

计算机组成原理

2017年修订

西南交通大学信息科学与技术学院
唐慧佳 hjtang@home.swjtu.edu.cn



第7章 输入输出系统

§ 7.1 外部设备概述

§ 7.2 主机与外设的连接

§ 7.3 程序查询方式及其接口

§ 7.4 中断系统和程序中中断方式

§ 7.5 DMA方式及其接口

§ 7.6 通道控制方式

§ 7.7 总线技术



§ 7.1 外部设备概述

外部设备：除主机以外，围绕主机设置的各种硬件装置。

外部设备主要完成数据的输入、输出、成批存储以及对信息加工处理的任务。

外部设备的主要类型：

- 1) 输入输出设备
- 2) 辅助存储器（后援存储器）
- 3) 终端设备
- 4) 过程控制设备
- 5) 脱机设备

§ 7.2 主机与外设的连接

7.2.1 输入输出接口（I/O接口）

I/O接口是主机和外设间的交接界面，可解决主机和外设间的信息交换，信息格式和工作速度的差异。

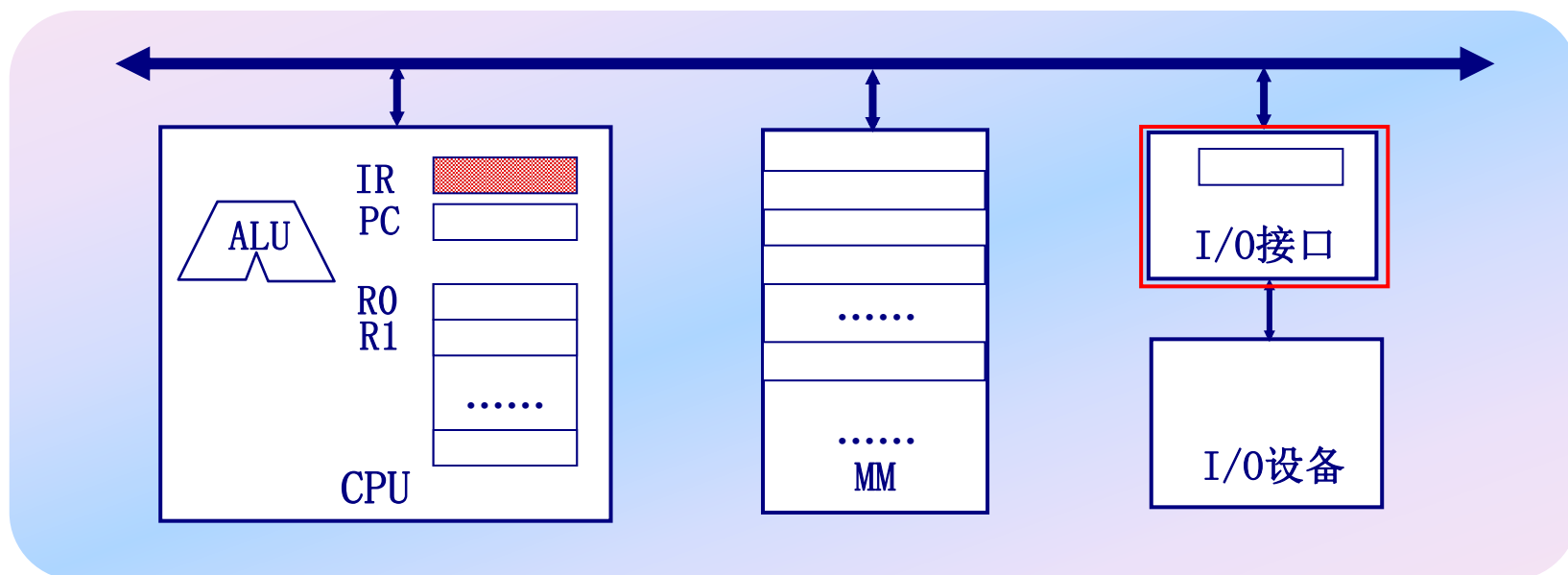
硬件接口——线路的连接逻辑

软件接口——信号传递协议

§ 7.2 主机与外设的连接

7.2.1 输入输出接口（I/O接口）

I/O接口是主机和外设间的交接界面，可解决主机和外设间的信息交换，信息格式和工作速度的差异。



7.2.2 接口的功能和基本组成

1. I/O接口的基本功能

(1) 寻址

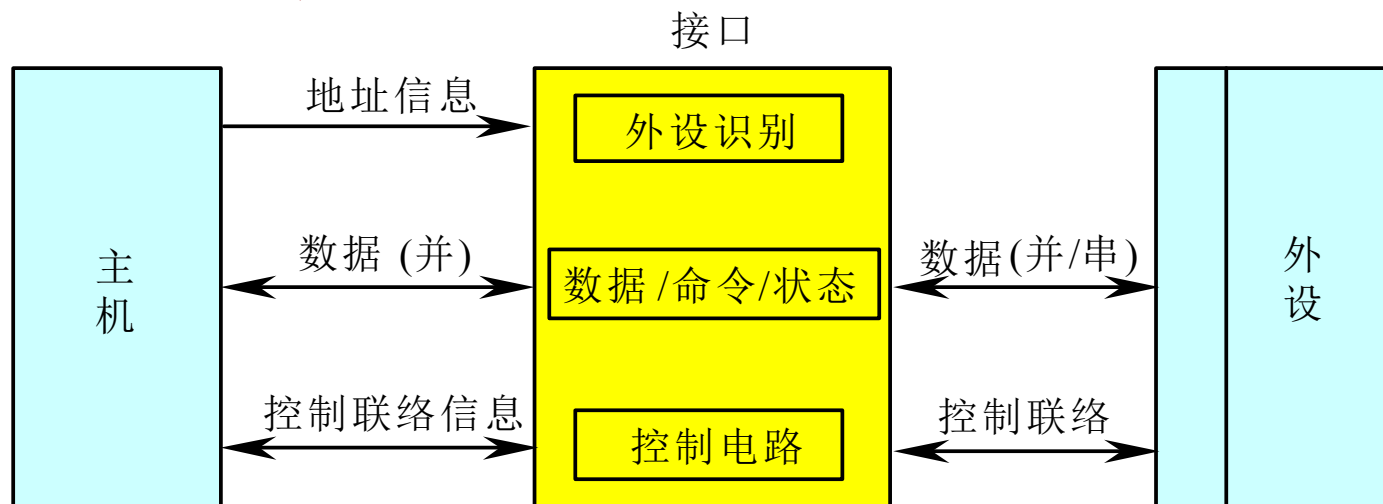
(2) 数据传送与缓冲——解决主机与外设的速度匹配问题

(3) 实现主机和外设的通信联络控制

(4) 传递控制命令和状态信息

(5) 提供驱动能力和数据格式变换

2. 接口的基本组成



端口： 接口电路中可以被CPU直接访问的寄存器。

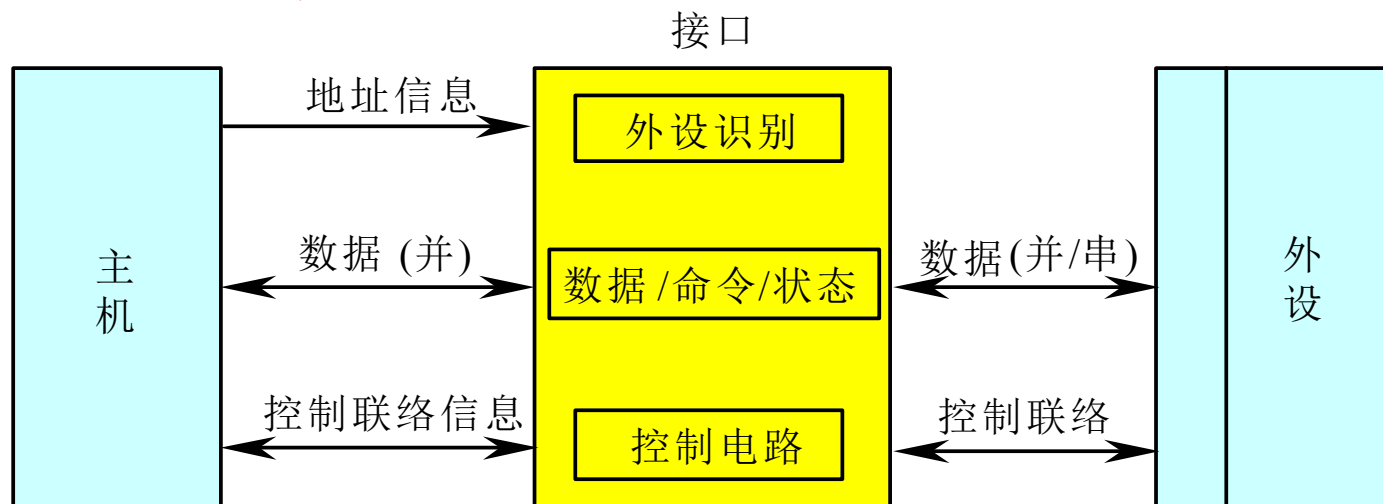
一个接口中包含：数据端口(数据寄存器)

命令端口(控制命令寄存器)

状态端口(状态寄存器)。

注： 状态信息和控制信息通常共用一个端口，称之为设备的控制/状态寄存器。

2. 接口的基本组成



端口： 接口电路中可以被CPU直接访问的寄存器。

CPU通过输入指令可以从有关端口中读取信息，
通过输出指令可以把信息写入有关端口。

3. 接口的类型

按数据传送方式分类

并行接口：接口与I/O之间并行传送。

串行接口：接口与I/O之间串行传送。

（注：接口与系统总线一般仍并行）

按主机访问I/O设备的控制方式分类

程序查询接口、中断接口、DMA接口、通道和IOP

按功能选择的灵活性分类

可编程接口和不可编程接口。

前者可用程序来改变接口的功能和操作方式。

7.2.3 外设的识别与端口寻址

1. 独立编址（I/O映射方式）

把I/O端口地址与存储器地址分别进行独立的编址；I/O地址和内存地址可能重叠。

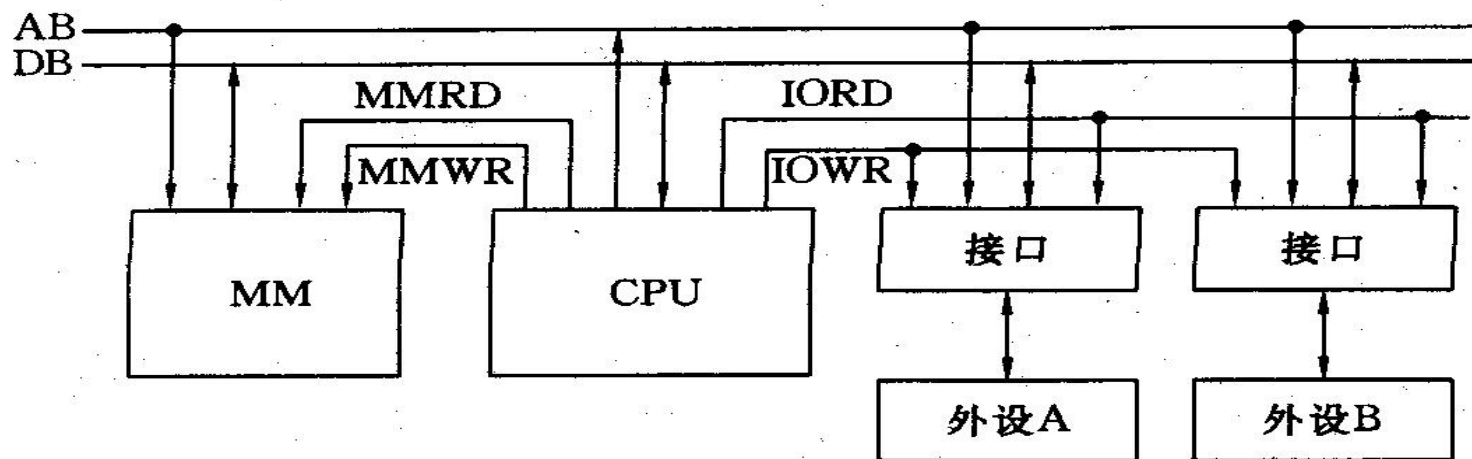
【例】**MOV** AX, [1000] ；访问内存、1000为内存地址
IN AX, [1000] ；访问I/O设备、1000为I/O地址

