

## I/O

1. I/O接口指主机与外设之间设置的硬件电路及相应的软件控制。

2 . I/O接口的功能与组成

选址(设备选择)

传送命令 命令寄存器,命令译码器

传送数据 数据缓冲寄存器

反映I/O设备工作状态 暂停、准备就绪、正在准备等状态

3 . 接口(interface)和端口(port)

端口=寄存器组(数据、控制、状态...) 接口=N个端口+控制逻辑

4 . 接口的类型

- 按数据传送方式分,有并行接口和串行接口。
- 按功能选择的灵活性分,有可编程接口和不可编程接口。  
不可编程接口可通过硬连线路逻辑来实现不同的功能
- 按通用性分类,有通用接口和专用接口。
- 按数据传送的控制方式分类,有程序型接口和DMA式接口

5 . I/O编址方式

统一编址

在主存储器的地址空间中划出某一区域专门作为外设地址区使用,需占用小部分存储空间。

独立编址

I/O端口和存储器分开编址(I/O Mapped I/O)。指令系统中分别设立面向存储器的指令和面向I/O操作的指令(IN指令和OUT指令)。优点:不占用主存空间 缺点:需专门的I/O指令,其寻址方式较简单,编程灵活性稍差

6 . 同步方式 主机较外设速度快,必须同步

立即响应方式:外设处于等待状态,CPU的I/O指令一到,立即响应

异步方式:

应答式(acknowledge)Ready:就绪 Strobe:选通 速度慢

同步式:主机与外设采用同步时标 并行,速度快

7 . 外设与主机连接的拓扑结构 总线式,星式联,级式

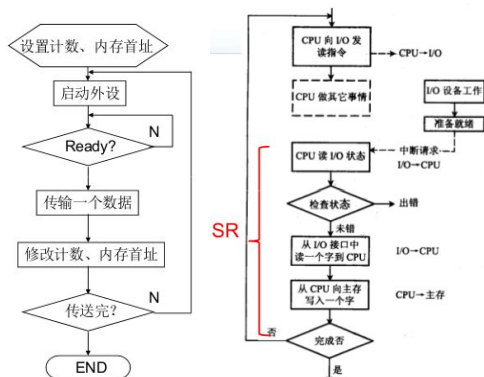
8 . 数据传输控制方式 程序查询方式 程序中断方式 直接内存访问方式

9 . 程序查询方式 由CPU控制数据传输的过程 处理器等待,效率低

接口电路 DBR:数据缓冲器,用于存放欲传送的数据。

10 . 程序中断方式

允许一个处理器“同时”执行多个任务



## 11. 键盘读键过程

- 按键 – 识别(查出按下的是哪一个键) – 将此键翻译成ASCII码

### 编码键盘与非编码键盘

- 编码键盘法:采用硬件电路确认哪个键按下的方法。
- 非编码键盘法:采用软件确认哪个键按下的方法。

## 1 2 . DMA方式 避免在I/O时占用CPU

需要解决访存时总线冲突问题 CPU在总线周期结束时让出总线 批量数据传输

## 1 3 . DMA概念

- 用途:高速外设直接与主存进行数据传输, CPU不用暂停现程序而为I/O服务
- 关键问题:DMA接口与CPU之间共享主存,存在总线争用、访存冲突问题

CPU暂停方式 周期窃取方式 交替访问方式

## 1 4 . CPU暂停方式过程

DMA向CPU申请总线 CPU暂停 DMA传输 DMA释放总线 CPU继续

- 并未真正实现并发

## 1 5 . 周期窃取/周期挪用(cycle stealing): 当I/O设备发出DMA请求时,I/O设备便挪用或窃取总线占用权一个或几个主存周期。

- 有三种情况

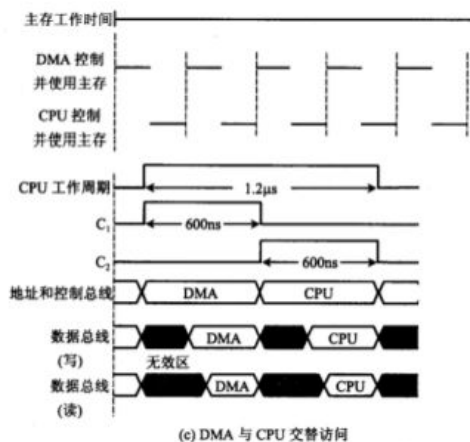
- CPU不访存(复杂指令mul):DMA使用
- CPU正在访存:DMA等待CPU完成(总线周期结束),然后获得总线使用权
- CPU与DMA同时发生:DMA优先,窃取一到二个存取周期(否则数据丢失)

• 应该指出,I/O设备每挪用一個主存周期都要申请总线控制权、建立总线控制权和归还总线控制权。因此,尽管传送一个字对主存而言之占用一个主存周期,但是对DMA接口而言,实质上要占2~5个主存周期。

## 1 6 . 交替访问——周期扩展

将CPU工作周期延长(或者CPU的工作周期本身比主存周期长得多),分成两段。下图为交替访问方式示意图。

这种方式不需要总线使用权的建立和归还过程,总线使用权是通过C 1 和C 2 分别控制的。实际上总线便成了C 1 和C 2 控制下的多路转换器。



## 1 7 . DMA接口的功能

向CPU申请DMA传送;

