

《微机原理与接口》

第8章 并行接口和8255

教师：苏曙光

华中科技大学软件学院

● 教学内容

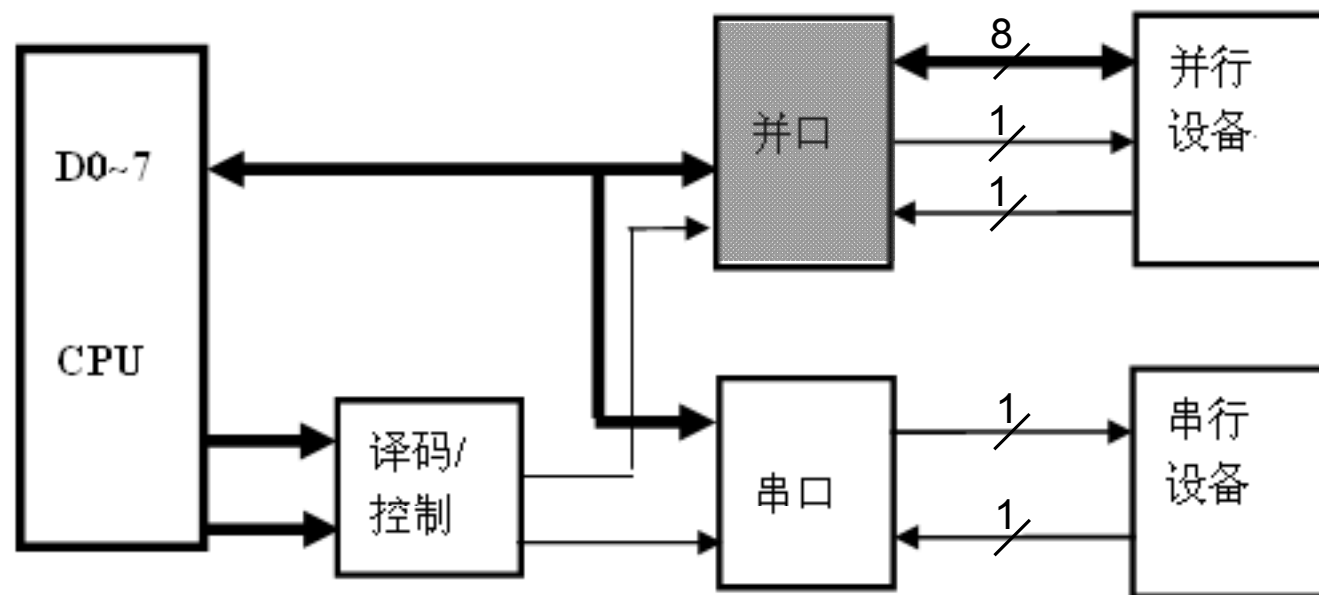
- 第1节 并行接口的概念
- 第2节 8255A的外部引脚和内部结构
- 第3节 8255A的控制字
- 第4节 8255A的工作方式



第1节 并行接口的概念

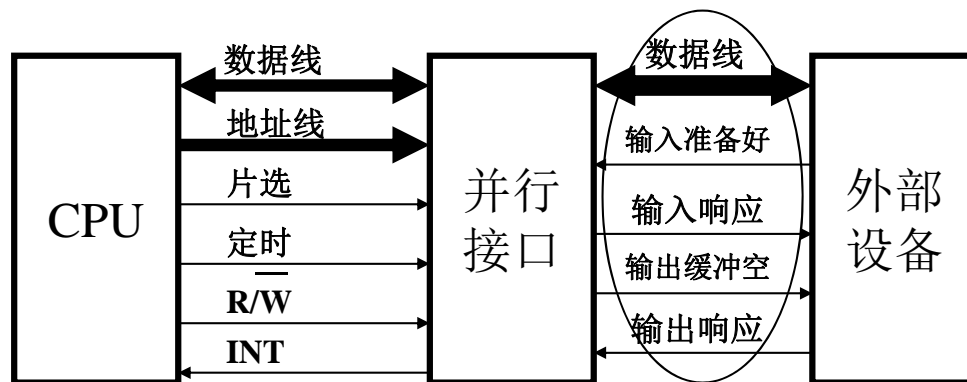
并行接口的特点

- 并口的位置

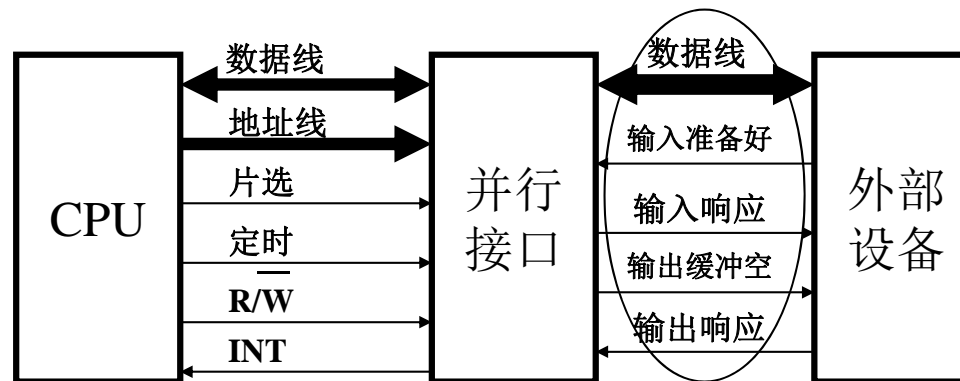
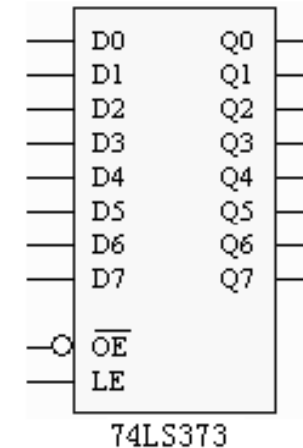


并行接口的特点

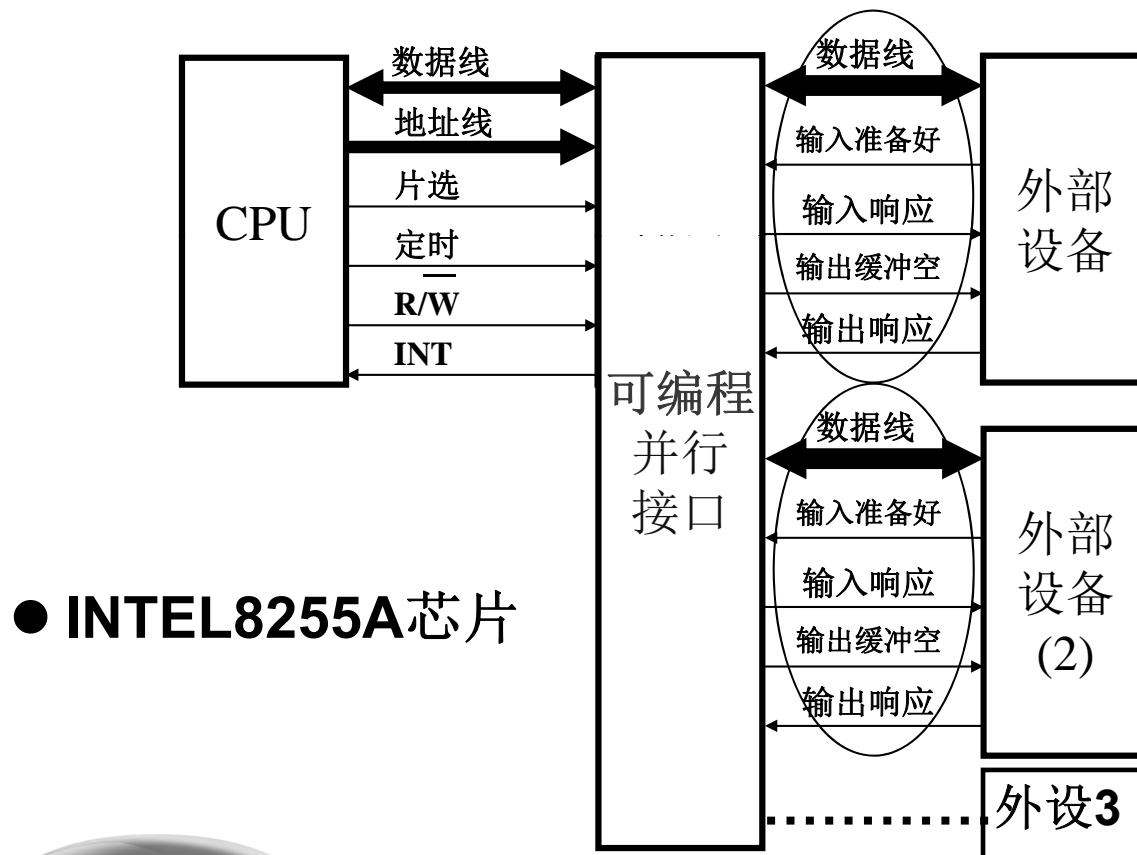
- 1) 数据线有多根，所有数据线一起传输。
 - 典型数据线的根数：8根（字节）或16根（字），也可自定
 - 对比串行接口：通过1根线传送信息
 - 8位或16位一起传输（即使只用到其中1位）。
 - 并行：特指接口与I/O设备之间。
- 2) 并行接口组成：数据线 + 至少两根握手信号线



- 3) 数据格式不受限制
- 4) 分类：硬连接并口和可编程并口。
 - 硬连接并口：
 - ◆ 工作方式及引脚功能固定(例373,244)
 - 可编程并口
 - ◆ 工作方式及引脚功能可编程设定 (例8255A)



可编程并口作用：扩充数据总线的数量和适应性





第2节 8255A的外部引脚和内部结构

8255A的基本特点

- 功能

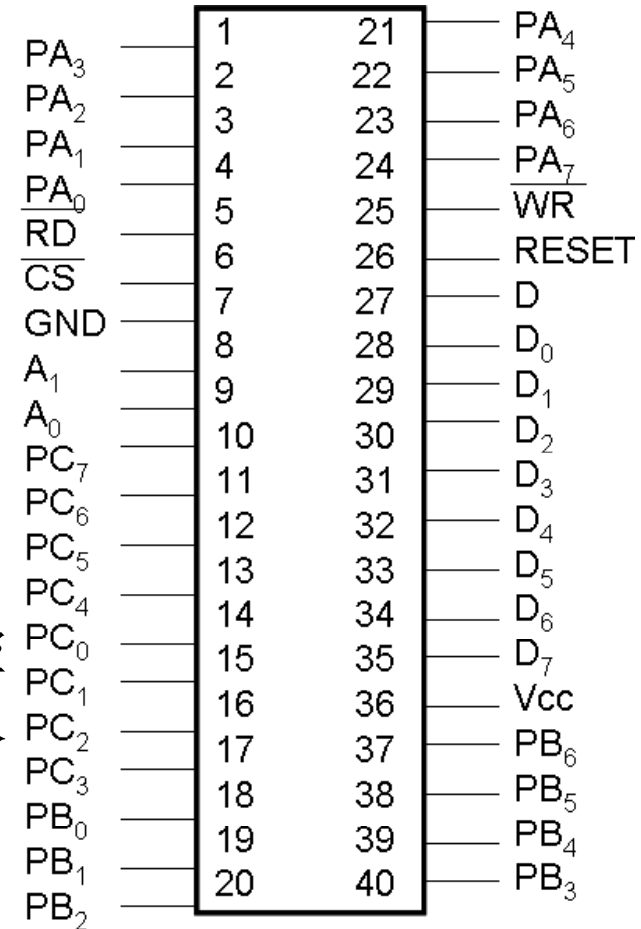
- 扩充数据总线的数量（1→3）： PA,PB,PC
- 适应输入/输出，查询，中断等多种应用场合
- 具有3个输入/输出并口（PA,PB,PC）

- 特点

- 可选择某一特定并口与外设连接；
- 每个并口可以指定输入/输出的方向；
- 与CPU之间的数据传送方式可选择；
- 有4个8位端口可以存取：
 - ◆ PA端口， PB端口， PC端口， 控制端口

8255的外部引脚（面向CPU侧）

- 数据总线：D0~D7
 - 命令、数据、状态等数据。
- 地址总线：A1, A0（CS）
 - 4个端口：PA, PB, PC, 控制口
- 控制总线：RD, WR
 - 读，写信号
- 其他总线：RESET, GND, VCC
 - RESET，高电平有效。清除控制寄存器并将三个端口置输入方式并且屏蔽中断请求。



