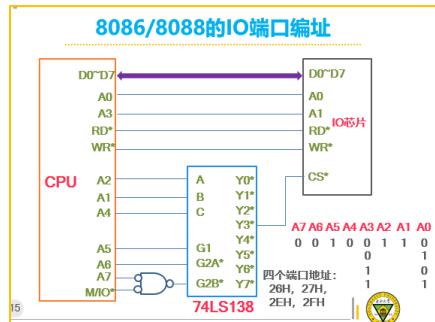


2¹⁶=64KB=65535 2²⁰=1MB 2²⁴=16MB

Chp6 输入输出, 中断

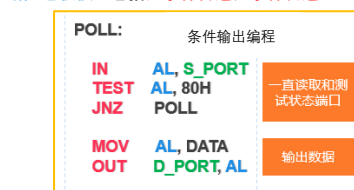
1.接口电路传输的信息包括: 数据信息(数字量,模拟量,开关量); 状态信息; 控制信息

2.端口编址:



3.程序传递:

通过 CPU 执行预先编制好的输入输出程序中的 I/O 指令来完成数据的传送。数据传送速度较低,数据输入输出的响应也较慢。包括无条件传送,条件传送。



4.中断传递:

中断是外设用来“主动”通知 CPU,准备发送或接收数据的一种方式。当中断发生时,CPU 暂停其现行程序,转而执行中断处理程序,完成 I/O 工作;当中断处理完毕后,CPU 又返回到原来的任务。

注意:两种 I/O 方式都需要 CPU 作为中介。程序的执行速度限定了传送的最大速度(约为几十 KB/秒)。

5.DMA 数据传送

传送过程:①接口准备就绪发出 DMA 请求 ②CPU 通过 HOLD 接收 DMA 发出的总线请求 ③HLDA 送回总线允许信号使 DMA 接管总线控制权 ④DMA 把地址送地址总线 ⑤DMA 请求被接口接收 ⑥内存把数据送数据总线 ⑦接口锁存数据 ⑧DMA 传送结束,将 HOLD 变低电平,撤销总线请求,让出总线权 ⑨CPU 将 HLDA 变为低电平恢复对总线的控制

6.中断及其相关

6.1 中断概述:

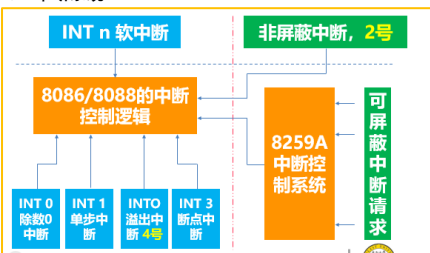
中断源: 外部设备,实时时钟,故障源。

中断存在优先级排队,高级可以打断低级

中断响应过程:关中断→保留断点→保护现场→转入中断服务程序→恢复现场→开中断,返回

可屏蔽中断 (INTR) 过程:①中断申请 (INTR) ②中断响应 INTR ③取中断类型号④把 PSW、CS、IP 压入堆栈,保存断点⑤清楚 IF 与 TF 标志⑥从断向量表中取新的 CS、IP 值⑦转入中断服务程序⑧开放中断⑨IRET 使旧 IP、CS、PSW 从堆栈中弹出⑩返回被中断程序

6.2 中断系统:



外部中断: 可屏蔽中断, 非屏蔽中断

中断的优先级:

优先级	中断名	中断类型	说明
高	除法错	类型0	高于可屏蔽中断 (软件中断)
	INT n	类型n	用户定义软件中断 (软件中断)
	INTO	类型4	溢出 (软件中断)
	NMI	类型2	非屏蔽中断 (硬件中断)
	INTR	由外设送入	可屏蔽中断 (硬件中断)
低	单步	类型1	调试用 (软件中断)

6.3 中断向量:

中断向量表:

共 256 类中断

每类中断的入口地址 (每类中断的入口地址 (每类中断的入口地址 (CS:IP) 占 4 个内存单元,共占用 1k 个内存单元,一般置于存储器的最低端,位于 00000-003FFH 的范围内。

向量的首地址: 类型号*4

6.4 中断时序响应: 第一个INTA周期通知响应成功,第二个INTA周期获得中断类型号

Chp7 可编程芯片

1.可编程中断控制器 8259A:

单片管理 8 级中断,9 片主从级联控制 64 级中断

1.1 芯片功能: 8259A 具有中断扩展、自动提供中断号、中断优先级裁决等中断管理功能。

1.2 芯片结构:



级联时从片的 INT 接主片的 IRx 引脚

中文	英文	缩写
中断屏蔽寄存器	Interrupt Mask Register	IMR
中断请求寄存器	Interrupt Request Register	IRR
中断服务寄存器	In-Service Register	ISR
中断结束	End Of Interrupt	EOI
中断请求	Interrupt Request	IR
初始化命令字	Initialization Command Word	ICW
操作命令字	Operation Command Word	OCW
优先级判别器	Priority Resolver	PR

寄存器功能:

总线缓冲器: 数据总线缓冲器为三态、双向、8位寄存器,数据总线D7~D0与CPU系统总线连接,构成CPU与8259A之间信息传送的通道。

读写控制逻辑

读/写控制逻辑用来接收CPU系统总线的读/写控制信号和端口地址选择信号,用于控制8259A内部寄存器的读/写操作。

级联缓冲比较

8259A可工作于单片或多片方式,级联缓冲/比较器提供多片8259A的管理和选择功能,其中一片为主片,其余为从片。

控制逻辑

负责向片内各部件发送控制信号,向CPU发送中断请求信号INT和接收CPU返回的INTA信号,控制8259A进入中断管理状态。

中断请求寄存器

IRR是一个8位寄存器,用于记录外部中断请求。当IRi请求时,IRR中的相应位Ii置1,在中断响应信号有效时,Ii被清除。

中断屏蔽寄存器

当IMR中的Di位为1时,对应的IRi请求被禁止;当IMR中的Di位为0时,则允许对应的中断请求进入。它可以由软件设置或清除。

优先级判别器

优先权判别器对IRR中记录的内容与当前ISR中记录的内容进行比较,以便选出当前优先级最高的中断请求。

中断服务寄存器

记录CPU当前正在服务的中断标志。IRi请求得到响应时,CPU的第一个INTA*将IRR中相应位置1,ISR复位由中断结束方式决定。

中断响应过程:

1)外设向 8259A 发出中断请求,IRR 相应位置 1。
2)在 IMR 和 PR 约束下,向 CPU 发出 INT 中断请求。
3)CPU 发出第一个 INTA*响应,ISR 相应位置 1。
4)CPU 发出第二个 INTA*,8259 发出中断类型号。
5)第二个 INTA*后,如果自动 EOI,会清除相应 ISR 位。

1.3 工作方式:

1.3.1 优先权方式:

1)普通全嵌套方式: 默认优先权管理方式,规定 IR0 最高,IR7 最低。

2)特殊全嵌套方式: 与完全嵌套方式基本类似,区别在于允许同级的中断请求进入。通常主片设置为特殊全嵌套方式,从片设置为普通全嵌套方式。

3)自动循环方式: 中断源优先级循环变化,又称等优先级方式。一个中断服务完成后其优先级自动降为最低,与之相邻低一级中断请求源为最高。开始时 IR0 优先级最高。

4)特殊循环方式: 与优先级自动循环方式相比,主要区别是可以编程设置开始的最低优先级。如,初始设 IR2 为最低优先级则 IR3 为最高优先级。

5)

1.3.2 结束中断方式:

注意:中断结束 (EOI): 中断处理结束后须将 ISR 相应位清 0,以开放同级或低级的中断请求。

1)自动中断结束: 自动结束方式(AEOI):CPU 响应中断请求时,在第二个 INTA 脉冲的后沿,由 8259A 自动将 ISR 的相应位清 0。

2)普通的 EOI: CPU 向 8259A 发出普通 EOI 命令时,8259A 把所有正在服务的中断中优先级最高的 ISR 位复位 (配合全嵌套优先权方式)。

3)特殊的 EOI: CPU 向 8259A 发送一条特殊 EOI 命令,命令中指出了要清除哪个 ISR 位 (用 3 位编码指定清除位),1.3.3 中断屏蔽方式:

1)普通屏蔽方式: 通过对中断屏蔽寄存器设定,实现对中断请求的屏蔽。中断屏蔽寄存器的每一位对应一个中断请求。1: 屏蔽,0: 开放。

2)特殊屏蔽方式: 提供了允许较低优先级的中断能够得到响应的特殊手段。假定当前正在处理 IR6,进入特殊屏蔽方式,这时除 IR6 和被 OCV1 屏蔽的中断,所有中断请求均能打断当前中断。其作用在于,可在程序的不同阶段任意改变中断级的优先次序

1.3.4 中断触发方式:

边沿触发: 8259A 的 IRQi 引脚上出现上升沿信号表示有中断请求

电平触发: 8259A 的 IRQi 引脚上出现高电平信号表示有中断请求。在第 1 个 INTA*结束前,IRi 必须保持高电平

1.3.5 数据线连接方式:

缓冲: 芯片缓冲。 非缓冲: 直接连数据总线

1.4 命令字:

1.4.1 初始化命令字

注意:用于初始化,从 ICW1 开始写。

ICW1 写入 A0=0 的端口

端口	D7	D6	D5	D4	D3	D2	D1	D0
A0	X	X	X	1(特征位)	1: 电平触发 0: 边沿触发	X	1: 单片 0: 多片	1: 写ICW4 0: 不写

ICW2

写入 A0=1 的端口,命令字为 IR0 对应的中断类型号。IR0 的中断类型号低三位必为 000,其它 IRi 根据 IR0 的中断类型号递增。

ICW3

主片: 对应位: 1--接了从片; 0--没有接从片

从片: 低三位 XXX: 对应主片的 IR0-IR7

写入 A0=1 的端口

ICW4 中断结束方式初始化命令字

端口	D7	D6	D5	D4	D3	D2	D1	D0
A0=1	0	0	0	0: 普通屏蔽 1: 特殊屏蔽	0X: 非缓冲 10: 缓冲/从片 11: 缓冲/主片		1: 自动EOI 0: 非自动EOI	1: 8086/8088 0: 8088/8088

1.4.2 控制/操作命令字

注意:初始化完成后,8259A 进入正常工作状态时,可以在应用程序中随时用操作命令字编程,改变 8259A 的工作方式或读出 8259A 内部的状态。可不按顺序写 OCV1 写入 A0=1 的端口,送入 IMR

中断屏蔽字: 1: 屏蔽 0: 允许

OCW2 写入 A0=0 的端口



注意:初始化后,如果采用了非自动中断结束,需要在中断服务子程序的最后发出 EOI 命令。

最常用的为普通 EOI 命令,即: OCW2 = 20H

OCW3 写入 A0=0 的端口

端口	D7	D6	D5	D4	D3	D2	D1	D0
A0=0	X		10: 清除特殊屏蔽 11: 设置特殊屏蔽	0		1	1: 查询命令 0: 非查询	10: 随后读IRR 11: 随后读ISR

注意:

1)OCW3 命令分为三类: 特殊屏蔽功能设定和清除、中断状态查询位、读 ISR 和 IRR 命令。

2)每次只能使用上述一个功能,除特征位,其它位为 0。

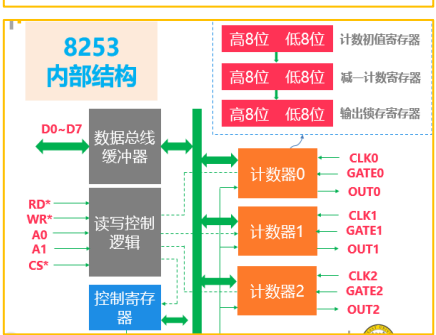
3)非查询: 读取 IRR 或 ISR

2.可编程定时器 8253:

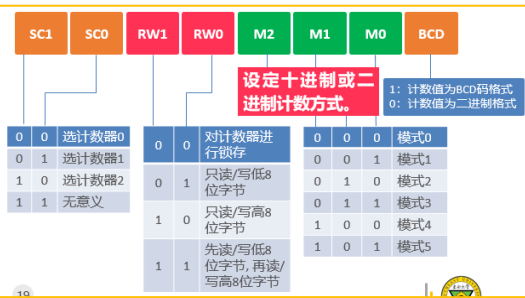
2.1 芯片功能:

8253 是一种可编程的计数器/定时器接口芯片,最高计数频率为 2MHz,可用于产生各种定时波形,也可用于对外部事件计数。内部有三个独立的计数器,通过设置控制字,各计数器可以工作于 6 种工作方式。独立计数器可以级联。

2.2 芯片结构:



2.3 控制字：写入 A0A1=11 的端口



注意：二进制计数最大值： $2^{16}=65536$ 。十进制计数值最大值：10000。最大值计数时从 0 开始计数。

初始化：写完控制字之后，计数初值写入对应的计数器端口。

2.4 工作方式：

2.4.1 模式 0

结束后输出高电平。可用于产生中断请求

GATE=1:

1) 写入控制字，OUT 端输出低电平为起始电平，装入计数初值 n，开始计数。

2) 写信号后沿经一个 CLK 装入计数初值，每过一个 CLK，在 CLK 下降沿计数器减 1。

3) n=0 时计数结束，OUT 由低电平变为高电平并保持，不开始重新计数。计数过程中若重新装入则重新计数

GATE=0:

停止计数，直至 GATE 恢复高电平，再继续计数。

2.4.2 模式 1

输出负脉冲，宽度由计数值决定。

1) 计数过程中仅有 1 个上升沿：

写入控制字，OUT 端输出高电平为起始电平。装入计数初值 n 后，必须等待 GATE 的上升沿来后才转入计数，这时 OUT 变低，开始计数，每一个计数脉冲，计数器值减 1。

