

用总线，它必须等待。这一机制称为周期窃取 (cycle stealing)，因为设备控制器偶尔偷偷溜入并且从CPU偷走一个临时的总线周期，从而轻微地延迟CPU。在块模式中，DMA控制器通知设备获得总线，发起一连串的传送，然后释放总线。这一操作形式称为突发模式 (burst mode)。它比周期窃取效率更高，因为获得总线占用了时间，并且以一次总线获得的代价能够传送多个字。突发模式的缺点是，如果正在进行的是长时间突发传送，有可能将CPU和其他设备阻塞相当长的周期。

在我们一直讨论的模型——有时称为飞越模式 (fly-by mode) 中，DMA控制器通知设备控制器直接将数据传送到主存。某些DMA控制器使用的其他模式是让设备控制器将字发送给DMA控制器，DMA控制器然后发起第2个总线请求将该字写到它应该去的任何地方。采用这种方案，每传送一个字需要一个额外的总线周期，但是更加灵活，因为它可以执行设备到设备的复制甚至是内存到内存的复制（通过首先发起一个到内存的读，然后发起一个到不同内存地址的写）。

大多数DMA控制器使用物理内存地址进行传送。使用物理地址要求操作系统将预期的内存缓冲区的虚拟地址转换为物理地址，并且将该物理地址写入DMA控制器的地址寄存器中。在少数DMA控制器中使用的一个替代方案是将虚拟地址写入DMA控制器，然后DMA控制器必须使用MMU来完成虚拟地址到物理地址的转换。只有在MMU是内存的组成部分（有可能，但罕见）而不是CPU的组成部分的情况下，才可以将虚拟地址放到总线上。

我们在前面提到，在DMA可以开始之前，磁盘首先要将数据读入其内部的缓冲区中。你也许会产生疑问：为什么控制器从磁盘读取字节后不立即将其存储在主存中？换句话说，为什么需要一个内部缓冲区？有两个原因。首先，通过进行内部缓冲，磁盘控制器可以在开始传送之前检验校验和。如果校验和是错误的，那么将发出一个表明错误的信号并且不会进行传送。

第二个原因是，一旦磁盘传送开始工作，从磁盘读出的数据就是以固定速率到达的，而不论控制器是否准备好接收数据。如果控制器要将数据直接写到内存，则它必须为要传送的每个字取得系统总线的控制权。此时，若由于其他设备使用总线而导致总线忙（例如在突发模式中），则控制器只能等待。如果在前一个磁盘字还未被存储之前下一个磁盘字到达，控制器只能将它存放在某个地方。如果总线非常忙，控制器可能需要存储很多字，而且还要完成大量的管理工作。如果块被放入内部缓冲区，则在DMA启动前不需要使用总线，这样，控制器的设计就可以简化，因为对DMA到内存的传送没有严格的时间要求。（事实上，有些老式的控制器是直接存取内存的，其内部缓冲区设计得很小，但是当总线很忙时，一些传送有可能由于超载运行错误而被终止。）

并不是所有的计算机都使用DMA。反对的论据是主CPU通常要比DMA控制器快得多，做同样的工作可以更快（当限制因素不是I/O设备的速度时）。如果CPU没有其他工作要做，让（快速的）CPU等待（慢速的）DMA控制器完成工作是无意义的。此外，去除DMA控制器而让CPU用软件做所有的工作还可以节约金钱，这一点在低端（嵌入式）计算机上十分重要。

5.1.5 重温中断

我们在1.4.5节中简要介绍了中断，但是还有更多的内容要介绍。在一台典型的个人计算机系统中，中断结构如图5-5所示。在硬件层面，中断的工作如下所述。当一个I/O设备完成交给它的工作时，它就产生一个中断（假设操作系统已经开放中断），它是通过在分配给它的一条总线信号线上置起信号而产生中断的。该信号被主板上的中断控制器芯片检测到，由中断控制器芯片决定做什么。

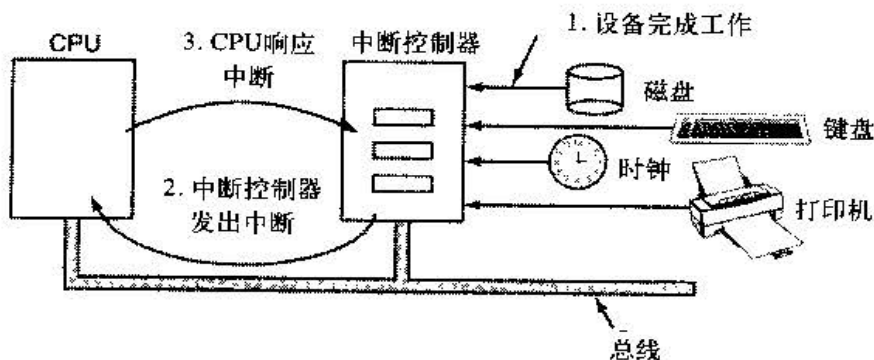


图5-5 中断是怎样发生的。设备与中断控制器之间的连接实际上使用的是总线上的中断线而不是专用连线