

I/O控制方式(1/3)

(1) 可编程I/O (轮询/查询)

由CPU代表进程给I/O模块发I/O命令，进程进入忙等待，直到操作完成才继续执行

(2) 中断驱动I/O

为了减少设备驱动程序不断地询问控制器状态寄存器的开销

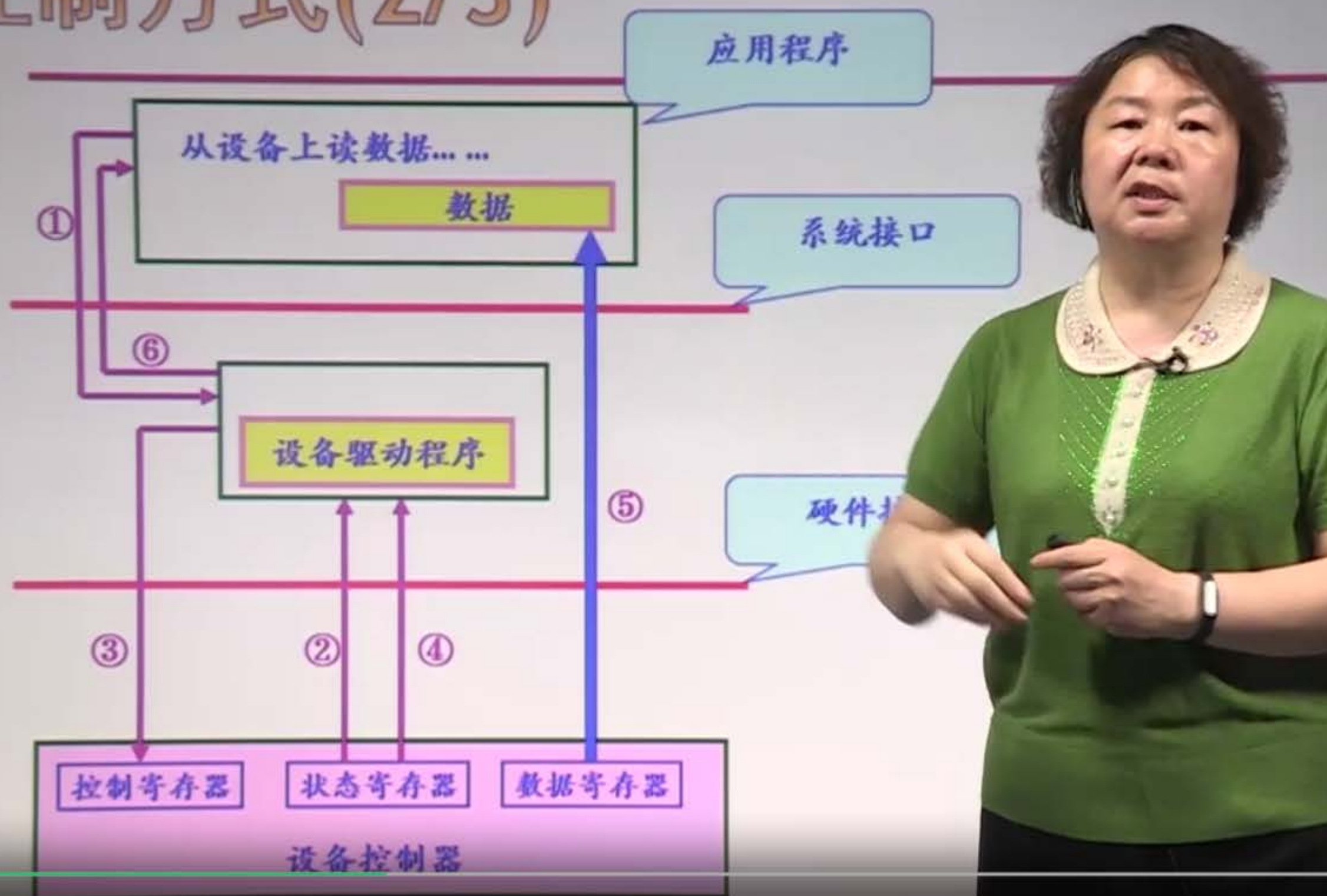
I/O操作结束后，由设备控制器主动通知设备驱动程序

(3) DMA

	无中断	使用中断
通过处理器实现 I/O-内存间的传送	可编程I/O	中断驱动I/O
I/O-内存间直接传 送		直接存储器访问 (DMA)



