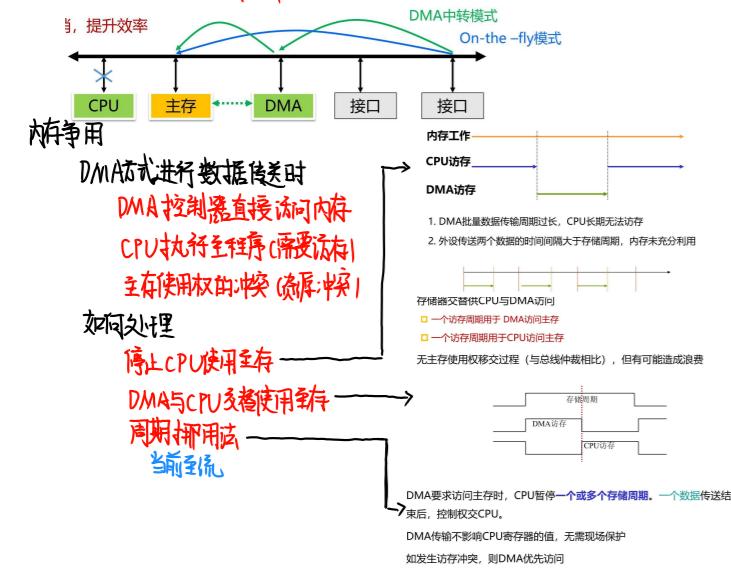


中断方式

传送一个数据执行一次中断服务子程序(含时钟指令)
效率低下,不适合高量数据传输的场合

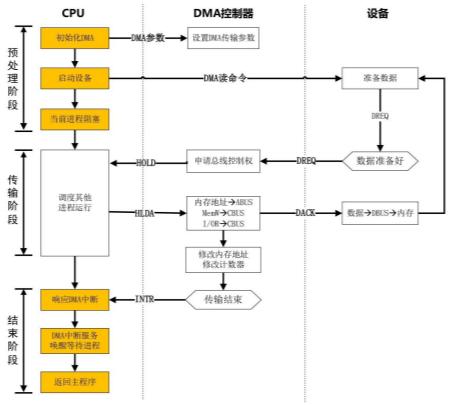
DMA方式

外设与主存间建立一个由硬件管理的数据通道(不占有单独通道)
CPU不介入外设与主存的数据传输,传输阶段不占用CPU时间
减少CPU开销,提升效率



DMA传输流程

- 准备阶段**
- CPU向DMA控制器发送DMA参数, DMA控制器设置DMA终结参数。
 - 设备准备数据。
- 传送阶段**
- CPU向DMA控制器发送DMA命令, DMA控制器向设备发送HOLD信号。
 - 设备申请总线控制权, DMA控制器向CPU发送DREQ信号。
 - 设备将数据准备好, DMA控制器向设备发送DACK信号。
 - 设备将数据写入内存, DMA控制器向CPU发送ACK信号。
- 结束阶段**
- 设备将数据读出内存, DMA控制器向设备发送DACK信号。
 - 设备向DMA控制器发送INTR信号, 表示传输结束。
 - CPU响应INTR信号, 处理中断。



通道方式: 实现DMA的集中管理

DMA接口起始准备到传输结束仍需CPU参与, 高速设备数据快
在频繁传送的情况下, CPU归功于时间不可忽视。

CPU与外设传输方式总结

■ 程序控制方式

- 简单输入输出: 外设永远准备好
- 查询式输入输出: 通过状态查询读取单个或批量数据
- 特点: 硬件开销小, CPU资源开销大

■ 中断方式

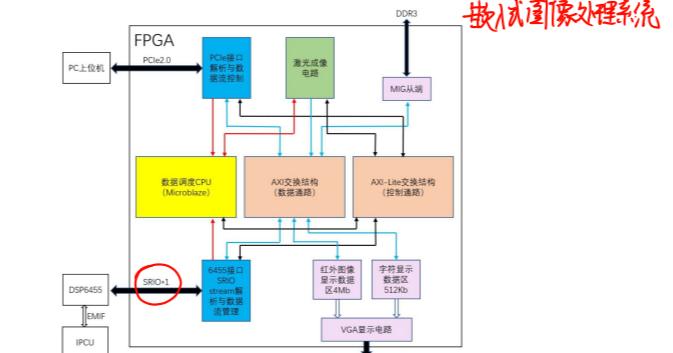
- 存在中断响应开销, 适用于随机发生的小批量数据传输

■ DMA方式

- 适用于大批量数据传输, 需要专用的DMAC模块支持

■ 通道方式

- 实现更大规模和更加复杂的数据传输, 可以将中断与DMA传输直接关联



设置专用的~~输出~~输出处理器(通道), 分担编辑输出管理的全部工作
吸引DMA技术, 增加软件管理, 没有专用通道指令
有层次性的I/O系统

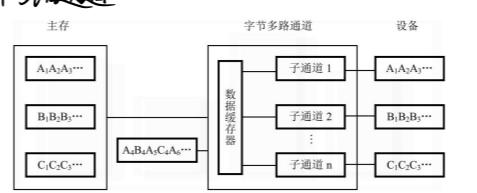
一个主机可连接许多通道; 一个通道可管理多个设备控制器; 一个设备控制器可控制多台设备

