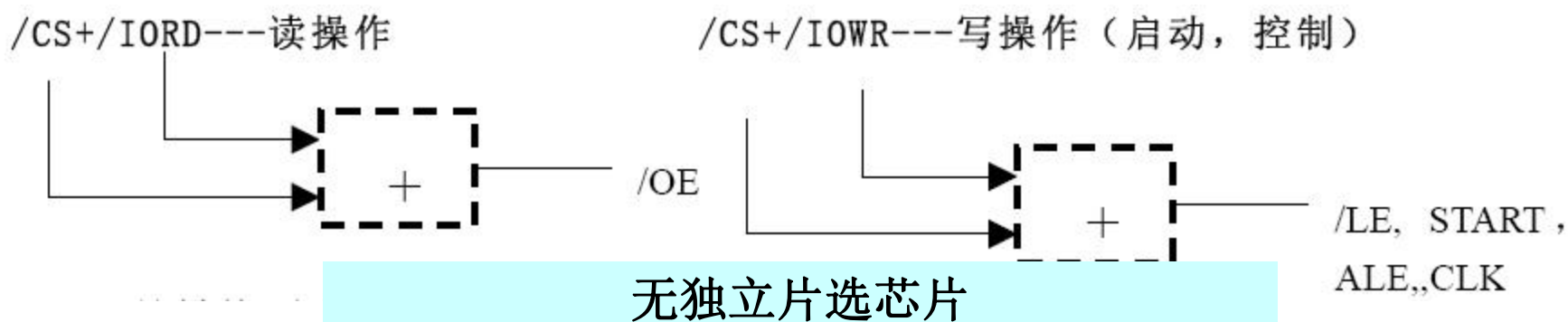
# I/O接口技术—译码与片选

\*所有接口电路的译码、指令操作（读图），简单互连

## \*设计I/O端口译码电路

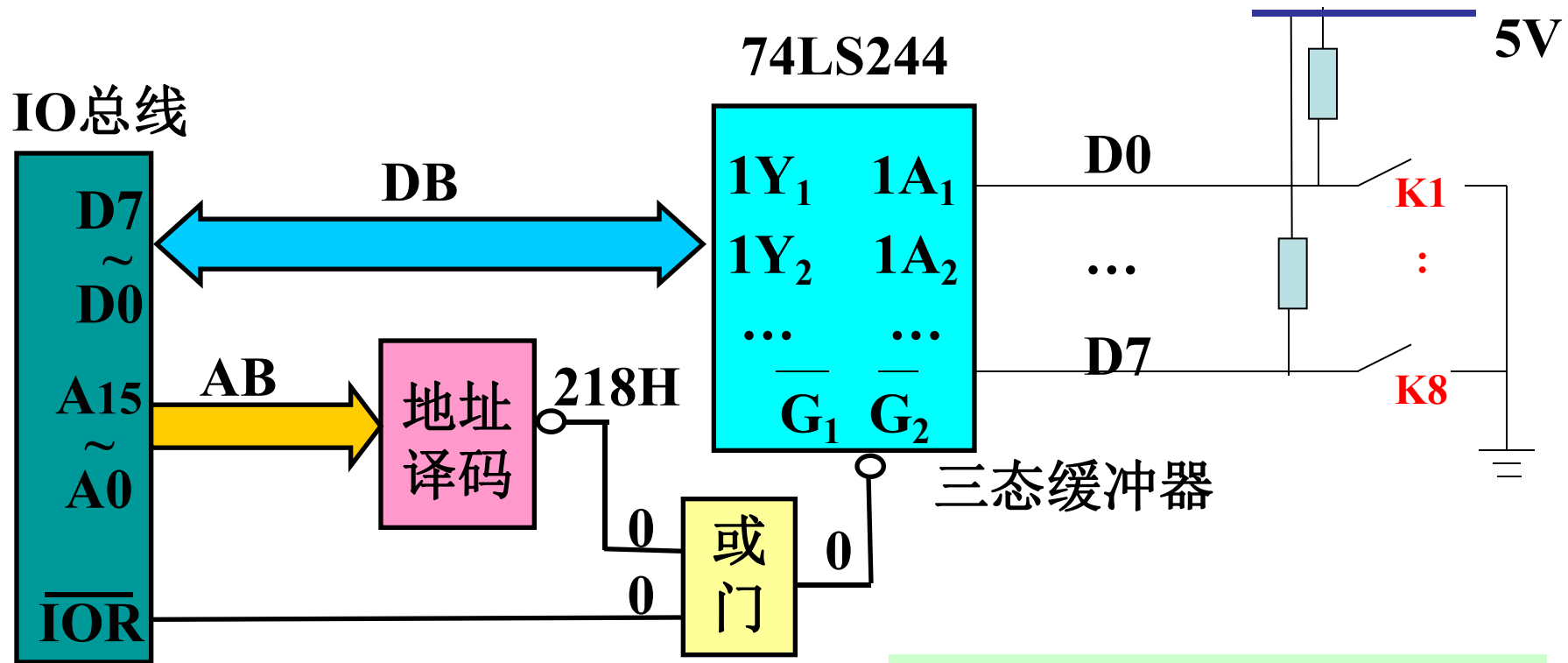
1. 根据端口地址确定地址信号A15~A0的条件取值，用门电路、译码器及组合实现满足条件情况的电路。
2. 端口的选通信号通常为低电平有效，除端口的地址信号参加译码外，控制信号 $\overline{\text{IOW}}$ 、 $\overline{\text{IOR}}$ （ $\text{IO}/\overline{\text{M}}$ 、 $\overline{\text{AEN}}$ 也可参加译码）