8255的外部引脚（面向外设侧）

- 三组并口线

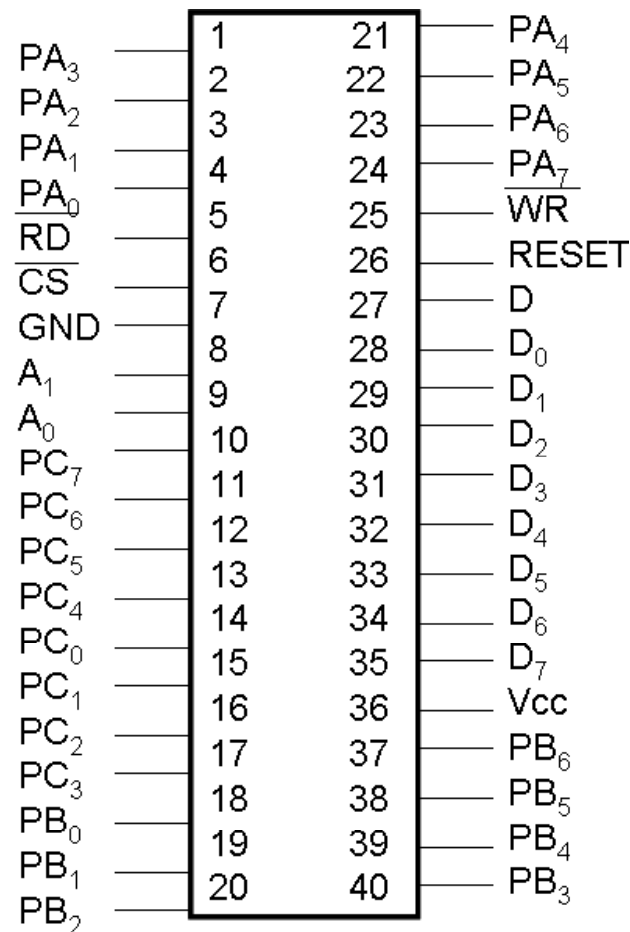
- PA0~PA7：对应PA端口

- PB0~PB7：对应PB端口

- PC0~PC7：对应PC端口

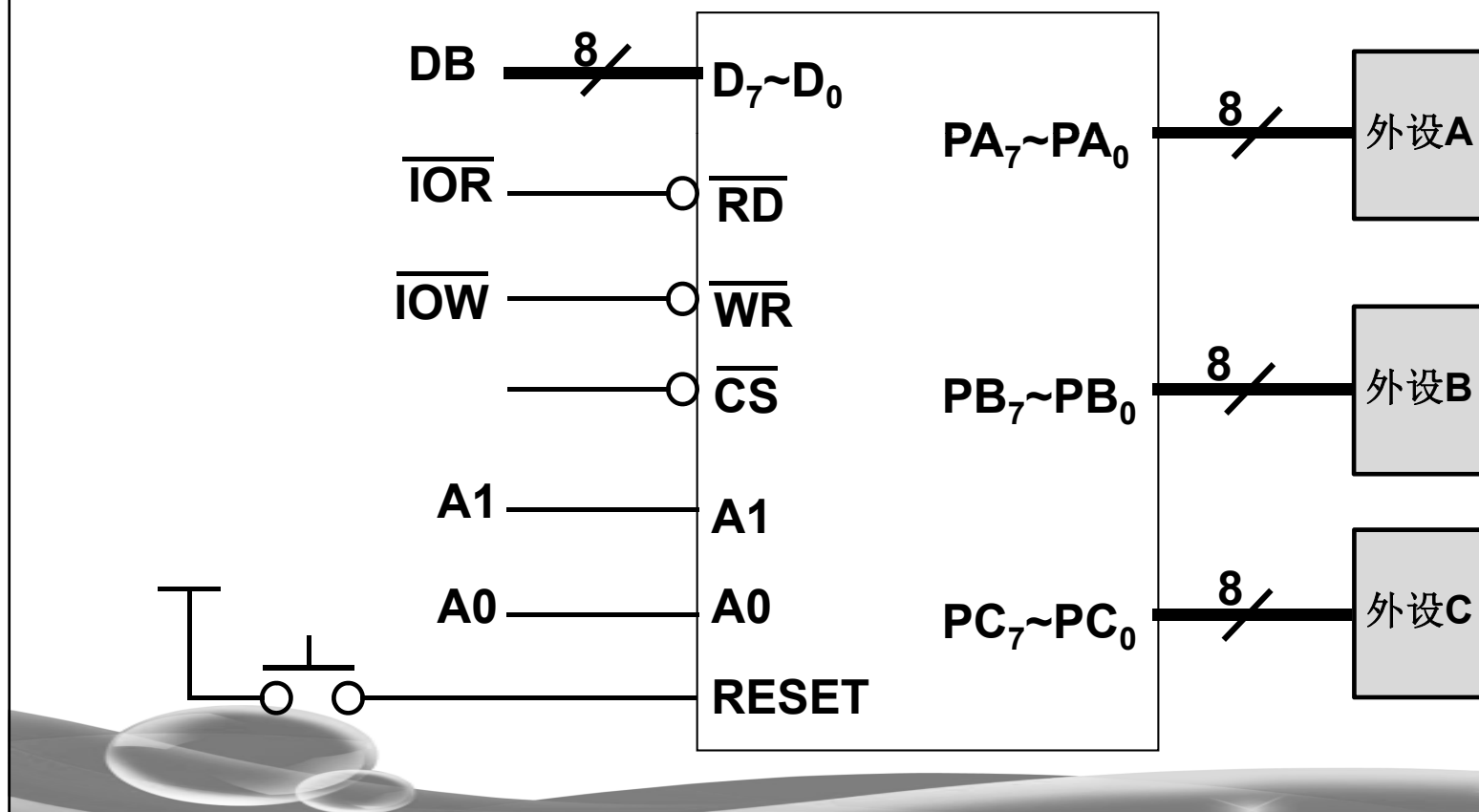
- 说明

- PA，PB与PC的作用和方式可以编程：既可以做8位并行数据线，也可以按位控制作控制线来用。



8255A与PC微机的典型连接方式

- (DB被扩充: 1→3)



● 8255A的操作与端口

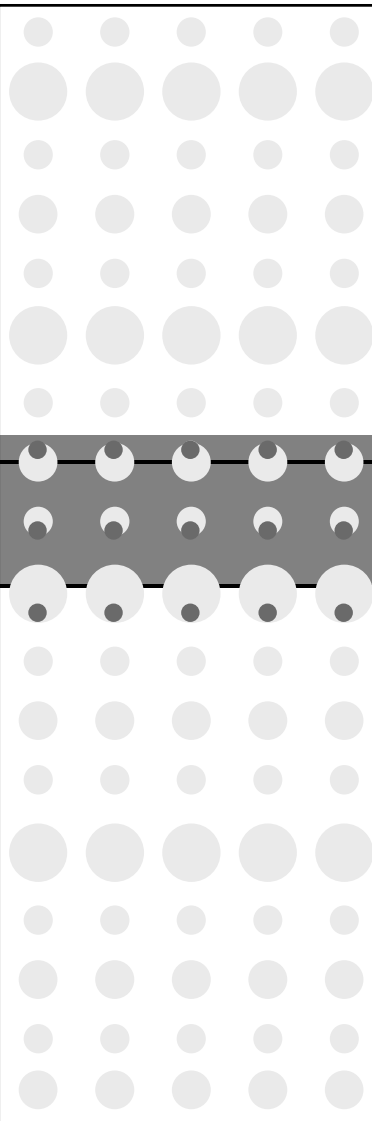
■ 4个连续端口：PA,PB,PC,控制口

$\overline{\text{CS}}$	$\overline{\text{RD}}$	$\overline{\text{WR}}$	A_1	A_0	操作
0	1	0	0	0	写端口A
0	1	0	0	1	写端口B
0	1	0	1	0	写端口C
0	1	0	1	1	写控制口
0	0	1	0	0	读端口A
0	0	1	0	1	读端口B
0	0	1	1	0	读端口C
0	0	1	1	1	无操作

● 分组结构

- 分组结构

 - PC_{4~7}与PA: A组
 - PC_{0~3}与PB: B组



第3节 8255A的控制字

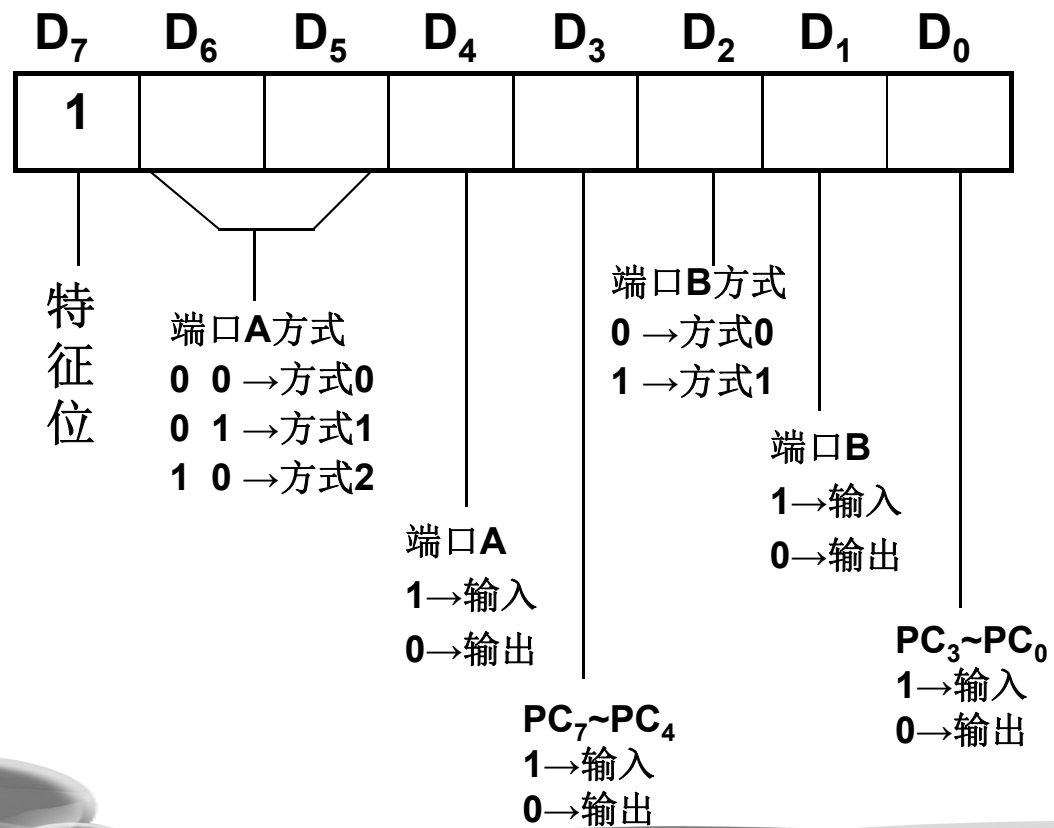
● 8255A的常用控制字

- 方式选择控制字：设置工作方式
- 按位置位/复位字：按位设定PC某位的值。
- 注意：控制字必须写在控制端口

● 8255的3种工作方式

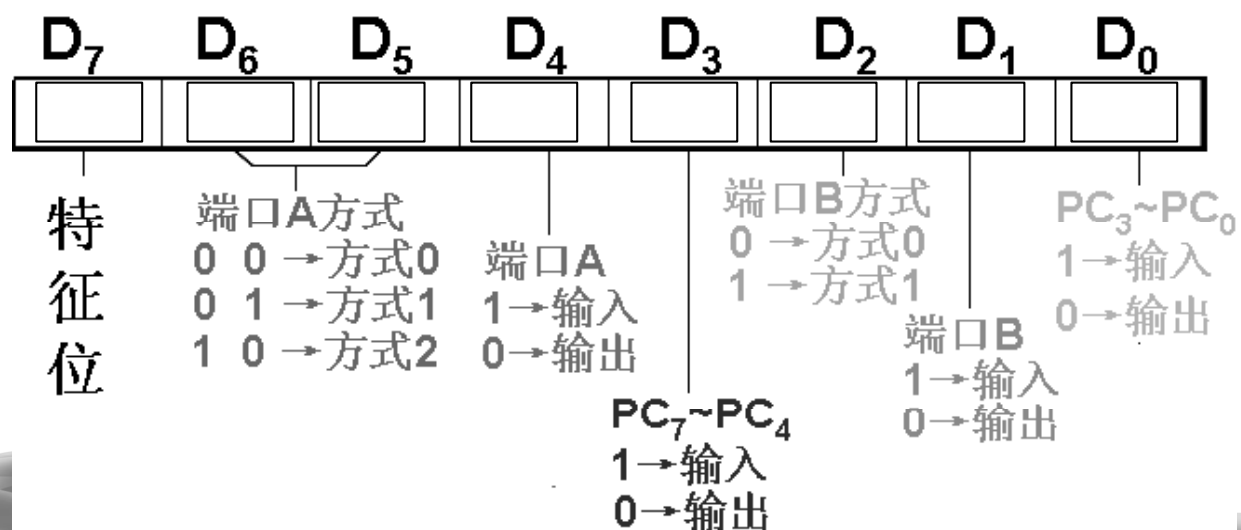
- 方式0(Mode 0) ——基本输入输出
- 方式1(Mode 1) ——选通输入输出
- 方式2(Mode 2) ——双向传送
- 三个并口（端口）可以独自设置不同的工作方式。
 - ◆ PA: 方式0、方式1、方式2;
 - ◆ PB: 方式0、方式1;
 - ◆ PC上部分: 方式随PA。
 - ◆ PC下部分: 方式随PB。
- 工作方式由方式选择控制字设定。

