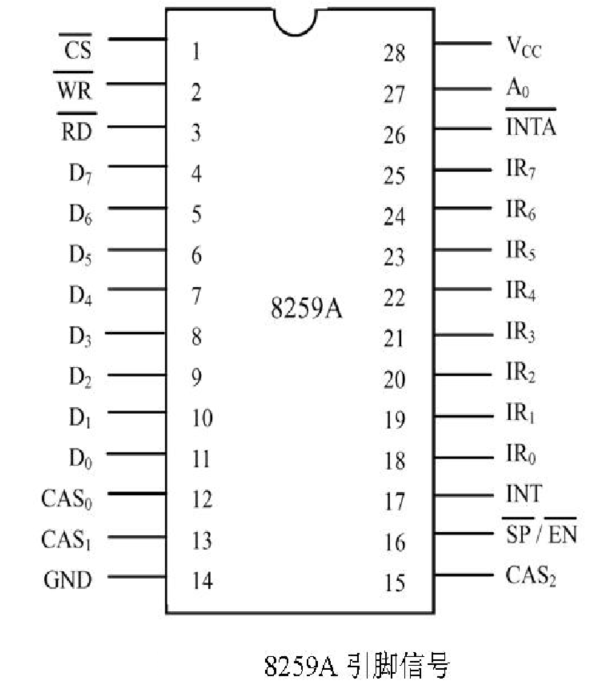
8259

可编程中断控制器8259A是Intel公司专为80x86 CPU控制外部中断而设计开发的芯片。　它将中断源优先级判优、中断源识别和中断屏蔽电路集于一体，不需要附加任何电路就可以对外部中断进行管理，单片可以管理8级外部中断，在多片级联方式下，可以管理多达64级的外部中断。

可编程中断控制器8259A是28引脚双列直插式芯片，单一＋5V电源供电．

引脚信号

8259A引脚信号如图所示：



-CS（chip select）：片选信号，输入，低电平有效。当有效时8259A被选中。

-WR（write）：写信号，输入，低电平有效。当有效时CPU对8259A进行写操作。

-RD（read）：读信号，输入，低电平有效。当有效时CPU对8259A进行读操作。

D7～D0（bidirectional data bus）：双向、三态数据线，与CPU系统总线连接。

A0（address line）：端口地址选择信号，输入，由8259A片内译码，选择内部寄存器。

CAS2～CAS0（cascade lines）：级联信号线，作为主片与从片的连接线，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。

-SP/-EN（slave program/enable buffer）：双向信号线，用于从片选择或总线驱动器的控制信号。当8259A工作于非缓冲方式时，-SP/-EN作为输入信号线，用于从片选择。级联中的从片-SP/-EN接低电平，主片-SP/-EN接高电平。当8259A工作于缓冲方式时，-SP/-EN作为输出信号线，用做8259A与系统总线驱动器的控制信号。

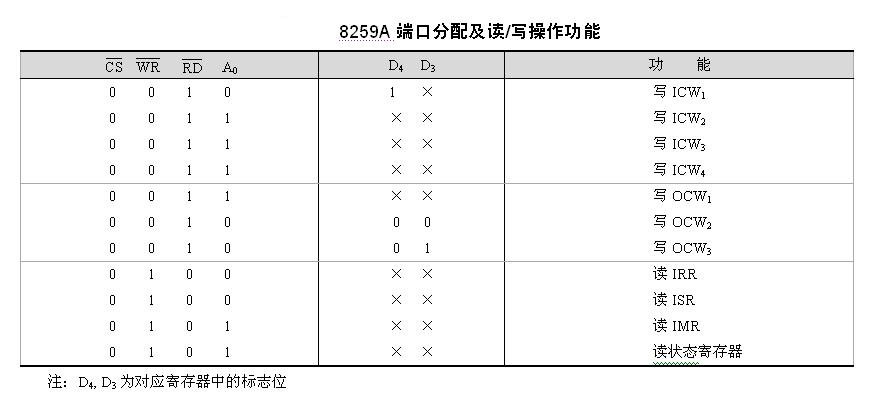
INT（interrupt request）：中断请求信号，与CPU的中断请求信号相连。

IR7～IR0（interrupt requests）：中断请求输入信号，由外设输入。

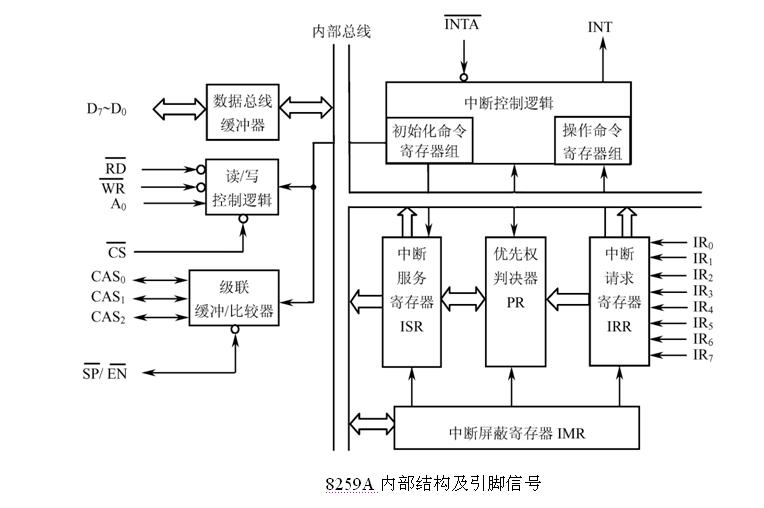
-INTA（interrupt acknowledge）：中断响应信号，与CPU的中断应答信号相连。