多种选择设计：可选常规，易实现



# \*简单I/O: 开关电平变换连接

\*输入:三态缓冲器(244), 输出:锁存器(273), 可编程并口芯片8255

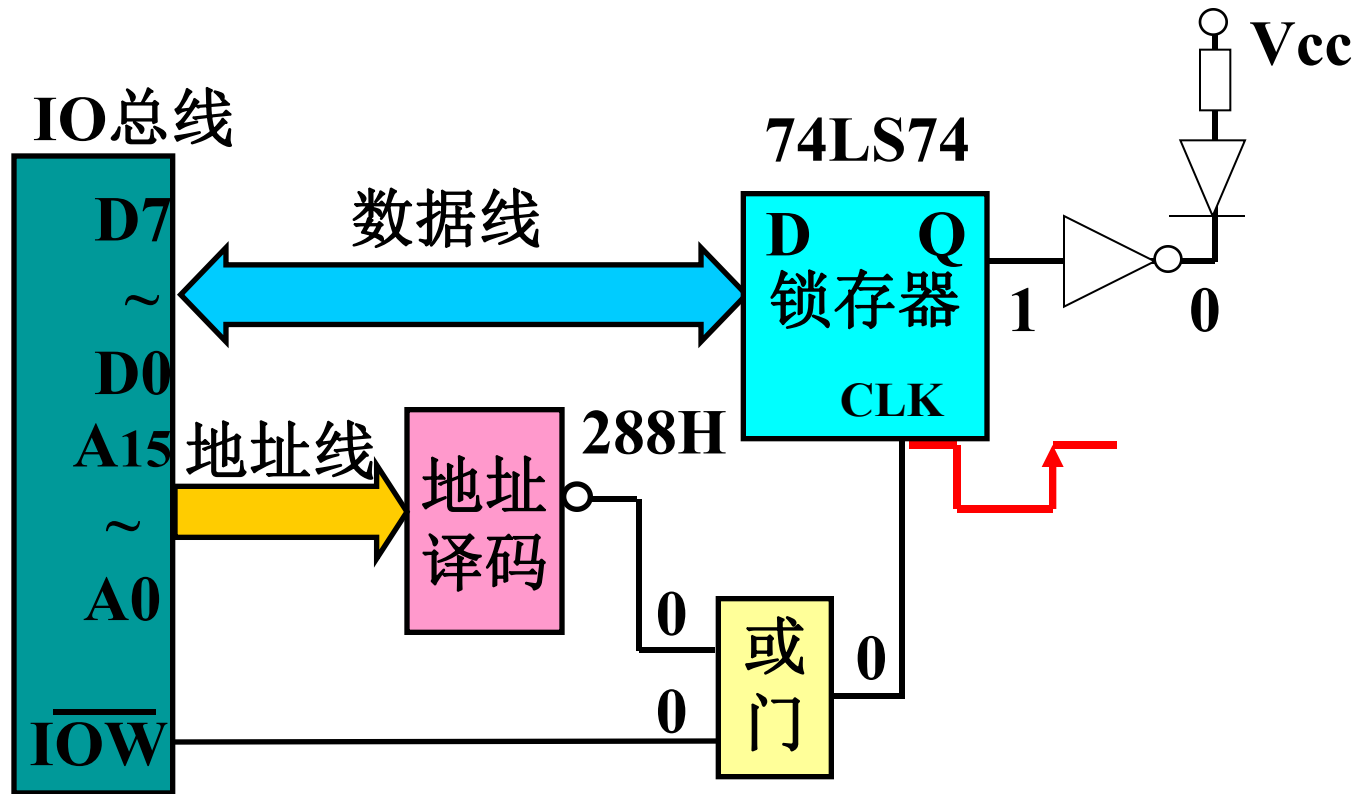


MOV DX, 218H  
IN AL, DX

简单I/O指令控制:  
输入三态缓冲



# \*简单I/O: LED/八段数码管的连接



```
MOV AL, 81H  
MOV DX, 288H  
OUT DX, AL
```

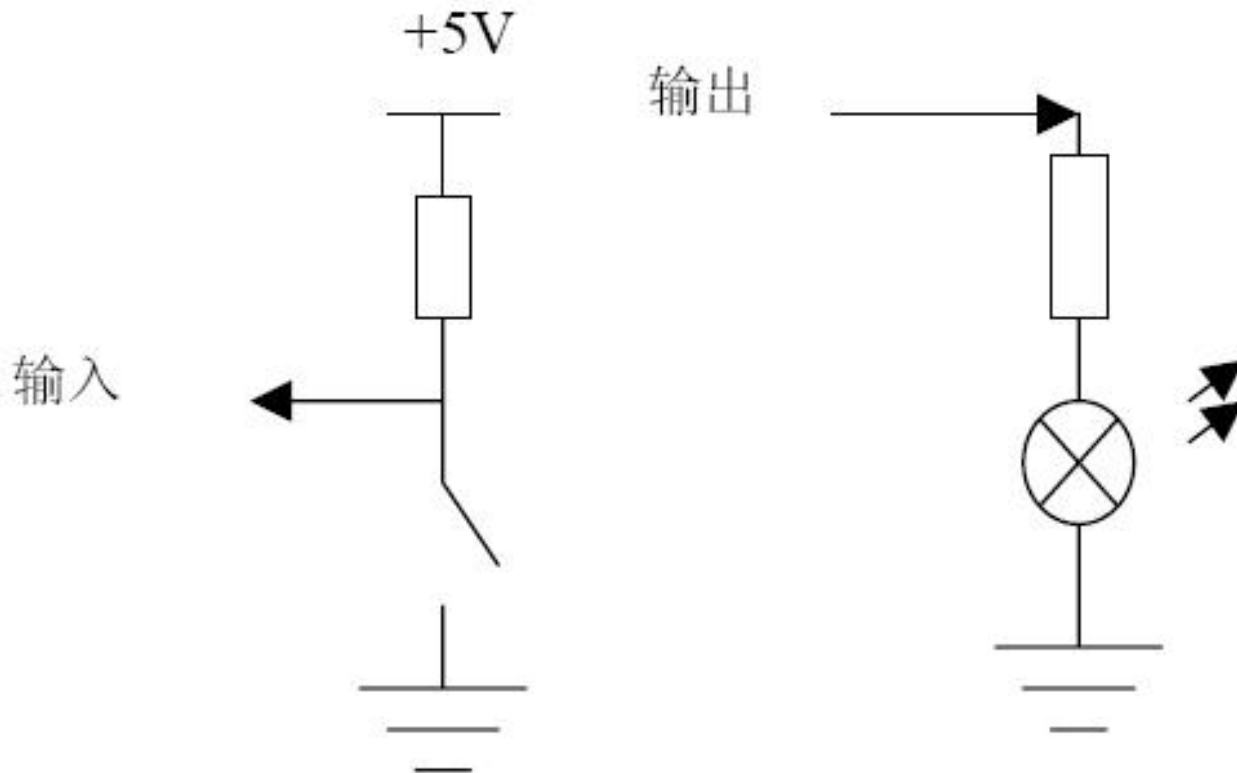
简单I/O控制:  
输出使能/锁存



# \*简单I/O的连接方法

硬件连线:

开关转换(上拉电阻)和LED驱动(限流电阻)



另: P247  
图5.11, 经  
反相器7406  
后再接LED  
接电阻到  
VCC

# 中断及中断控制器8259

中断概念: CPU与外设交换信息的一种重要方式

==》硬件手段, 改变CPU执行程序的顺序 (程序流)

了解: 中断屏蔽; 中断优先级; 中断嵌套;

掌握: 中断类型, 中断概念; 中断处理过程, 中断服务程序;

应用: 外部中断的随机性

## 分类

### (1)外部中断

不可屏蔽中断NMI/可屏蔽中断INTR

### (2)内部中断

- INT n: 软中断

- CPU的某些运算错误引起的中断: 除法错、溢出

- 由调试程序debug设置的中断: 单步/断点

# 中断向量表

**\*中断向量表：**中断服务程序入口地址表。8086/8088系统允许处理256种类型的中断，对应类型号为0~FFH。在存储器的00000H~003FFH占1K字节空间存放中断向量。

9. 中断类型码为40H的中断服务程序入口地址存放在中断向量表中的起始地址是\_\_\_\_\_。答案：C

A. DS:0040H

B. DS:0100H

C. 0000H:0100H

D. 0000H:0040H

## 03年考题

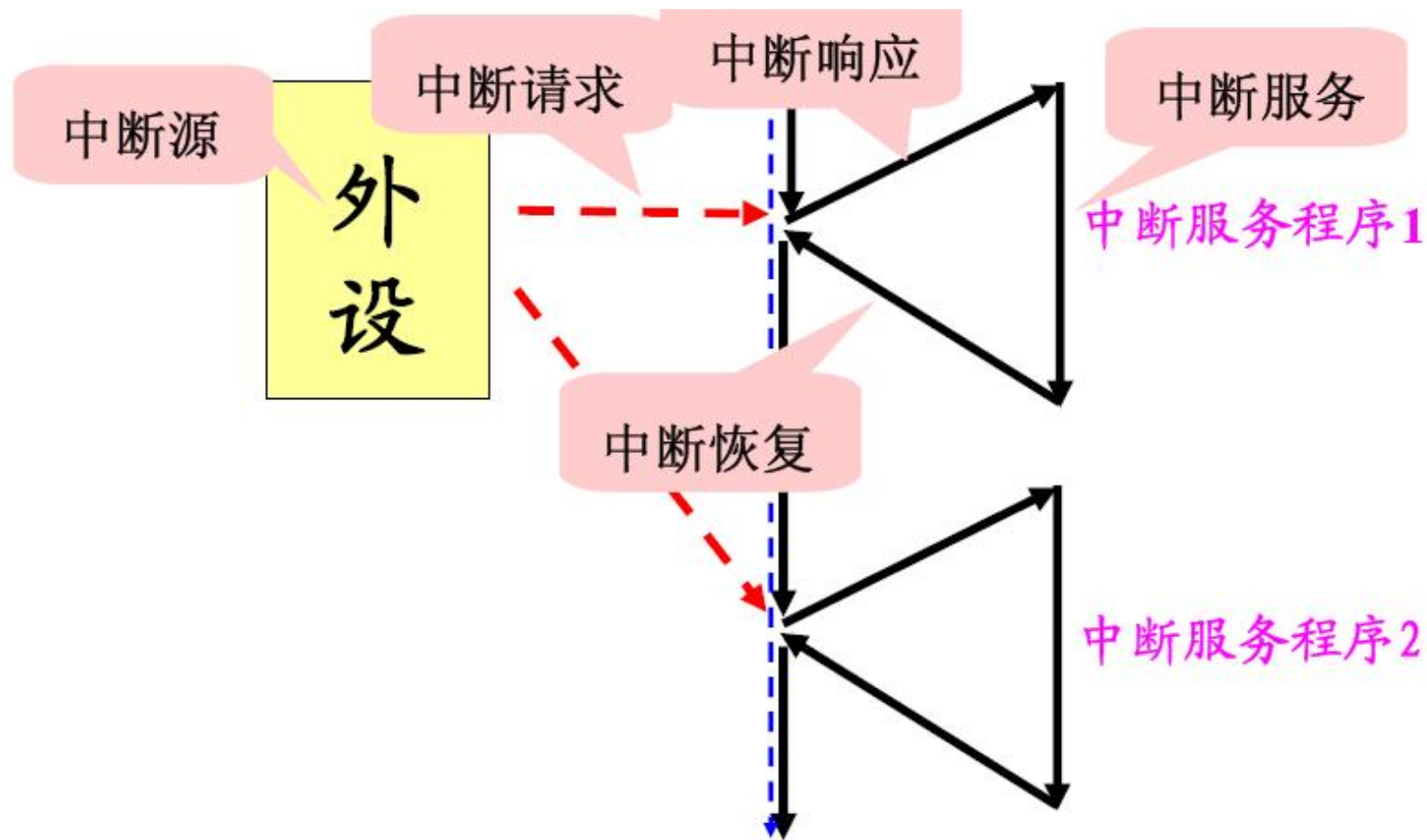
10. 8086/8088 中断系统可处理 256 个中断源，其中断向量的作用是\_\_\_\_\_C\_\_\_\_\_。

A. 确定中断源    B. 指定对应中断保护数据区的地址    C. 指定对应中断服务程序的地址    D. 存放对应中断响应执行的第一条指令

# 中断响应原理

中断实现：

中断源---中断请求---中断响应---中断服务---中断恢复



CPU响应中断三个条件：外设提出中断申请；本中断未被屏蔽；CPU中断允许。

# 中断响应过程

## \*CPU步骤:

- (1) 获取中断类型号;
- (2) 将标志寄存器**FLAGS**的值入栈;
- (3) 将中断允许标志**IF**和单步标志**TF**清0——屏蔽外部其它中断请求, 避免CPU以单步方式执行中断处理程序;
- (4) 保护断点——将当前下一条指令的**CS**和**IP**的值入栈;
- (5) 根据中断类型号到中断向量表中找到中断向量, 转入相应中断服务子程序  
(0: 4\*n取中断向量→CS:IP)
- (6) 中断处理程序结束后, **IRET**从堆栈中依次弹出**IP**、**CS**和**FLAGS**, 然后返回主程序断点处, 继续执行原来的程序

# 中断响应过程（续）

\*CPU如何获得中断类型号？

□INT n;