计数到 0，OUT 变成高电平，负脉冲结束，脉冲宽度=tc×n (tc 为时钟周期)。

在计数过程中，若 GATE 变低，不影响计数。

2) 计数过程中超过 1 个上升沿：

在计数过程中，若再次产生 GATE 的上升沿触发，则要重新装入 n 值。在触发脉冲上升沿之后的一个 CLK 脉冲的下降沿，计数器重新开始计数。可以通过该方法改变脉冲的宽度。

如果是计数结束后，再次产生 GATE 的上升沿触发，则需要重新装入 n 值计数。

2.4.3 模式 2

分频器，产生分频负脉冲序列。

1) GATE=1:

写入控制字，OUT 端输出高电平为起始电平，装入计数初值 n，开始计数。

每一个计数脉冲下降沿 n 减 1，当 n 减至 1 时，OUT 变低，n 减为 0 时，OUT 变高，产生一个与时钟脉冲周期一样宽的负脉冲。

接着自动装入 n 连续计数，输出频率为：fclk/n。

计数过程中，允许重新装入新的 n 值，下一个计数周期按新的 n 值计数。

2) GATE=0:

计数过程中，若 GATE=0，停止计数，并强迫 OUT 输出高电平，在 GATE 变为高电平后，重新装入 n 值，开始计数。

2.4.4 模式 3

分频器，输出方波，频率为 CLK/n

1) GATE=1:

写入控制字后，OUT 端输出低电平作为起始电平，装入计数初值 n 后，变为高电平。

n 为偶数，每个时钟脉冲下降沿 n 值减 1，至 n/2 后，电平变为低电平，并继续减 1 计数至 0，然后改变 OUT 电平，重新装入 n，开始计数。

n 为奇数，输出高电平宽度为(n+1)/2，低电平宽度为(n-1)/2 的方波。

2) GATE=0:

停止计数，并强迫 OUT 输出高电平，在 GATE 变高后，重新将 n 装入，开始计数。

2.4.5 模式 4

软件触发选通脉冲。

GATE=1:

写入控制字后，OUT 端变为高电平，写入初始值后，计数器作减 1 计数，OUT 电平保持不变。

计数器减至 0 时，OUT 端输出一个脉冲周期的负脉冲，然后停止计数，只有输入新的计数值后，才能开始新的计数。

GATE 变为 0:

计数过程中，若 GATE 变低电平，停止计数，在其变高后，重新将 n 装入，开始计数。

2.4.6 模式 5

硬件触发选通脉冲

写入控制字后，OUT 端变为高电平，写入初始值 n 后，必须等待 GATE 的上升沿触发才转入计数。

计数器减至 0 时，OUT 端输出一个脉冲周期的负脉冲。然后 n 值自动装入计数器，但要等 GATE 的上升沿来后才再次开始计数。

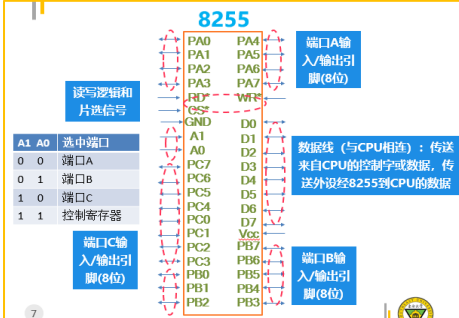
计数过程中，若 GATE 变低电平，不影响计数，但其上升沿将使得 n 重新装入计数器，开始计数。

如果是计数结束后，再一次产生 GATE 的上升沿，则再次计数。

注意：1. 方式 2 和方式 3 自动重复计数，常用作分频，要求 GATE=1。2. 方式 1 和方式 5，虽然不自动重复，但是，可通过产生 GATE 上升沿即通过硬件重复触发。3. 方式 0 用于计数结束产生中断。方式 1 产生一个宽度为 nT 的负脉冲。4. 方式 4 和方式 5 波形相同，产生选通信号，但门控信号 GATE 要求不同，方式 4 要求 GATE=1，方式 5 要求 GATE 上升沿。

3. 可编程并口 8255:

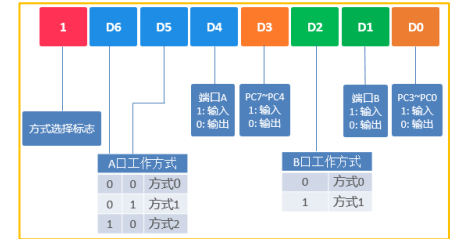
3.1 芯片结构



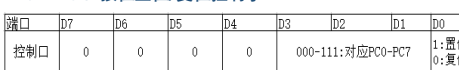
内部分为两组：A 组控制：控制部件+A 口+C 口高 4 位
B 组控制：控制部件+B 口+C 口低 4 位