I/O控制方式(2/3)



应用程序

系统接口

硬件

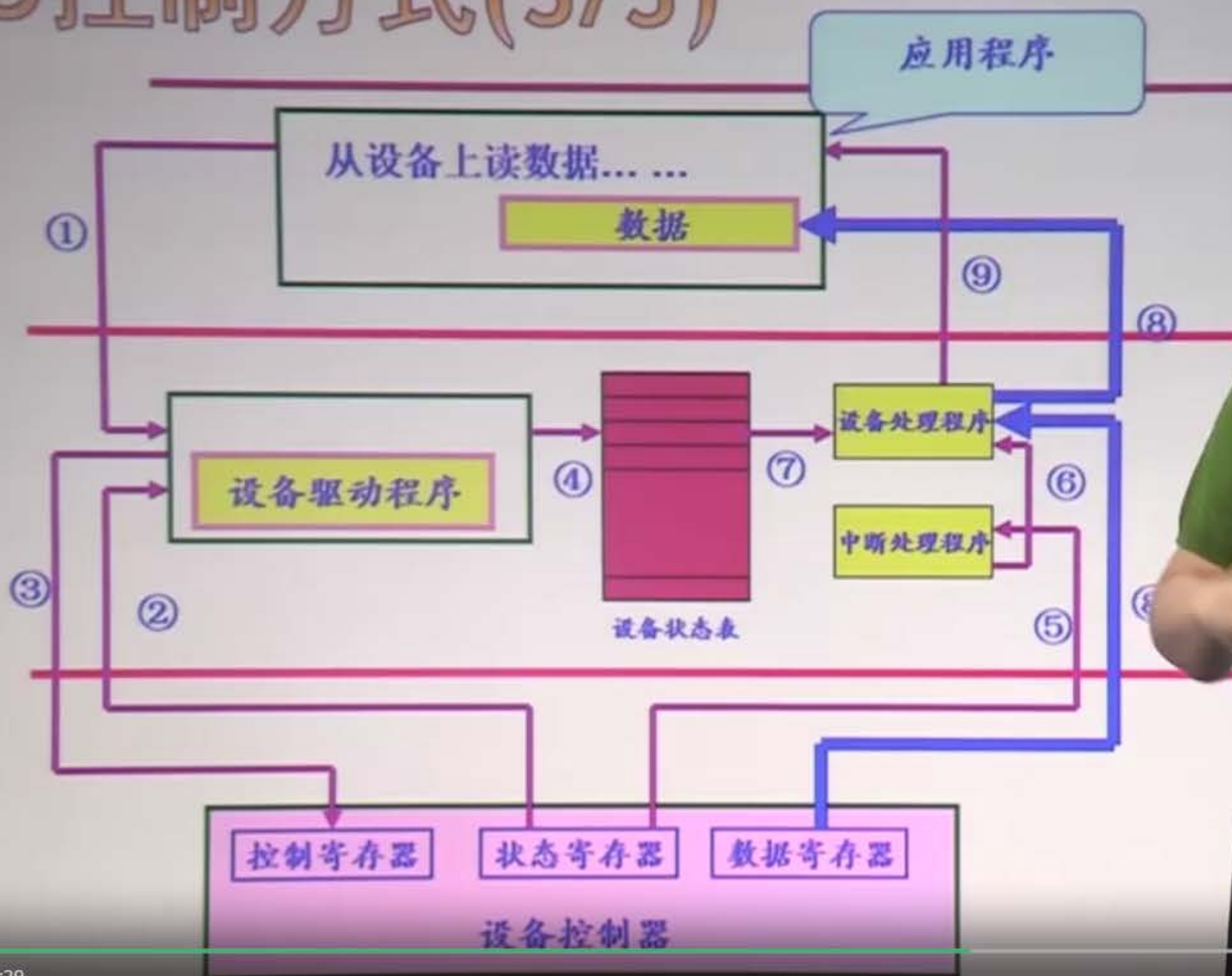
控制寄存器

状态寄存器

数据寄存器

设备控制器

I/O控制方式(3/3)