□除法=0，INTO溢出=4，单步=1，断点=3；

□不可屏蔽中断NMI，自动获得2；

□可屏蔽中断，由接口电路（8259）通过数据总线向CPU发；

保护断点：FLAGS入栈；IF=0 TF=0；CS、IP入栈；自动  
保护现场：PUSH指令将某些寄存器的内容压入堆栈；手动

- 在中断服务程序中STI，允许中断嵌套
- 在中断服务程序中一定要手动保护现场

# 中断处理程序

2011年考题

## 实验四（2）注意保护现场：相关的通用寄存器+段寄存器

```
INT_PROC PROC FAR
```

```
PUSH AX
```

```
PUSH CX
```

```
PUSH DX
```

←补充**PUSH DS**      补充 **STI** ;开中断

```
MOV AX,DATA
```

```
MOV DS,AX
```

```
DEC IRQ_TIMES
```

```
MOV CX, 0FFFFH
```

```
LOOP2: NOP
```

```
LOOP LOOP2
```

```
POP DX
```

```
POP CX
```

←补充**POP DS**      补充 **CLI**;关中断

```
MOV AL,20H
```

```
OUT 20H,AL      ;写OCW2
```

```
POP AX
```

```
IRET
```

```
INT_PROC ENDP
```



# i8259A可编程中断控制器

多中断源如何协调？查询、优先级？屏蔽？自动？

中断控制逻辑→专用IC i8259

\*（了解）i8259A两步编程：

1、初始化编程——一次写入（编程）

初始化命令字ICW1~4

2、工作编程——可多次写入（编程）

工作命令字OCW1~3

初始化编程顺序：

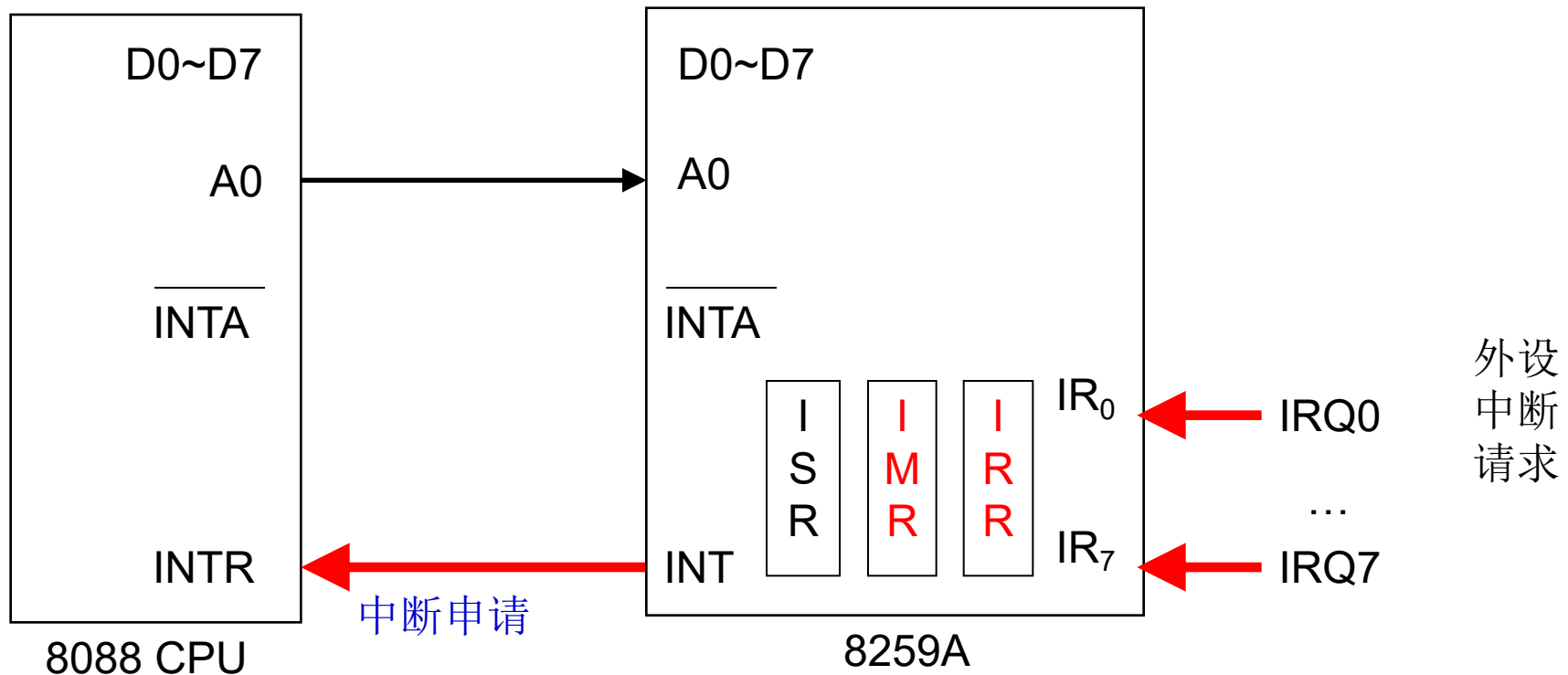
ICW1 → ICW2

→[多片时写ICW3] ->[需要时写ICW4]

# 8259的中断响应顺序

(1) 当有一条或多条中断请求 $IR_0 \sim IR_7$ 变高，**IRR**寄存器相应位置位；

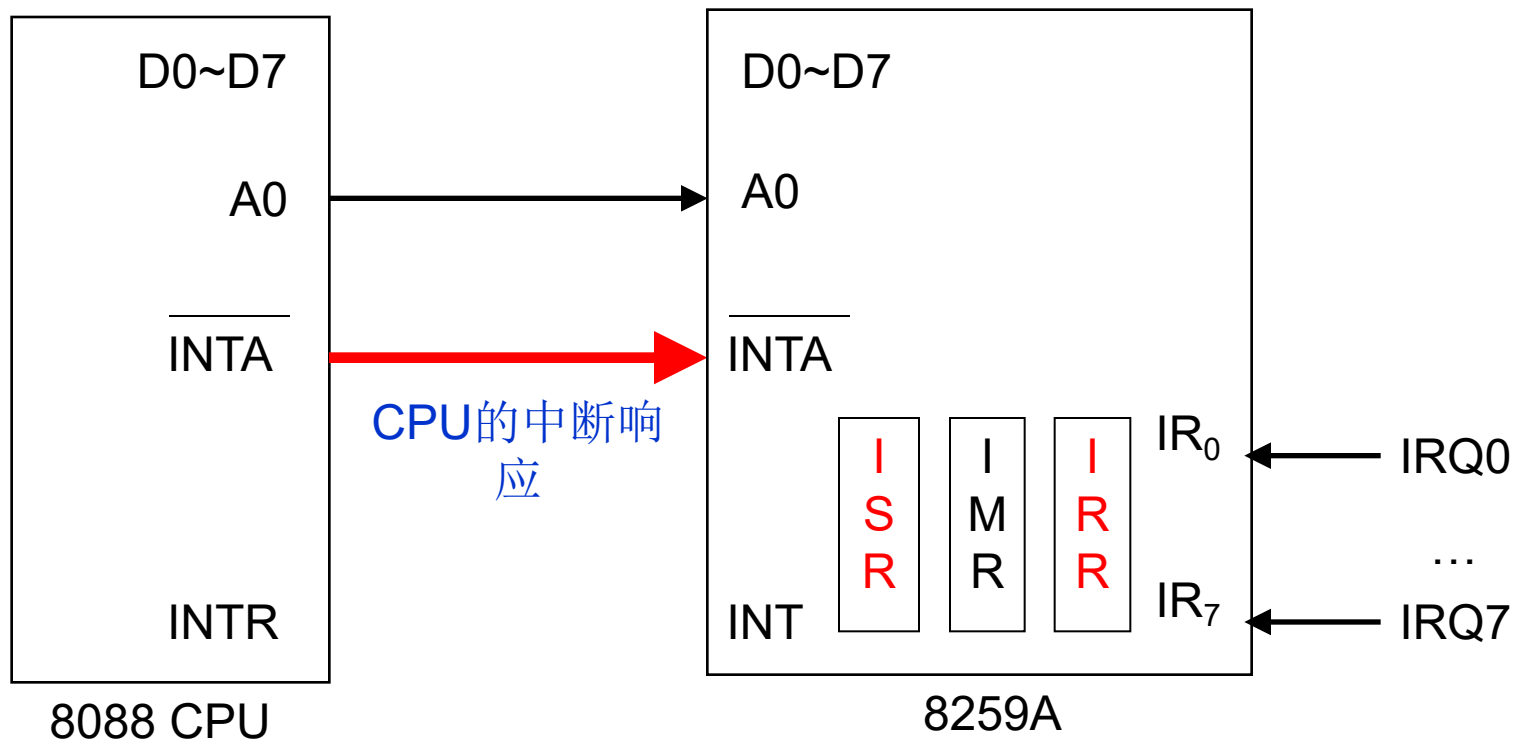
(2) 若这些请求至少一条中断允许（**IMR**没有屏蔽），则8259由**INT**引脚向CPU发出**中断请求**信号；