7.2.3 外设的识别与端口寻址

1. 独立编址 (I/O映射方式)

把I/O端口地址与存储器地址分别进行独立的编址；
I/O地址和内存地址可能重叠。

🌀 **特点：** I/O地址不占用内存地址空间，但需要专门的I/O指令进行访问，用专门的硬件信号线来区分访存还是访问I/O操作。



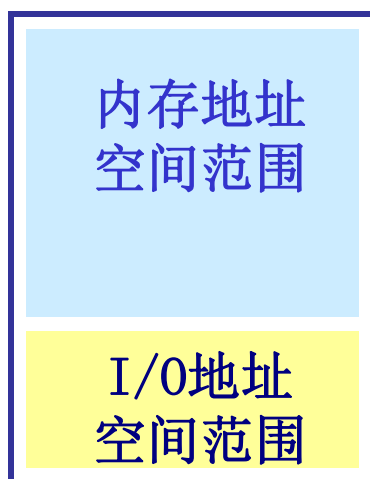
(b) 单独编址的单总线结构

7.2.3 外设的识别与端口寻址

2. 统一编址（存储器映射方式）

将I/O接口中的端口看作存储器单元，与主存储器单元统一编址，I/O地址与内存地址不重叠。

例：MOV AX, [1000] ; 1000可能为内存地址
; 也可能为I/O地址

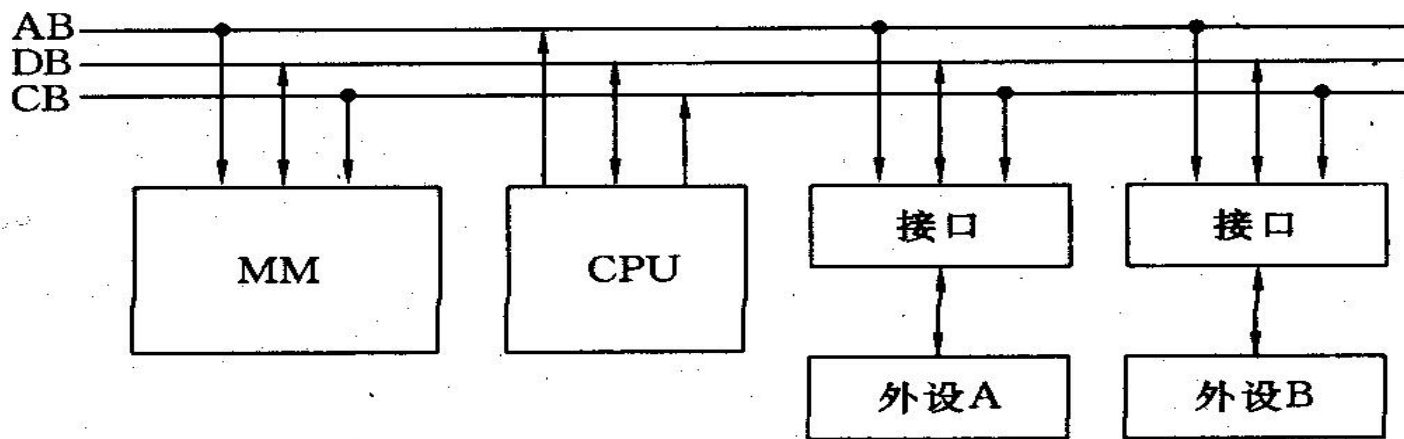


7.2.3 外设的识别与端口寻址

2. 统一编址（存储器映射方式）

将I/O接口中的端口看作存储器单元，与主存储器单元统一编址，I/O地址与内存地址不重叠。

🌀 **特点：**不需要设置专门的I/O指令，可利用访存指令访问I/O接口中的寄存器，程序设计简单。但外设占用了存储器的地址空间。



(a) 统一编址的单总线结构

课堂练习

1. 在统一编址方式下，下面哪一个说法正确（ ）
 - A 一个具体地址只能对应输入输出设备
 - B 一个具体地址只能对应内存单元
 - C 一个具体地址既可对应内存单元又可对应输入输出设备
 - D 一个具体地址只对应I/O设备或者只对应内存单元。
2. 在独立编址方式下，存储单元和I/O设备是靠（ ）来区别的。
 - A 不同的地址代码
 - B 不同的地址总线
 - C 不同的指令或不同的控制信号
 - D 上述都不对

课堂练习（续）

3. 数据格式的串——并转换是（ ）功能。

- A 并行接口
- B 串行接口
- C 同步接口
- D 异步接口



4. I/O接口中数据缓冲器的作用是（ ）

- A 暂存外设和CPU之间传送的数据
- B 暂存外设的状态
- C 暂存外设的地址
- D 上述都不对

§ 7.3 程序查询方式及其接口

7.3.1 程序查询方式

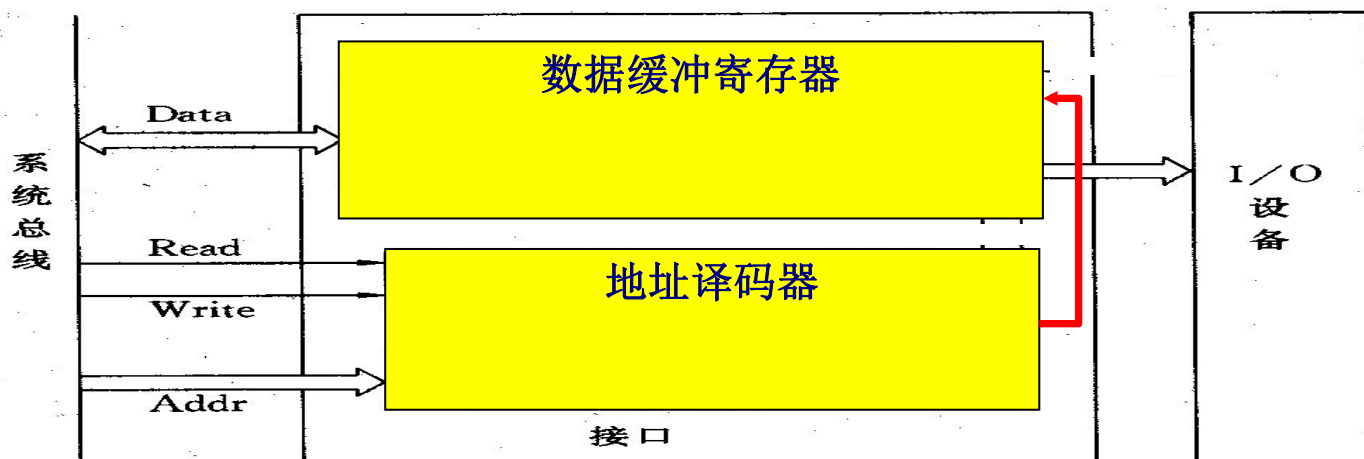
基本思想：由CPU执行一段输入输出程序来实现主机与外设之间数据传送控制。即外设的整个工作过程均在CPU的监控之下。

根据CPU与外设之间传送数据的时机是由CPU决定还是由外设决定的，程序直接控制方式可分为无条件传送和有条件传送（即程序查询）方式。

7.3.1 程序查询方式

1. 无条件传送方式（接口最简单）

CPU无需询问I/O接口的状态，可随时直接输入或输出数据。如信号灯控制。



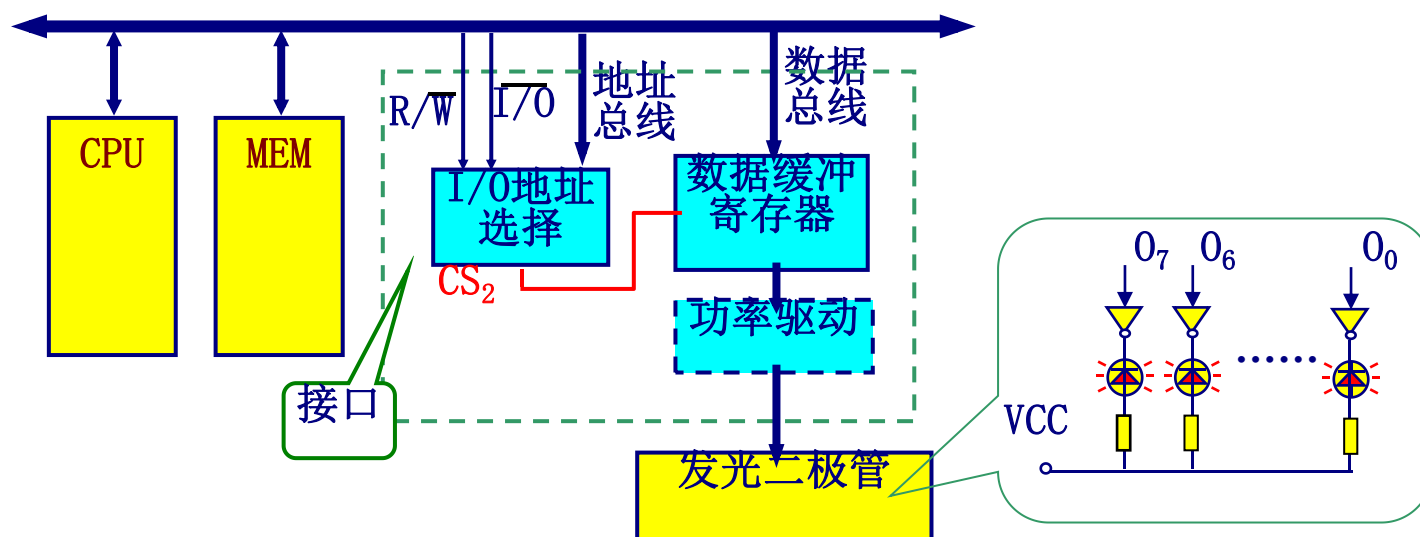
(a) 程序查询方式的接口

【例】向端口FEH输出数据(8位)，显示在发光二极管上。

假定：I/O设备独立编址，I/O地址为8位(A_{7-0})