I/O部件的演化

性能/CPU与
I/O分离

1. CPU直接控制外围设备
2. 增加了控制器或I/O部件, CPU使用非中断的可编程I/O
CPU开始从外部设备接口的具体细节中分离出来
3. 与2相同, 但采用了中断方式
CPU无需花费等待执行一次I/O操作所需的时间, 效率提高
4. I/O部件通过DMA直接控制存储器
可以在没有CPU参与的情况下, 从内存中移出或者往内存中移入一块数据, 仅仅在传送开始和结束时需要CPU干预
5. I/O部件增强为一个单独的处理器, 有专门为I/O设计的指令集; CPU指导I/O处理器执行内存中的一个I/O程序。I/O处理器在没有CPU干涉的情况下取指令并执行这些指令
6. I/O部件有自己的局部存储器(其本身就是一台计算机)
使用这种体系结构可以控制许多I/O设备, 并且使需要CPU参与程度降到最小(通常用于控制与交互终端的通信, I/O处理器负责大多数控制终端的任务)



下面我们来介绍一下I/O控制方式，第一种方式呢是叫可编程I/O，也就是我们常说的轮询或者是查询，它是由CPU代表进程给I/O模块发出了I/O指令，然后，进程呢进入一个忙等状态，直到操作完成之后，这个进程才能够继续执行。第二种方式呢称之为中断驱动I/O，那么因为第一种方式设备驱动程序是不断地去询问设备的状态，因此呢，带来了很大的开销，为了减少这个开销呢，就引入了中断驱动I/O。当I/O操作结束后由设备控制器主动地来通知设备驱动程序说这次结束，而不是设备驱动程序不断地去轮询看看设备的状态。第三种方式是DMA方式，直接的内存存取，是由一个专门的控制器来完成数据从内存到设备或者是从设备到内存的传输工作。这里给出了一张表，总结了这三种方式的一些特点。下面我们看一下轮询方式的工作过程。应用程序提出了一个读数据的请求，那么操作系统，设备驱动程序会去检查一下设备的状态，如果状态是没有问题的，那么就给这个设备发出相应的控制命令，然后就不断地去测试这个设备是否完成了这次执行过程，这个④实际上就是一个轮询，不断地去测试是否完成，如果完成了就把数据送给应用程序。然后，应用程序就可以继续进行相应的处理，这就是轮询的一个工作过程。下面我们看一下中断驱动I/O的工作过程。首先，用户程序提出I/O请求，那么设备驱动程序会去检查设备的状态，如果设备已经准备好，那么就向设备提出控制命令，提出控制命令之后就把这个状态记录在设备的状态表里头，这个时候，CPU可以继续做别的事情。当设备完成了工作之后会给CPU发中断信号，那么转入了中断处理程序，中断处理程序发现这是一个正常地完成了控制命令的信号之后就会去把结果提交给设备管理程序，而设备管理程序会从刚才记录的这张设备状态表里头取得是哪一个请求的完成。接着根据这次请求把相应的数据送到应用程序。最后呢，通知应用程序可以继续执行，这就是中断驱动I/O，它的一个处理过程。下面我们来看一下I/O部件的演化过程，最早呢，是CPU直接控制外部设备，然后就演化到增加了相应的控制器或I/O部件使得CPU可以使用非中断的可编程I/O来控制外部设备，也就是前面我们所介绍的轮询方式。这个时候，CPU已经开始从外部设备接口的具体细节当中分离出来了，但是，由于是轮询，所以还占用着CPU，所以又演化到使用了中断方式，也就是CPU无需花费等待执行一次I/O的时间，这样就提高了效率。接着I/O部件就演化到了DMA这种方式，直接地内存存取，也就是说没有CPU的参与的情况下，DMA可以直接从内存当中取一堆数据送给CPU或者是把一些数据写入内存。在DMA的工作方式下，CPU的干预仅仅是在传送数据的一开始和结束的时候，那

么再演化I/O部件就增强了一个单独的处理器，有专门为I/O设计的指令，那么，CPU实际上就是指导这个I/O处理器来执行内存的一个I/O程序，所以I/O处理器在没有CPU干预的情况下，可以去取指令然后执行这些指令。最后，I/O部件具有了自己的存储器，那么它实际上就是一台计算机了。那么，通过这样一个体系结构通过这样的计算机，就可以控制各种各样的外部设备，并且使得对CPU的需要越来越少，那么它的功能就变得越来越强大，那么这就是I/O部件的演化过程。当然啦，在不同的计算机系统当中，可以采用不同的I/O部件。主要的，这种演化过程就体现了性能这上面的考虑，使得CPU与I/O的操作分离，CPU做它要做的一些运算的事情，而I/O这部分通过不同的I/O部件来控制I/O的传输过程。