通道功能

根据CPU要求, 组织设备与系统连接和通信;
接收通道指令, 向设备发出操作命令;
指出数据在设备中的位置和主存缓冲区内的位置, 组织设备与主存间的数据传输。
向CPU反映设备、设备控制器及通道本身的状态信息。
将外设和通道的中断请求, 按次序及时报告CPU。
设备控制器(通道与设备之间)是通道对外部设备进行具体控制的机构。
■ 将通道发出的命令转换为设备能接受的控制信号
■ 向通道反映设备的状态
■ 将设备的各种电平信号转换成通道识别的标准逻辑信号。

通道类

1. 字节多路通道



- 包括若干子通道, 每个子通道服务于一个慢速设备
- 在一段时间内交替执行多个设备的通道子程序
- 传输单位是字节
- 宏观上这些设备并行工作

2. 选择通道

字节多路通道

- 适合慢速设备, 不适合高速设备
- 高速设备传送两个字节的时间很短

选择通道

- 设备以成批数据连续传送方式占用通道, 直到指定数量的数据全部传送完毕, 通道才转为其它设备服务。
- 选择通道在物理上可以连接多个设备, 但设备不能同时工作。
- 选择通道只有一个子通道, 它适用于大批量数据的高速传送。



3. 成组多路通道

通道能高速传送数据, 但设备辅助操作时间不能有效利用

- 如硬盘启动后, 平均定位时间较长, 磁带机磁头定位时间更长, 可达几分钟。导致通道处于等待状态。

为利用这段时间, 将字节多路和选择通道折中, 称为成组多路通道。

- 多个设备以数据组(块)为单位交叉使用通道。
- 设备占用通道时, 连续传送一组数据, 然后将出让通道使用权
- 数据组的大小因设备而异, 有256B、512B或1KB等
- 某设备执行辅助操作时, 通道暂时断开与该设备的连接, 相应程序挂起

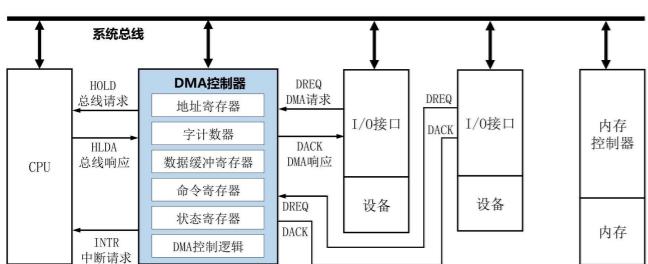
传送效率高, 硬件复杂度高

通道发展

IOP: I/O处理器 → 作为CPU的部件工作, 提供高速DMA能力

PPU: 外围处理器 → 用于大型计算机, 独立于CPU实现数据传输、码制转换、格式处理、数据块校验等

DMA控制器



DMA主要操作过程

一、准备阶段

主机通过CPU指令向DMA接口发送必要的传送参数, 启动DMA工作

- 数据传送的方向
- 数据块在主存的首地址
- 数据在外设首地址上的地址
- 数据的块容量
- 地址变化的方式

二、传送阶段

宏观: DMA连续传送一批数据
微观: 每传送一个数据发一次DMA请求

传输过程(周期挪用)

- 设备准备数据。设备接收到CPU的DMA命令后就可以开始准备数据。
- 设备发送DMA请求。数据准备后就通过DREQ控制线向DMAC发出DMA请求。
- DMAC申请总线。DMAC收到DMA请求后立即置HOLD信号置“1”, 向CPU申请总线控制权。
- 总线授权。CPU在机器周期内响应回复使用申请, 让出总线控制权, 发出总线授权信号HLDIA通知DMAC。
- DMA数据传输。收到HLDIA后, 将内存地址放置在地址总线上; 设置控制总线读写控制信号, 向设备DMA应答信号DACK。设备收到DACK信号后和内存完成子机操作的数据交换。
- 传输结束后, DMAC通过INTR信号发一个EOP的DMA中断清除信号, 告知CPU传输完成。

三、结束阶段

在正常结束(完成一批数据传送)和非正常结束(DMA故障)时进入结束阶段。

- DMA向主机发出中断请求。
- CPU查询DMA接口状态, 据状态进行不同处理(执行中断服务)。