VCC：＋5V电源输入信号。　　 GND：电源地

有关寄存器的端口地址分配及读/写操作功能见下表。



其内部结构及引脚信号如图6.9所示。



内部结构

（1）数据总线缓冲器

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

（2）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。

（3）级联缓冲/比较器

　　8259A既可以工作于单片方式，也可以工作于多片级联方式，级联方式硬件连接如图6.10所示。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。

（4）中断控制逻辑

　　中断控制逻辑按照编程设定的工作方式管理中断，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回送的中断响应信号，控制8259A进入中断管理状态。

（5）中断请求寄存器（interrupt request register，IRR）

　　IRR是一个8位寄存器，用于记录外部中断请求。其中D7～D0分别与外部中断请求信号IR7～IR0相对应，当IRi（i＝0～7）有请求（电平或边沿触发）时，IRR中的相应位Di置1，在中断响应信号有效时，Di被清除。

（6）中断服务寄存器（interrupt service register，ISR）

　　ISR是一个8位寄存器，用于记录CPU当前正在服务的中断标志。当外部中断IRi（i＝0～7）的请求得到CPU响应进入服务时，由CPU发来的第一个中断响应脉冲将ISR中的相应位Di（i＝0～7）置1，而ISR的复位则由8259A中断结束方式决定。若定义为自动结束方式，则由CPU发来的第二个中断响应脉冲的后沿将Di复位为0；若定义为非自动结束方式，则由CPU发送来的中断结束命令将其复位。

（7）中断屏蔽寄存器（interrupt mask register，IMR）

　　IMR是一个8位寄存器，用来存放IR7～IR0的中断屏蔽标志。它的8个屏蔽位D7～D0与外部中断请求IR7～IR0相对应，用于控制IRi的请求是否允许进入。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，通过编程设定屏蔽字，可以改变原来的优先级别。

（8）优先权判决器（priority register，PR）

　　优先权判决器对IRR中记录的内容与当前ISR中记录的内容进行比较，并对它们进行排队判优，以便选出当前优先级最高级的中断请求。如果IRR中记录的中断请求的优先级高于ISR中记录的中断请求的优先级，则由中断控制逻辑向CPU发出中断请求信号INT，中止当前的中断服务，进行中断嵌套。如果IRR中记录的中断请求的优先级低于ISR中记录的中断请求的优先级，则CPU继续执行当前的中断服务程序。

8259A的工作方式

8259A的中断管理功能很强，单片可以管理8级外部中断，在多片级联方式下最多可以管理64级外部中断，并且具有中断优先权判优、中断嵌套、中断屏蔽和中断结束等多种中断管理方式。

中断优先权方式

8259A中断优先权的管理方式有固定优先权方式和自动循环优先权方式两种。

1）固定优先权方式

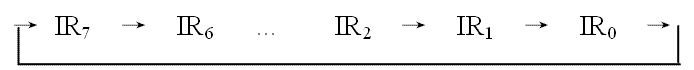
　　在固定优先权方式中，IR7～IR0的中断优先权的级别是由系统确定的。

　　它们由高到低的优先级顺序是：IR0, IR1, IR2, …, IR7, 其中，IR0的优先级最高，IR7的优先级最低。当有多个IRi请求时，优先权判决器（PR）将它们与当前正在处理的中断源的优先权进行比较，选出当前优先权最高的IRi，向CPU发出中断请求INT，请求为其服务。

（2）自动循环优先权方式

　　在自动循环优先权方式中，IR7～IR0优先权级别是可以改变的。

　　其变化规律是：当某一个中断请求IRi服务结束后，该中断的优先权自动降为最低，而紧跟其后的中断请求IR(i＋1)的优先权自动升为最高，IR7～IR0优先权级别按如下所示的右循环方式改变。



　　假设在初始状态IR0有请求，CPU为其服务完毕，IR0优先权自动降为最低，排在IR7之后，而其后的IR1的优先权升为最高，其余依次类推。这种优先权管理方式，可以使8个中断请求都拥有享受同等优先服务的权利。

　　在自动循环优先权方式中，按确定循环时的最低优先权的方式不同，又分为普通自动循环方式和特殊自动循环方式两种。

　　普通自动循环方式的特点是：IR7～IR0中的初始最高优先级由系统指定，即指定IR0的优先级最高，以后按右循环规则进行循环排队。

　　而特殊自动循环方式的特点是：IR7～IR0中的初始最低优先级，由用户通过置位优先权命令指定。

中断嵌套方式

8259A的中断嵌套方式分为完全嵌套和特殊完全嵌套两种。

（1）完全嵌套方式

　　完全嵌套方式是8259A在初始化时自动进入的一种最基本的优先权管理方式．

　　其特点是：中断优先权管理为固定方式，即IR0优先权最高，IR7优先权最低，在CPU中断服务期间（即执行中断服务子程序过程中），若有新的中断请求到来，只允许比当前服务的中断请求的优先权“高”的中断请求进入，对于“同级”或“低级”的中断请求禁止响应。

（2）特殊完全嵌套方式

　　特殊完全嵌套方式是8259A在多片级联方式下使用的一种最基本的优先权管理方式。

　　其特点是：中断优先权管理为固定方式，IR7～IR0的优先顺序与完全嵌套规定相同；与完全嵌套方式不同之处是在CPU中断服务期间，除了允许高级中断请求进入外，还允许同级中断请求进入，从而实现了对同级中断请求的特殊嵌套。

　　在级联方式下，主片通常设置为特殊完全嵌套方式，从片设置为完全嵌套方式。当主片为某一个从片的中断请求服务时，从片中的IR7～IR0的请求都是通过主片中的某个IRi请求引入的。因此从片的IR7～IR0对于主片IRi来说，它们属于同级，只有主片工作于特殊完全嵌套方式时，从片才能实现完全嵌套。