● 方式选择控制字



- 例子：初始化8255：A口方式1，输入；C口上部输出；B口方式0，输出；C口下部输入。8255地址：300H~303H

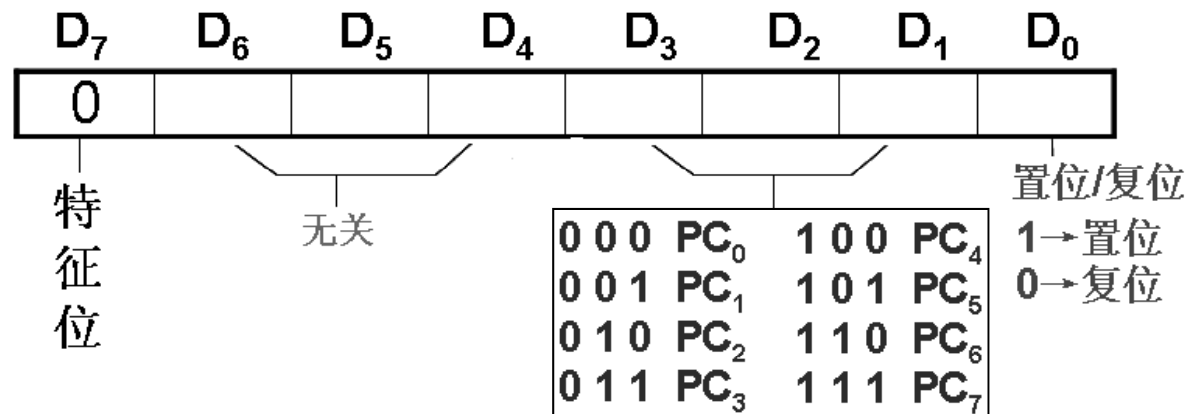
```
MOV DX, 303H    ; 8255A命令口地址
MOV AL, B1H     ; 初始化命令
OUT DX, AL      ; 命令口
```



● 按位置位/复位控制字

■ 作用：按位设定PC某一位电平的高或低。

■ 格式：8位：最高位固定为0。



■ 例：PC₃置位：00000111B

■ 例：PC₃复位：00000110B

- 例：把PC₂置成高电平。命令字：**00000101B或05H**

MOV DX, 303H ;8255A命令口地址

MOV AL, **05H** ;使PC₂=1的命令字

OUT DX, AL ;送到命令口

；若置PC₂低电平： 00000100B或**04H**

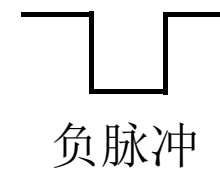
区别如下置PC₂ 高电平？

MOV DX, 302H ; PC口的地址

MOV AL, 00000100 ; PC₂为高电平

OUT DX, AL

- 例子：利用**PC₇**产生负脉冲作打印机接口数据选通信号。
;假定PC7初始缺省为高电平

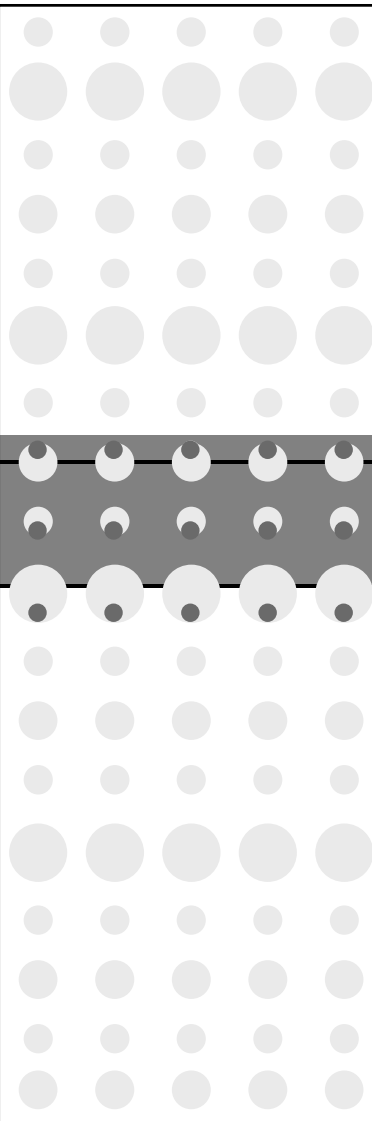


- 例：驱动喇叭产生不同声音【方波，频率， PC_6 】



● 按位置位/复位字与方式控制字的使用

- 1、该字须跟在方式选择控制字之后写入控制端口。
- 2、按位置位/复位写入命令口，不要写入PC口
- 3、按位置位/复位不影响工作方式
- 4、D7位区分两个命令



第4节 8255的工作方式

● 0方式（基本I/O方式）

- 指无条件传送或查询传送。

- ◆ 无条件传送：不用联络信号（状态信息）。

- ◆ 查询传送：但是联络信号可由任意空闲引脚充当。

- PA, PB和两个PC4位都可以设定为0方式

- 可以由用户指定输入或者输出方向。

- 单向输入或输出

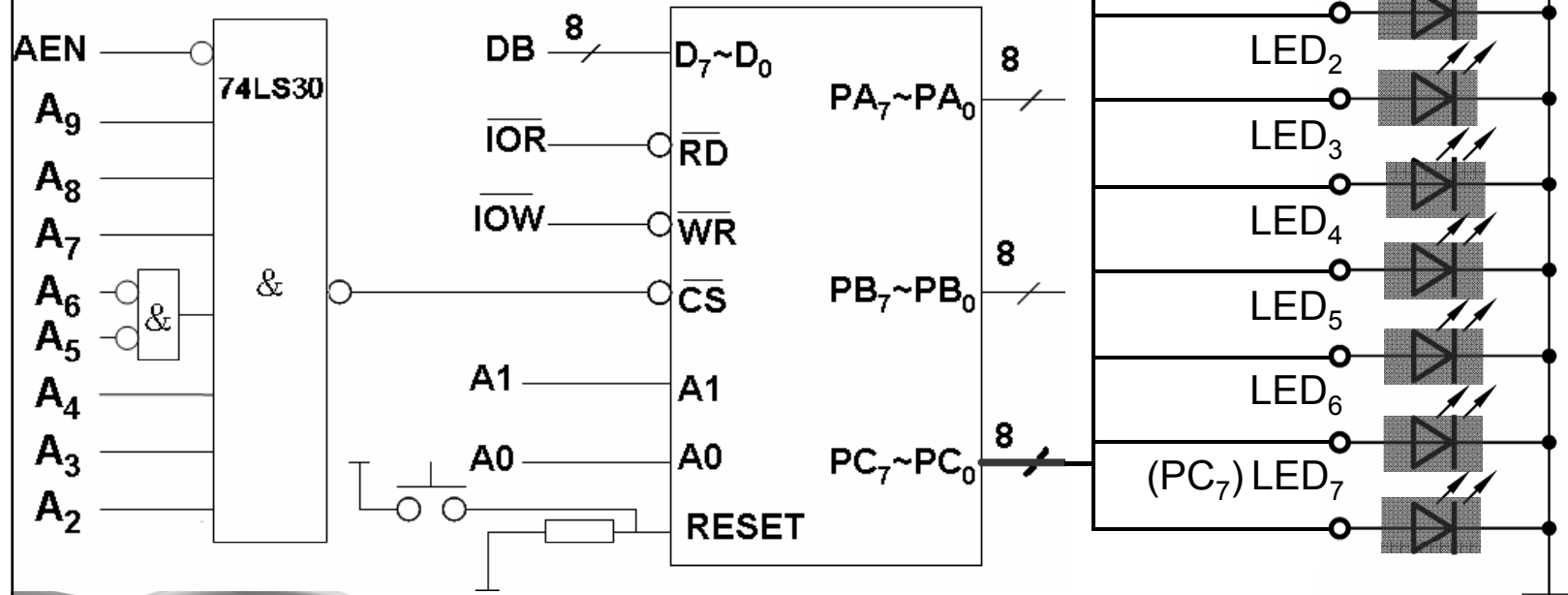
- 输出有锁存能力，输入有缓冲能力而无锁存能力。

思考（1）如何设置方式字？（2）程序流程？
（3）8255的地址=？

● 方式0例子

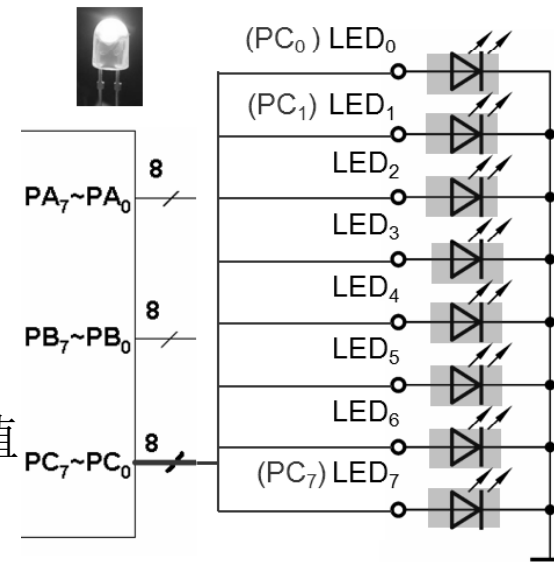
■ 8255PC接8只LED，依次亮灭

◆ $PC_i = 1$ 亮； $PC_i = 0$ 灭



注意：8255A地址: 39CH

```
MOV DX,39FH ;控制端口地址
MOV AL,80H
OUT DX,AL
MOV DX,39EH ;PC端口地址
MOV AL,0000 0001B ;设定PC的值
AGAIN:OUT DX,AL ;点亮LED
        DELEY_1000MS ;延时1秒
        ROL AL,1 ;逻辑左移（LED下行），改变PC值
        JMP AGAIN
```