# 8259的中断响应顺序

(3) 若CPU中断开 ( $IF=1$ )，则执行完当前指令后用 $\overline{INTA}$ 信号**应答**；

(4) 8259收到应答后， $IRR$ 对应位**清零**， $ISR$ 中最高优先权对应位**置位**；

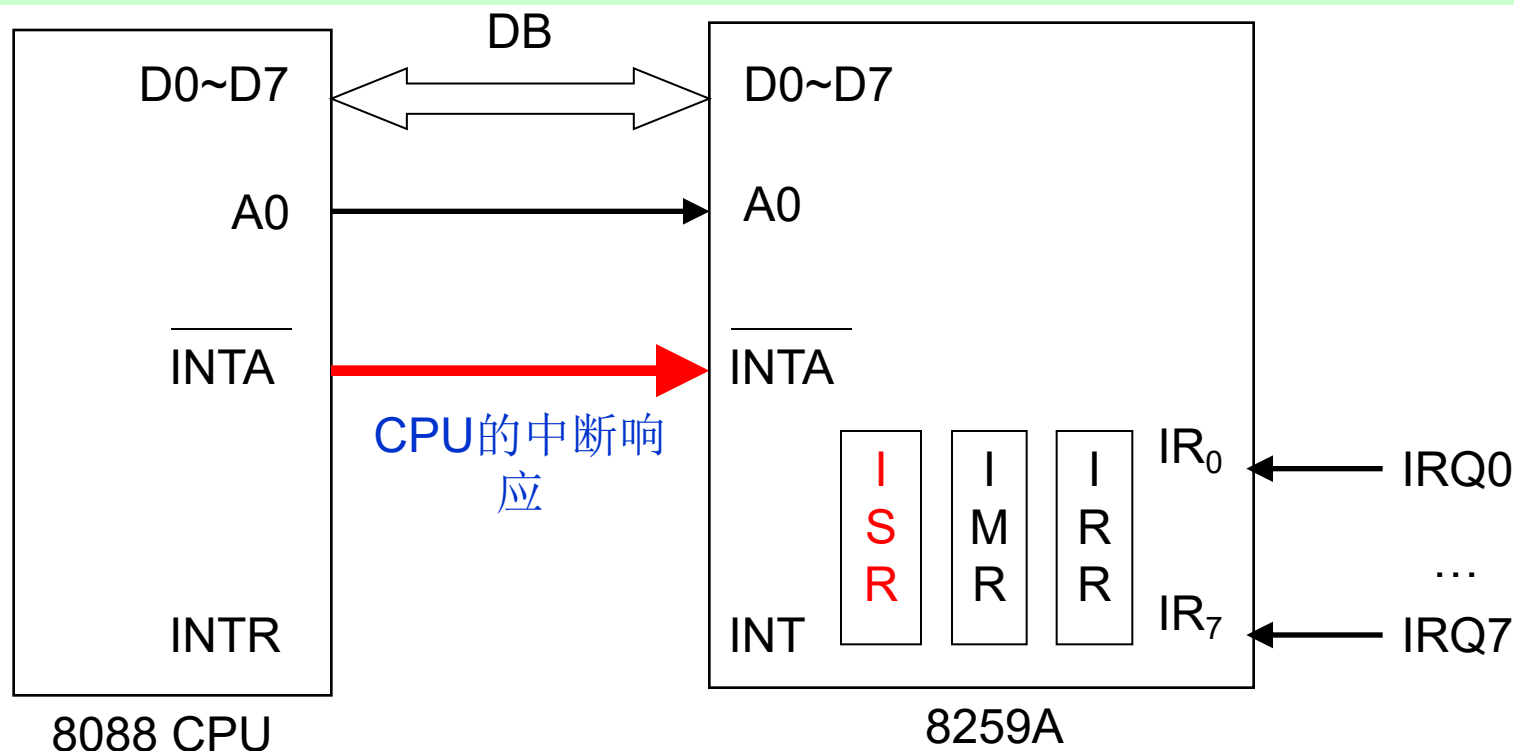


外设  
中断  
请求

# 8259的中断响应顺序

(5) CPU发出第2个 $\overline{\text{INTA}}$ 脉冲，(6) 8259向数据总线送出8位中断类型码，CPU读之并乘4，找到相应中断服务程序入口地址，转入（CPU进入中断周期后自动关中断）。

(7) CPU给8259发EOI，使ISR中相应位复位，然后返回。



外设  
中断  
请求

比一般的软中断指令要多一个IO总线读周期

2(随堂测试)、在由i8259中断控制器管理的、且允许中断嵌套的一个系统应用中，\_\_不是程序员在进入中断服务程序后应该完成的工作。（多选）

A

添加开中断指令STI，允许中断嵌套

B

编写指令，手动保护现场寄存器

C

编写指令，手动保护断点

D

编写指令，让i8259向CPU发送中断类型号n。

提交

# ICW2中断类型号寄存器

D7	D6	D5	D4	D3	D2	D1	D0
T7	T6	T5	T4	T3	×	×	×

地址：A0=1

对于8086/8088，D7~D3是用户编程确定的中断类型号高五位（T7~T3），D2~D0不写，响应时根据输入中断源0~7，自动填入中断类型号。例如IR2的请求，D2~D0会被自动填入010。

例：

MOV AL, 00001000B ;中断类型号从08H开始

OUT 21H, AL

各引脚输入的中断请求安排型号：

n=08H—IR0

n=09H—IR1

...

## 选择题(13年考题)

8086/8088处理器系统通过可编程中断控制器8259A响应外部中断，如果系统设计其IRQ0对应的中断类型号是16，则IRQ2 中断对应的中断矢量存放在

- ☐ A 0000:0012H
- ☐ B 0000:0028H
- ☐ C 0000:0030H
- ☒ D 0000:0048H

提交



# T/C 8253：定时器/计数器

## 背景

为CPU和外部设备提供实时时钟：

\*可编程定时器/计数器(减法)，T/C**差别**（CLK）：

□ **Timer** ——为CPU和外设提供实时时钟、定时或延时控制，如定时中断、定时检测、定时扫描；

□ **Counter**——对外部事件计数

□ 硬件级连的概念\*

□ 定时中断\*

# T/C 8253: 定时器/计数器

- ◆控制字：先低后高字节，BIN/BCD；
- ◆六种工作方式：  
波形特征、计数初值（n可编程设置）与产生脉冲频率的关系、CLK/GATE对OUT的作用、触发方式；  
软件触发—写入初值，硬件触发—Gate正跳变 (08考题)
- ◆方式0:计满n+1脉冲则out产生高电平
- ◆递减计数，计数值最大时应置时间常数为0(和LOOP CX相同)

# T/C 8253: 定时器/计数器

掌握六种方式:

- ◆方式0、1和方式4、5不能自动重复，常用于计数；
- ◆方式2、3能自动重复，常用于定时；

常用:

方式0: 计完 $n+1$ 个脉冲发中断请求，软件启动

方式2: 速率发生器，对clk进行 $n$ 分频

方式3: 方波发生器，区分 $n$ 奇数/偶数情况

# 8255A工作方式

- **方式0：**基本输入输出方式
  - ◆ 适用于无条件传送和查询方式的接口电路
- **方式1：**选通输入输出方式
  - ◆ 适用于查询和中断方式的接口电路
- **方式2：**双向选通传送方式
  - ◆ 适用于与双向传送数据的外设
  - ◆ 适用于查询和中断方式的接口电路

# 并行I/O接口 i8255

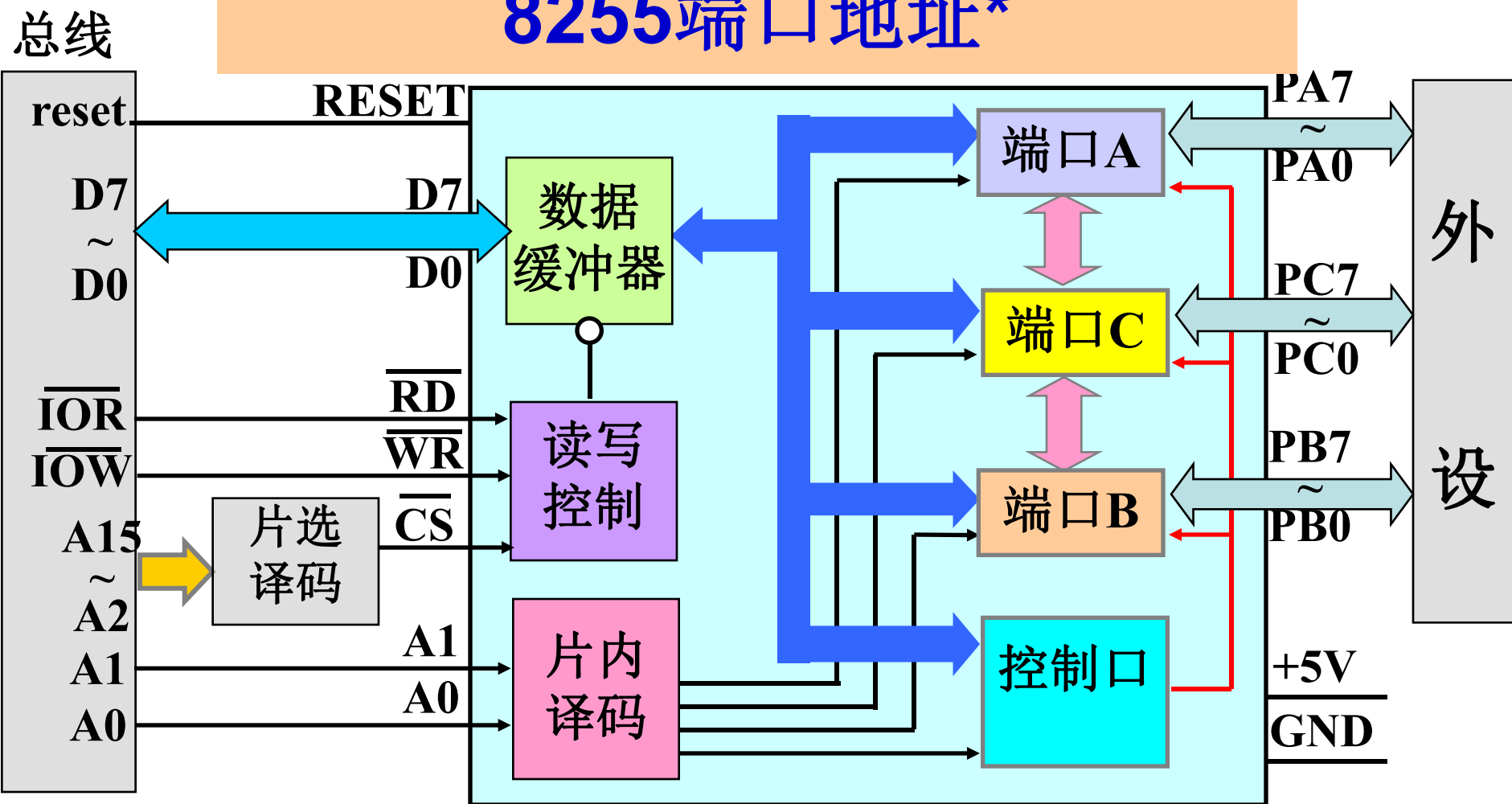
- ◆方式0：基本直接I/O，含输出锁存/输入缓冲无锁存；
  - ◆方式1：选通I/O(A口/B口，C口联络)；
  - ◆方式2：双向I/O方式(A口，C口联络)；
- 方式1/2带INTR，可中断；IBF/OBF信号的作用(输入/输出缓冲区满)；而方式0只能查询。