3.2 控制字

3.2.1 工作方式控制字



3.2.2 C 口按位置位/复位控制字



注意：以上的两个控制字均写入控制口 (11)

3.3 工作方式

3.3.1 方式 0 (端口 ABC)

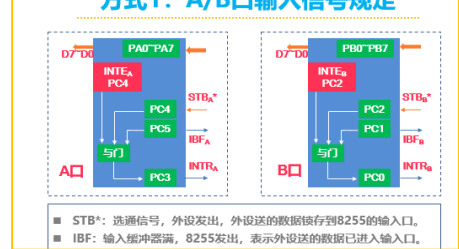
基本输入输出，任意端口均可以作为输入/输出

注意：C 口的高低 4 位可以分开使用，与 AB 组合起来共 16 种；方式 0 可用于无条件传送或查询式传送。

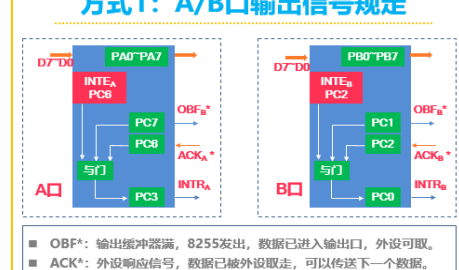
3.3.2 方式 1 (端口 AB)

选通的输入输出方式

方式 1: A/B 口输入信号规定



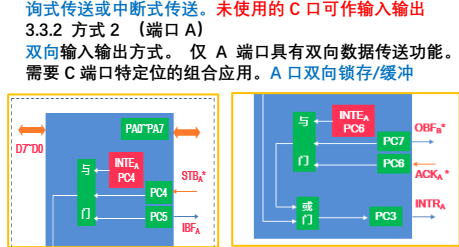
方式 1: A/B 口输出信号规定



注意：方式 1 要求外设提供选通信号 STB*或 ACK*的能力。INTE (中断允许) 信号无引出，通过控制口对 C 口相应的位置/复位，设置允许或不允许。方式 1 可用于查询式传送或中断式传送。未使用的 C 口可作输入输出

3.3.2 方式 2 (端口 A)

双向输入输出方式。仅 A 端口具有双向数据传送功能。需要 C 端口特定位的组合应用。A 口双向锁存/缓冲



注意：未被占用的 C 端口可以单独作为输入/输出

A 端口方式 2、B 端口方式 1；PC0, PC1, PC2 可以用作输入输出。

4. 可编程串口 8250:

4.1 串行通信原理

4.1.1 串行通信：数据在单根数据线上逐位传送。

4.1.2 工作方式：

异步工作方式：不发数据时处于高电平。有数据时，先发送一个低电平为起始位，然后发数据位、停止位。不要求

时钟相同，传输速率双方约定。

同步工作方式：先发送一到两个特殊同步字符，当收发同步后，连续发送一块数据。收发双方必须用同一时钟协调，

以确定数据中每一位的位置。

4.1.3 示例：



4.1.4 波特率

波特率：每秒传送数据的位数，单位波特，它用于表示数据

传送的速率。若一个串行字符要 10 位，每秒传送 240 个字符。则波特率=10×240=2400 波特

4.1.5 通信协议

收发双方的同步方式、传输速率、通信报文格式；传输控制

步骤、及控制字符定义；差错检验方式、数据编码。

注意：传输电缆线上的 1 为低电平，0 为高电平

4.2 芯片结构

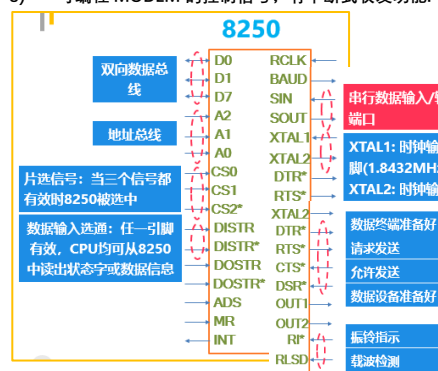
8250 特性：

1) 支持异步串行通信规程，发送时可自动插入起始位、

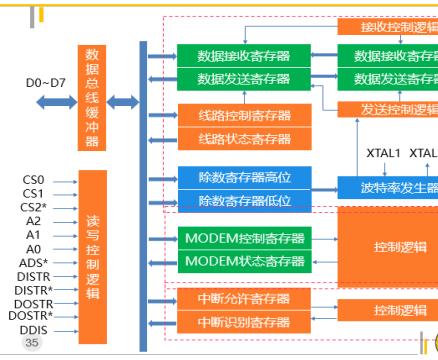
停止位和奇偶校验位，接收时能自动删除。

2) 内部具有可编程的时钟产生电路，发送和接收都采用双缓冲结构。

3) 可编程 MODEM 的控制信号，有中断式收发功能。



编程结构：



4.3 工作过程

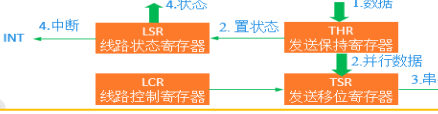
4.3.1 发送数据

CPU 将数据发送给 8250 的 THR; THR 把数据送至 TSR.

LSR 中"THR 空"状态位置位；TSR 根据 LCR 规定的格式从

低到高逐位发送数据；LSR 中"THR 空"可产生中断，也可

查询以连续发送。



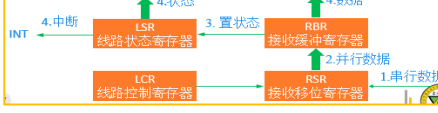
4.3.2 接收数据

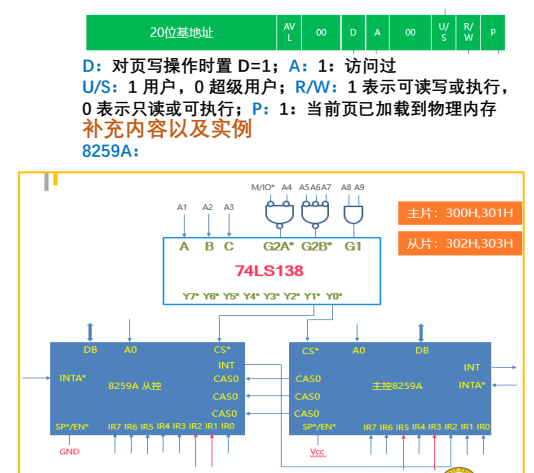
SIN 引脚上的串行数据逐位进入 RSR; RSR 根据 LCR 中

规定接收一个完整数据并送至 RBR; RBR 收到 RSR 的数

据，将 LSR 中"RBR 满"状态位置位；LSR 中"RBR 满"可产生

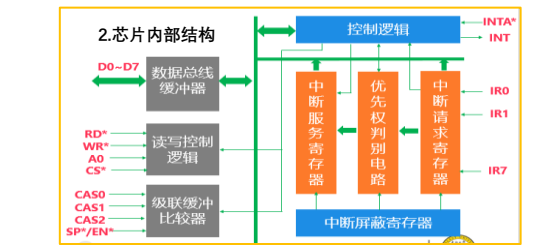
中断，也可查询以连续接收。





1. OCW3 的读/写操作与内存的读/写类似。但是，需要特别注意的是，同样的地址，读出来的值和写进去的值可能意义是不同的：1. A0=1，读出 IMR。2. A0=0，读出 IRR、ISR 或中断状态。需要两步：写 OCW3；从端口读

读IRR	读IMR, 关IR4, IR3	读ISR	读中断状态
MOV AL, 0AH OUT 20H, AL IN AL, 20H	IN AL, 21H OR AL, 18H OUT 21H, AL	MOV AL, 08H OUT 20H, AL IN AL, 20H	MOV AL, 0CH OUT 20H, AL IN AL, 20H



