

第九章 输入/输出系统

2. 简单回答下列问题。(参考答案略)

- (1) 什么是接口？其基本功能有哪些？按数据传送方式分有哪两种接口类型？
- (2) 串行接口和并行接口的特点各是什么？
- (3) CPU 如何进行设备的寻址？I/O 端口的编址方式有哪两种？各有何特点？
- (4) 什么是程序查询 I/O 方式？说明其工作原理。
- (5) 什么是中断 I/O 方式？说明其工作原理。
- (6) 什么叫向量中断？说明在向量中断方式下形成中断向量的基本方法。
- (7) 对于向量中断，为什么 I/O 模块把中断向量放在总线数据线上而不是放在地址线上？
- (8) 在多周期处理器中并不是每个时钟周期后都允许响应中断。为什么？如果在一条指令执行过程中，CPU 为了响应中断而停止操作，会产生什么问题？
- (9) 什么是可屏蔽中断？什么是非屏蔽中断？
- (10) 为什么在保护现场和恢复现场的过程中，CPU 必须关中断？
- (11) DMA 方式能够提高成批数据交换效率的主要原因何在？
- (12) CPU 响应 DMA 请求和响应中断请求有什么区别？为什么通常使 DMA 请求具有高于中断请求的优先权？
- (13) 在 DMA 接口中，什么时候给出“DMA 请求”（或“总线请求”）信号？什么时候给出“中断请求”信号？CPU 在什么时候响应 DMA 请求？在什么时候响应中断请求？

3. 假设某计算机带有 20 个终端同时工作，在运行用户程序的同时，能接受来自任意一个终端输入的字符信息，并将字符回送显示（或打印）。每一个终端的键盘输入部分有一个数码缓冲寄存器 $RDBR_i (i=1\sim 20)$ ，当在键盘上按下某一个键时，相应的字符代码即进入 $RDBR_i$ ，并使它的“完成”状态标志 $Done_i (i=1\sim 20)$ 置 1，要等处理器把该字符代码取走后， $Done_i$ 标志才置 0。每个终端显示（或打印）输出部分也有一个数码缓冲寄存器 $TDBR_i (i=1\sim 20)$ ，并有一个 $Ready_i (i=1\sim 20)$ 状态标志，该状态标志为 1 时，表示相应的 $TDBR_i$ 是空着的，准备接收新的输出字符代码，当 $TDBR_i$ 接收了一个字符代码后， $Ready_i$ 标志才置 0，并送到终端显示（或打印），为了接收终端的输入信息，处理器为每个终端设计了一个指针 $PTR_i (i=1\sim 20)$ 指向为该终端保留的主存输入缓冲区。

处理器采用下列两种方案输入键盘代码，同时回送显示（或打印）。

- ① 每隔一固定时间 T 转入一个状态检查程序 $DEVCHC$ ，顺序地检查全部终端是否有任何键盘信息要输入，如果有，则顺序完成之。
- ② 允许任何有键盘信息输入的终端向处理器发出中断请求。全部终端采用共同的向量地址，利用它使处理器在响应中断后，转入一个中断服务程序 $DEVINT$ ，由后者询问各终端状态标志，并为最先遇到的请求中断的终端服务，然后转向用户程序。

要求画出 $DEVCHC$ 和 $DEVINT$ 两个程序的流程图。

参考答案（参见课程网站中第八章的例题 10）

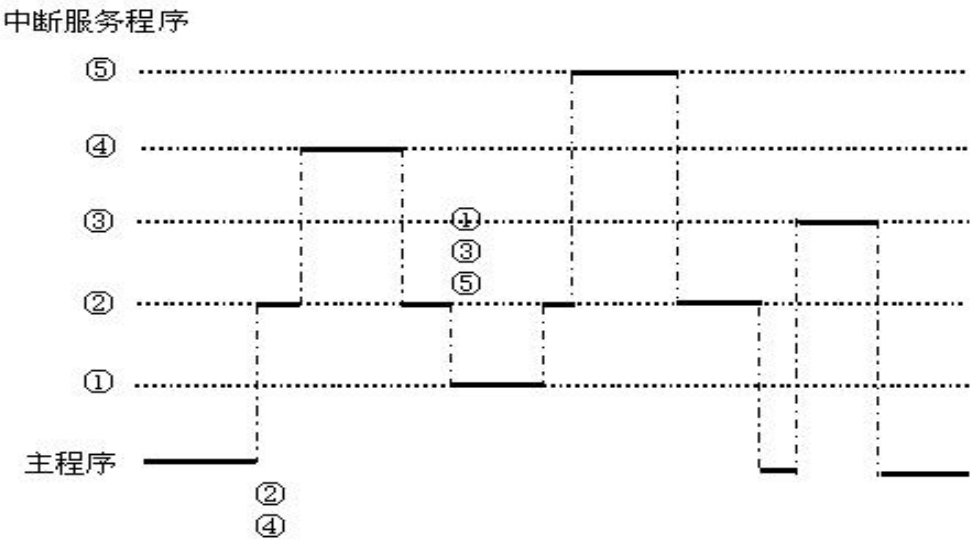
4. 若某计算机有 5 级中断，中断响应优先级为 $1>2>3>4>5$ ，而中断处理优先级为 $1>4>5>2>3$ 。
- 要求：
- ① 设计各级中断处理程序的中断屏蔽位(令 1 为屏蔽，0 为开放)；
 - ② 若在运行主程序时，同时出现第 4、2 级中断请求，而在处理第 2 级中断过程中，又同时出现 1、5、3 级中断请求，试画出此程序运行过程示意图。