8255控制字：

- (1)方式字，特征位D7=1；
- (2)PC口位控功能字，特征位D7=0，允许CPU用输出指令单独对C口的某一位写入“1”或“0”，正确的方法是通过写8255的控制寄存器方式写入；

C口某一位脉冲的产生：位0-1-0

# 8255端口地址\*



片内译码电路

A1A0=

数据端口A、B、C 每个端口8位，通过编程设定其为 输入口或输出口→和外设传送信息

00 A口； 01=B口； 10 C口； 11 控制口

# 例题

（2017年）8088CPU 通过并行接口 i8255A 与打印机连接的基本系统连线如图 1(a)所示，打印机与 CPU 的主要接口信号时序如图 1(b)所示。其中，/STB 是数据选通脉冲，打印机在其上升沿读入数据。/BUSY 信号变高，表示打印机正忙。因此 CPU 为了每发一个字节数据，需要在查询 /BUSY 状态，当 /BUSY 信号为低时，送出待打印数据，随后发出 /STB 信号上升沿以使打印机读取数据



# 例题

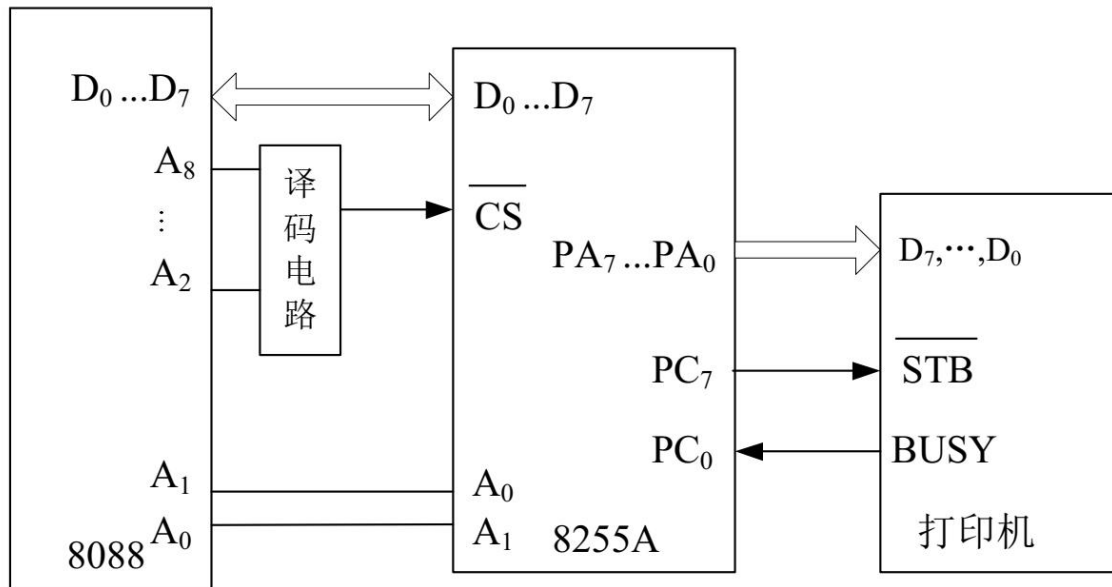


图 1 (a) 打印机接口图

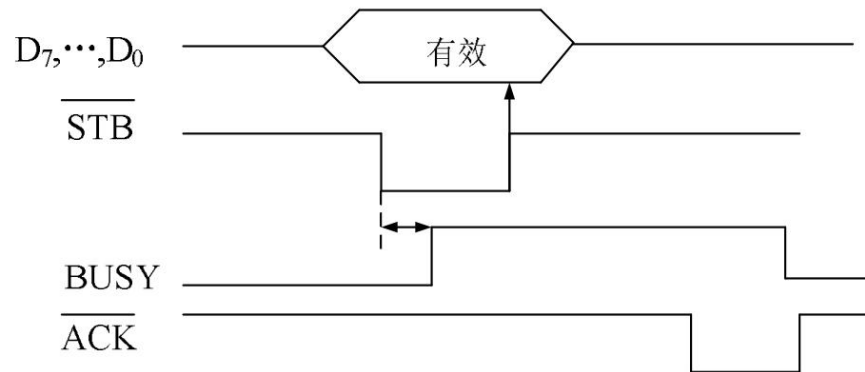


图 1 (b) 打印机时序图

# 例题

(1)说明 8255 的 A 口和 C 口的工作方式？（2 分）

A 口方式 0 输出（1 分），C 口方式 0，上半口输出，下半口输入（1 分）

# 例题

(2)采用常用门电路与译码器 74LS138 组合实现译码电路。设计 A15-A9 未参加译码，当 A8~A2 =1010100B 时译码电路输出低电平用于 8255 的片选/CS。写出译码电路设计过程并画出连线图。（8 分）。由于 A15-A9 未参加译码，则 8255 地址重叠，写出任意四个 8255A 口地址（4 分）

	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>7</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>	
8255PA	*	*	*	*	*	*	*	1	0	1	0	1	0	0	0	0	150H
8255PB	*	*	*	*	*	*	*	1	0	1	0	1	0	0	1	0	152H
8255PC	*	*	*	*	*	*	*	1	0	1	0	1	0	0	0	1	151H
8255PZ	*	*	*	*	*	*	*	1	0	1	0	1	0	0	1	1	153H

译码电路（例）：（编码：6 根线：6 分；译码 1 分，说明 1 分）

G1	G2A	G2B	C	B	A	/Y4
A8&A6	A7	A5	A4	A3	A2	/CS

8255PA 地址重叠：例 150H；350H;550H;750H (各 1 分)；地址错有过程 2 分

# 例题

(3)按照图 1 地址译码连线, 分别 写出 A15-A9 全 0 时 8255 的 A、B、C 口和控制端口的地址, (注意 8255 和 CPU 的 A1A0 是交叉反接的, 正常情况下 8255 A1A0=00A 口, =01B 口, =10C 口, =11 控制口)。 (4 分)

	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>7</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>	
8255PA	*	*	*	*	*	*	*	1	0	1	0	1	0	0	0	0	150H
8255PB	*	*	*	*	*	*	*	1	0	1	0	1	0	0	1	0	152H
8255PC	*	*	*	*	*	*	*	1	0	1	0	1	0	0	0	1	151H
8255PZ	*	*	*	*	*	*	*	1	0	1	0	1	0	0	1	1	153H

**150H, 152H, 151H, 153H (各 1 分)**

# 例题

(4)写出数据段中定义变量的伪指令片段，BUF 存有字符串“HELLO”，带有回车（ASCII 码为 0DH）、换行（ASCII 码为 0AH）字符结束。字节变量 NUM 存有字符串长度。（3分）

**BUF DB 'HELLO',0DH,0AH ; 2分**

**NUM DB \$-BUF; 或 DB 7, DB NUM-BUF ; 1分**

# 例题

(5)编写实现打印寄存器 AL 中字符的子程序（过程）PRTCHAR，已知 8255A口地址 PA8255， C 口地址 PC8255，控制口地址 P8255C（8 分）。

； AL=待打印字符

PRTCHAR PROC ;或 PRTCHAR:， PROC/ENDP 1 分

PUSH AX ; POP 保护 1 分

PRT1: MOV DX, PC8255

IN AL, DX ; 1 分

TEST AL, 01

JNZ PRT1 ; Busy ; 1 分

MOV DX, P8255C

MOV AL, 0\*\*\*1110B ;例\*\*\*=0，位控 PC7=0

OUT DX, AL ; 1 分

POP AX

MOV DX, PA8255

OUT DX, AL ; 字符 1 分

NOP ; 延时 可额外加 1 分（抵扣）

MOV AL, 0\*\*\*1111B ;例\*\*\*=0，位控 PC7=1

MOV DX, P8255C

OUT DX, AL ; 1 分

RET ;1 分

PRTCHAR ENDP

# 例题

(6)根据图 1 所示接口电路和（4）中定义的变量，利用 PRTCHAR 写出顺序打印出一行字符串“HELLO”的汇编指令程序片段。不要求写出 8255 初始化编程。（6 分）

```
MOV CL, NUM          ;1 分; MOV CH,0  LOOP 也可
MOV SI, offset BUF    ;1
LOP: MOV AL, [SI]      ;1
CALL PRTCHAR          ;1
INC SI                ;1
DEC CL                ; loop
JNZ LOP               ;1
```