中断屏蔽方式

中断屏蔽方式是对8259A的外部中断源IR7～IR0实现屏蔽的一种中断管理方式，有普通屏蔽方式和特殊屏蔽方式两种。

（1）普通屏蔽方式

　　普通屏蔽方式是通过8259A的中断屏蔽寄存器（IMR）来实现对中断请求IRi的屏蔽。由编程写入操作命令字OCW1，将IMR中的Di位置1，以达到对IRi（i＝0～7）中断请求的屏蔽。

（2）特殊屏蔽方式

　　特殊屏蔽方式允许低优先级中断请求中断正在服务的高优先级中断。这种屏蔽方式通常用于级联方式中的主片，对于同一个请求IRi上连接有多个中断源的场合，可以通过编程写入操作命令字OCW3来设置或取消。

　　在特殊屏蔽方式中，可在中断服务子程序中用中断屏蔽命令来屏蔽当前正在处理的中断，同时可使ISR中的对应当前中断的相应位清0，这样一来不仅屏蔽了当前正在处理的中断，而且也真正开放了较低级别的中断请求。

　　在这种情况下，虽然CPU仍然继续执行较高级别的中断服务子程序，但由于ISR中对应当前中断的相应位已经清0，如同没有响应该中断一样。所以，此时对于较低级别的中断请求，8259A仍然能产生INT中断请求，CPU也会响应较低级别的中断请求。

中断结束方式

中断结束方式是指CPU为某个中断请求服务结束后，应及时清除中断服务标志位，否则就意味着中断服务还在继续，致使比它优先级低的中断请求无法得到响应。中断服务标志位存放在中断服务寄存器（ISR）中，当某个中断源IRi被响应后，ISR中的Di位被置1，服务完毕应及时清除。

8259A提供了以下三种中断结束方式：

（1）自动结束方式

　　自动结束方式是利用中断响应信号INTA¯¯¯¯¯¯¯¯¯¯的第二个负脉冲的后沿，将ISR中的中断服务标志位清除。

　　这种中断服务结束方式是由硬件自动完成的，需要注意的是：ISR中为“1”位的清除是在中断响应过程中完成的，并非中断服务子程序的真正结束，若在中断服务子程序的执行过程中有另外一个比当前中断优先级低的请求信号到来，因8259A并没有保存任何标志来表示当前服务尚未结束，致使低优先级中断请求进入，打乱正在服务的程序，因此这种方式只适合用在没有中断嵌套的场合，即，这种中断方法仅仅适用于单级中断。

（2）普通结束方式 (非自动中断结束方式EOI)

　　普通结束方式是通过在中断服务子程序中编程写入操作命令字OCW2，向8259A传送一个普通EOI（end of interrupt）命令（不指定被复位的中断的级号）来清除ISR中当前优先级别最高位。

　　由于这种结束方式是清除ISR中优先权级别最高的那一位，适合使用在完全嵌套方式下的中断结束。因为在完全嵌套方式下，中断优先级是固定的，8259A总是响应优先级最高的中断，保存在ISR中的最高优先级的对应位，一定对应于正在执行的服务程序。

（3）特殊结束方式 (非自动中断结束方式EOI)

　　特殊结束方式是通过在中断服务子程序中编程写入操作命令字OCW2，向8259A传送一个特殊EOI命令（指定被复位的中断的级号）来清除ISR中的指定位。

　　如果采用了循环优先级方式工作，则中断源的优先级在动态变化，且此时的正在服务的中断服务程序的优先级可能不是最高的，因此，必须使用指定中断结束命令来明确ISR中的待清理的位置。

　　由于在特殊EOI命令中明确指出了复位ISR中的哪一位，不会因嵌套结构出现错误。因此，它可以用于完全嵌套方式下的中断结束，更适用于嵌套结构有可能遭到破坏的中断结束。

中断触发方式

8259A中断请求输入端IR7～IR0的触发方式有电平触发和边沿触发两种，由初始化命令字ICW1中的LTIM位来设定。

当LTIM设置为1时，为电平触发方式，8259A检测到IRi（i＝0～7）端有高电平时产生中断。在这种触发方式中，要求触发电平必须保持到中断响应信号有效为止，并且在CPU响应中断后，应及时撤销该请求信号，以防止CPU再次响应，出现重复中断现象。

当LTIM设置为0时，为边沿触发方式，8259A检测到IRi端有由低到高的跳变信号时产生中断。

总线连接方式

8259A数据线与系统数据总线的连接有缓冲和非缓冲两种方式。

　（1）缓冲方式

　　如果8259A通过总线驱动器和系统数据总线连接，此时，8259A应选择缓冲方式，SP¯¯¯¯¯/EN¯¯¯¯¯¯引脚作为输出端，用于精致外接总线驱动器输入/输出的方向。当定义为缓冲方式后，即为输出引脚。在8259A输出中断类型号的时候，输出一个低电平，用此信号作为总线驱动器的启动信号。

　（2）非缓冲方式

　　8529A的数据总线具有双向、三态缓冲功能，因此，它不需要外加总线驱动器就可直接挂到数据总线上使用。总线驱动器是起缓冲和隔离作用，所以这种连接方式成为非缓冲方式。在这种方式下，SP¯¯¯¯¯/EN¯¯¯¯¯¯引脚为输入引脚，在级联方式下，用高/低电平通知8259A工作在主/从片方式下。

8259A初始化编程

在8259A内部有两组寄存器，一组为命令寄存器，用于存放CPU写入的初始化命令字ICW1～ICW4（initialization command words）；另一组为操作命令寄存器，用于存放CPU写入的操作命令字OCW1～OCW3（operation command words）。