输出指令为 OUT (设备地址), A ; A为CPU中的累加器

输出指令对应总线上控制信号 $\overline{I/O}=0$, $R/\overline{W}=0$

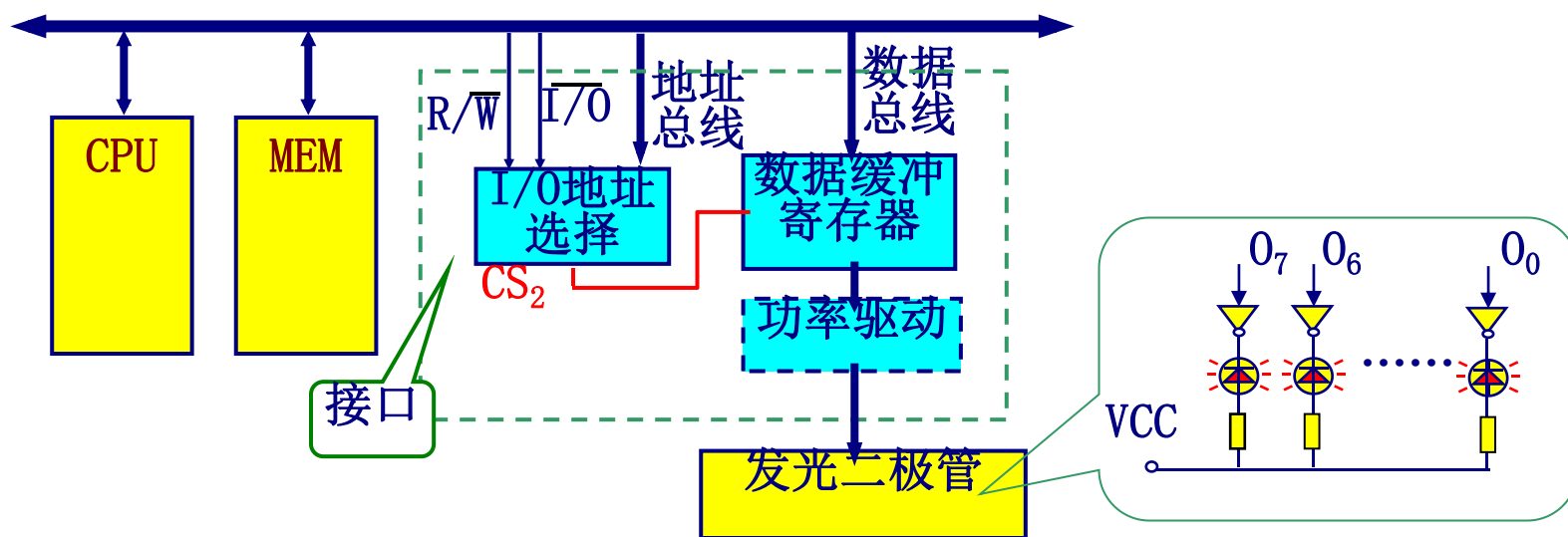


【例】向端口FEH输出数据(8位)，显示在发光二极管上。

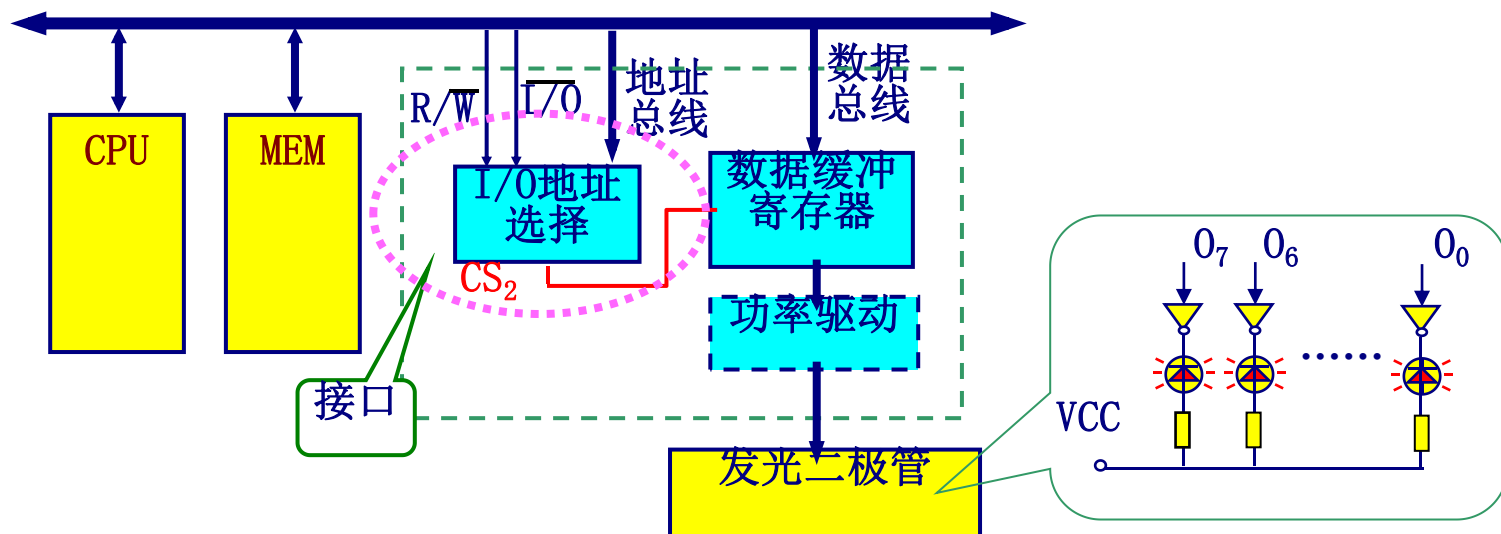
假定：I/O设备独立编址，I/O地址为8位(A_{7-0})

输出指令为 OUT (设备地址), A ; A为CPU中的累加器

输出指令对应总线上控制信号 $\overline{I/O}=0$, $R/\overline{W}=0$



【例】向端口 FEH 输出数据 (8 位)，显示在发光二极管上。



① I/O 地址选择

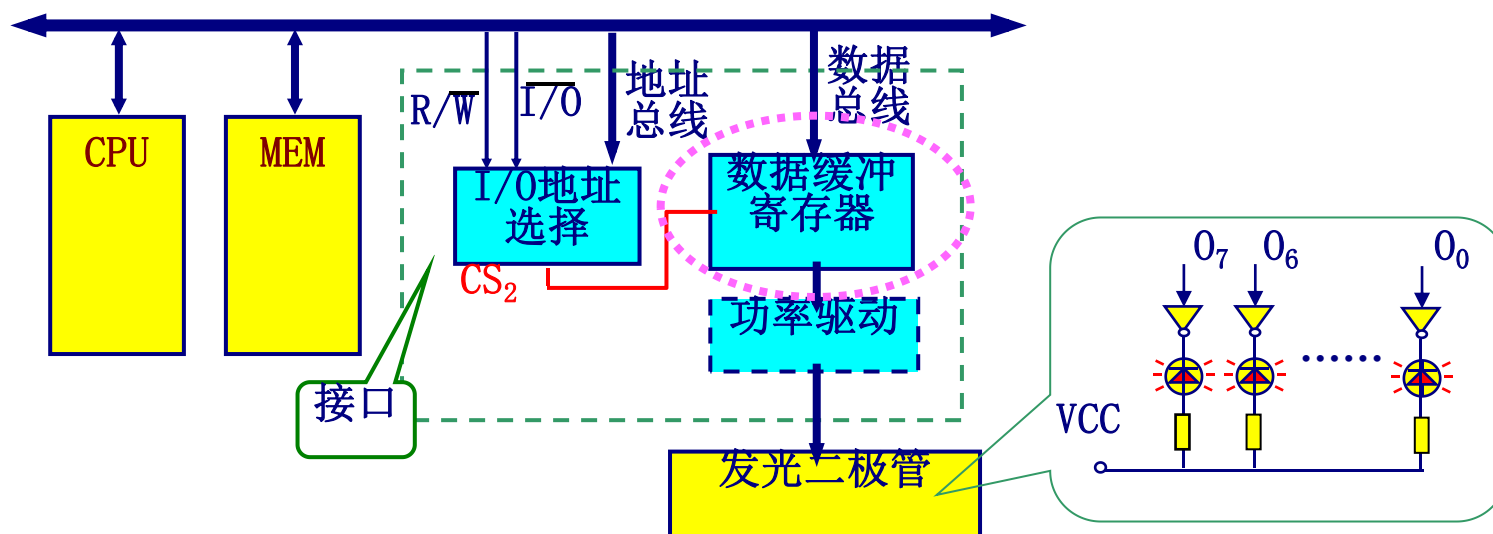
∵ CPU 通过执行指令 “OUT (0FEH), A” 实现输出，此时 CPU 将向地址线发地址值 FEH，向控制线发相应的控制信号，向数据线发送数据，