# 串行I/O接口 8250/8251

基本概念\*：同步/异步通信，单工/双工，调制解调，RS232电平标准，异步通信数据格式：起始位(1)，数据位(5-8，先低后高)，奇偶校验\*1，停止位(1-2)，==> 每个字符至少10位

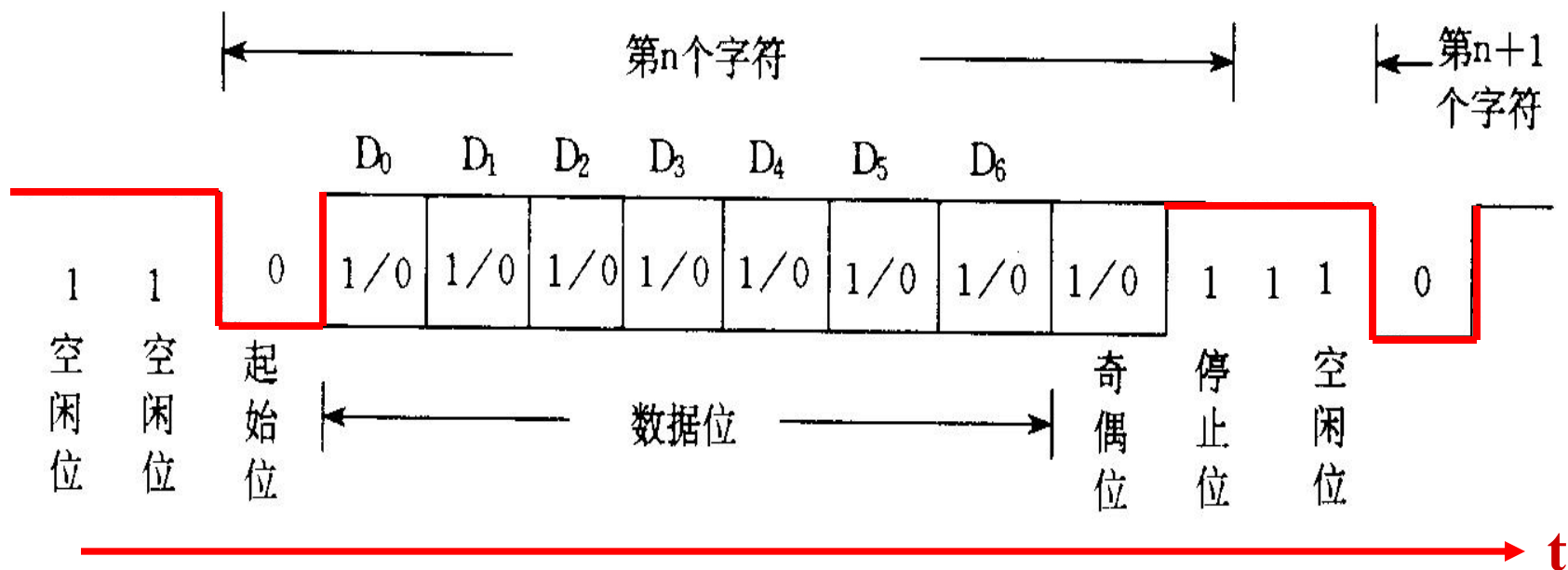
- 波特率：每秒钟传送的二进制脉冲的数目，即1波特=bit/s
- 字符速率：每秒所传输的字符数。

# 例题

01年考题:

13.若系统采用异步串行数据传送方式, 每个字符传送格式为数据位8位, 奇偶校验位1位, 停止位2位, 如其波特率为9600BPS, 则每秒钟最多能传输的字符数是\_\_\_\_\_。

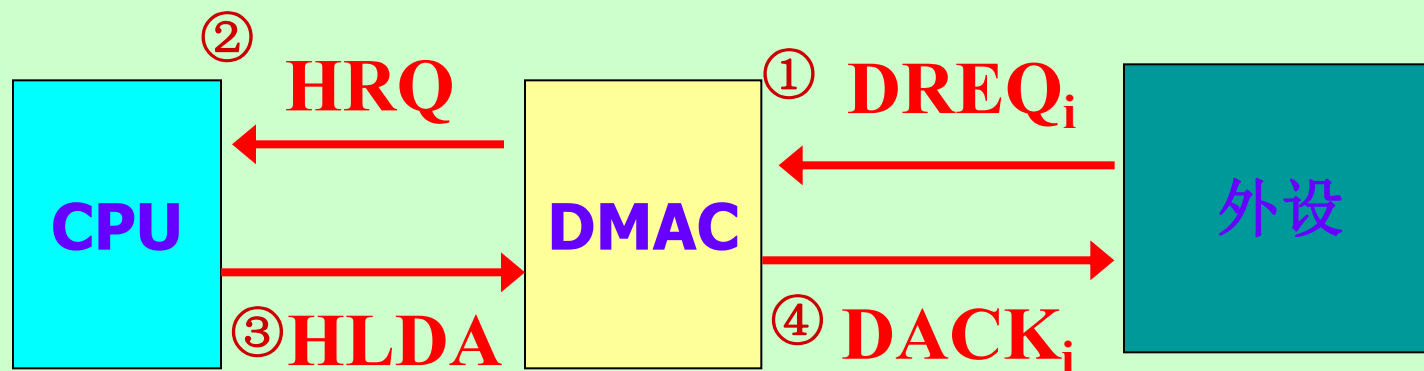
A. 9600    B. 1200    C. 872    D. 800    E. 738



串行异步通信格式 (数据)

# DMAC 直接存储器访问 8237

- ◆基本概念、DMAC的作用（原理）：**I/O外设与存储器之间高速数据交换**；
- ◆外设或存储器通过AEN信号区分MPU/DMAC之一控制总线；



$i=0,1,2,3$  4个16位通道，每个通道可独立响应外部DMA请求

# DMAC 直接存储器访问 8237

## 02年考题:

10. 在8086/8088系统中, 不需要CPU控制, 也不经过CPU的数据输入输出方式是\_\_。

A. 无条件传送   B. 查询传送   C. DMA传送   D. 中断传送

## 11年B卷:

15. 单i8237DMA控制器可支持4个通道的DMA, 若采用DMA控制器两级级连方式, 则8237最多可以实现\_\_通道的DMA。

(A)4      (B)8      (C) 12      (D)16      (E)64

# 复习课（4-6章）

第4部分：半导体存储器

第5部分：数字量输入输出

总线、接口概念

\*接口电路（芯片）、端口地址

\* 数据传送方式

\* 中断电路及其处理

\*定时/计数器电路与应用

\* 并行接口电路与应用

\* 串行接口电路与应用

DMA电路与应用

第6部分：模拟量输入输出

可能级联：  
**8253**两个通道，  
**DMA 8237**，  
中断管理器**8259**

# 模拟量接口A/D---D/A

基本概念\*:

- ◆多路转换开关MUX和采样保持器S/H的主要作用;
- ◆ADC—双积分、逐次逼近、V/F变换型
- ◆ADC0809: 8位8通道逐次逼近型模数转换器;
- ◆AD574A: 12位单通道;
- ◆精度: 相对满量程而言, 分辨率: 对信号灵敏度, 与位数相关;
- ◆Vref对转换精度的影响;

$$V_o = -V_{ref} \frac{R_f}{R} \frac{D}{256}$$

- ◆模拟地-数字地: 相同基准, 单点相连

掌握: 接口编程——ADC/DAC的转换控制

01年考题:

16. ADC0809是8位逐次比较式模数转换器，在接口系统中设计Vref(REF+)选用+4.00V电压基准源(REF-接模拟地)，如果读取的转换数字量=40H，则此时输入信号电压为\_\_\_\_\_。

- ☐ A 4.00V
- ☐ B 3.00V
- ☐ C 2.00V
- ☒ D 1.00V

提交

11年B卷:

14. 模拟量输入输出通道中多路转换开关(Multiplexer)的作用是\_\_\_\_\_。

- ☐ A 配合逐次比较式ADC
- ☒ B 从多路模拟信号中选择一路输出
- ☐ C 从多路数字信号中选择一路输出
- ☐ D 实现输入信号的程控放大调理

提交



# 模拟量接口 A/D---D/A

**接口电路简单互连：** 接口电路与简单I/O类似

(1) 数据线：8位，D7—D0 通常同名互连

(2) 经译码电路产生片选信号，其中若干连至芯片  
/CS, /CE **74LS138**

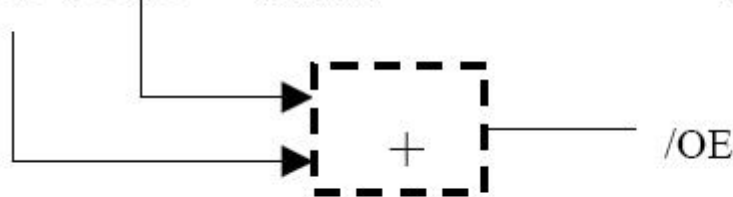
(3) 控制信号线：

/RD, /IORD 接/OE(输出允许)

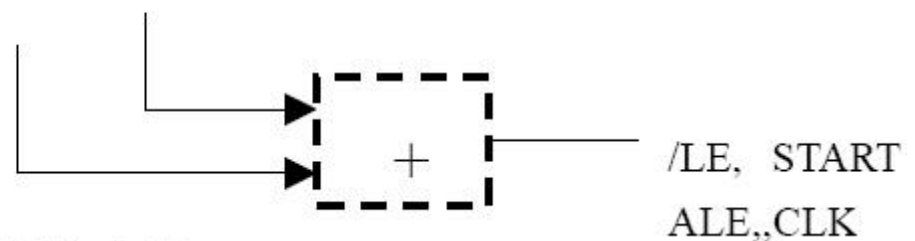
/WR, /IOWR 接/WE(写入允许,/LE,START,ALE等)