初始化命令字ICW的格式

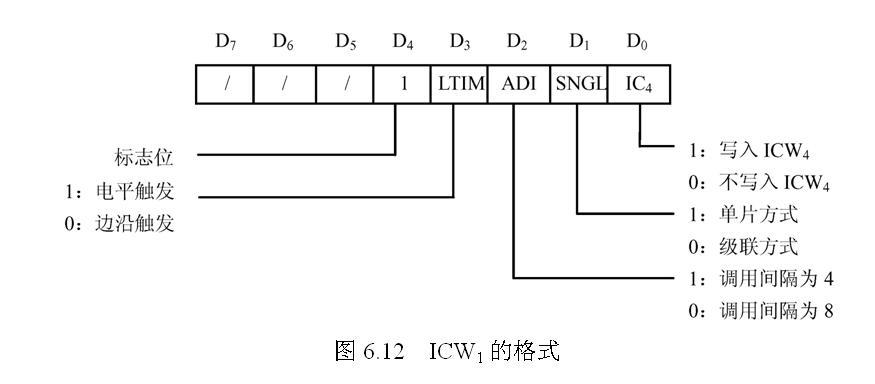
当地址线A0为1时，8259A提供了4个（ICW1～ICW4）初始化命令字，并规定了严格的初始化步骤。

8259A是中断系统的核心器件，对它的初始化编程要涉及中断系统的软、硬件的许多问题，而且一旦完成初始化，所有硬件中断源和中断处理程序都必须受其制约。

（1）ICW1的格式

用于触发方式和级联方式的设置，规定写入偶地址端口，即A0=0的端口。

　　ICW1的格式如图所示。



这里写图片描述

IC4 （ICW4 needed /no ICW4 needed）：指示在初始化时是否需要写入命令字ICW4。在80x86 CPU系统中需要定义ICW4，设IC4＝1。

SNGL（single/cascade mode）：指示8259A在系统中使用单片还是多片级联。SNGL＝1为单片，SNGL＝0为多片级联。

ADI（call address interval）：设置调用时间间隔，在80×86 CPU中无效，一般设为0。

LTIM（level/edge triggered mode）：定义IRi的中断请求触发方式。LTIM＝1为电平触发，LTIM＝0为边沿触发。

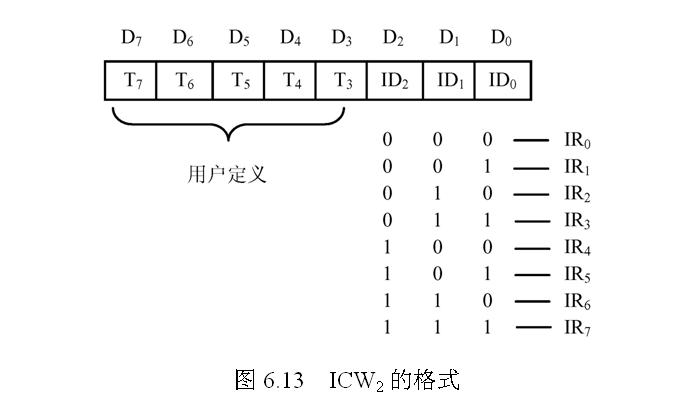
D4：ICW1的标志位，恒为1。

D5～D7：未用，通常设置为0。

（2）ICW2的格式

ICW2用于设置中断类型号，格式如下图所示。

这里写图片描述



ICW2中的低3位ID2～ID0由中断请求输入端IRi（i＝0～7）的编码自动引入，高5位T7～T3由用户编程写入。若ICW2写入40H时，则IR0～IR7对应的中断类型号为40H～47H。

（3）ICW3的格式

ICW3是级联命令字，在级联方式下才需要写入。ICW3规定写入奇地址。主片和从片所对应的ICW3的格式不同，主片ICW3的格式如图6.14所示，从片ICW3的格式如图6.15所示。

这里写图片描述

主片ICW3的格式

这里写图片描述

这里写图片描述

从片ICW3的格式

这里写图片描述

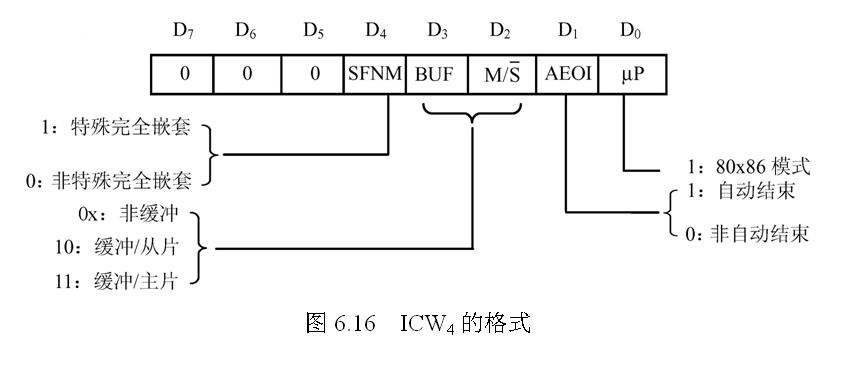
S7～S0与IR7～IR0相对应，若主片IRi（i＝0～7）引脚上连接从片，则Si＝1，否则Si＝0。

ID2～ID0是从片接到主片IRi上的标识码。例如，当从片的中断请求信号INT与主片的IR2连接时，ID2～ID0应设置为010，D7～D3未用，通常设置为0。

在中断响应时，主片通过级联信号线CAS2～CAS0送出被允许中断的从片的标识码，各从片用自己的ICW3和CAS2～CAS0进行比较，二者一致的从片被确定为当前中断源，可以发送该从片的中断类型码。

（4）ICW4的格式

ICW4用于设定8259A的结束中断方式、缓冲方式和嵌套模式，规定写入奇地址。其格式如图6.16所示.