课堂练习：方式0的应用

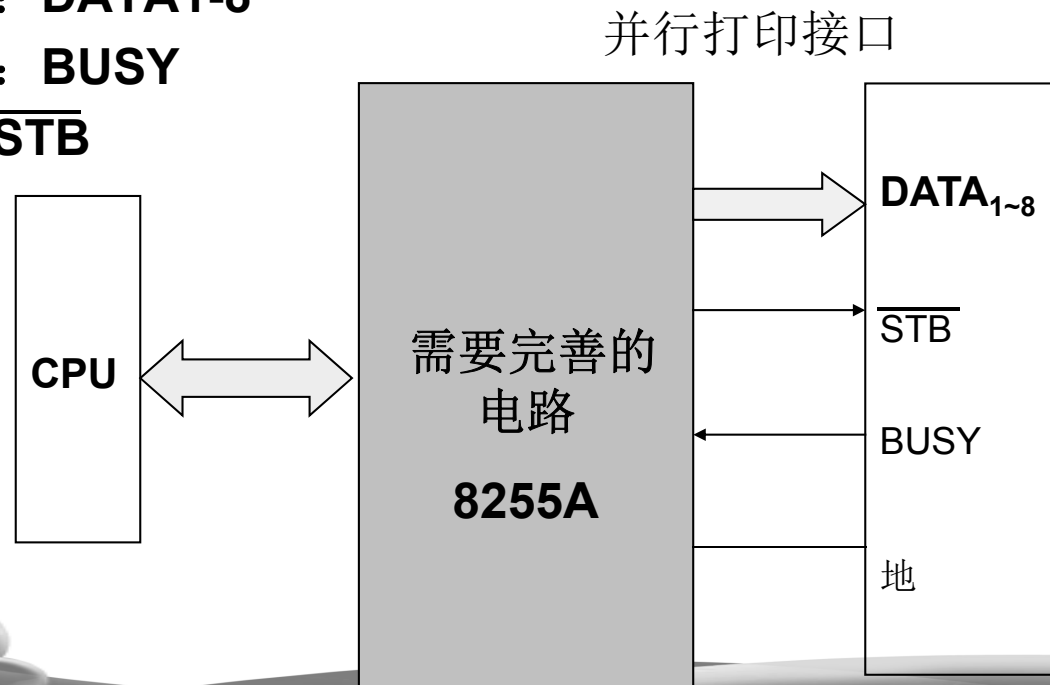
● 例子：并行打印机接口设计（查询方式）

■ 设计一个并行打印接口，CPU采用查询方式把存放在BUF缓冲区的256个字符（ASCII码）送去打印机进行打印。

■ 数据线：DATA1-8

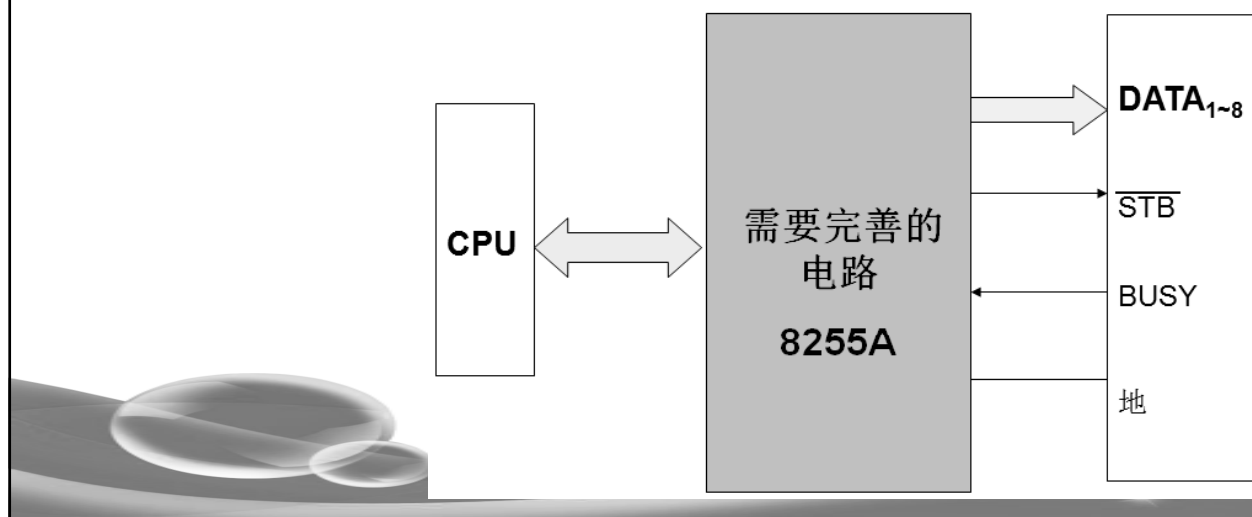
■ 状态忙：BUSY

■ 选通： $\overline{\text{STB}}$



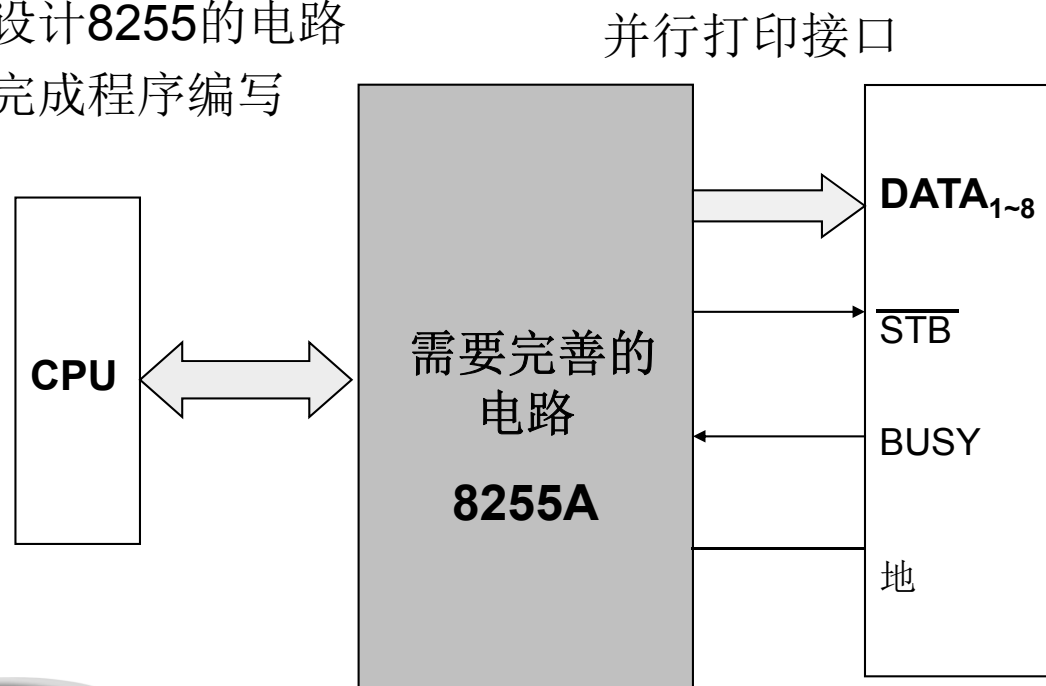
打印机与CPU之间传送数据的过程

- ① **CPU**首先查询**BUSY**。
 - 若**BUSY**=1，表示打印机忙，则等待；
 - 若**BUSY**=0，表示打印机不忙，则送新的打印数据。
- ②通过接口把数据送到**DATA_{1~8}**上（**PA**端口）
- ③ **STROBE**(负脉冲)把数据打入到打印机数据缓冲器。
- ④ 打印机收到数据后，发出“忙”（**BUSY**=1)信号
- ⑤每当一个字符打印完毕，撤消“忙”信号，置**BUSY**=0 。
- 重复上述工作，直到把全部字符打印出来。



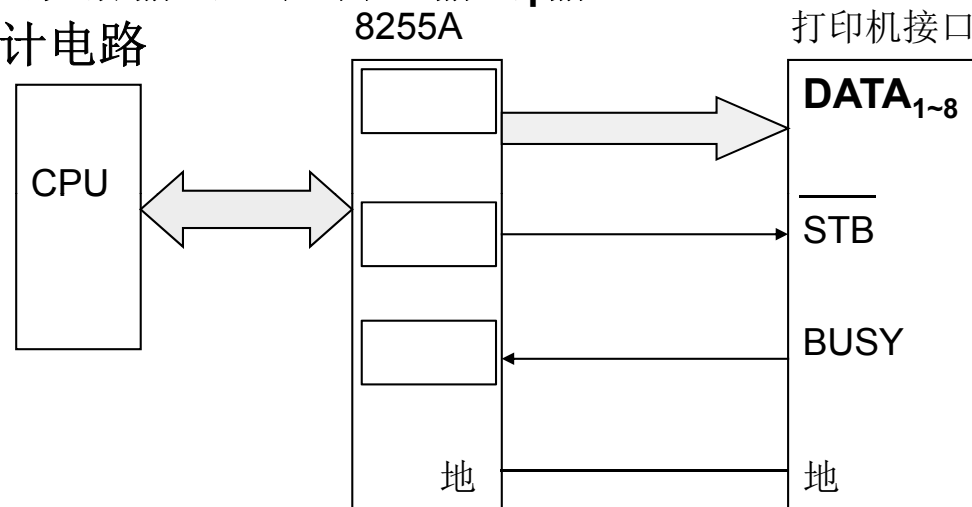
● 需要完成的工作

- 选定8255工作方式
- 设计8255的电路
- 完成程序编写



- 0方式：数据输出，控制：输出|输入

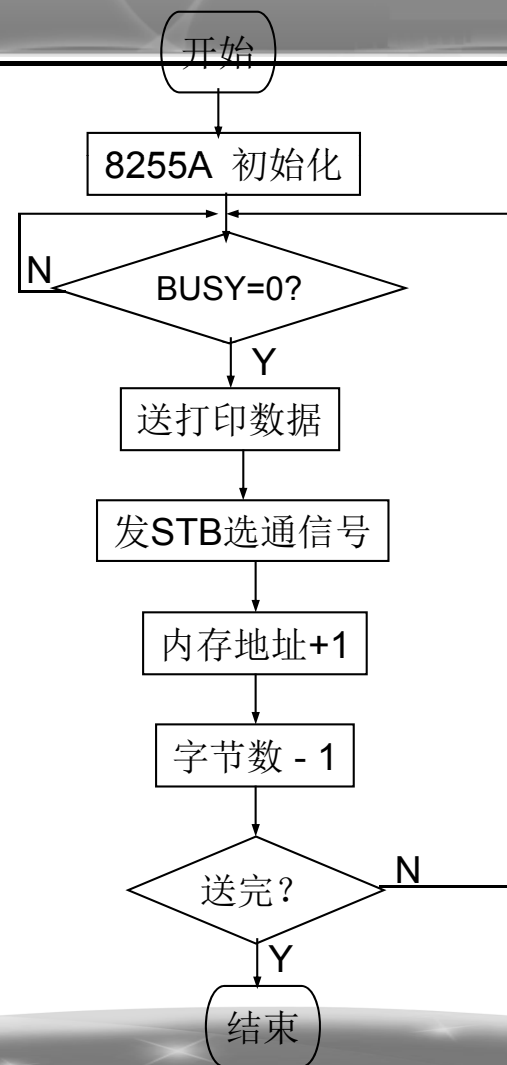
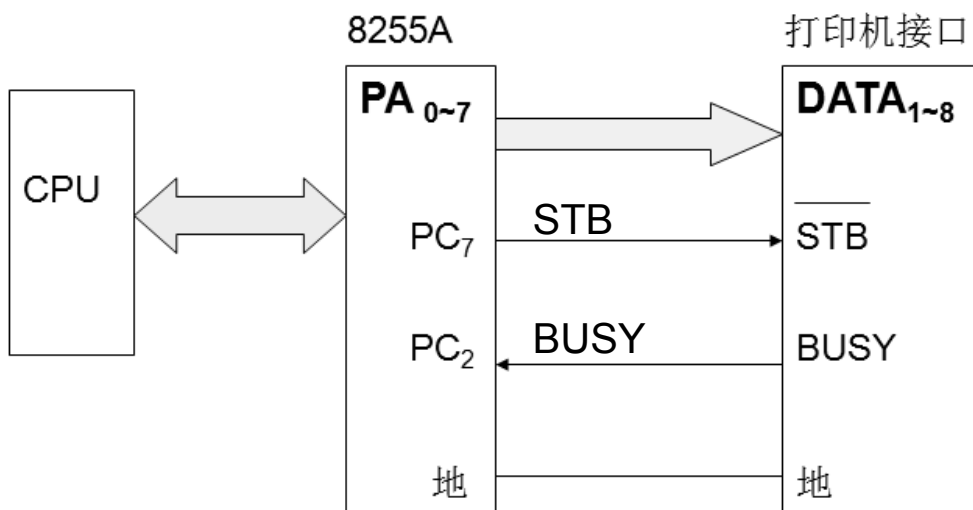
- 硬件设计电路



- PA输出，输出8位打印数据；
- PC₇输出。数据选通信号（STB）；
- PC₂输入。接收忙信号(BUSY)。
- 注意：PC₇和PC₂并非固定作控制用。这是0方式特点。

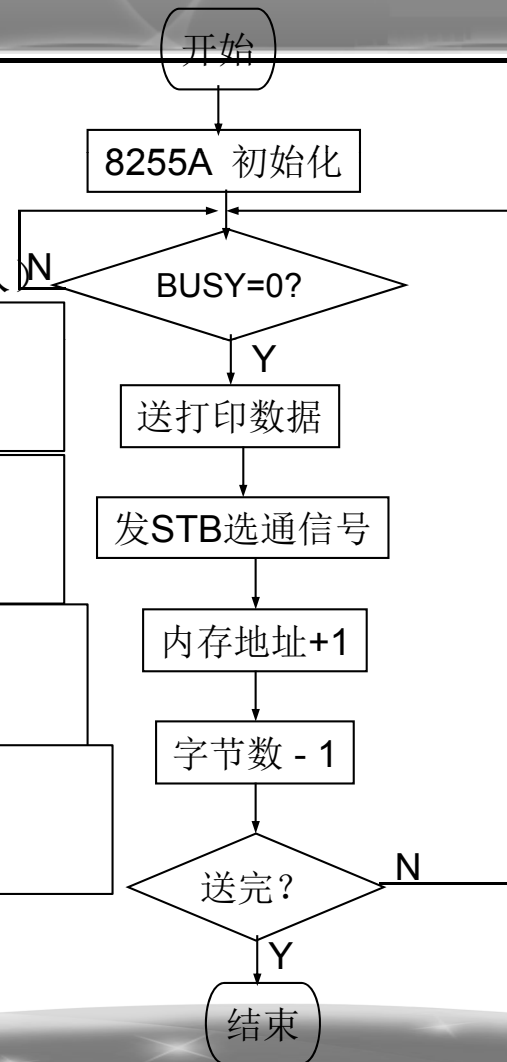
软件实现【8255地址300H ~ 303H】

● 流程



软件实现【8255地址300H ~ 303H】

```
MOV DX,303H      ;8255A命令口
MOV AL,10000001B ;工作方式字
OUT DX,AL ;A口0方式输出，C4~7输出，C0~3输入
```



软件实现(续)

