输入输出

I/O

I/O接口：I/O控制器、设备控制器，负责协调主机与外部设备之间的数据传输。
I/O控制器种类多，用于控制USB设备、SATA设备的IO接口等（I/O控制器是悲集成在主板上的芯片）

以下将I/O简写为IO

IO控制方式

数据流：输入设备 -> IO接口的数据寄存器 -> 数据总线 -> CPU某寄存器 -> 主存(变量的对应位置)

1. 程序查询方式：CPU不断轮询检查IO控制器中低"状态寄存器"，检测到状态为"已完成"后，再从数据寄存器取出输入数据。
2. 程序中断方式：等待键盘IO时，CPU可以先去执行其他程序，键盘IO完成后，IO控制器向CPU发出中断请求，CPU相应中断请求并取走输入数据。

DMA控制方式

DMA接口即是DMA控制器，也是一种特殊的IO控制器。

主存与高速IO设备间有一条直接数据通路（DMA总线）。CPU向DMA接口发出"读写"命令，病知名主存地址、磁盘地址、读写数据量等参数。
DMA控制器自动控制控制磁盘与主存的数据读写，每完成一整块数据读写（如1KB为一整块），才向CPU发出一次中断请求。

通道控制方式

通道是具有特殊功能的处理器，能对IO设备进行统一管理。可理解为性能并不很强的CPU，可以识别并执行一系列通道指令，通道指令种类、功能通常更单一。

IO软件

通常用IO指令和通道指令实现主机和IO设备的信息交换

1. IO指令：CPU指令的一部分。操作码|命令码|设备码
2. 通道指令：通道能识别的指令，通道程序提前编制好放在贮存中

外部设备

外部设备（外围设备）是除主机外，能直接或间接与计算机交换信息的装置。包括有：
输入设备，输出设备，外存设备

输入设备：鼠标，键盘，等

输出设备：
显示器 [屏幕大小，分辨率，灰度级（常见于黑白显示器），刷新率，显示存储器（VRAM，刷新存储器，把一帧图像信息存储在刷新存储器中，其存储容量VARM容量 = 分辨率 × 灰度级位数，带宽 = 分辨率 × 灰度级

位数 × 帧频）];
打印机 [打击式和非打击式，串行和行式，针式、喷墨式、激光];

IO接口

IO接口（IO控制器、设备控制器）协调主机与外部设备之间的数据传输。

IO接口可以通过数据缓冲寄存器让主机和外设工作速度匹配；通过状态寄存器反馈设备的各种错误、状态信息，供CPU查用；接收从控制总线发来的控制信号、时钟信号；串并、并串等格式转换；实现主机-IO接口-IO设备间的通信。

内部接口：内部接口与系统总线相连，实质上是内存与CPU相连。

外部接口：外部接口通过接口电缆与外部相连，外部接口的数据传输可以是串并相互转换的。

- 接口Interface
    - 端口Port（寄存器）
        - 数据端口：读&写
        - 控制端口：写
        - 状态端口：读
    - 逻辑控制
- IO端口是指接口电路中可以被CPU直接访问的寄存器

对端口编址

有同一编址和独立编址两种方式。

统一编址：IO端口的地址和内存地址是同一套（存储器映射方式），靠不同的地址码区分内存和IO设备，
程序设计灵活性高，读写控制逻辑电路简单，
但端口占用了主存地址空间，外设寻址时间长。

独立编址：IO端口的地址和内存地址不是同一套，需要专门的IO指令访问IO端口，
使用专用IO指令，程序编制清晰，端口地址为树梢，地址译码速度高，不占用主存空间，
但IO指令类型少，一般只对端口进行传送操作，程序设计灵活性差，需要CPU提供存储器读写、IO设备读写两组信号，增加逻辑电路的复杂性。