∴ I/O 地址选择逻辑（正逻辑）应为：

$$CS_2 = A_7 A_6 A_5 A_4 A_3 A_2 A_1 A_0 (\overline{I/O}) (\overline{R/W})$$

使输出接口的数据缓冲寄存器从系统总线的数据线上接收数据。

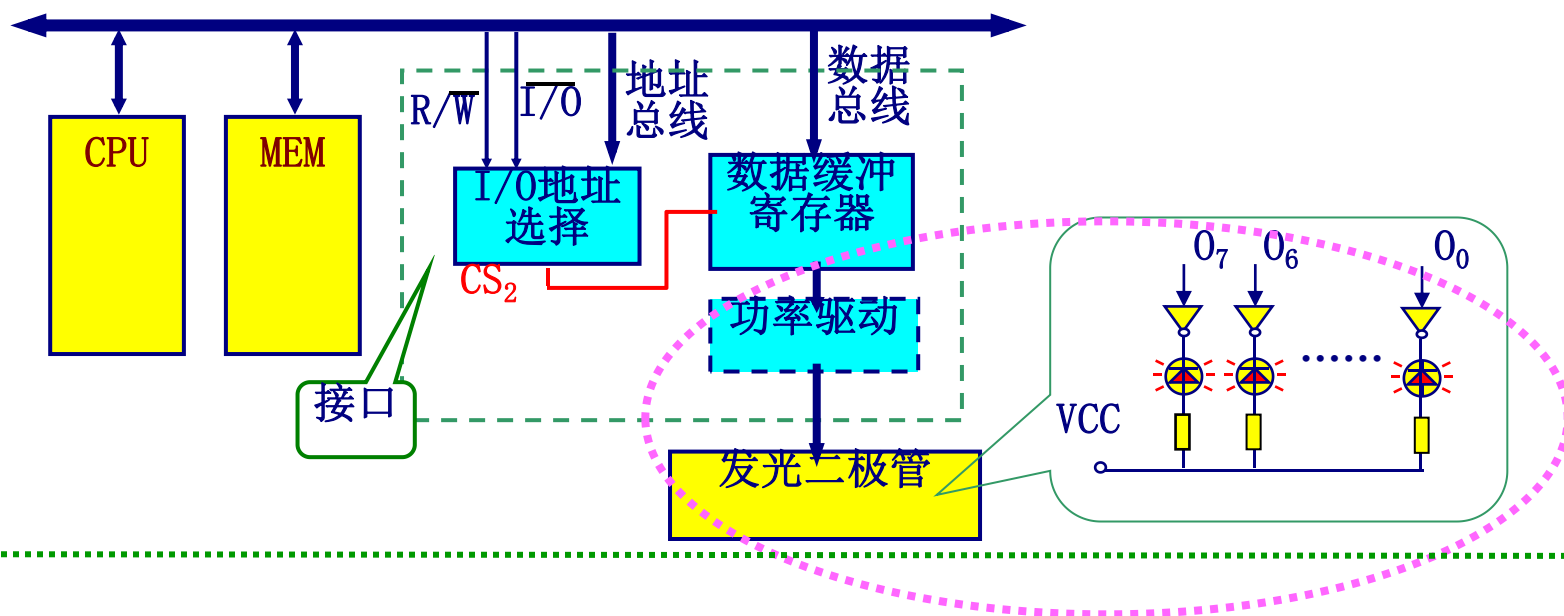
【例】向端口 FEH 输出数据 (8 位)，显示在发光二极管上。



②数据缓冲寄存器

可用74LS273(八D锁存器)等。其脉冲打入端由地址选择电路进行控制。

【例】向端口 FEH 输出数据 (8 位)，显示在发光二极管上。



③ 驱动与显示

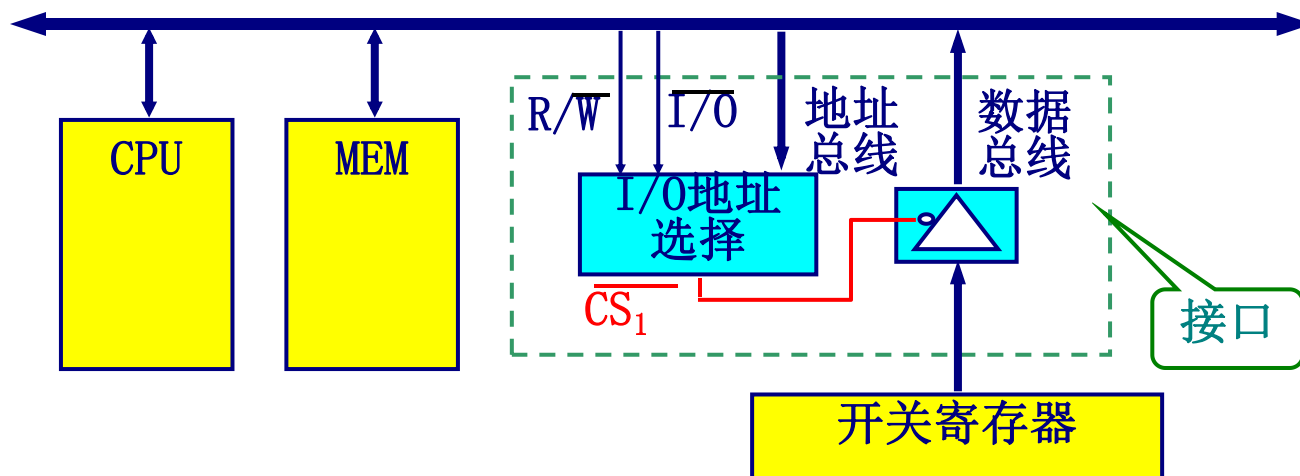
输出为“1”的位对应的发光二极管发亮。

例：从开关寄存器(8 位)输入数据，输入端口地址为 FEH.

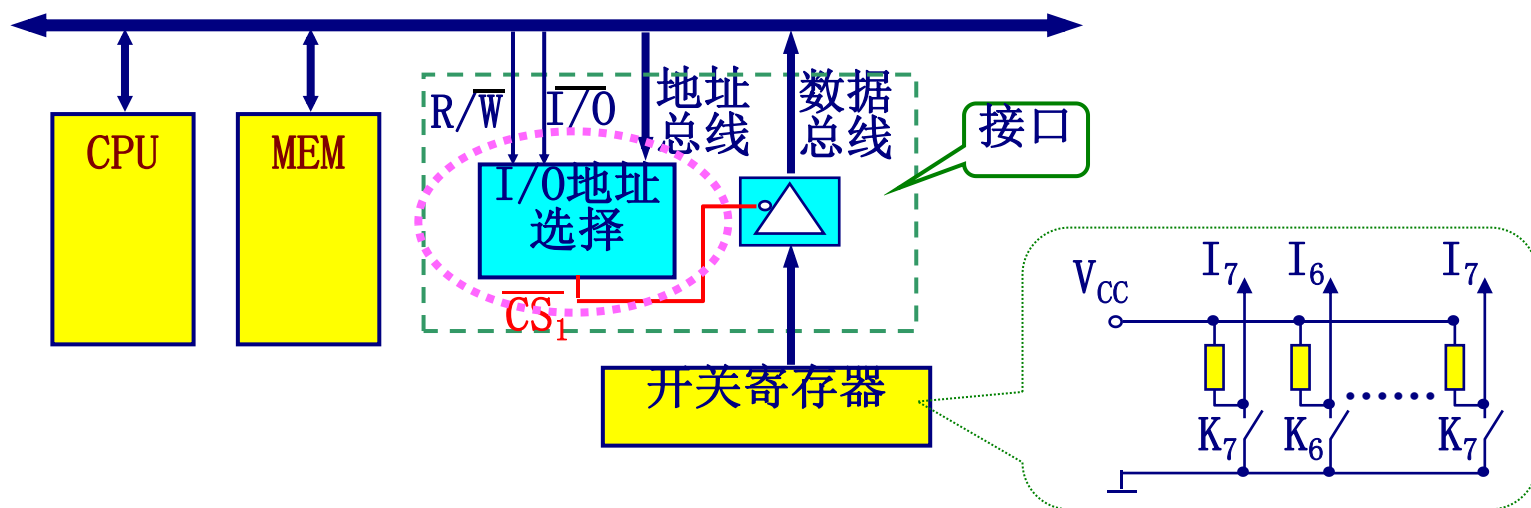
假定：I/O 设备独立编址，I/O 地址为 8 位 (A_{7-0})

输入指令为 IN A, (设备地址) ； A 为 CPU 中的累加器

输入指令对应总线上控制信号 $\overline{I/O}=0$, $R/\overline{W}=1$



例：从开关寄存器(8 位)输入数据，输入端口地址为 FEH.



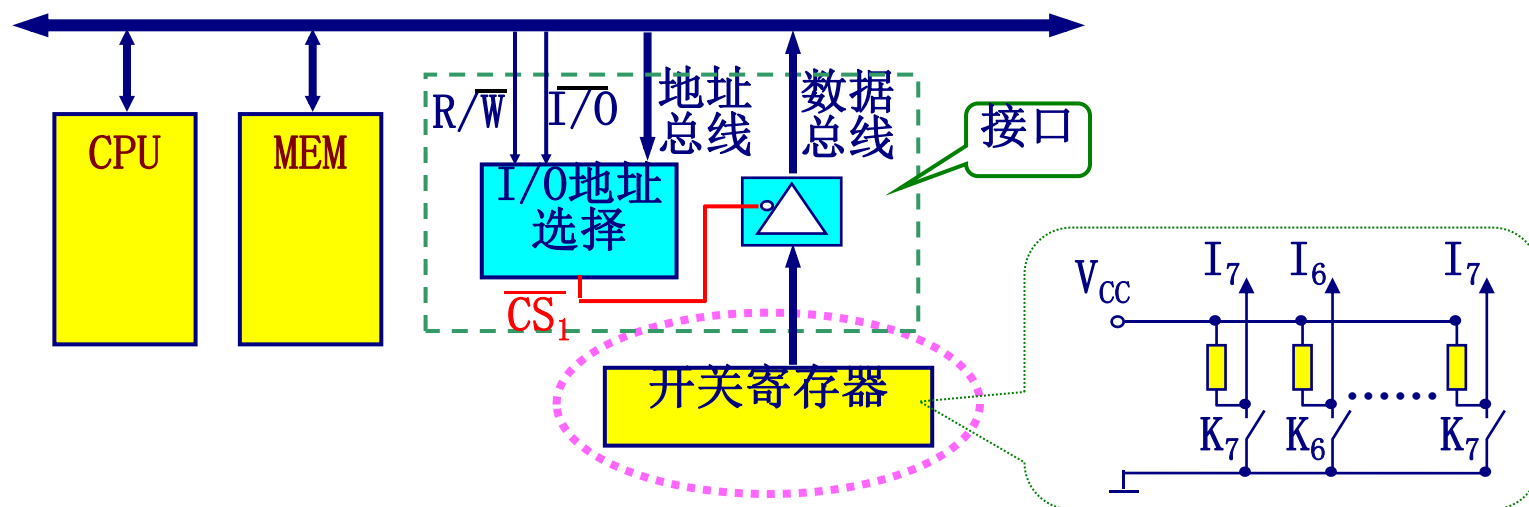
① I/O 地址选择

∴ CPU 通过执行指令 “IN A, (0FEH)” 实现输入，此时 CPU 将向地址线发地址值 FEH，向控制线发相应的控制信号，

∴ I/O 地址选择逻辑（负逻辑）应为：

$$\overline{CS_1} = \overline{A_7 A_6 A_5 A_4 A_3 A_2 A_1 A_0} \cdot (\overline{I/O}) \cdot (R/\overline{W})$$

例：从开关寄存器(8 位)输入数据，输入端口地址为 FEH.