在CPU允许DMA工作时,处理总线控制权的转交,避免因进入DMA工作而影响CPU正常活动或引起总线竞争;

在DMA期间管理系统总线,控制数据传送;确定数据传送的起始地址和数据长度,修正数据传送过程中的数据地址和数据长度;

在数据块传送结束时,给出DMA操作完成的信号

#### 1 8 . 简单的DMA接口的基本组成 DMA接口也称作DMA控制器(DMAC)

- 主存地址寄存器(Address Register) 每传输一个数据(字节、字),地址加1
- 字计数器(word counter)
- 数据缓存寄存器(Buffer Reg) 完成数据格式转换等 数据传输不一定经过DMAC
- DMA控制逻辑 – DREQ-DACK、HRQ-HLDA 优先级控制
- 设备地址寄存器(DAR) 外设中数据块地址或当前设备的设备号
- 中断逻辑——不一定在DMAC中 请求CPU进行DMA“后处理” DMA出错处理

#### 1 9 . DMA工作过程

- 预处理 初始化 启动设备开始传输
- 数据传输

外设准备好数据

外设通过DMAC申请总线使用权 DREQ HRQ HLDA DACK

DMAC接管总线,控制完成数据传输

释放总线 HLDA

- 后处理 请求中断服务,对读入的数据进行处理

#### 2 0 . DMA控制器(DMAC)

- 功能 申请总线 控制总线 控制传输过程(地址、计数、R/W命令) I/O device 释放总线
- I/O 内存与外设、内存与内存、外设与外设
- 数据大小 单字节、块传输、请求传输

#### 2 1 . DMA系统连接方式

- 系统中存在多个DMA通道和多个I/O设备,如何连接? 优先级问题
- DMA和CPU:多个DMAC
  - 公共请求方式:链式/级联
  - 独立请求方式:星型连接
- DMA和设备: 一个DMAC多个设备
  - 选择型:DMAC“软”选择响应设备
  - 多路型:“硬”选择响应设备
    - 链式多路型
    - 独立请求多路型
- 分布式?集中式?
  - 分布式:公共请求,链式多路型
  - 集中式:独立请求,选择型

#### 2 2 . DMA与cache一致性

- 方式一:禁止DMA目标地址范围内的cache功能。设置这些内存页uncached
- 方式二:Flush与Invalidate指令操作

#### 2 3 . 中断控制方式与DMA方式有何异同?

DMA是程序中断传送技术的发展,在硬件逻辑机构的支持下,以更快的速度,更简便的形式传送数据,二者的区别为:

(1)中断方式通过**程序**实现数据传送,而DMA方式直接靠**硬件**来实现。

(2)CPU对中断的**响应是在执行完一条指令之后**,而对DMA的响应则可以在**指令执行过程中的任何两个存储周期之间**。

(3)中断方式具有**数据传送和处理异常事件**的能力,而DMA只能进行**数据传送**。

(4)中断方式**必须切换程序,要进行CPU现场的保护和恢复**,DMA仅挪了一个存储周期,**不改变CPU现场**。

(5)**DMA请求的优先权比中断请求高**,CPU优先响应DMA请求,是为了避免DMA所连接的高速外设丢失数据。

2 4 . 设备驱动程序 操作特定设备的程序:字符设备、块设备,将所有设备映射成“文件”

功能:设备管理,读写操作,并发控制

2 5 . I/O应用程序编程模型

- 多I/O并发:磁盘、网络、外设...

- 单线程I/O 程序查询方式:顺序轮询式 阻塞式I/O?

- 多线程 每个线程负责一个I/O任务 同步式I/O(阻塞式) 线程切换开销大

- 事件驱动 单线程 异步I/O:事件队列循环

2 6 . 具有通道结构的阶段

- 对大型系统,设备多,数据传输频繁,DMA造成的总线冲突仍然影响CPU的效率。

- 解决办法:采用I/O通道方式进行数据交换。执行专用指令,完成数据交换,专用处理器IOP,受主CPU控制(启停等)

2 7 . 通道与DMA的区别

- 对CPU而言,通道比DMA具有更强的独立处理I/O的能力。

DMA方式是通过DMA控制器控制总线,在外设和主存之间直接实现I/O传送;而通道通过执行通道程序进行I/O操作的管理。

- DMA控制器通常只控制一台或多台同类的高速设备,而通道可控制多台同类或不同类的设备。

2 8 . 通道的工作过程

CPU执行I/O指令 当需要进行I/O操作时,CPU按约定的格式准备好命令和数据,编制好通道程序,然后通过执行I/O指令(如:START I/O,TEST I/O,HALT I/O等)来启动通道。I/O指令应给出通道开始工作所需的全部参数,如:通道执行何种操作,在哪个通道和设备上进行操作等。CPU启动道后,通道和外部设备将独立进行工作。

处理来自通道的中断请求 当通道和外设发生异常或结束处理时,通道采用“中断”方式向处理器报告。

2 9 .