IO接口工作原理

1. 发命令：发送命令字到IO控制寄存器，箱设备发送命令（需要驱动程序的协助）
2. 读状态：从状态寄存器读取状态字，会哦的设备或IO控制器的状态信息
    - 控制寄存器、状态寄存器在使用时间上是错开的，所以有的IO接口会将二者合一
    - IO控制器中的各种寄存器称为IO端口
3. 读写数据：从数据缓冲寄存器发送或读取数据，完成主机与外设的数据交换

IO方式

程序查询方式

|<-CPU执行现行程序(启动IO)->|<-CPU查询等待(IO准备)->|<-CPU控制数据传送(数据传送)->|<-CPU执行现行程序|

- 预设值传送参数；启动外设
- 取外设状态
- 外设准备就绪?
    - 否：回到取外设状态
    - 是：继续
- 传送一次数据；修改传送参数
- 传送完否?
    - 否：回到取外设状态
    - 是：继续
- 结束

接口设计简单、设备量少；
但CPU在信息传送过程中要发飞很多时间用于查询和等待，而且一段时间内只能和一台外设交换信息，效率大大降低。

e.g. 每个查询要100个时钟周期，CPU时钟频率=50MHz，CPU每秒对鼠标进行30次查询，硬盘以32位字长单位传输数据（每32位被CPU查询一次），传输率=2×2^20B/s

解：一个时钟周期=1/50Mhz=20ns，一个查询操作耗时100×20ns=2000ns；
鼠标每秒查询耗时=30×2000ns=60000ns，
查询鼠标的时间比率=60000ns/1s=0.006%；
硬盘每秒查询=(2×2^20B)/4B=2^19次，
（1B = 8bit，8位，32位即为4B）
查询硬盘耗时=2^19×2000ns=512×1024×2000ns≈1.05×10^9ns，
查询硬盘花费时间比率=(1.05×10^9ns)/1s=1.05%

程序中断方式

中断是计算机执行现行程序的过程中，出现某些急需处理的异常情况或特殊请求，CPU暂时终止现行程序，转去对异常情况或特殊请求进行处理，处理后CPU自动返回现行程序的断点处继续执行。
工作流程如下：

1. 中断请求：中断源向CPU发送中断请求信号
2. 中断响应：响应中断的条件
    - 中断判优：多个中断源同时提出请求时通过中断判优逻辑响应一个中断源
3. 中断处理：
    - 中断隐指令
    - 中断服务程序

中断的类别：

- 中断（广义的中断）
    - 内中断（衣长、例外、陷入）
        - 资源中断——指令中断
        - 强迫中断
            - 硬件故障
            - 软件中断
    - 外中断（狭义的中断）
        - 外设请求

        - 人工干预
- 外中断可细分为：
    - 非屏蔽中断：关中断时也会被响应（例如掉电）
    - 可屏蔽中断：关中断时不会被响应

CPU会通过关中断指令进入原子操作。
(原子操作：CPU执行某一操作时不受其它指令影响)

中断判优（当有多个中断信号时判定哪个优先处理）：
可用硬件实现、软件实现。
硬件实现通过硬件排队器实现，它皆可以设置在CPU中，也可以分散在各个中断源中；
软件实现通过查询程序实现。

中断判优的优先级排序（靠左的优先）：

1. 硬件故障，软件中断
2. 非屏蔽中断，屏蔽中断
3. DMA请求，IO设备传送的中断请求
4. 高速设备，低速设备
5. 输入设备，输出设备
6. 实时设备，普通设备

中断隐指令的主要任务：

1. 关中断：保护中断现场
2. 保存断点：保存原程序的断电（程序计数器PC的内容），可以放入堆栈或存入指定单元
3. 引出中断服务程序：取出中断服务程序的入口地址并传送给程序计数器PC

由硬件产生向量地址，再由向量地址找到入口地址。

中断服务程序的主要任务：

1. 保护现场：保存通用寄存器和状态寄存器的内容，以便返回原程序后恢复CPU环境。可以用堆栈或特定存储单元
2. 中断服务（设备服务）：中体部分，如通过程序控制需打印的字符代码送入打印机的缓冲存储器
3. 恢复现场：通过栈指令或取数指令吧之前保存的信息送回寄存器中