② 数据缓冲寄存器

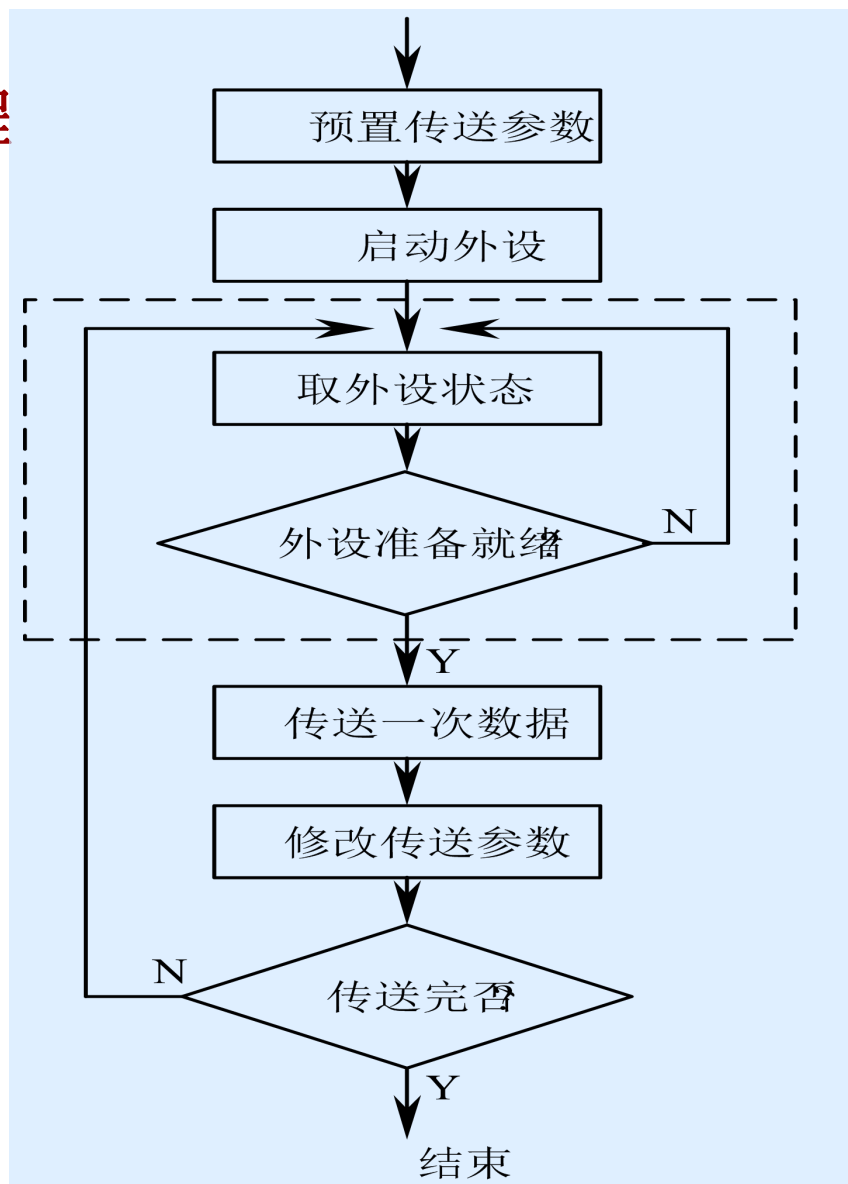
由于输入设备（开关寄存器）随时都可以提供数据，所以不必再设数据缓冲寄存器。

7.3.1 程序查询方式

2. 程序查询方式的工作流程

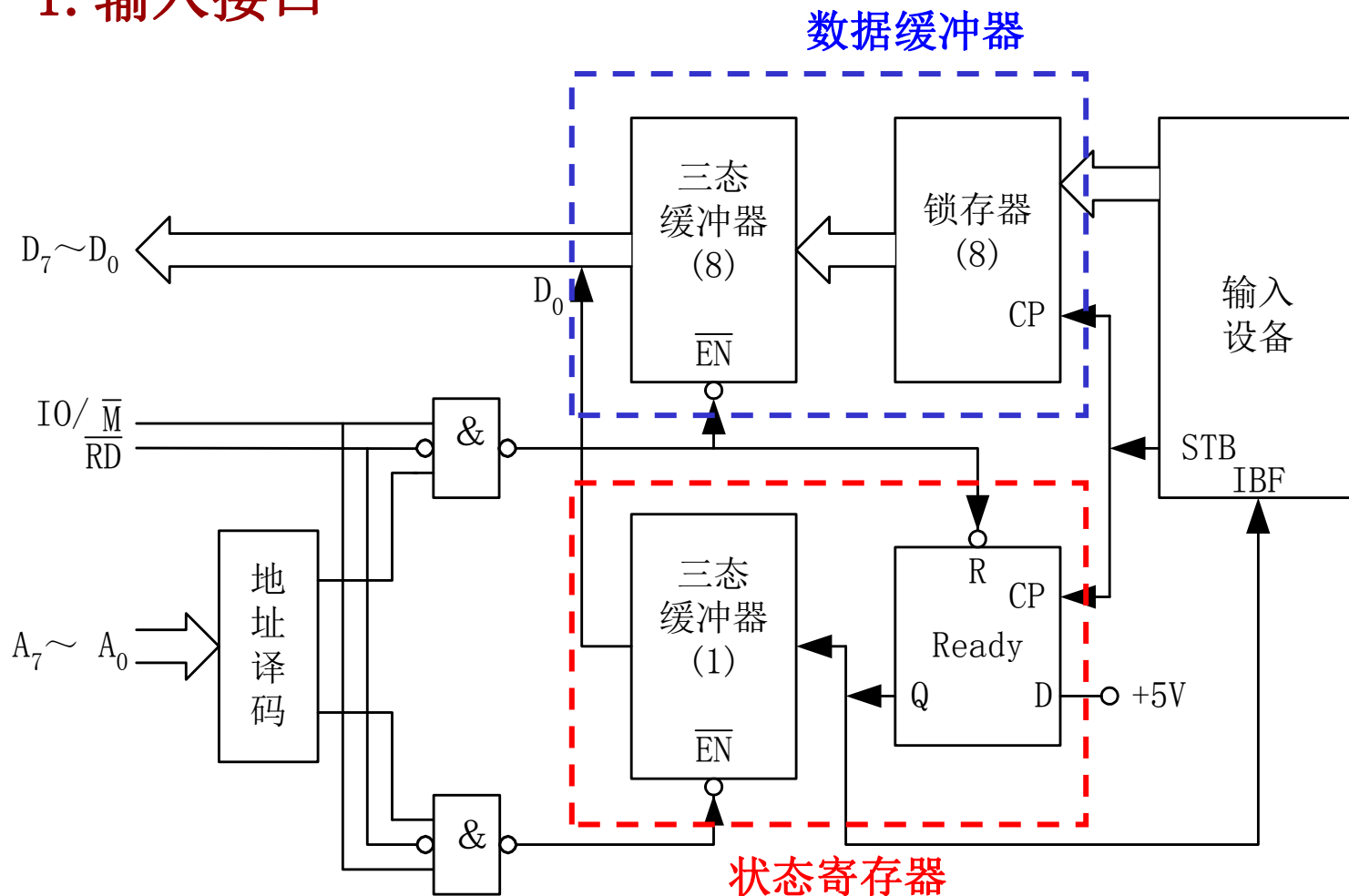
CPU需要根据外设的工作状态来决定是否进行数据传送，随时对接口状态进行查询。

特点：简单，不需要复杂的硬件接口。但CPU完全被外设占有，不能做其他任何事情，效率低，浪费了许多CPU时间。

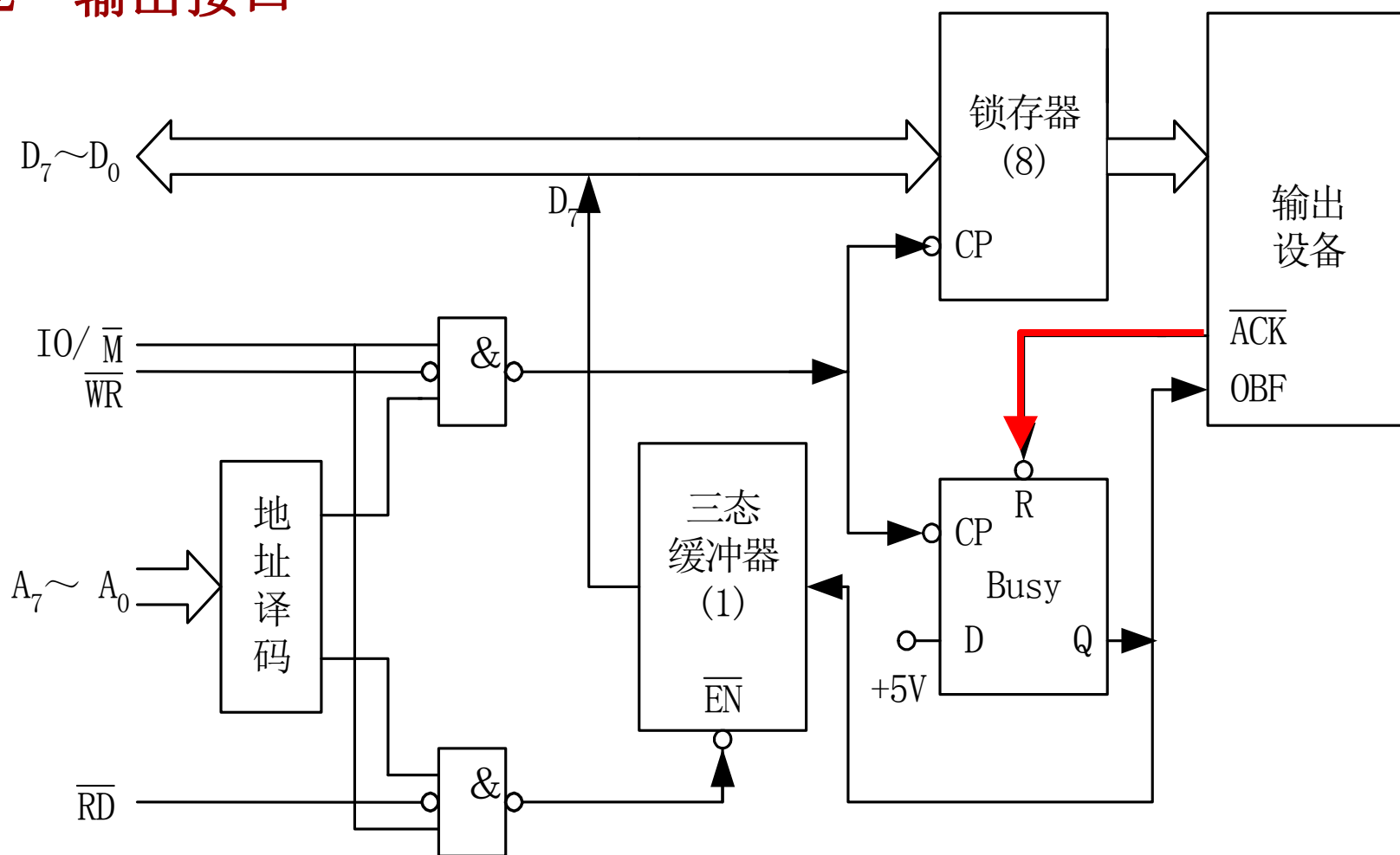


7.3.2 程序查询方式接口

1. 输入接口



2 输出接口



思考题： P348 1, 2, 3



§ 7.4 中断系统和程序中中断方式

7.4.1 中断的基本概念

1. 中断的提出

基本思想：CPU启动外设后，不再查询等待I/O接口，而是继续执行程序处理其他事务，I/O接口在必要的时候主动向CPU发信息（中断请求）。

