

7.1对输入和输出操作进行硬件和软件的控制就是所谓的输入/输出控制,即I/O控制.I/O控制不仅要使外设和主机联系起来,构成一个"系统",而且要使系统具有较高的吞吐能力和工作效率。

7.2在程序终端控制方式中,外部设备在完成了数据传送的准备工作后,主动向CPU提出传送请求,CPU暂停原执行的程序,转向信息交换服务.在这种方式下,CPU的效率得以提高,这是因为设备在数据传送准备阶段时,CPU仍在执行原程序;此外,CPU不再向程序直接控制方式下那样被状态外设独占,它可以同时与多台设备进行数据传送,与程序查询方式相同的是在信息传送阶段,CPU仍要执行一段控制程序,还没有完全摆脱I/O操作的具体管理。

7.3通道方式利用了DMA技术,再加上软件,形成一种新的控制方式.通道是一种简单的处理机,它有指令系统,能执行程序,它的独立工作的能力比DMA强,能对多台不同类型的设备统一管理,对多个设备同时传送信息.而外围处理机的结构更接近于一般的处理机,它可完成I/O通道所要完成的I/O控制,还可完成码制变幻,格式处理,数据块的检错,纠错等操作,因而可以说外微处理机结构就是一般的小型计算机。

7.4外设有两种编址方法,一种是统一编址,一种是单独编址.所谓统一编址法,就是将I/O设备中的控制寄存器,数据寄存器等等也和内存单元一样看待,将它们和内存单元一起编排地址,它的特点就是访问内存单元的指令就可以访问I/O设备的某个

寄存器,不需要专门的I/O指令组.而在单独编址的方法中,内存单元与I/O设备的地址是分开的,访问内存单元与访问I/O设备的指令是不同的,有两套指令系统。

7.5外设采用程序中断方式传送数据时分为以下10个步骤:

- 1.程序启动外设,将该外设接口的"忙"标志BS置"1","准备就绪"标志RD清"0";
- 2.接口向外设发出启动信号;
- 3.数据由外设传送到接口的缓存寄存器;
- 4.当外设动作结束或缓冲寄存器数据填满时,设备向接口送出以控制信号,将数据"准备就绪"标志RD置"1";
- 5.当允许中断标志EI为"1"时,接口向CPU发出中断请求信号;
- 6.在一条指令执行末尾CPU检查中断请求线,将中断请求线的请求信号送到中断请求触发器IR;

7.如果中断屏蔽触发器IM为"0",则CPU在一条指令结束后受理外设的中断请求,向外设发出相应中断信号并关闭中断;

- 8.然后转向该设备的中断服务程序入口;
- 9.中断服务程序用输入指令把接口中数据缓冲寄存器的数据读至CPU中的累加器或寄存器中;
- 10.最后CPU发出控制信号C将接口中的BS和RD标识复位,一次数据传送结束。

采用程序中断方式的接口应由准备就绪标志触发器,允许中断触发器,忙标志触发器,数据缓冲寄存器,中断向量逻辑,和设备选择等组成。

其程序中断控制方式中的接口的示意图见p241的图7.15。

7.6所谓中断判优就是判别中断条件,确定中断源,并转入相应的中断服务程序入口,这也是中断处理首先要解决的问题.大致有三种不同的方法。

1.查询法:这是最简单的实现方法,在这种查询方式下,CPU首先转向固定的中断查询程序入口,执行该程序,可以确定相应的中断请求,查询的顺序决定了设备中断优先权.这种软件查询方法是用于低速和中速设备.它的优点是中断条件标志的优先级可用程序任意改变,灵活性好,缺点是设备多时速度慢。

#### 2.串行排队链法

与向量中断:该方法是由硬件实现的具有公共请求线的判优选择方式.向量中断方式是为每一个中断源设置一个中断向量的方式,中断向量包括了该中断源的中断服务程序入口地址.它完全由硬件直接产生中断响应信号,经过中断排队和编码逻辑,又被选中的设备直接送回中断向量。

3.独立请求法:该判优方法中具有较高优先权的中断请求自动封锁比它优先权低的所有中断请求,该方法速度快,但连线多,逻辑电路复杂。

7.7

中断请求、中断允许、禁止和中断的响应都是由硬件实现的。中断处理过程中，关中断、保存断点一般由硬件实现，判别中断条件所用的串行排队连法也由硬件实现。

执行中断服务程序由软件完成。

保存现场可以由硬件完成，也可以由软件完成。

## 7.8

DMA方式中的中断称为简单中断,在DMA方式的I/O过程中,主机响应中断后不要执行服务程序,而是让出一个或几个存取周期供I/O设备与主存直接交换数据,此时,CPU可以暂停运行,也可以执行非访问内存储器操作。