; 不忙, 则向A口送数

MOV DX,300H

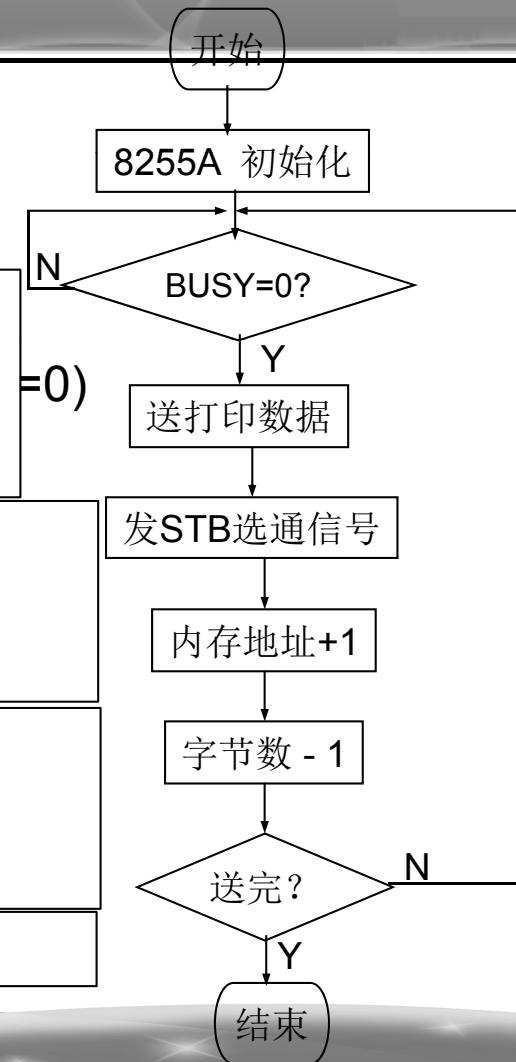
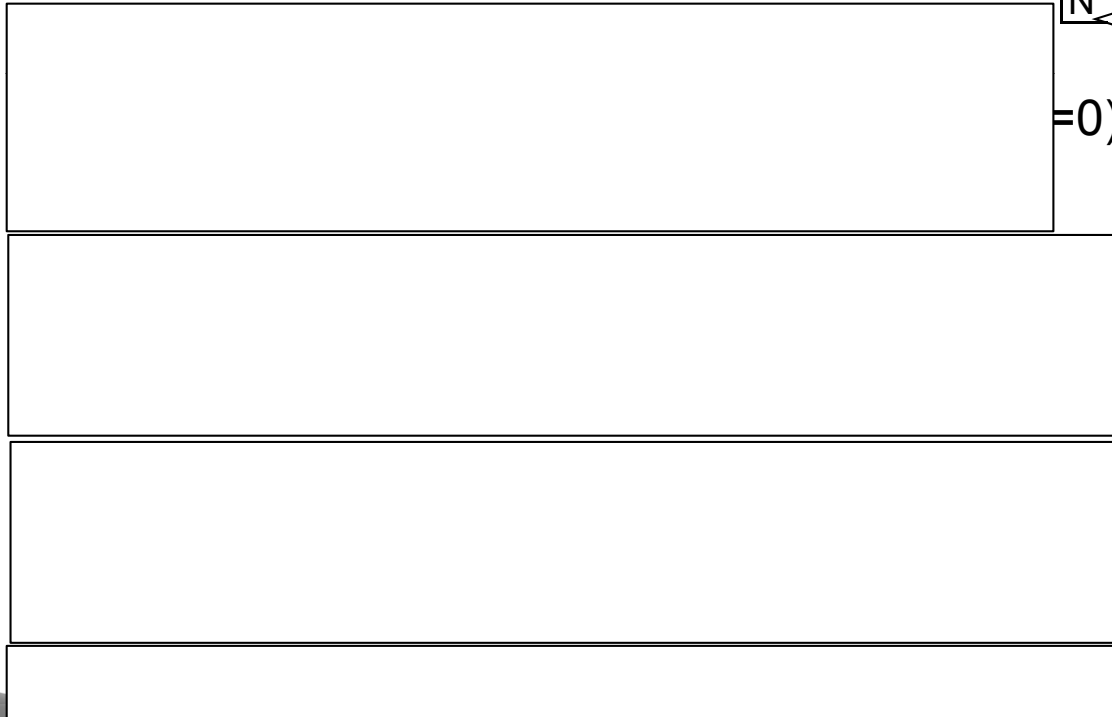
; PA口地址

MOV AL,[SI]

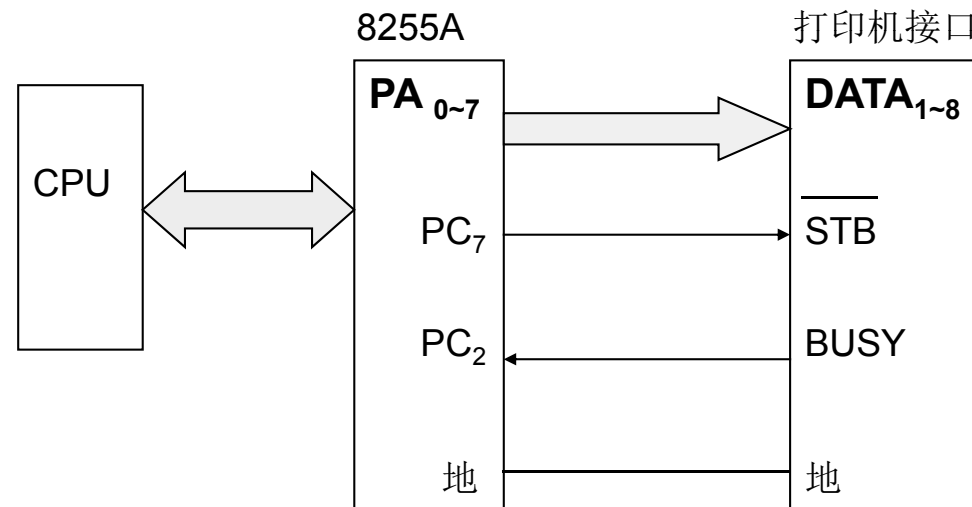
; 从内存取数

OUT DX,AL

; 送数到A口



- 思考：PC2 能不能换成PC4或PC5或PC6？



● 1方式（选通I/O方式、应答方式）

- 常用于查询（条件）传送或中断传送。

- ◆ 需设置专用联络线，联络线间有固定时序。

- ◆ 输入和输出都有锁存能力。

- PA和PB为数据口，PC部分引脚做联络信号

- ◆ A组：PA和PC₃~PC₅ 或PC₃, PC₆, PC₇ 3位；

- ◆ B组：PB和PC₀~PC₂。

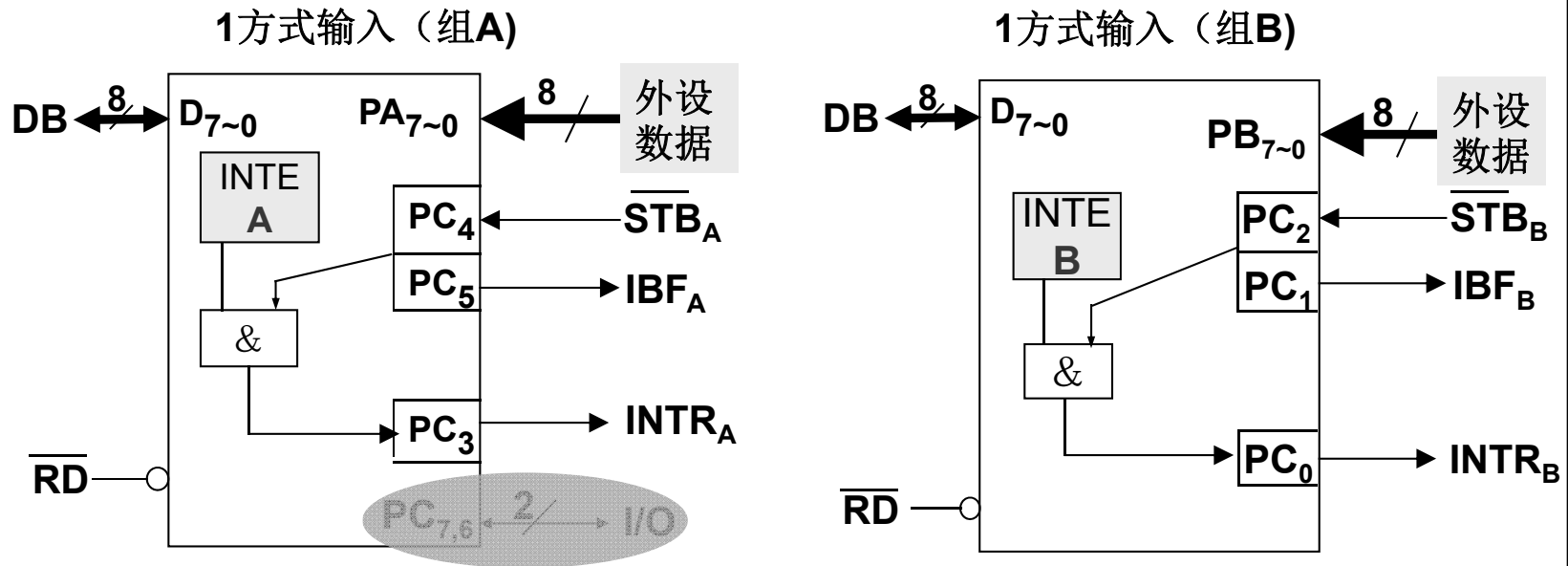
- 传送过程会产生状态字，可作查询或中断用。

- ◆ 从PC口读取状态字。

- 单向传送

1方式输入的联络信号定义及时序

- 输入：外设数据 → 送到**8255** → **CPU**读取
- **PA**用**PC_{3~5}**做专用联络线，**PB**利用**PC_{0~2}**做专用联络线。
- 注意：PC_{7,6}在方式1中是自由I/O引脚



● 1方式输入时的联络信号

■ $\overline{\text{STB}}$: STROBE,选通信号, 低电平有效。

◆ 外设将数据锁存到PA口和PB口。

■ IBF (Input Buffer Full), 缓冲器满高有效

◆ 8255给外设: 数据未被CPU取走。

■ INTR (Interrupt Request), 高电平有效

◆ 8255给CPU的“中断”信号。INTR变高3个必要条件

□ $\overline{\text{STB}}$ 变高;

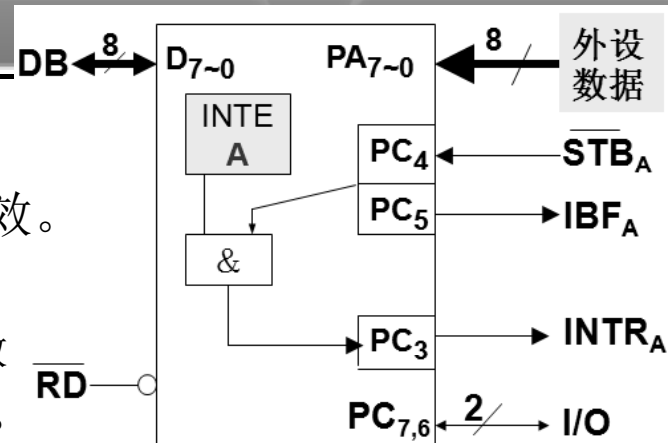
□ IBF变高;

□ 中断请求被允许 ($\text{INTE} = 1$)