/CS+/IORD---读操作

/CS+/IOWR---写操作（启动，控制）



无独立片选芯片



# 模拟量接口A/D---D/A

根据指定端口地址，利用I/O指令接口应用程序片断编程：

◆ **DA**：先送出8位/12位数据，再启动DAC

例：

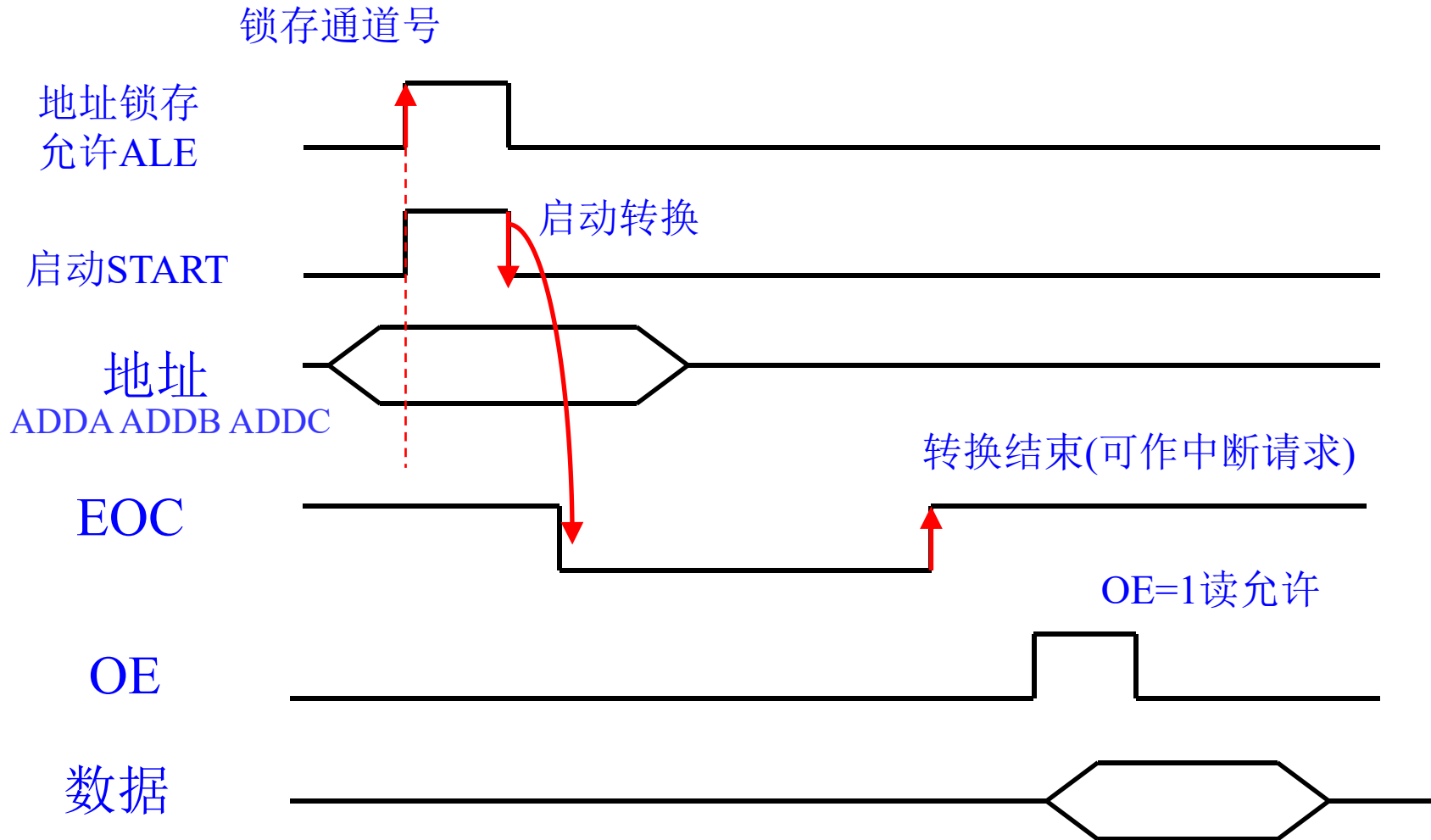
◆ 设CS的端口地址为320H，XFER的端口地址为321H。  
编写数据通过DAC0832进行D/A转换输出的程序段。  
(双缓冲)

MOV DX, 320H	； 输入寄存器
MOV AL, DATA	； DATA为被转换的数据
OUT DX, AL	； 数据送入输入寄存器
INC DX	； 指向DAC寄存器
OUT DX, AL	； 选通DAC寄存器，启动 D/A转换

↖  
值任意

# 模拟量接口 A/D---D/A

◆ AD: 先启动转换START/ALE, 延时或判断EOC再读取结果



# 模拟量接口A/D---D/A

提示:

◆ AD/DA程序片断运行的方式（除初始化芯片和单元外），一般为定时执行或中断执行；

例如：定时触发DAC

```
MOV DX, OUTPORT  
OUT DX, AL
```

◆ 如需要记忆以前信息（用于比较等），一般存放在内存单元中而非寄存器中。

例如：按设计要求处理后保存到变量单元或输出

```
MOV DX, INPORT  
IN AL, DX  
MOV V1, AL
```

# 例题

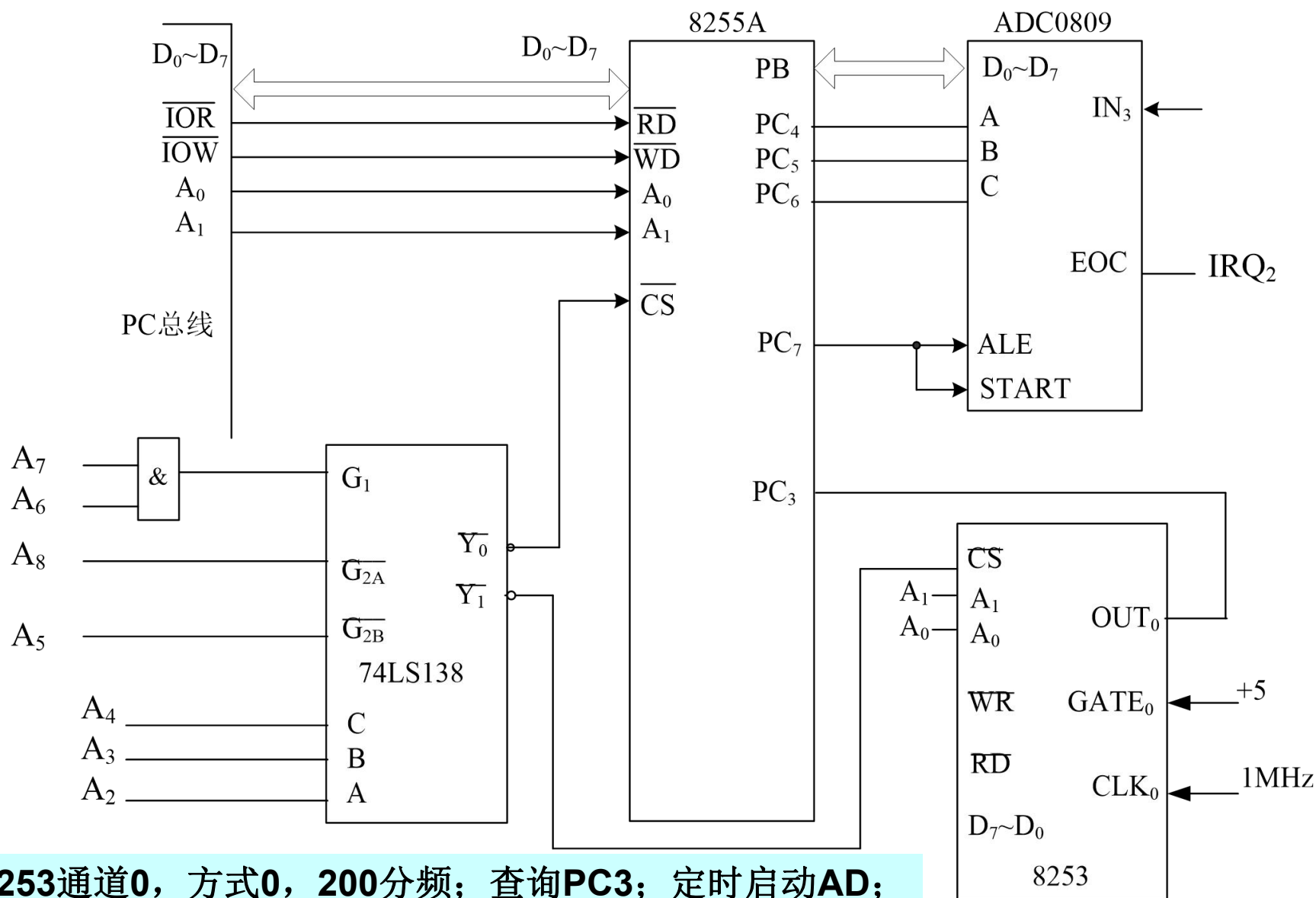
课本P344：图6.27， ADC0809+i8255

## 习题6.4：综合设计型

ADC0809设计一个数据采集系统，采用中断方式，EOC接至8259A的IRQ<sub>2</sub>，每隔200μs采样一个数据。试完成：

- (1) 硬件设计，画出连线图（不包括8259A）；
- (2) 软件设计，包括8255A、8253的初始化及中断服务程序。

考试不会要求任何芯片的初始化编程！



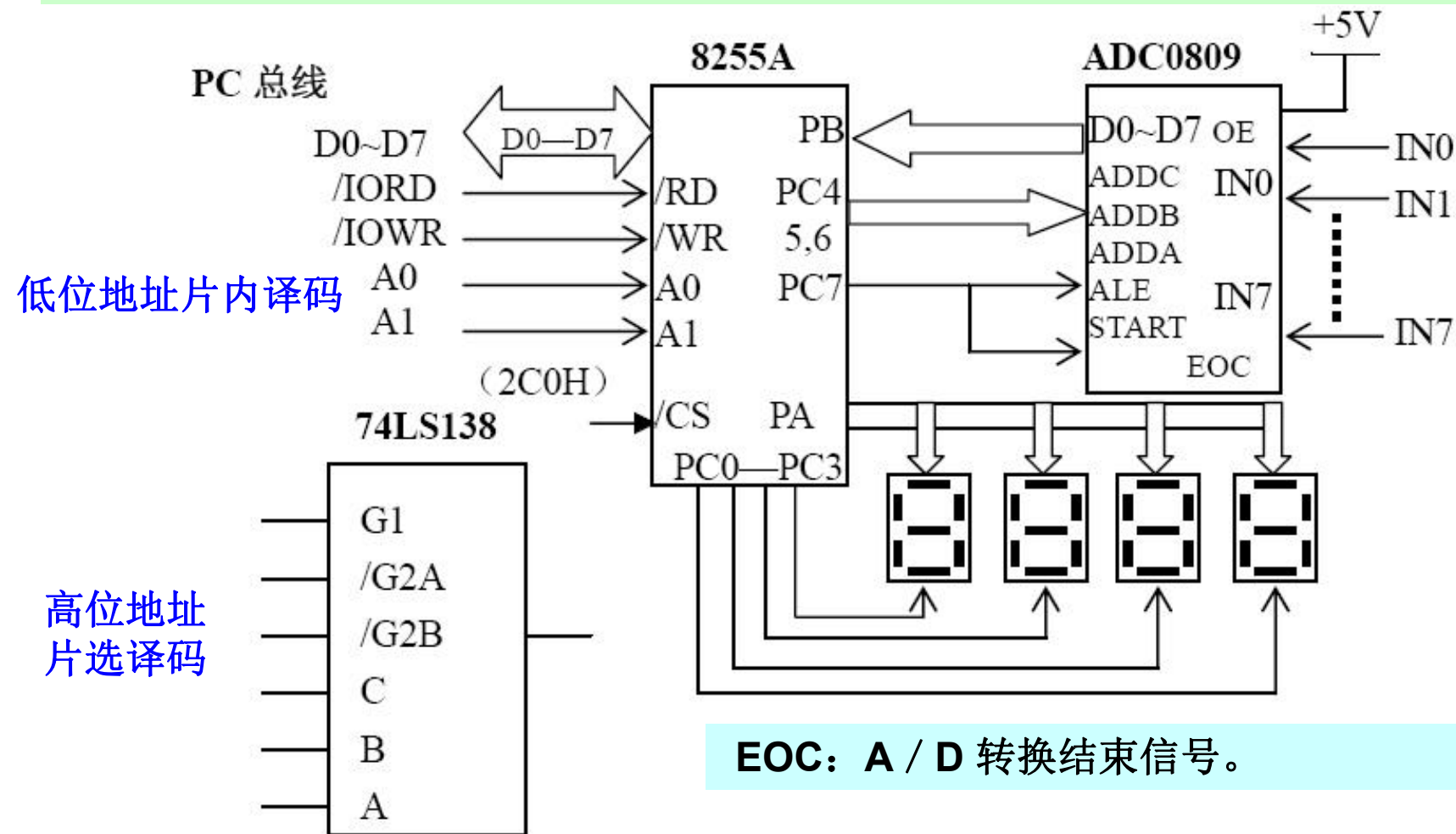
**8253通道0，方式0，200分频；查询PC3；定时启动AD；  
中断服务程序读AD结果**

**A15 A14 A13 A12 A11 A10 A9 A8 A7 A6 A5 A4 A3 A2 A1 A0**

**0 1 1 0 0 0 0 0 0 0 0C0H选中8255A**

# 习题 08年试题

8086/8088 系统采用ADC0809 和8255 接口芯片采集8 个输入通道的信息，并转换成对应的电压数字量，输出到4 个LED 显示器显示（其中：1 个LED 显示通道数0-7，3 个LED 显示相应通道的电压值）。

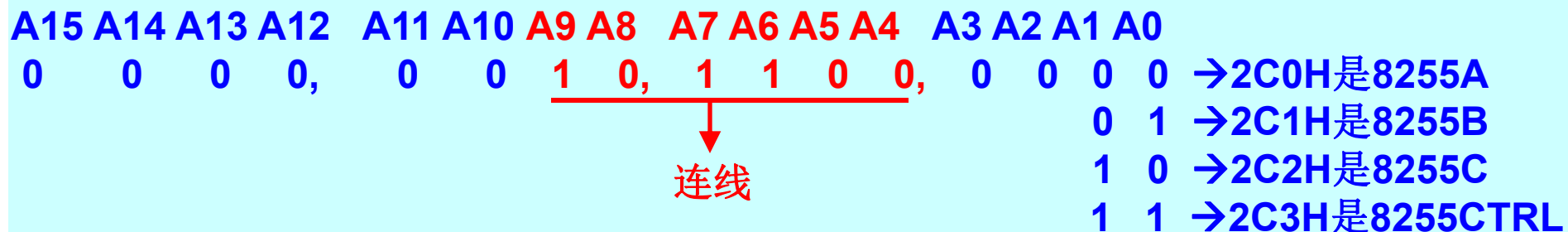


# 习题 08年试题(35分)

(1)说明8255A 的PA、PB 和PC 口的工作方式及各口的作用；  
答：

- PA 口工作于方式0 输出，给4 个数码管提供段码；
- PB 口工作于方式0 输入，读取ADC0809 的转换结果；
- PC 口工作于方式0 输出，PC0—PC3 给4 个数码管提供位码，PC4—PC6给ADC0809 提供地址信号，PC7 给ADC0809 提供地址锁存信号ALE和启动转换信号START。

(2)设计8255 端口地址2C0H, 用A9-A4 译码，译码器用74LS138，写出设计过程并画出连线图；





## (3)简述CPU 控制ADC 工作原理。

答：PC4—PC6 给ADC0809 提供地址信号，确定IN0——IN7 中的某通道；PC7 给ADC0809 提供启动转换信号，启动ADC，同时锁存通道地址；通过PB 口读取0809 的ADC 结果。

# (续)

(4)根据图中连线,编写选择IN4 通道进行转换的程序片断,并将转换结果存入单元VIN4 中.(不考虑PC 口对显示部分的影响)

**P8255C EQU 2C2H**

**MOV DX, P8255C ;C口地址2C2H**

**MOV AL, 00010000B ;选择IN4, PC6 5 4=001B, (设PC4接ADDC)**

**OUT DX, AL**

**MOV AL, 10010000B**

**OUT DX, AL**

**MOV AL, 00010000B ; ALE/START 正脉冲启动ADC, 同时锁存通道**

**OUT DX, AL**

**;也可以用位控字方式产生通道选择锁存信号, 写入控制端口2C3H**

**CALL delay ;延迟读结果**

**MOV DX, P8255B ;B口地址2C1H**

**IN AL, DX**

**MOV VIN4, AL**

# 习题 08年试题(35分)

## (5)简述CPU 控制数码管显示工作原理

答：某通道的ADC 结果经CPU 换算后，成为3 位八段码，连同表示该通道位一起，共4 位八段码，分别在4 个数码管上显示。具体：PA 口给4 个数码管循环提供段码，同时PC0—PC3 给4 个数码管循环提供位码，循环显示4 位BCD 码，使人眼感觉4 个数码管同时显示。

# 习题 08年试题(35分)

(6)设计指定通道(0-7)的转换-显示子程序DISPADCH 的结构(功能和参数传递关系),(不编具体程序),编写周期性显示各通道和电压值的主程序片断,其中可直接调用上述DISPADCH 以及延时程序。

```
AGAIN: MOV CX, 8
        MOV AH, 0    ; 任何寄存器或单元
LP1:    PUSH AX       ; 以AH为入参
        CALL DISPADCH
        POP AX
        INC AH        ;
        LOOP LP1      ;
        JMP AGAIN     ; 可以不压栈
```

# 习题 08年试题(35分)

(7)如果CPU 采用中断方式读取ADC 转换结果，如何实现？

答：中断方式读取ADC 结果：将ADC0809 的EOC 信号连在DMAC8259的某一个输入引脚，例如IR2，作为中断请求信号，申请CPU 中断。若只读一次：在主循环中启动START/ALE，在CPU 中断服务程序中，读取ADC 结果。

# 总结——课程目标回顾

- 本课程为自动化专业必修的一门主要的专业基础课，也是电子信息与电气大类专业校级平台课程。课程有较强的工程实践性，教学目的是使学生通过本课程的学习，在已学数字逻辑、电子线路和程序设计的基础上，**深入掌握微处理器和基本接口的结构与工作原理，学会用汇编语言编程、调试和硬件控制基本方法，正确建立微型计算机系统的整体概念，具备计算机硬件应用分析问题和解决问题的初步能力**，以便为今后嵌入式系统、单片机、DSP原理、检测、控制等后续课程学习以及计算机软硬件应用系统设计和科研打下必要的计算机原理与接口技术基础。

考试时间第15周周五  
考前答疑时间地点待定

祝大家考出好成绩！