这里写图片描述

μP（microprocessor）：设置CPU模式。μP＝1为80x86模式，μP＝0为8080/8085模式。

AEOI（auto end of interrupt）：设置8259A的中断结束方式。AEOI＝1为自动结束方式，AEOI＝0为非自动结束方式。

M/S¯¯¯（master/slave）：选择缓冲级联方式下的主片与从片。M/S¯¯¯＝1为主片，INTA¯¯¯¯¯¯¯¯¯¯＝0为从片。

BUF（buffer）：设置缓冲方式。BUF＝1为缓冲方式，BUF＝0为非缓冲方式。

SFNM（special fully nested mode）：设置特殊完全嵌套方式。SFNM＝1为特殊完全嵌套方式（同级别的中断也可以打断），SFNM＝0为非特殊完全嵌套方式

D7～D5：未定义，通常设置为0。

需要注意：当多片8259A级联时，若在8259A的数据线与系统总线之间加入总线驱动器，SP¯¯¯¯¯/EN¯¯¯¯¯¯引脚作为总线驱动器的控制信号，D3位BUF应设置为1，此时主片和从片的区分不能依靠SP¯¯¯¯¯/EN¯¯¯¯¯¯引脚，而是由M/S¯¯¯来选择，当M/S¯¯¯＝0时为从片；当M/S¯¯¯＝1时为主片。如果BUF＝0，则M/S¯¯¯定义无意义。

操作命令字OCW的格式

操作命令字有OCW1, OCW2和OCW3。

OCW1的格式

OWC1为中断屏蔽字，规定写入奇地址，写入中断屏蔽寄存器（IMR）中，对外部中断请求信号IRi实行屏蔽。

这里写图片描述

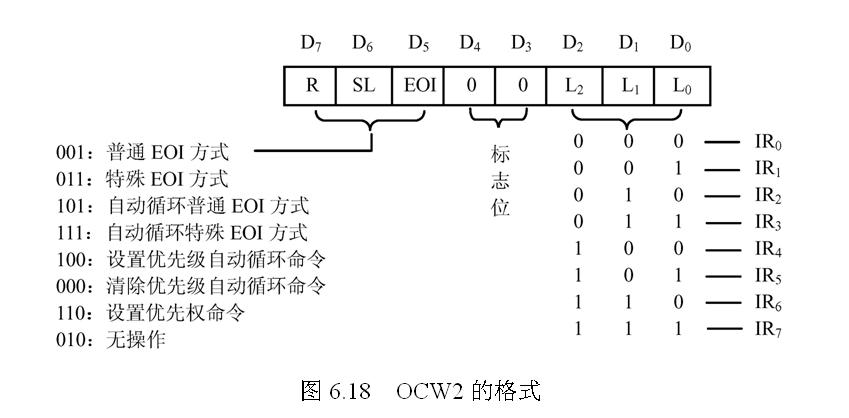
这里写图片描述

当某位Mi（interrupt mask）为1时，则对应的IRi请求被禁止；当Mi为0时，则对应的IRi请求被允许。在工作期间可根据需要随时写入或读出。

OCW2的格式

OCW2用于设置中断优先级方式和中断结束方式，其格式如图6.18所示。

这里写图片描述



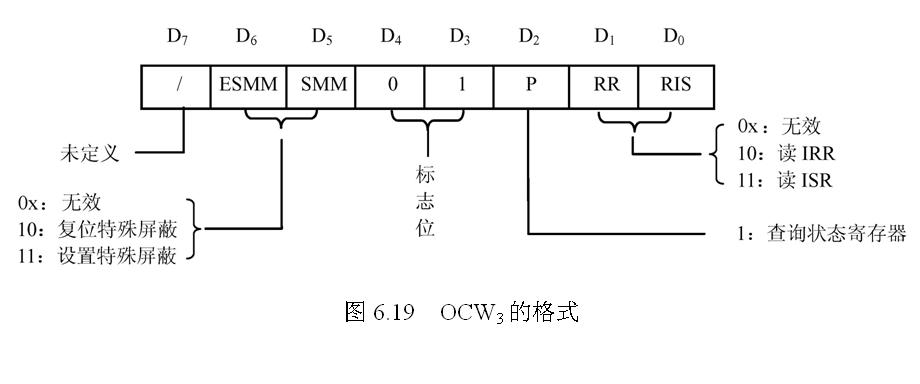
L2～L0（IR level to be acted upon）：8个中断请求输入端IR7～IR0的标志位，用来指定中断级别。L2～L0指定的中断级别是否有效，由SL（specific level）位控制。当SL＝1时，L2～L0定义有效；当SL＝0时，L2～L0定义无效。

EOI（end of interrupt）：中断结束命令。若EOI＝1时，在中断服务子程序结束时向8259A回送中断结束命令EOI，以便使中断服务寄存器（ISR）中当前最高优先权位复位（普通EOI方式），或由L2～L0表示的优先权位复位（特殊EOI方式）。

R（rotation）：设置优先权循环方式位。R＝1为优先权自动循环方式；R＝0为优先权固定方式。D4,D3为OCW2标志位。

OCW3的格式

OCW3用于设置或清除特殊屏蔽方式和读取寄存器的状态，规定写入偶地址。格式如图6.19所示。



这里写图片描述

RR（read register command）：读ISR和IRR命令位，RIS（read interrupt register select）读寄存器选择位。当RR＝1，RIS＝0时，读取IRR命令；当RR＝1，RIS＝1时，读取ISR命令。在进行读ISR或IRR操作时，先写入读命令OCW3，然后紧接着执行读ISR或IRR的指令。