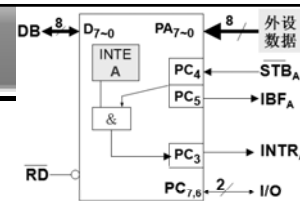
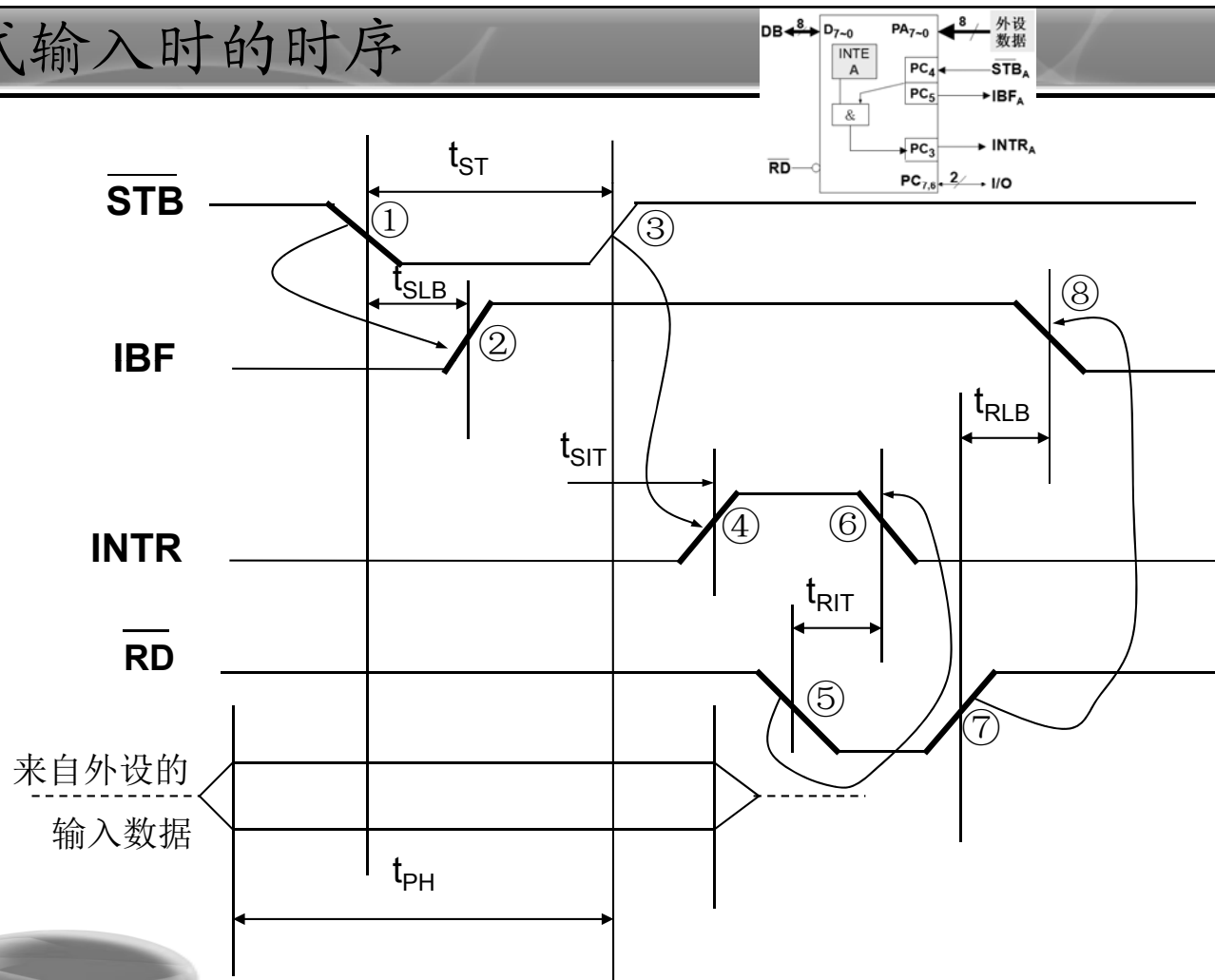
■ INTE: 中断允许 (INTE_A , INTE_B 允许PA或PB产生中断?)

◆ A组 INTE_A 由PC₄按位置位/复位来控制。

◆ B组 INTE_B 由PC₂按位置位/复位来控制。



1方式输入时的时序

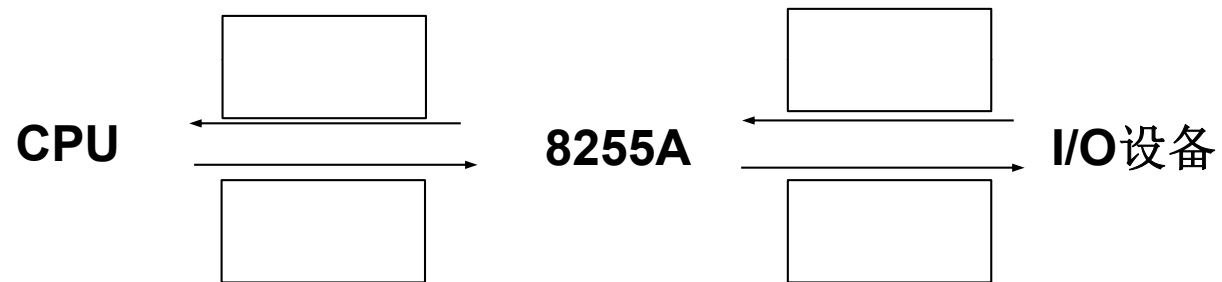


● 1方式输入时的时序

- 1、外设准备好数据，发送 $\overline{\text{STB}}$ 信号，把数据输入8255。
- 2、在 $\overline{\text{STB}}$ 下降沿，数据已锁存到8255缓冲器后，引起IBF变高，表示8255A“输入缓冲器满”，禁止输入新数据。
- 3、在 $\overline{\text{STB}}$ 上升沿后，在中断允许（ $\text{INTE}=1$ ）的情况下IBF的高电平产生中断请求，使INTR上升变高，通知CPU，接口中已有数据，请求CPU读数。CPU接受中断请求后，转到相应的中断子程序。在子程序中执行IN指令，将缓冲器中的数据取走。
- 4、CPU读操作（ $\overline{\text{RD}}$ ）信号的下降沿使INTR复位，撤消中断请求，为下一次中断请求作好准备。
- 5、 $\overline{\text{RD}}$ 信号的上升沿延时一段时间后清除IBF使其变低，表示接口的输入缓冲器变空，允许外设再次输入新数据。

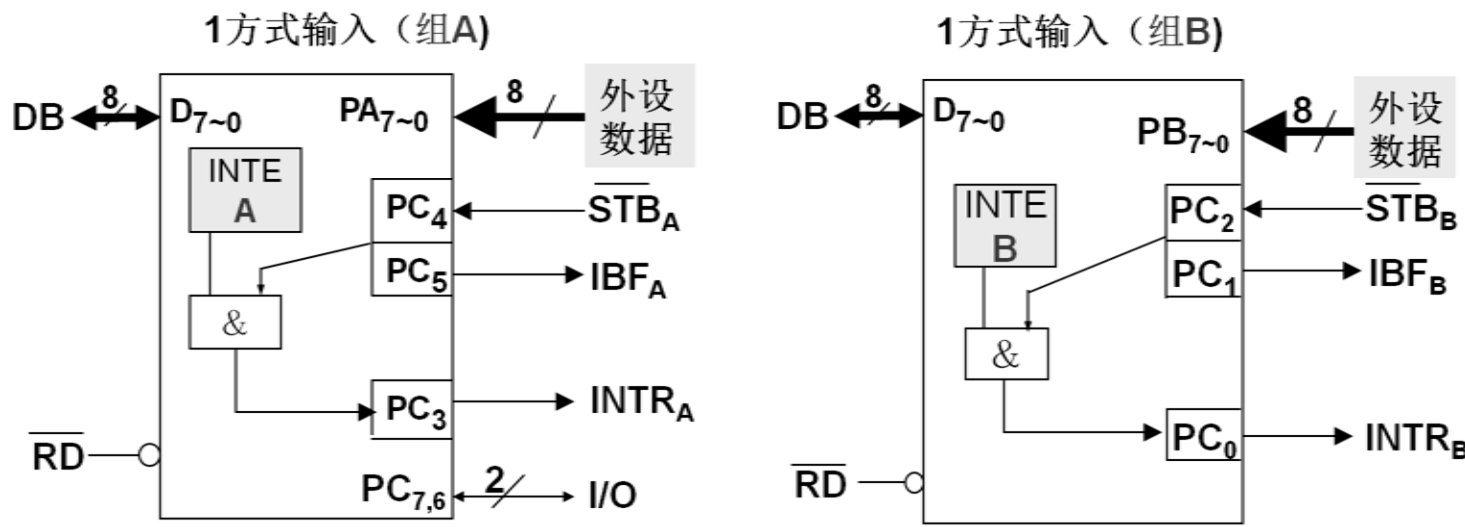
● 1方式输入时的时序

■ 数据从I/O设备发出，通过8255A，送到CPU。



补充说明

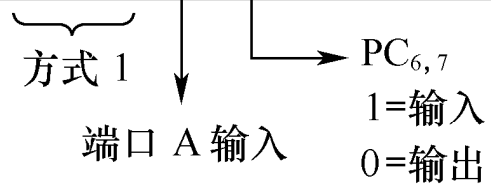
- 若 **A组**和**B组**都工作于方式1，则**PC**还留下**PC₆₋₇**2位，可以做普通**I/O**用。
 - 这2位可以由方式控制字**D3**位指定作为输入或输出。
 - 也具有按位置位/复位功能。
- 若**A组**或**B组**只有一组工作于方式1，则余下的**13**位可以工作在方式0，可以做普通**I/O**用。



● 1方式输入时的A,B两组的方式控制字

A 组方式控制字

D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀
1	0	1	1	I/O	×	×	×

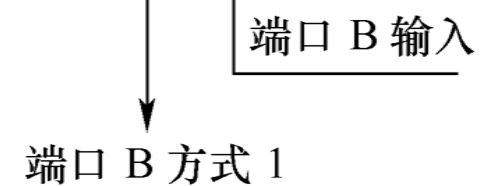


● 注意：D3位

■ PC_{7,6}

B 组方式控制字

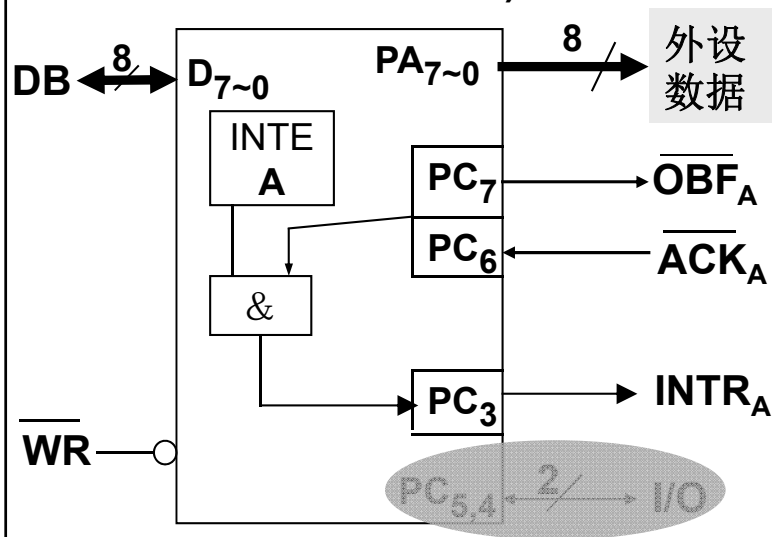
1	×	×	×	×	1	1	×
---	---	---	---	---	---	---	---



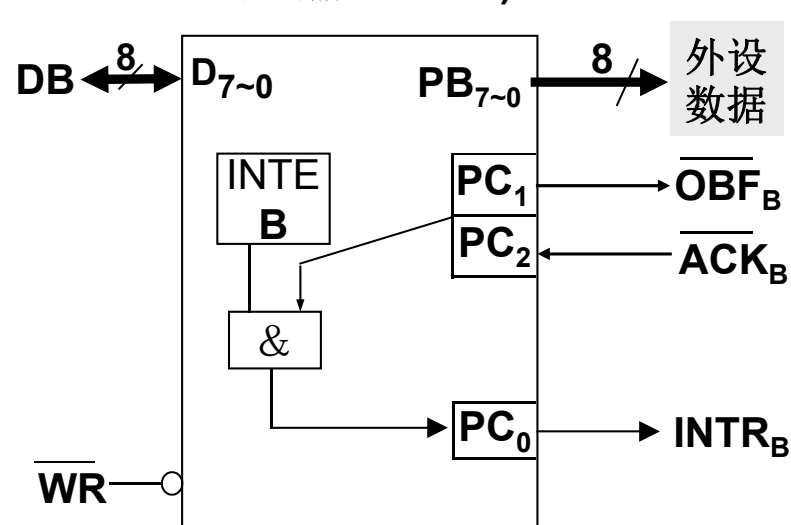
1方式输出的联络信号定义及时序

- 输出: CPU数据 → 写入8255 → 外设
- **PA**用 $PC_{3,6,7}$ 3位,**PB**用 $PC_{0\sim2}$ 3位做专用联络信号。
- 注意: $PC_{5,4}$ 是自由I/O引脚

1方式输出 (组A)



1方式输出 (组B)