提供多片8259A的管理和选择功能，主片为输出，从片为输入，主片通过CAS2~CAS0的编码选择和管理从片。

PR 在多重中断的时候，判断新的中断是否允许打断正在处理的中断

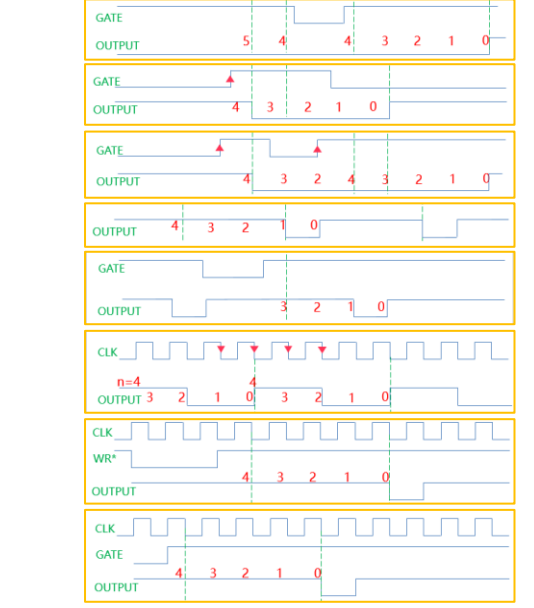
3. 第一个 INTA*周期，ISR 置 1 后，对应的 IRR 恢复为 0

4. EOI 的作用：在第二个 INTA*后，将 ISR 对应位置 0

5. ICW2 示例：中断类型码应设为 08H~0FH，ICW2 设置为 08H。ICW2 的低 3 位与中断类型码数值无关。

6. OCW3 的查询命令：使用查询命令前，CPU 首先必须关中断。一旦 CPU 发出查询命令，8259A 把 IN 指令的 RD* 脉冲当作响应信号，如有中断请求，就使 ISR 相应位置 1，并将查询字送到 DB 上，供 CPU 从 A0=0 的端口读取。

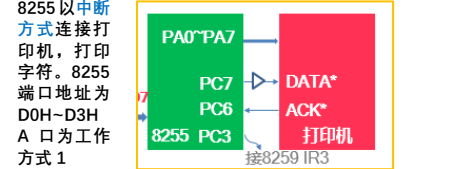
8253 定时器：工作模式的波形示意



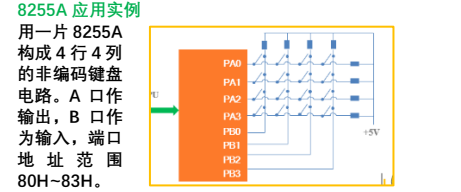
■ 0号计数器，采用二进制计数，计数初值为16位。

■ 写出读取计数器计数值的程序。

```
MOV AL, 0000000B  
OUT 43H, AL  
IN AL, 40H  
MOV AH, AL  
IN AL, 40H  
XCHG AL, AH
```



```
BUF DB 'Hello world'  
.....  
MOV AL, 0A0H ;方式字设置  
OUT 0D3H, AL  
MOV AL, 0CH ;PC6置1，开中断  
OUT 0D3H, AL  
LEA SI, BUF ;数据存储器区设置  
MOV CX, 11 ;等待  
LOP: JNZ LOP  
INT_PRT PROC FAR  
MOV AL, [SI]  
OUT 0D0H, AL ;送打印数据  
INC SI  
DEC CX  
IRET  
INT_PRT ENDP
```

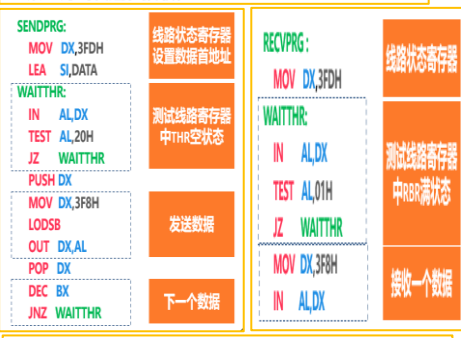


```
判断是否有键按下：  
MOV AL, 82H ; 初始化 8255A 口输出，B 口输入，方式 0。  
OUT 83H, AL  
MOV AL, 00H  
OUT 80H, AL ; 使 A 口输出全为低电平。  
IN AL, 81H ; 读取 B 口。  
AND AL, 0FH ; 屏蔽高 4 位。  
CMP AL, 0FH ; 判断低 4 位是否有低电平。  
JZ LOOA  
CALL D20MS ; 若有，则延时 20ms 再次读取（排除干扰）  
IN AL, 81H  
AND AL, 0FH  
CMP AL, 0FH ; 再次读取并判断。  
JZ LOOA ; 若仍有低电平，则确认有键按下，转入 STRAT。  
判哪个键按下：  
START: MOV BL, 4 ; 行数放入 BL。  
MOV BH, 4 ; 列数放入 BH。  
MOV AL, 1111110B ; 先扫描 0 行。  
MOV CL, 0FH ; 设置键盘屏蔽码。  
MOV CH, 0FFH ; 设置起始键号。  
LOP1: OUT 80H, AL ; 扫描一行。  
ROL AL, 1 ; 修改扫描码，准备扫描下一行。  
MOV AH, AL ; 保存扫描码。  
IN AL, 81H ; 读列值 (B 口)。  
AND AL, CL  
CMP AL, CL ; 判断是否有低电平 (是否有 0)。  
JNZ LOP2 ; 有列值为 0，转 LOP2，找列线。  
ADD CH, BH ; 若无键按下，修改键号。  
MOV AL, AH ; 恢复扫描码，准备扫描下一行。  
DEC BL ; 行数减 1。  
JNZ LOP1 ; 未扫完，转 LOP1。  
JMP START ; 重新扫描。  
判断出按下键后，确定键号：  
LOP2: INC CH ; 键号加 1。  
ROR AL, 1 ; 检查对应列线。  
JC LOP2 ; 若无键按下，查下一列。  
MOV AL, CH ; 若找到按键，键号送 AL。  
CMP AL, 0  
JZ KEY0 ; 根据键号，转向相应的处理程序。  
JZ KEY1  
; 省略。  
CMP AL, 0EH  
JZ KEY14  
JMP KEY15
```