而读取IMR寄存器的内容，不需要事先发出OCW3指定命令，直接通过读奇地址端口就可以。

例如，设8259A的两个端口地址为20H和21H，OCW3, ISR和IRR共用一个地址20H。

读取ISR内容的程序段为

MOV　　　　AL, 00001011B

OUT　　　　20H, AL　　　 ; 读ISR命令写入OCW3

IN　　　　AL, 20H　　 ; 读ISR内容至AL中读取IRR内容的程序段为

MOV　　　　AL, 00001010B

OUT　　　　20H, AL ; 读IRR命令写入OCW3

IN　　　　 AL, 20H ; 读IRR内容至AL中

1

2

3

4

5

6

1

2

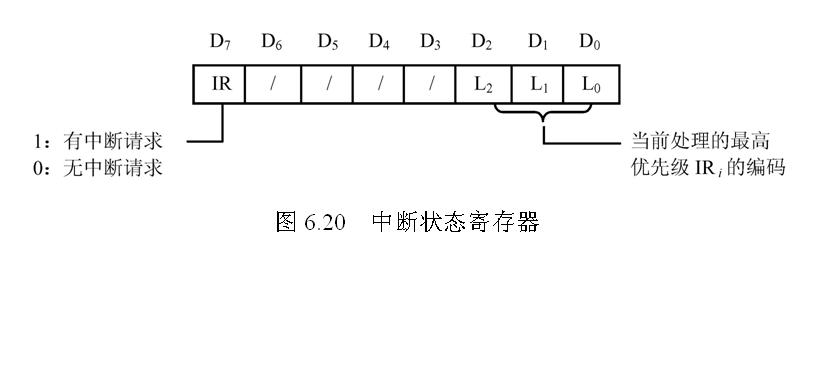
3

4

5

6

P（poll command）：为中断状态查询位。当P＝1时，可通过读入状态寄存器的内容，查询是否有中断请求正在被处理，如有则给出当前处理中断的最高优先级。中断状态寄存器如图6.20所示。