参考答案：

中断程序级别	屏蔽字				
	1 级	2 级	3 级	4 级	5 级
第 1 级	1	1	1	1	1
第 2 级	0	1	1	0	0
第 3 级	0	0	1	0	0
第 4 级	0	1	1	1	1
第 5 级	0	0	0	1	1

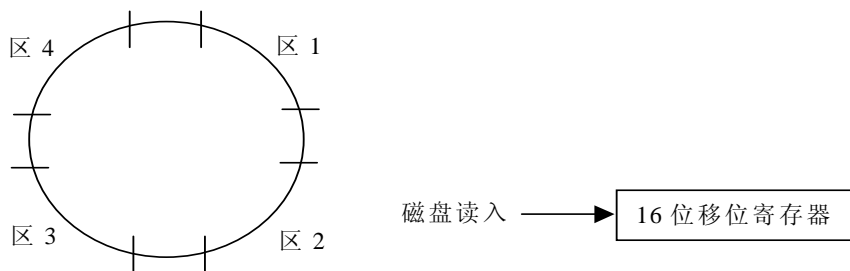
(假定 1 是屏蔽，0 是开放)

上述第 5 级中断的屏蔽字是错误的。



5. 设有一磁盘面共有磁道 200 道，盘面总存储容量为 1.6 兆字节，磁盘旋转一周时间 25ms，每道有 4 个区，每两个区之间有一个间隙，磁头通过每个间隙需 1.25ms。问：该磁盘通道所需最大数据传输率是多少（字节/秒）？假如有人为上述磁盘机设计了一个与计算机之间的接口，如下图所示，磁盘读出串行送入一个移位寄存器，每当移满 16 位后向处理器发出一个请求交换数据的信号。处理器响应请求交换的信号。并取走寄存器的内容后，磁盘机读出再串行送入下一个 16 位的字，如此继续工作，如果现在已知处理器在接到请求交换的信

号以后，最长响应时间是 3 微妙，这样的接口设计能否正确工作？应如何改进？



参考答案：

每个磁道的存储容量： $1.6 \times 10^6 / 200 = 8000B$

区容量： $1.6 \times 10^6 / (4 \times 200) = 2000B$

区时间： $(25 - 1.25 \times 4) / 4 \text{ ms} = 5\text{ms}$

最大数据传输率： $2000B / 5\text{ms} = 4 \times 10^5 \text{ 字节/秒}$

因此，传送 1 位的时间为： $10^6 / 8 \times 4 \times 10^5 = 0.31 \text{ 微妙} < 3 \text{ 微妙}$

而传送一个字（16 位）需 $2 / 4 \times 10^5 = 5 \text{ 微妙} > 3 \text{ 微妙}$

所以可以增加一个 16 位数据缓冲器。当 16 位移位寄存器装满后，送入数据缓冲寄存器，在读出下一个 16 位数据期间（5 微妙），上次读出的 16 位数据从数缓器中被取走（3 微妙）。

6. 假定某计算机字长 16 位，主存周期为 500 毫微秒，运算器一次定点加法时间等于 250 毫微秒，配置的磁盘旋转速度为每分钟 3000 转，盘组数据面的每一磁道存两个数据块，每一块 8000 字符，同一磁道两个数据块之间相隔间隙的越过时间为 2 毫秒。问：(1) 磁盘机在一个磁道上读出的最大字符率是多少？(2) 当磁盘机按最大字符率与主机交换数据时，主存频带空闲百分比是多少？(3) 设有一条直接寻址的“存储器-存储器”SS 型加法指令，在无 I/O 操作打扰时的执行时间等于多少？当磁盘 I/O 操作与一连串这种 SS 型加法指令同时进行，问加法指令执行时间平均是多少？（假定采用多周期处理器方式，CPU 时钟周期等于主存存储周期）