■ 8250 的端口为：3F8H~3FEH。

■ 假定要发送的字节数放在 BX 中，要发送的数据顺序存放在以 DATA 为首地址的内存区中。

■ 以查询式发送和接收。



例子：中断式收发

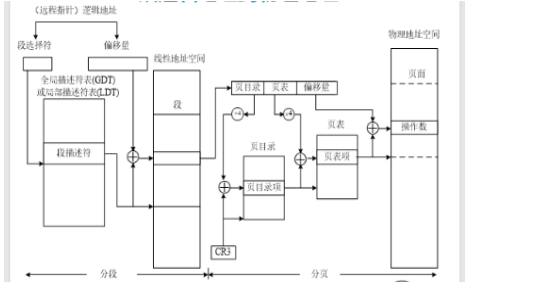
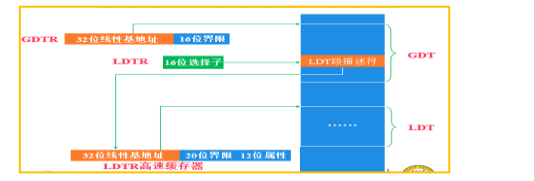
```
TRINT: .....  
MOV DX, 3F2H  
IN AL, DX  
TEST AL, 02H  
JZ RCVE  
MOV AL, DATA  
MOV DX, 3F0H  
OUT DX, AL  
JMP CONT  
RCVE: MOV DX, 3F0H  
IN AL, DX  
CONT:
```

假设只允许收发中断

8250 的 MCR:

无中断方式	中断方式	循环自测
MOV AL, 03H MOV DX, 3FCH OUT DX, AL	MOV AL, 0BH MOV DX, 3FCH OUT DX, AL	MOV AL, 13H MOV DX, 3FCH OUT DX, AL

中断方式：设定 OUT2*有效，它通过三态门控制向 PC 申请中断。



描述符表分类：全局，局部，中断。存放描述符的特殊段。最大 64KB，最多存放 8192 个描述符。

虚拟 8086 模式：多任务运行 DOS；与 8086 兼容又不同于实地址

非屏蔽中断：NMI 中断，不会被 IF 屏蔽，实际运行时优先级高于 INTR 上的所有中断

内部中断（软件中断）：

- 除法错误 INT0：除数为 0 或商数超出范围。
- 溢出 INTO：OF=1 时转入对应中断。
- 单步中断 INT1：TF=1，单步工作。
- 断点中断：INT3：一般用于调试程序。

代码查看：

SAL/SAR d, count: 算数左/右移。移出的位进入 CF，移入的位保持符号位不变

SHL/SHR d, count: 逻辑左/右移。移出的位进入 CF，移入的位用 0 填充

注意：1. 若 count 大于 1，则将 count 放入 CL 中。2. 影响 OF 和 CF，若仅移 1 位，且结果使最高位发生变化，则置 OF 为 1，若移多位，OF 无效。3. D 允许为 8/16 位寄存器或存储器

NEC d: 取负指令，CF 置 1，影响其余 5 位

CMP d, s: 比较指令，d-s，不储存结果但是影响标志位

注意：1. 目标操作数不允许是立即数 2. 源和目标操作数不能同时为寄存器操作数

MUL/IMUL s: 不带符号/带符号乘法。分为字节/字乘法。AX*S 送入 DX, AX; AL*S 送入 AH, AL

注意：1. AL (AX) 为隐含的被乘数寄存器 2. SRC 不能为立即数 4. 除 CF 和 OF 外，对其它状态标志位无意义

MUL 对标志位的影响：CF OF= 0 0 乘积的高一半为 0 (字节操作的 (AH) 或字操作的 (DX)) ; CF OF= 1 1 乘积的高一半不为 0

DIV/IDIV s: 不带符号/带符号除法。分为字节/字除法。

注意：1. AX (DX, AX) 为隐含的被除数寄存器；2. AL (AX) 为隐含的商数寄存器；3. AH (DX) 为隐含的余数寄存器；4. SRC 不能为立即数；5. 对所有状态标志位均无定义