这里写图片描述

在读取中断状态字时，先写入中断查询命令，然后读取中断状态字，程序如下：

　　MOV　　　　AL, 00001111B

　　OUT　　　　20H, AL 　　　　; 读中断状态字命令写入OCW3

　　IN　　　　 AL, 20H 　　　　; 读中断状态字

1

2

3

1

2

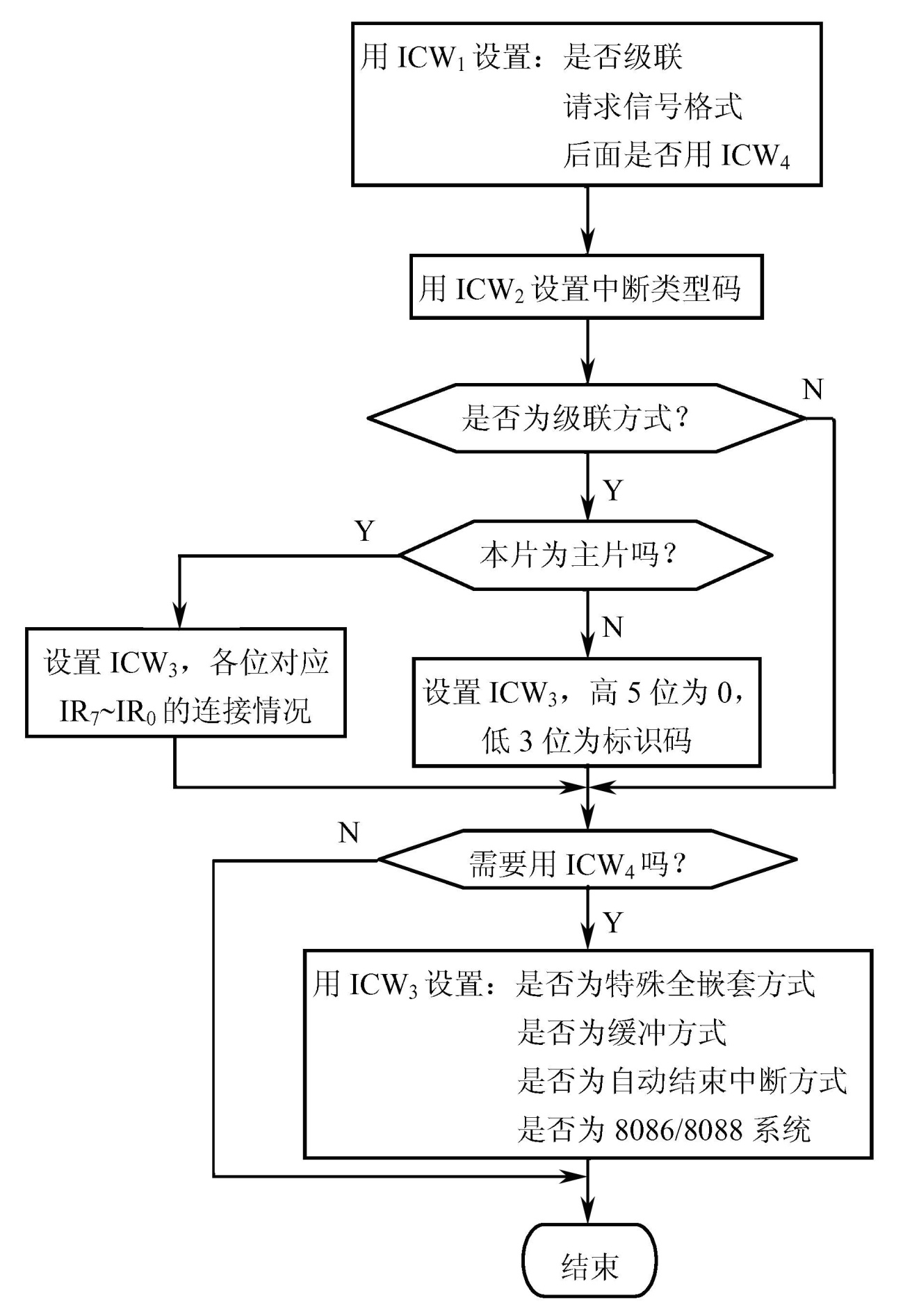
3

ESMM（enable special mask mode）与SMM（special mask mode）组合可用来设置或取消特殊屏蔽方式。当ESMM＝1，SMM＝1时，设置特殊屏蔽；当ESMM＝1，SMM＝0时，取消特殊屏蔽。

8259A的初始化编程

8259A的初始化编程需要写入初始化命令字ICW1～ICW4，对它的连接方式、中断触发方式和中断结束方式进行设置。但由于ICW1～ICW4使用两个端口地址，即ICW1用A0＝0的端口，ICW2～ICW4使用A0＝1的端口，因此初始化程序应严格按照系统规定的顺序写入，即先写入ICW1，接着写ICW2, ICW3, ICW4。

　　操作命令字OCW1～OCW3的写入比较灵活，没有固定的格式，可以在主程序中写入，也可以在中断服务子程序中写入，视需要而定。下面通过例子来说明如何编写8259A的初始化程序。



这里写图片描述

【例6.1】　 某微机系统使用主、从两片8259A管理中断，从片中断请求INT与主片的IR2连接。设主片工作于特殊完全嵌套、非缓冲和非自动结束方式，中断类型号为40H，端口地址为20H和21H。从片工作于完全嵌套、非缓冲和非自动结束方式，中断类型号为70H，端口地址为80H和81H。试编写主片和从片的初始化程序。

　　根据题意，写出ICW1, ICW2, ICW3和ICW4的格式，按图6.21的顺序写入。编写初始化程序如下：

主片8259A的初始化程序如下：

　　MOV　　　AL, 00010001B　　　　　　　　; 级联, 边沿触发, 需要写ICW4

　　OUT　　　20H, AL　　　　　　　　　　　; 写ICW1

　　MOV　　　AL, 01000000B　　　　　　　　; 中断类型号40H

　　OUT　　　21H, AL　　　　　　　　　　　; 写ICW2

　　MOV　　　AL, 00000100B　　　　　　　　; 主片的IR2引脚接从片

　　OUT　　　21H, AL　　　　　　　　　　　; 写ICW3

　　MOV　　　AL, 00010001B　　　　　　　　; 特殊完全嵌套、非缓冲、自动结束

　　OUT　　　21H, AL　　　　　　　　　　　; 写ICW4

从片8259A初始化程序如下：

　　MOV　　　AL, 00010001B　　　　　　　　; 级联, 边沿触发, 需要写ICW4

　　OUT　　　80H, AL　　　　　　　　　　　; 写ICW1

　　MOV　　　AL, 01110000B　　　　　　　　; 中断类型号70H

　　OUT　　　81H, AL　　　　　　　　　　　; 写ICW2

　　MOV　　　AL, 00000010B　　　　　　　　; 接主片的IR2引脚

　　OUT　　　81H, AL　　　　　　　　　　　; 写ICW3

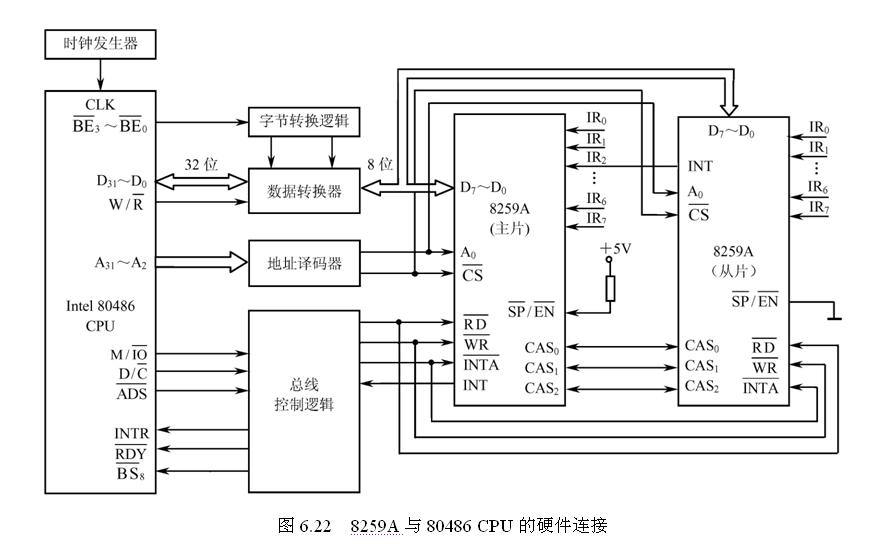
　　MOV　　　AL, 00000001B 　　　　　　 　; 完全嵌套、非缓冲、非自动结束

　　OUT　　　81H, AL 　　　　　　　　　 　; 写ICW4

在Intel 80486 CPU系统中，使用2片8259A管理中断，采用级联方式。主片中的8个中断请求IR7～IR0除IR2扩展从片以外，其他均为系统使用，从片中的8个中断请求IR7～IR0供用户使用。