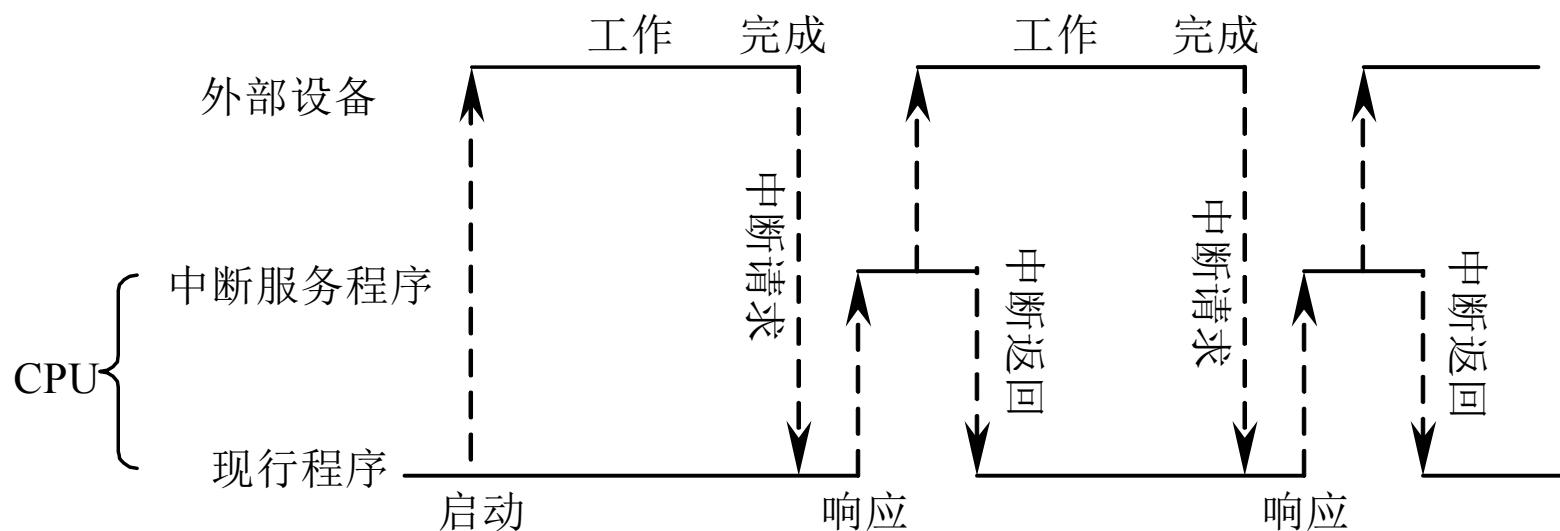
中断(程序中中断)：指CPU在执行现行程序的过程中，出现了某些突发事件急待处理，CPU必须暂停正在执行的程序，转去处理突发事件，处理结束后又返回到原程序被中断的位置继续执行。

中断的引入，使CPU能与多台外设并行工作，并具有了处理突发事件的能力。

2. 中断的特征

- (1) 程序切换； 现执行程序→中断服务程序→现执行程序
- (2) 随机性。

中断系统是计算机实现中断功能的软、硬件总称。一般在CPU中配置中断机构，在外设接口中配置中断控制器，在软件上设计相应的中断服务程序。



3. 中断的基本类型

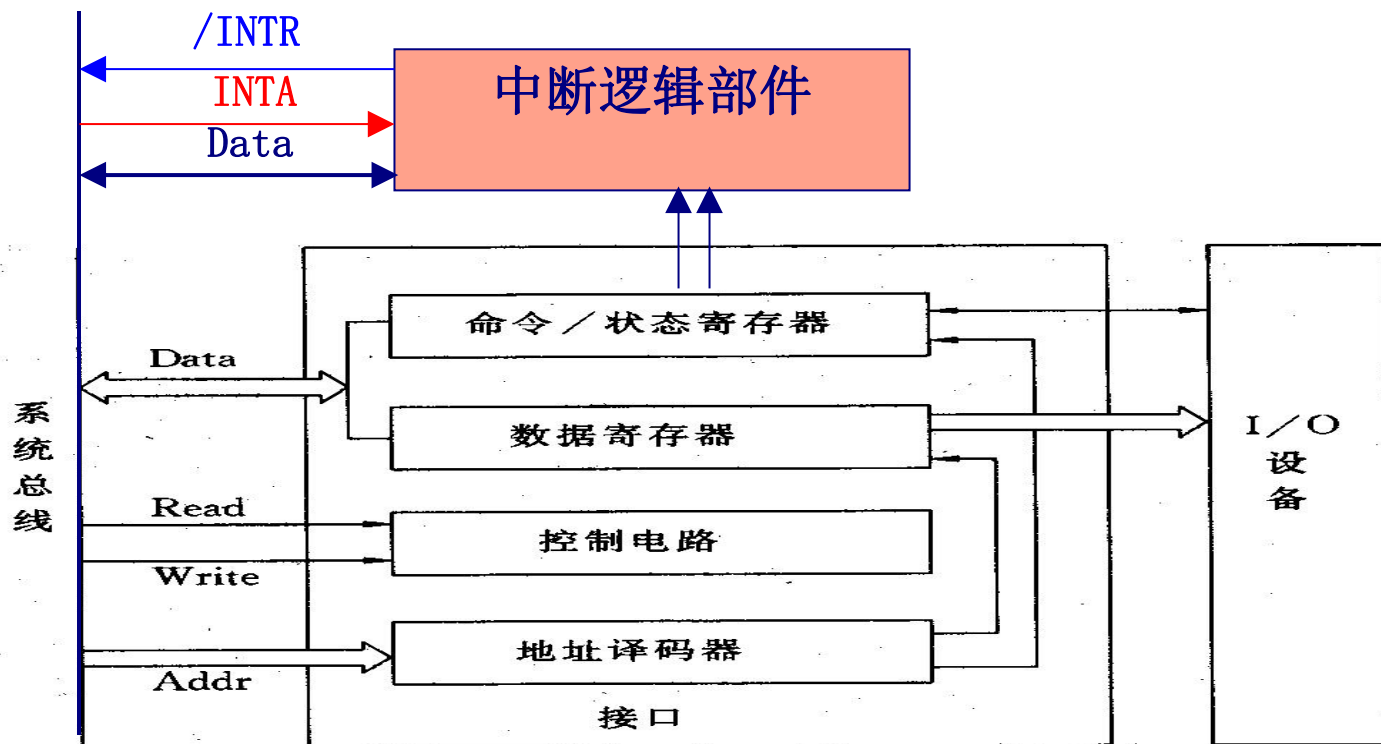
按不同的分类方法，有：

- (1) 自愿中断 / 强迫中断
- (2) 程序中断 / 简单中断(早期DMA方式的一种叫法)
- (3) 内中断 / 外中断
外中断又可以分为不可屏蔽中断和可屏蔽中断
- (4) 向量中断 / 非向量中断
- (5) 单重中断和多重中断