● 1方式输出时的联络信号

■ $\overline{\text{OBF}}$ ，“输出缓冲器满”低有效

◆ CPU已将数据写到8255输出口

■ $\overline{\text{ACK}}$ ，外设对8255A应答低有效。

◆ 外设已经取走8255端口的数据。

◆ $\overline{\text{ACK}}$ 下降一段时间后使 $\overline{\text{OBF}}$ 变高，为下次输出作好准备。

■ INTR，8255A给CPU的“中断请求”信号，高电平有效。

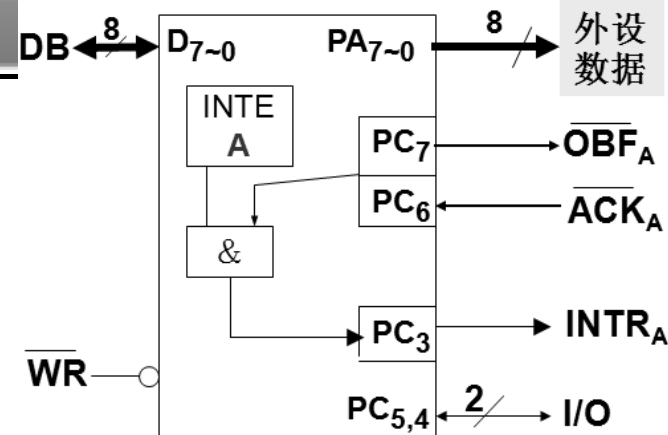
◆ 作为中断请求信号，请求CPU向8255A输出数据。

◆ INTR变高条件： $\overline{\text{OBF}}$ 、 $\overline{\text{ACK}}$ 和 INTE都为高电平。

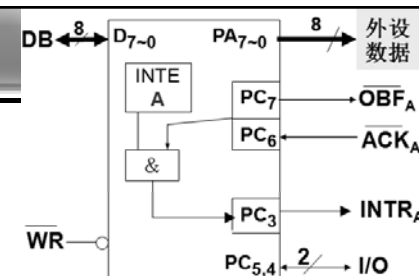
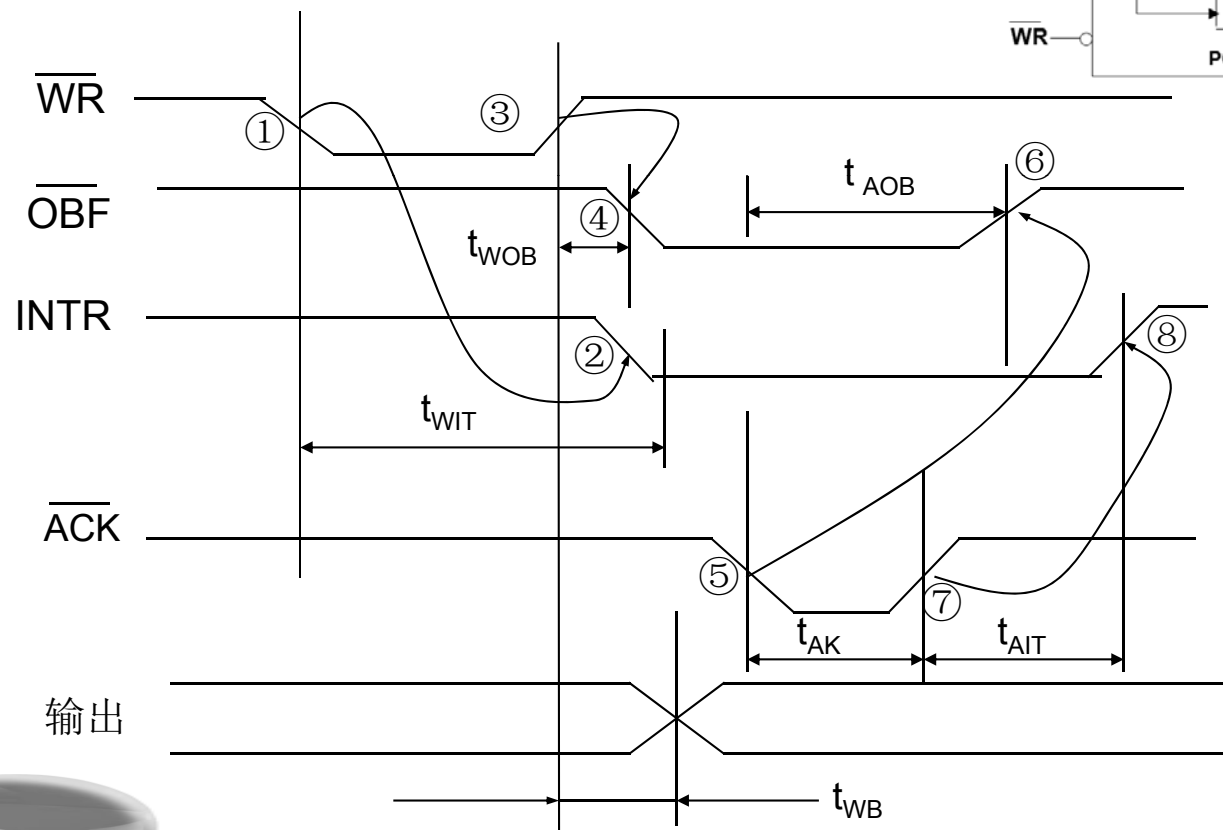
■ INTE：中断允许

◆ A组 INTE_A 由 PC_6 按位置位/复位来控制。

◆ B组 INTE_B 由 PC_2 按位置位/复位来控制。



● 1方式输出时的时序（假定在中断方式下）

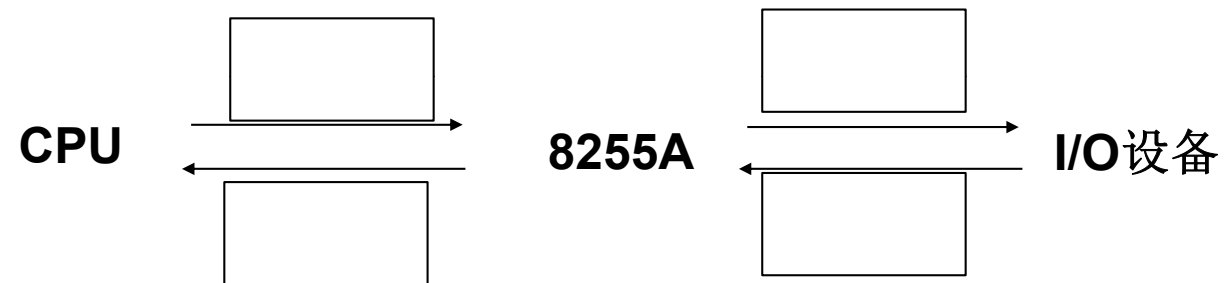


● 1方式输出时的时序（假定在中断方式下）

- 1、数据输出时，CPU准备好数据并写到8255输出数据寄存器。当CPU向8255A写完一个数据后， \overline{WR} 的上升沿使 \overline{OBF} 有效，表示8255A的输出缓冲器已满，通知外设读取数据。
- 2、 \overline{WR} 的下降沿同时使中断请求INTR变低，封锁中断请求。
- 3、外设得到 \overline{OBF} 有效的通知后，开始读数。当外设读取数据后，用 \overline{ACK} （低电平）回答8255A，表示数据已收到。
- 4、 \overline{ACK} 下降沿将 \overline{OBF} 置高，表示输出缓冲器变空，为下一次输出作准备。
- 5、在中断允许的情况下 \overline{ACK} 上升沿使INTR变高，产生中断请求。CPU在中断服务程序中，向8255A写(OUT)新数据。

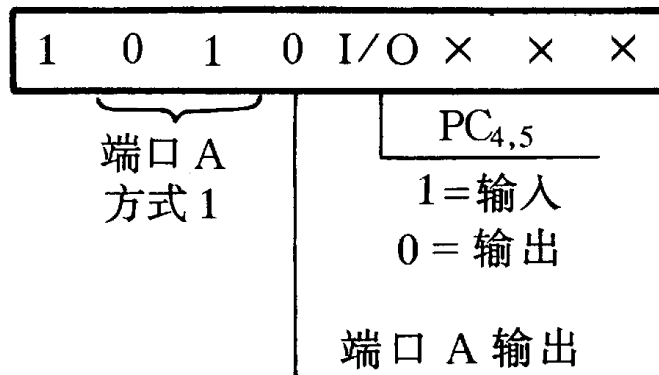
● 1方式输出时的时序

■ 数据从CPU，通过8255A，送到I/O设备有如下4步：



● 1方式输出时的A,B两组的方式控制字

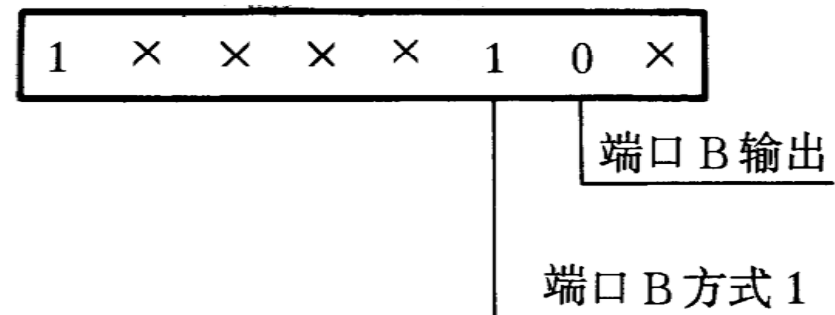
A 组方式 1 控制字



● 注意: D3位

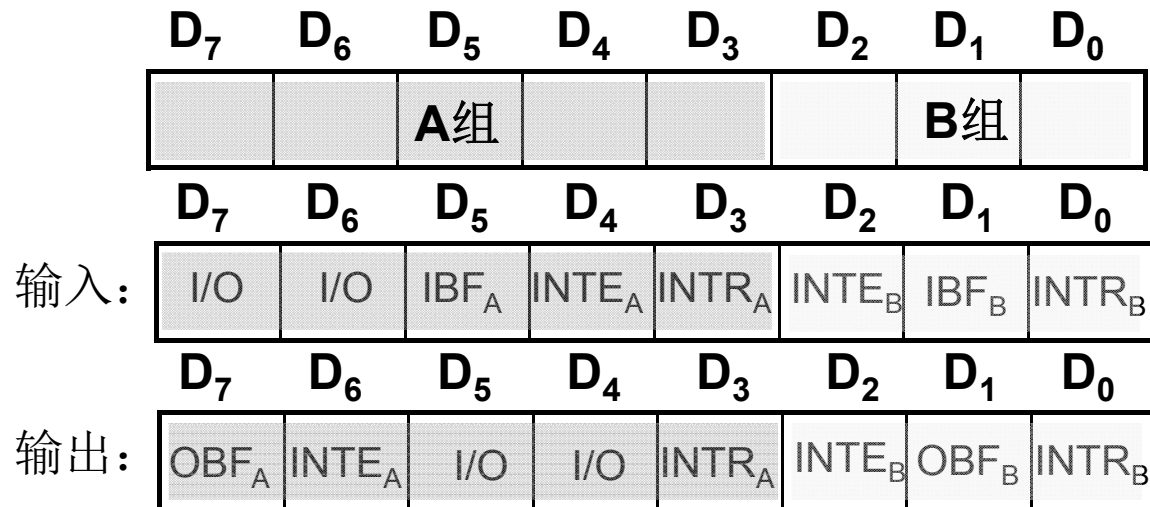
■ PC_{5,4}

B 组方式 1 输出控制字



1方式的状态字

- 提供8255的状态；辅助CPU识别中断源（不提供中断矢量）
- 状态字从PC口读取，与外部引脚无关。
- 状态字的格式



- 说明

- 供查询的状态位有INTR、IBF、OBF等。
- 在查询方式下，一般查询INTR位（比IBF、OBF 两位可靠）
- INTE位由用户用置位/复位指令设定，而非I/O操作自动产生。

1方式的应用和设计

● 基本思路

- (1) 确定A组和B组的输入/输出;
- (2) 把PC联络线与外设连好, 与CPU相应引脚连好。
- (3) 确定是采用中断方式还是查询方式和CPU通信

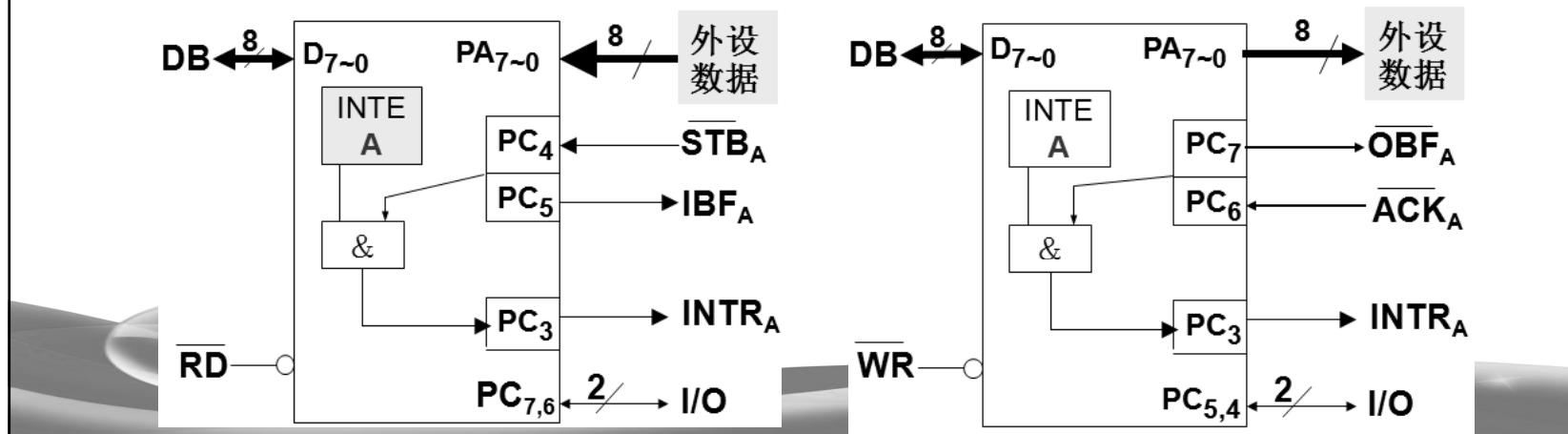
◆ 中断方式: 把INTR线接到CPU或8259A的INTR;