硬件连接如图6.22所示。

这里写图片描述



8259A的硬件连接比较简单，不需要附加其他电路就可以与CPU直接连接。

系统的工作过程是：当80486 CPU响应8259A的中断请求INT时，向8259A回送中断响应信号INTA¯¯¯¯¯¯¯¯¯¯，自动进入中断响应总线周期，进行中断处理。

INTA¯¯¯¯¯¯¯¯¯¯是由INTA¯¯¯¯¯¯¯¯¯¯，D/C¯¯¯和W/R¯¯¯信号通过总线控制逻辑产生，其编码见表2.6，当M/IO¯¯¯¯¯,D/C¯¯¯和W/R¯¯¯的编码是000时，为中断响应周期。INTA¯¯¯¯¯¯¯¯¯¯时序如图6.23所示：

这里写图片描述

在第一个INTA¯¯¯¯¯¯¯¯¯¯周期，CPU向8259A发送第一个INTA¯¯¯¯¯¯¯¯¯¯脉冲，表示响应INT请求，并利用这个信号将8259A的请求信号IRi复位，同时使总线锁存信号LOCK¯¯¯¯¯¯¯¯¯¯有效，以禁止其他总线主控设备争用总线。

在第二INTA¯¯¯¯¯¯¯¯¯¯周期，8259A将中断类型号n通过数据总线送至CPU，由CPU自动完成向量表地址4n运算后，从中断向量表中取出相应的中断向量送入CS:IP中，继而转去执行中断服务子程序。

同时，还利用第二个INTA¯¯¯¯¯¯¯¯¯¯周期的后沿将INT复位。在自动结束方式中，也是利用这个后沿，将ISR中的中断服务标志位清除。

例6.2】　设计一个中断处理程序。要求中断请求信号以跳变方式由IR2引入（可为任一定时脉冲信号），当CPU响应IR2请求时，输出字符串“8259A INTERRUPT!”，中断10次，程序退出（设8259A的端口地址为20H和21H，中断类型号为40H）。

　　中断处理程序如下：

DATA　　　SEGMENT

　　MESS　　　DB'8259A? INTERRUPT!', 0AH, 0DH, '$'

　　COUNT　　 DB 　10　　　　　　　　　; 计数值为10

　　DATA　　　ENDS

　　STACK　　 SEGMENT　STACK

　　STA　　　 DB　100H DUP(?)

　　TOP 　　　EQU　LENGTH　　STA

　　STACK　　 ENDS

　　CODE 　　 SEGMENT

　　ASSUME　　CS:CODE, DS:DATA, SS:STACK

　　MAIN:　　 CLI

　　MOV　　　 AX,DATA

　　NOV 　　　DS, AX

　　MOV　　　 AX, STACK

　　MOV 　　　SS, AX

　　MOV　 　　SP, TOP

　　MOV　　 　AL, 13H 　　　　　　　 ; 8259A初始化

　　OUT　　　 20H, AL　　　　　　　　 ; 单片, 边沿触发

　　MOV 　　　AL, 40H　　　　　　　　 ; 中断类型号40H

　　OUT　　　 21H, AL

　　MOV 　　　AL, 01H　　　　　　　　 ; 非自动结束

　　OUT　　　 21H, AL

　　MOV 　　　AX, SEG INT-P　　　　　 ; 设置中断向量

　　MOV　　　 DS, AX 　　　　　　　　 ; 中断服务子程序入口段基址送DS

　　MOV　　　 DX, OFFSET INT-P 　　　 ; 中断服务子程序入口偏移地址送DX

　　MOV　　　 AL, 42H 　　　　　　　　; IR2的中断类型号42H送AL

　　MOV　　　 AH, 25H 　　　　　　　　; 25H功能调用

　　INT 　　　21H

　　IN　　　　AL, 21H　　　　　　　　 ; 读IMR

　　AND　　　 AL, 0FBH　　　　　　　　; 允许IR2请求中断

　　OUT 　　　21H, AL 　　　　　　　　; 写中断屏蔽字OCW1

　　WAIT1:　　STI 　　　　　　　　　　; 开中断

　　CMP　　　 COUNT,0 　　　　　　　　; 判断10次中断是否结束

　　JNZ 　　　WAIT1 　　　　　　　　　; 未结束, 等待

　　MOV　　　 AX, 4C00H　　　　　 　　; 结束, 返回DOS

　　INT 　　　21H

　　INT-P　　 PROC　　　　　　 　 ; 中断服务子程序

　　PUSH 　　 DS　　　　　　　　 ; 保护现场

　　PUSH 　 　AX

　　PUSH 　　 DX

　　STI　　　　　　　　　　　　　　 　; 开中断

　　MOV　　　 DS, AX

　　MOV　　　 DX, OFFSET MESS

　　MOV 　　　AH, 09H

　　INT　　　 21H

　　DEC 　　　COUNT　　　　　　　 　; 控制10次循环

　　JNZ　　　 NEXT

　　IN 　　　 AL, 21H　　　　　　　　 ; 读IMR

　　OR 　　　 AL, 04H　　　　　　 ; 屏蔽IR2请求

　　OUT 　　　21H, AL

　　NEXT:　　 CLI　　　　　　　　　　 ; 关中断

　　MOV 　　　AL, 20H　　　　　　 ; 写OCW2, 送中断结束命令EOI

　　OUT　　　 20H, AL

　　POP 　　　DX　　　　　　　　　　 ; 恢复现场

　　POP　　　 AX

　　POP 　　　DS

　　IRET 　　　　　　　　　　　　　　 ; 中断返回

　　INT-P　　 ENDP

　　CODE　　　ENDS

　　END　　　 MAIN