向量中断：发出中断请求的外设主动向CPU发出一个识别代码（称为中断向量），CPU通过中断向量识别各个中断源，并产生中断服务程序的入口地址。

7.4.2 中断接口模型和中断的全过程

1. 向量中断接口模型

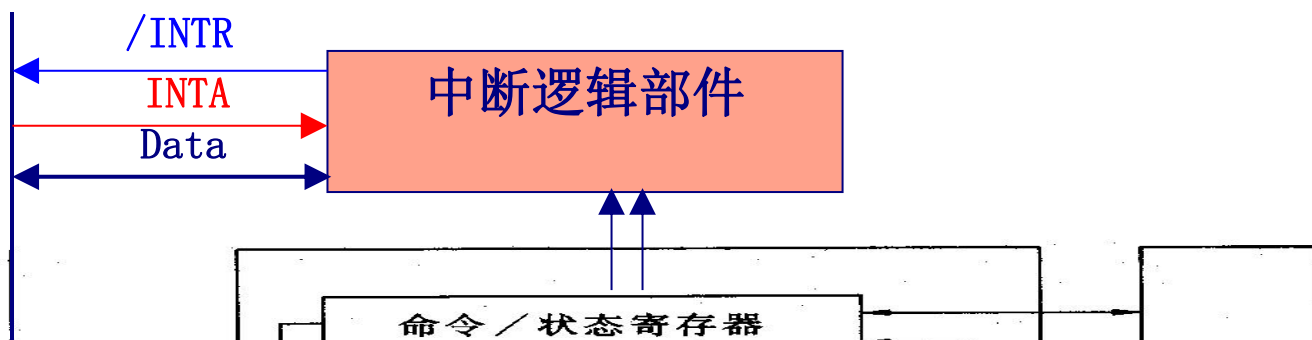


(a) 程序查询方式的接口

比程序查询方式的接口多了一个中断逻辑部件。

7.4.2 中断接口模型和中断的全过程

1. 向量中断接口模型



中断控制机构包括：

1) 中断请求电路

向CPU发出中断请求信号

2) 中断优先电路

保证优先级高的中断源先获得服务

3) 向量地址形成部件

产生向量中断时需要的向量地址。

7.4.2 中断接口模型和中断的全过程

2. 中断全过程

中断全过程分为5个阶段：

中断请求

中断判优

中断响应

中断处理

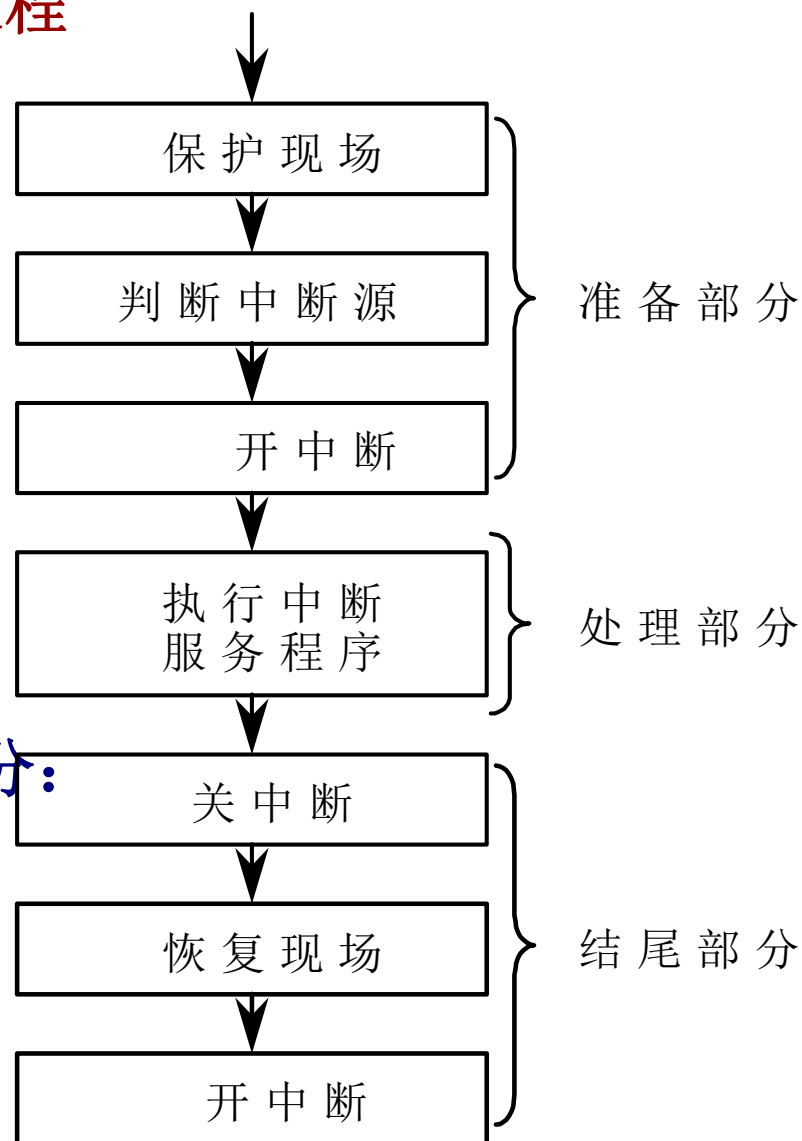
中断返回

其中中断处理又分为三个部分：

准备部分

处理部分

结尾部分



7.4.3 中断请求和中断判优

1. 中断源和中断请求信号

中断源：引起计算机中断的事件。

为了记录中断事件并区分不同的中断源，可采用中断请求触发器（INTR）来记录中断请求。当某一个中断源有中断请求时，其相应的中断请求触发器置成“1”状态。

2. 中断请求信号的传送

- 1) 独立请求线
- 2) 公共请求线
- 3) 二维结构

3. 中断优先级与判优方法

当同时有多个中断请求时，需根据中断的性质和处理的轻重缓急安排优先级。一般：

- ✓ 故障引起的中断优于I/O操作
- ✓ 非屏蔽优于可屏蔽
- ✓ 高速I/O的中断优于低速I/O
- ✓ 输入设备的中断优于输出设备的中断

3. 中断优先级与判优方法

当同时有多个中断请求时，需根据中断的性质和处理的轻重缓急安排优先级。

中断判优的方法可由硬件实现，也可通过软件实现。

(1) 软件判优法

通过执行查询程序逐个检测中断请求寄存器的各位状态，检测顺序按优先级大小排列

特点：简单，可以灵活地修改中断源的优先级别；
判优速度慢且占用CPU时间。

3. 中断优先级与判优方法

(1) 软件判优法

通过执行查询程序逐个检测中断请求寄存器的各位状态,检测顺序按优先级大小排列

特点: 简单,可以灵活地修改中断源的优先级别;
判优速度慢且占用CPU时间。

(2) 硬件判优电路

根据中断请求信号的传送方式不同,有不同的优先排队电路。

现代微机采用中断优先权编码电路,可通过编程规定其优先级:循环优先级/固定优先级等。

硬件判优特点: 节省CPU时间,速度快。

4. 关中断和中断屏蔽

(1) 开中断与关中断

由CPU中的中断允许触发器（EINT）控制：

EINT=1时开中断（受理中断请求）；

EINT=0，关中断。

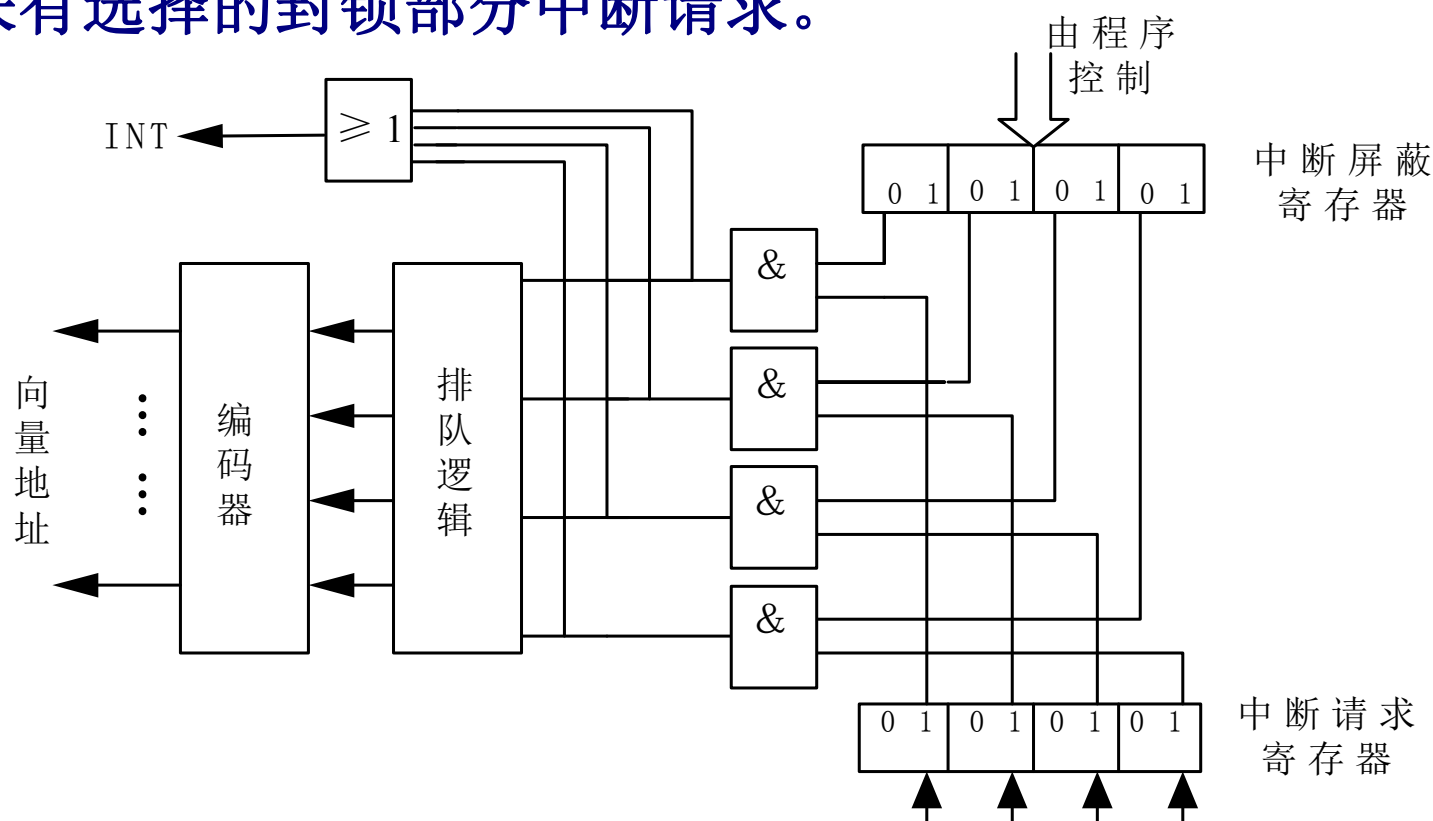
(2) 中断屏蔽

用程序方式有选择地封锁部分中断源发出中断请求，使之不能真正送到CPU去，这就是中断屏蔽。

4. 关中断和中断屏蔽

(2) 中断屏蔽

例如，对每个中断源设置一个中断屏蔽触发器（MASK），来有选择的封锁部分中断请求。



7.4.4 中断响应和中断处理

1. CPU响应中断的条件

(1) CPU接收到中断请求信号

中断源要能发出中断请求(未被屏蔽)，且能被CPU接收到。

(2) CPU允许中断（即开中断，在CPU内部）。

但非屏蔽中断不受此限制。

(3) 一条指令执行完毕，且没有DMA请求

这是CPU响应中断请求的时间限制条件。

7.4.4 中断响应和中断处理

2. 中断周期（又称为中断响应周期）

是CPU响应中断后从原程序转向中断处理程序的过渡阶段，它完成**中断隐指令**的操作。

中断隐指令：

指CPU响应中断之后，经过某些操作转去执行中断服务程序。这些操作是由硬件直接实现的，称为中断隐指令。

中断隐指令完成的操作主要有：

1) 保存断点

将PC值压入堆栈，也可以存入主存的特定单元中

2) 关中断

为了保护中断现场（即CPU的主要寄存器状态）期间不被新的中断所打断。

注：并不是所有的计算机都在中断隐指令中由硬件自动地关中断。

3) 引出中断服务程序

获取中断服务程序的入口地址。

注意：中断隐指令不是指令系统中真正的指令，它没有操作码，是不允许、也不可能为用户使用的特殊指令。

7.4.4 中断响应和中断处理

3. 进入中断服务程序

(1) 软件方法

由中断隐指令控制进入一个中断总服务程序，在那里判优、寻找中断源并且转入相应的中断服务程序。

特点：方便、灵活，硬件极简单，但效率是比较低的。

(2) 硬件向量中断法

在CPU响应某一中断请求时，需要中断源将中断向量传送给CPU，以引导CPU确定中断处理程序的入口地址，这就是中断向量的呈送。

7.4.4 中断响应和中断处理

4. 中断现场的保护和恢复

中断现场：指发生中断时CPU的主要状态，包括断点和一些通用寄存器的状态。

5. 多重中断（中断嵌套）

较低的中断处理程序可被新的优先级较高事件所中断。

要使计算机具有中断嵌套的能力，有两个关键点：

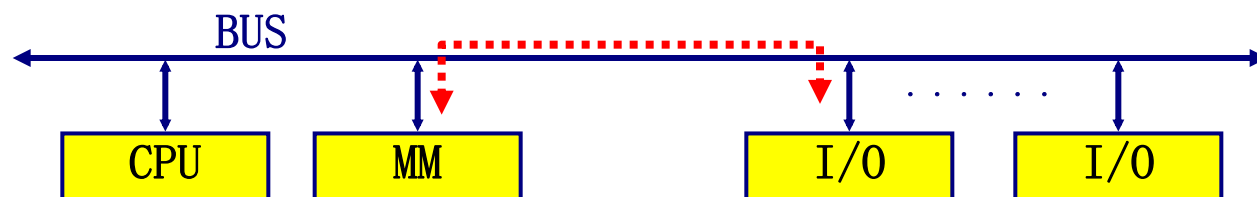
- 1) 要能保护多个断点（通常用堆栈来保存断点）；
- 2) CPU进入中断处理程序后，系统必须处于开中断状态。