DMA传送与中断传送相比有以下不同点:1.中断传送需要保存CPU现场并执行中断服务程序,时间开销较大,而DMA由硬件实现,不需要保存CPU现场,时间开销较小;2.中断传送只能在一个指令周期结束后进行,而DMA传送则可以在两个及其周期之间进行。

## 7.9

采用DMA方式传送一批数据需要程序中断。因为在DMA方式中，当数据传送完毕时，DMA控制器向CPU报告DMA操作结束，CPU会以中断方式响应DMA结束请求，并由CPU在中断程序中进行结束后的处理工作。

DMA接口由以下几个部分组成：

- (1) 内存地址计数器。保存内存数据缓冲的首地址。
- (2) 字计数器。每完成一个字或字节的传送后，该计数器减1，计数器为0时，传送结束，发一个信号到中断机构。
- (3) 中断机构。可向CPU提出中断请求。
- (4) 控制/状态逻辑。
- (5) 数据缓冲寄存器。
- (6) DMA请求标志。

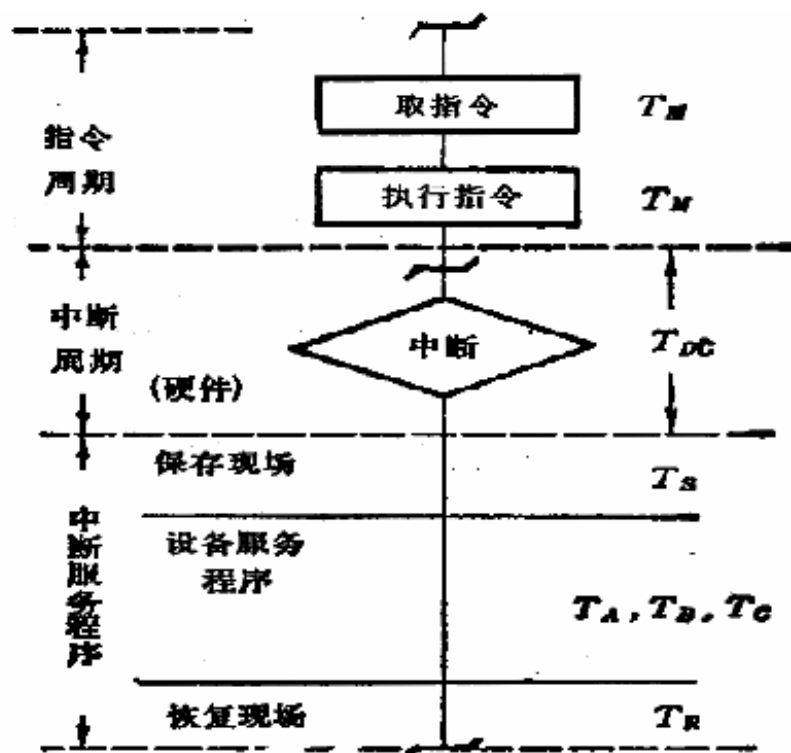
## 7.10

当CPU对设备B的中断请求进行服务时，如果设备A提出请求，CPU不能响应。因为在同一中断级中，执行某一中断源的中断服务程序不允许其它中断源打断，即使是优先权高的中断源也不允许打断它。

要使设备B的中断请求及时得到响应。可将设备B从第2级取出来，单独放在第3级上，使第3级的优先级最高即可。

## 7.11

假设执行一条指令的时间也为 $t_1$ ，恢复现场所需的时间也为 $t_2$ 。中断处理过程和各个时间段如下图所示：



当三个设备同时发出中断请求时，分别处理设备A、D、G的时间如下：

$$t_A = 2t_1 + t_4 + t_2 + T_A + t_2$$

$$t_D = 2t_1 + t_4 + t_2 + T_D + t_2$$

$$t_G = 2t_1 + t_4 + t_2 + T_G + t_2$$

处理三个设备所需的总时间为  $T = t_A + t_D + t_G$

因此达到中断饱和的最小时间为  $T$ ，即中断极限频率为  $f = 1/T$ 。

7.12 逻辑图如图所示：