中断处理的过程：

- 取指令；执行指令
- 是否中断?
    - 否：返回取指令
    - 是：继续
- 中断隐指令（中断周期）
    - 中断响应，程序断点进站，关中断，向量地址->PC
- 中断服务程序
    - 保护现场；设备服务；恢复现场
    - 开中断，中断返回
        - 返回取指令

多重中断

执行中断程序时响应新的中断请求。CPU需要满足下列条件才能具备多重中断的功能：

1. 在中断服务程序中提前设置开中断指令
2. 优先级别高的中断源有权中断优先级别低的中断源

每个中断源都有一个屏蔽触发器，1表示屏蔽该中断源的请求，0表示可以正常申请，所有屏蔽触发器组合一起，构成一个屏蔽字寄存器，屏蔽字寄存器的内容称为屏蔽字。

e.g.：四个中断源A、B、C、D的硬件排队次序为A>B>C>D，中断服务程序时间为20us，CPU在5us开始运行B、10us运行D、35us运行A、60us运行C。要求终端次序改为D>A>C>B，写出中断源对应屏蔽字，并按时间轴画出CPU执行程序轨迹。

解：

中断源	屏蔽字(ABCD)
A	1110
B	0100
C	0110
D	1111

程序执行轨迹：
|5-B-10-D-30-B-35-A-55-B-60-C-80-B-85|

DMA方式

DMA方式可以控制数据传输的方式，DMA控制器与主存每次传送完一整块数据后才向CPU发出中断请求。通常用于控制块设备（例如磁盘）

CPU向DMA控制器致命要输入还是输出，要传送多少个数据，数据在主存、外设中的地址（1、2传送前，3、4传送时，5传送后）：

1. 接受外设发出的DMA请求（外设传送一个字的请求），并向CPU发出总线请求。
2. CPU响应此总线请求，发出总线响应信号，接管总线控制权，进入DMA操作周期。
3. 确定传送数据的主存储单元地址和长度，并能自动修改主存地址计数器和传送长度计数。
4. 规定数据在主存和外设间的传送方向，发出读写控制信号，执行数据传送操作。
5. 向CPU报告DMA操作的结束。

传送过程：

- 预处理：
  - 主存起始地址 -> AR
  - IO设备起始地址 -> DAR
  - 传送数据个数 -> WC
  - 启动IO设备
- 数据传送：
  - 继续执行主程序
  - 同时完成一批数据的传送
- 后处理：

    - 中断服务程序
    - 做DMA结束处理
- 继续执行主程序

DMA传送方式:
主存和DMA控制器之间有一条数据通路，因此主存和IO设备之间交换信息时，不通过CPU。dna当IO设备和CPU同时访问主存时，可能发生冲突，为了有效地使用主存，DMA控制器和CPU通常采用以下方式使用主存:

1. 停止CPU访问主存
    - 控制简单
    - 但CPU处于不工作状态或保持状态，未充分发挥对主存的利用率
2. DMA与CPU交替访存
    - 不需要总线的使用权申请、建立和归还过程
    - 但硬件逻辑更为复杂
3. 周期挪用（周期窃取）
    - DMA访问主存有三种可能:
        1. CPU此时不访存（不冲突）
        2. CPU正在访存（存取周期结束让出总线）
        3. CPU与DMA同时请求访存（IO访存优先）

DMA方式和中断方式对比

-	**中断**	**DMA**
数据传送	程序控制 程序切换 -> 保存和恢复现场	硬件控制 CPU只需进行预处理和后处理
中断请求	传送数据	后处理
响应	指令执行周期结束后响应中断	每个机器周期结束均可，总线空闲时即可相应DMA请求
场景	CPU控制，低速设备	DMA控制器控制，高速设备
优先级	优先级低于DMA	优先级高于中断
异常处理	能处理异常事件	仅传送数据