参考答案：

(1) 转一圈的时间为： $60 \times 1000 / 3000 = 20\text{ms}$

区间隙时间为： $(20 - 2 \times 2) / 2 = 8\text{ms}$

最大字符传输率为： $8000 / 8 = 1000 \text{ 字符/毫秒}$

(2) 当一个字符为 16 位时，则每个字符产生一次 DMA 请求，故每隔 $10^6 / 1000 = 1000\text{ns}$ 产生一次 DMA 请求；即每 2 个存储周期中被 DMA 挪用 1 次，故主存频带空闲比是 50%。



当一个字符为 8 位时，则每 2 个字符产生一个 DMA 请求，故每隔 2000ns 产生一个 DMA 请求，即每 4 个存储周期中有一次被 DMA 挪用，故主存频带空闲比是 75%。



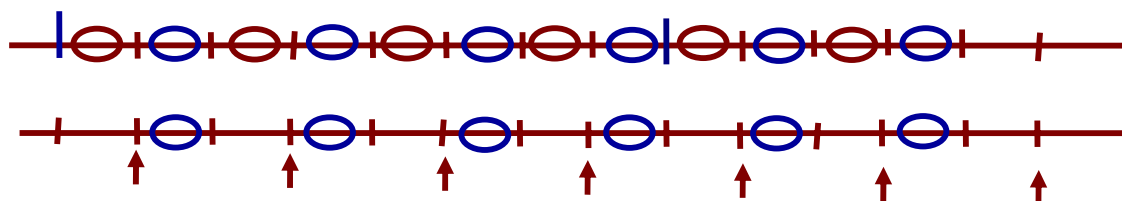
(3) 无 I/O 干扰时，执行一条直接寻址的 SS 型加法指令的时间为：

取指 500ns+取源 500ns+取目 500ns+执行 500ns+存结果 500 = 2.5 μ s



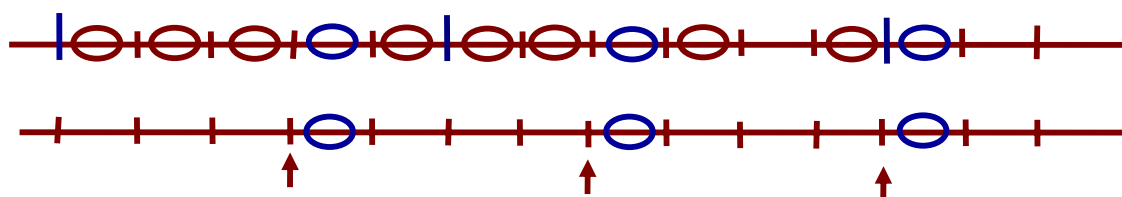
当磁盘 I/O 操作与一连串这种 SS 型加法指令同时进行，则可能因为 DMA 访存冲突而使指令的执行时间被延长。

假定一个字符为 16 位，则每隔 1000ns 产生一个 DMA 请求，要求访问主存，同时 CPU 执行指令也要求访问主存，此时，存储周期的占用情况如下：



由此可见，每个 SS 型加法指令周期一共需要 8 个存储周期的时间 $500\text{ns} \times 8 = 4\mu\text{s}$

假定一个字符为 8 位，则每隔 2000ns 产生一个 DMA 请求，要求访问主存，同时 CPU 执行指令也要求访问主存，此时，存储周期的占用情况如下：



由此可见，最好的情况是在 SS 型加法指令执行过程中没有访存冲突，此时最快，需 5 个存储周期的时间： $500\text{ns} \times 5 = 2.5\mu\text{s}$ ；最坏的情况是有一次访存冲突，此时最慢，需 6 个存储周期的时间： $500\text{ns} \times 6 = 3\mu\text{s}$ 。

8. 某终端通过 RS-232 串行通信接口与主机相连，采用起止式异步通信方式，若传输速率为 1200 波特，采用两相调制技术。传输数据为 8 位代码、无校验位、停止位为 1 位。则传送一个字节所需时间约为多少？若传输速度为 2400 波特，停止位为 2 位，其他不变，则传输一个字节的时间为多少？

参考答案：

采用两相调制技术，所以，波特率=比特率

(a) 1200 波特时，一个字符共占：1+8+1=10 位

所以一个字符所需时间约为： $10 \times (1/1200) = 8.3$ 毫秒

(b) 2400 波特时，一个字符共占：1+8+2=11 位

所以一个字符所需时间约为： $11 \times (1/2400) = 4.6$ 毫秒