□ 8255不提供中断矢量, 可结合系统的中断控制器使用。

◆ 查询方式, 把INTR线空着, 程序查寻状态字INTR位。

□ 查询INTR位比查询IBF位或OBF位可靠性更高。

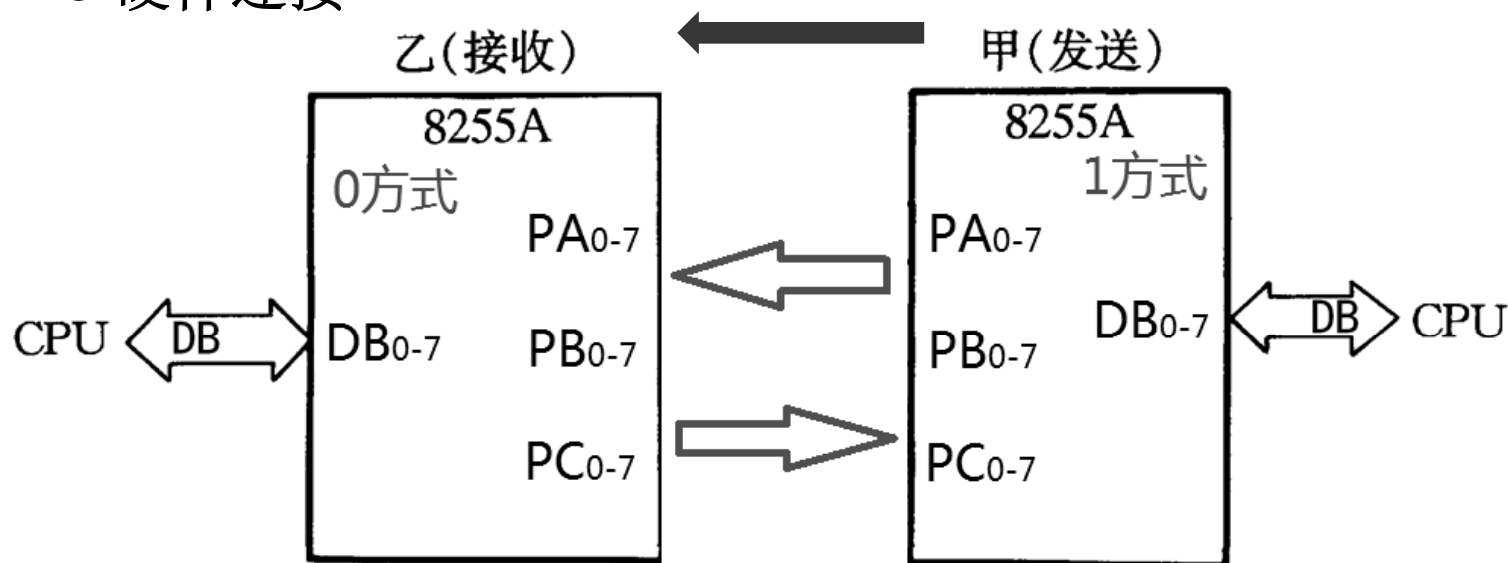
□ 注意: 查询INTR位时, 要确保INTE已经置位。



1方式应用例子

- 甲乙两机（查询方式）传送**1K**字节
 - 甲机发送（方式1）→ 乙机接收（方式0）。
 - 8255端口地址：300H~303H。

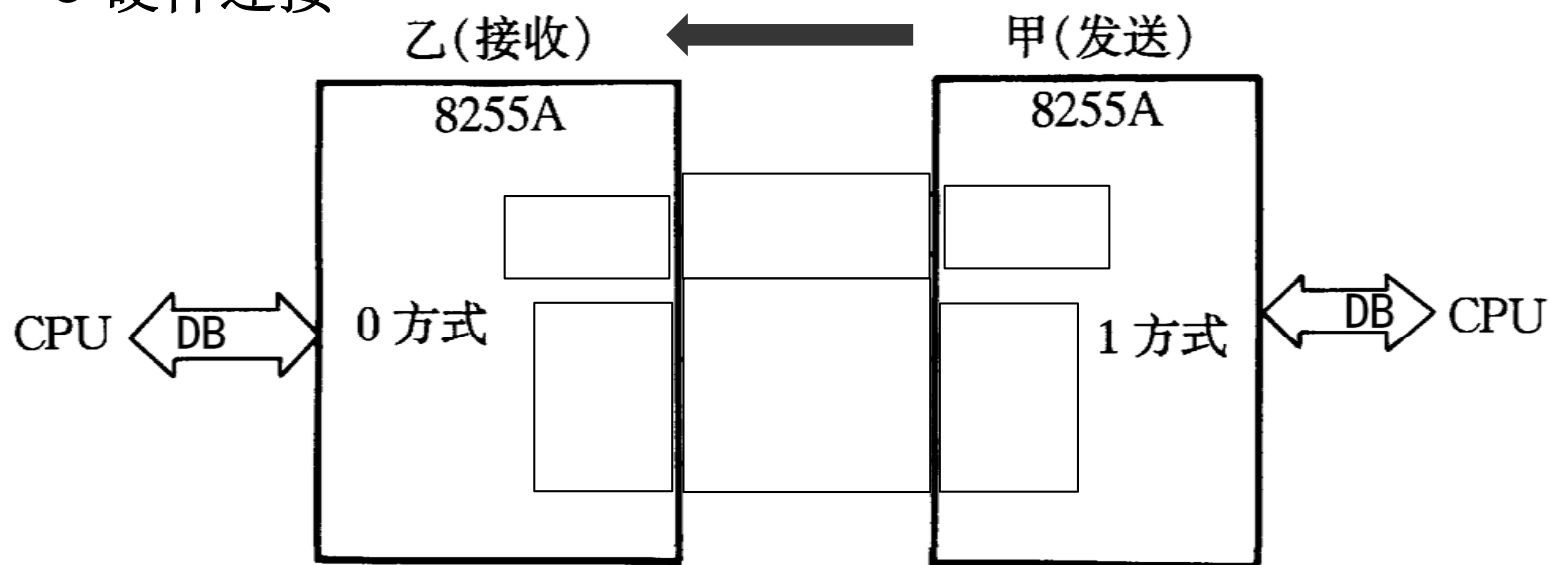
- 硬件连接



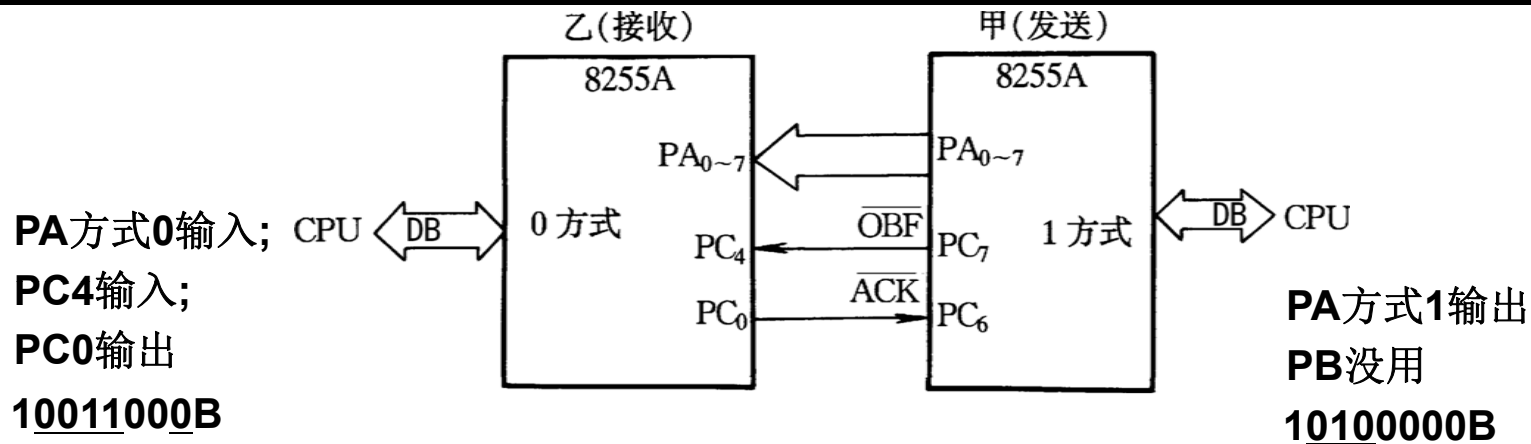
1方式应用例子

- 甲乙两机（查询方式）传送**1K**字节
 - 甲机发送（方式1）→ 乙机接收（方式0）。
 - 8255端口地址：300H~303H。

- 硬件连接



确定方式控制字和软件流程



乙机流程:

1. 设置控制字, PC_0 置1, 初始化:
(初始化数据首址和数量)
2. 查询 PC_4 (是否有数据? 负脉冲)
有: **PA** 接收1个数据
 PC_0 产生负脉冲
3. GOTO 2

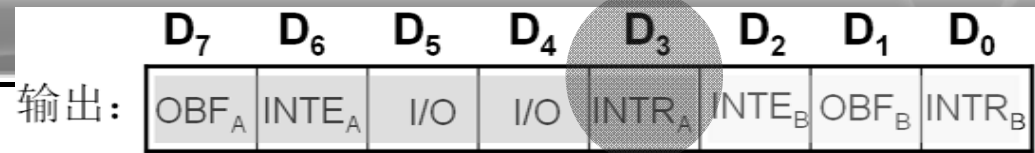
甲机流程:

1. 设置控制字, 使能中断, 初始化:
(初始化数据首址和数量)
2. **PA** 口发送1个数据
($OBF \rightarrow$ 负脉冲)
3. 轮询 $INTR_A$, 决定是否发新数据
是: GOTO 2
否: GOTO 3

课堂作业：补全程序，每行都有（姓名，学号，班级）

● 甲机发送程序（方式1）

MOV	DX, 303H	; 8255A命令口
MOV	AL, 10100000B	; 方式字：端口A方式1
OUT	DX, AL	; 输出方式字
MOV	AL, 01000000B;	; 置输出中断使能INTEA=1
OUT	DX, AL	; PC ₆ 置“1”
MOV	AX, 030H	; 待发送数据的首地址
MOV	ES, AX	
MOV	BX, 00H	; 待发送数据的偏移
MOV	CX, 3FFH	; 置发送字节数
MOV	DX, 300H	; 置8255A数据字地址
MOV	AL, ES:[BX]	; 取第1个待发送的数据
OUT	DX, AL	; 输出第1个数据
INC	BX	; 指向下一个数
DEC	CX	; 字节数减 1



输出:

```

L: MOV     DX, 302H      ; 8255A状态口 (即PC)
   IN      AL, DX        ; 输入状态
   AND     AL, 08H       ; 检查INTRA: 是否已被乙方接收?
   JZ      L             ; 若无中断请求则等待
   MOV     DX, 300H      ; 置数据口地址, 准备输出下一个数
   MOV     AL, ES:[BX]    ; 取数据
   OUT     DX, AL        ; 输出数据
   INC     BX            ; 指向下一个数
   DEC     CX            ; 字节数减 1
   JNZ     L             ; 若未发送完, 继续循环
   MOV     AX, 4C00H
   INT     21H           ; 发送完成, 返回DOS

```

● 甲机发送程序（方式1）

```

MOV          ; 8255A命令口
MOV          ; 方式字：端口A方式1
OUT          ; 输出方式字
MOV          ; 置输出中断使能INTEA=1
OUT          ; PC6置“1”
MOV     AX, 030H      ; 待发送数据的首地址
MOV     ES, AX
MOV     BX, 00H      ; 待发送数据的偏移
                ; 置发送字节数
                ; 置8255A数据字地址
MOV     AL, ES:[BX]   ; 取第1个待发送的数据
                ; 输出第1个数据
                ; 指向下一个数
                ; 字节数减1
    
```

甲机流程：

- 1.设置控制字，使能中断，初始化
- 2.PA口发送1个数据
（OBF → 负脉冲）
- 3.轮询INTR_A，决定是否发新数据
是：GOTO 2
否：GOTO 3

```

L: MOV     DX, 302H      ; 8255A状态口（即PC）
                ; 输入状态
                ; 检查INTRA：是否已被乙方接收？
                ; 若无中断请求则等待
MOV     DX, 300H      ; 置数据口地址，准备输出下一个数
                ; 取数据
OUT     DX, AL          ; 输出数据
                ; 指向下一个数
                ; 字节数减1
                ; 若未发送完，继续循环
MOV     AX, 4C00H
INT     21H            ; 发送完成，返回DOS
    
```

PA方式1输出

PB没用

10100000B

8255端口地址：300H~303H

输出：

D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀
OBF _A	INTE _A	I/O	I/O	INTR _A	INTE _B	OBF _B	INTR _B