§ 7.5 DMA方式及其接口

7.5.1 DMA方式的基本概念

1. DMA方式及应用场合

DMA方式是在外设和主存间开辟一条“直接数据通道”，在不需CPU干预也不需要软件介入的情况下在两者间进行的高速数据传送方式。



一般用在主存与高速外设间的简单数据传送。如：磁盘、光盘等外设接口；网络通信接口；高速数据采集接口等。

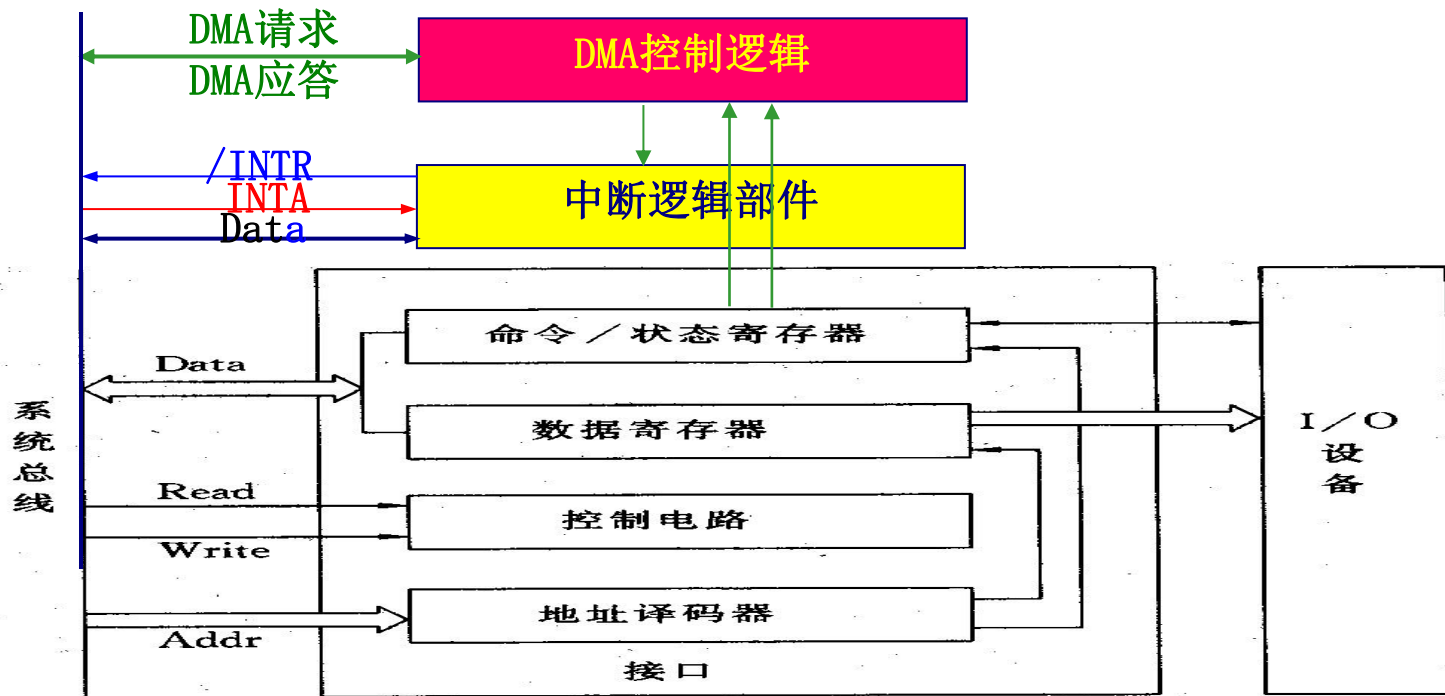
DMA方式用于传送整批数据。

整批数据传送开始前要通过程序进行**预处理**，结束后通过中断方式进行**后处理**。

在整批数据传送期间，每传一个数据都要向CPU发一次DMA请求。数据传送通常采用周期挪用工作方式。

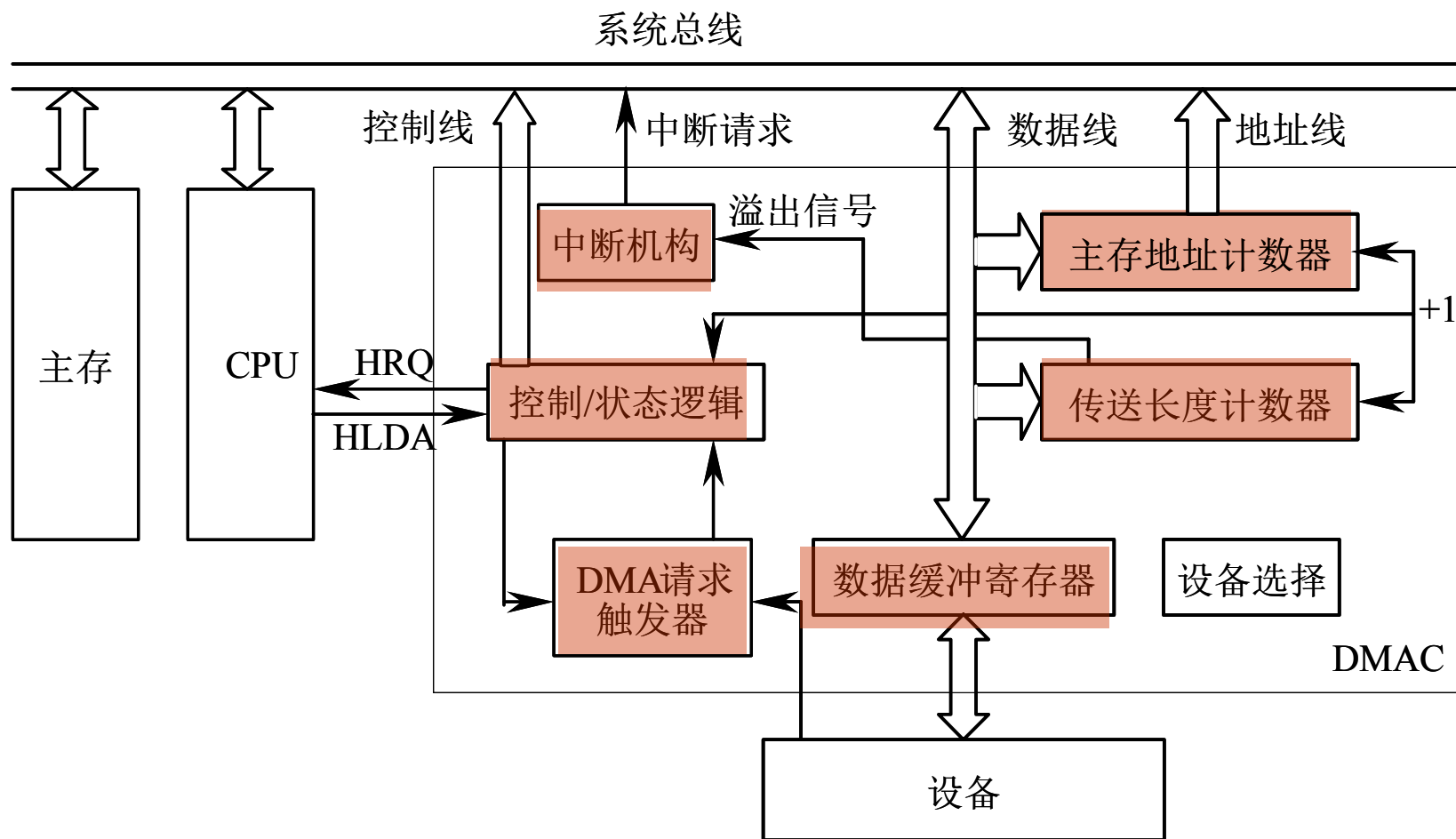
7.5.2 DMA接口（又称为DMA控制器）

DMA控制器必须具有控制系统总线的能力, 能够像CPU一样输出地址信号, 接收或发出控制信号, 输入或输出数据信号。



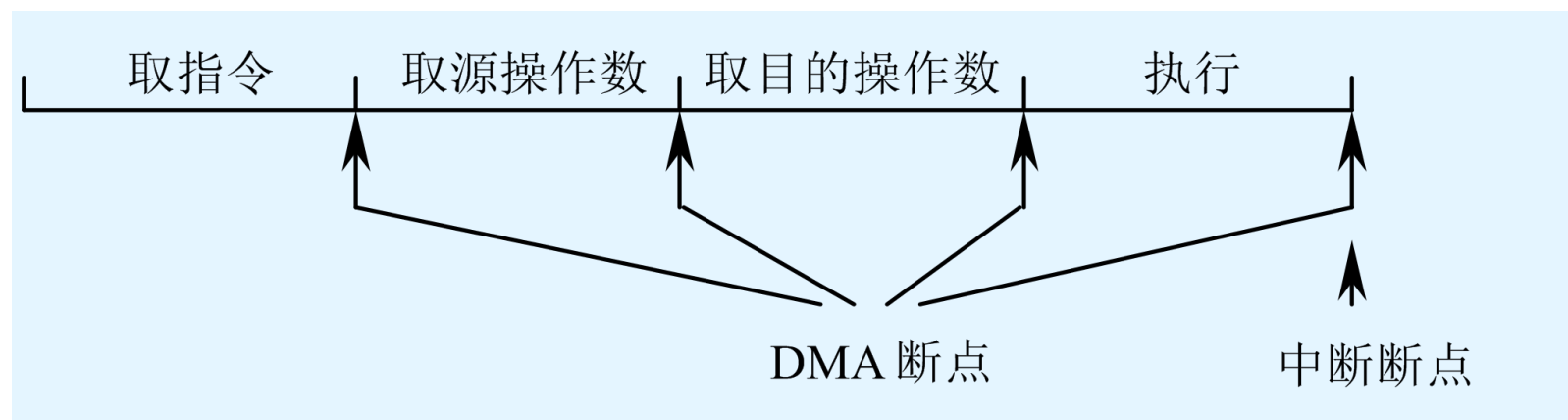
比中断方式多了DMA控制逻辑部件, 该部件包括:
内存地址寄存器和长度寄存器。

DMA控制器的基本组成



7.5.3. DMA和中断的区别

	中断方式	DMA方式
程序转换	有程序切换、需要保护和恢复现场	无程序切换、不占用CPU任何资源
响应时间	只能发生在每条指令执行完毕时	可以发生在每个机器周期结束时
优先级	DMA请求的优先级高于中断请求	
处理能力	有对异常事件处理能力	仅局限于完成传送数据块的I/O操作



§ 7.6 通道控制方式

7.6.1 通道的基本概念

通道是一种高级的I/O控制部件，基本任务是通过执行程序来管理输入输出工作。

1. 通道控制方式与DMA方式的区别

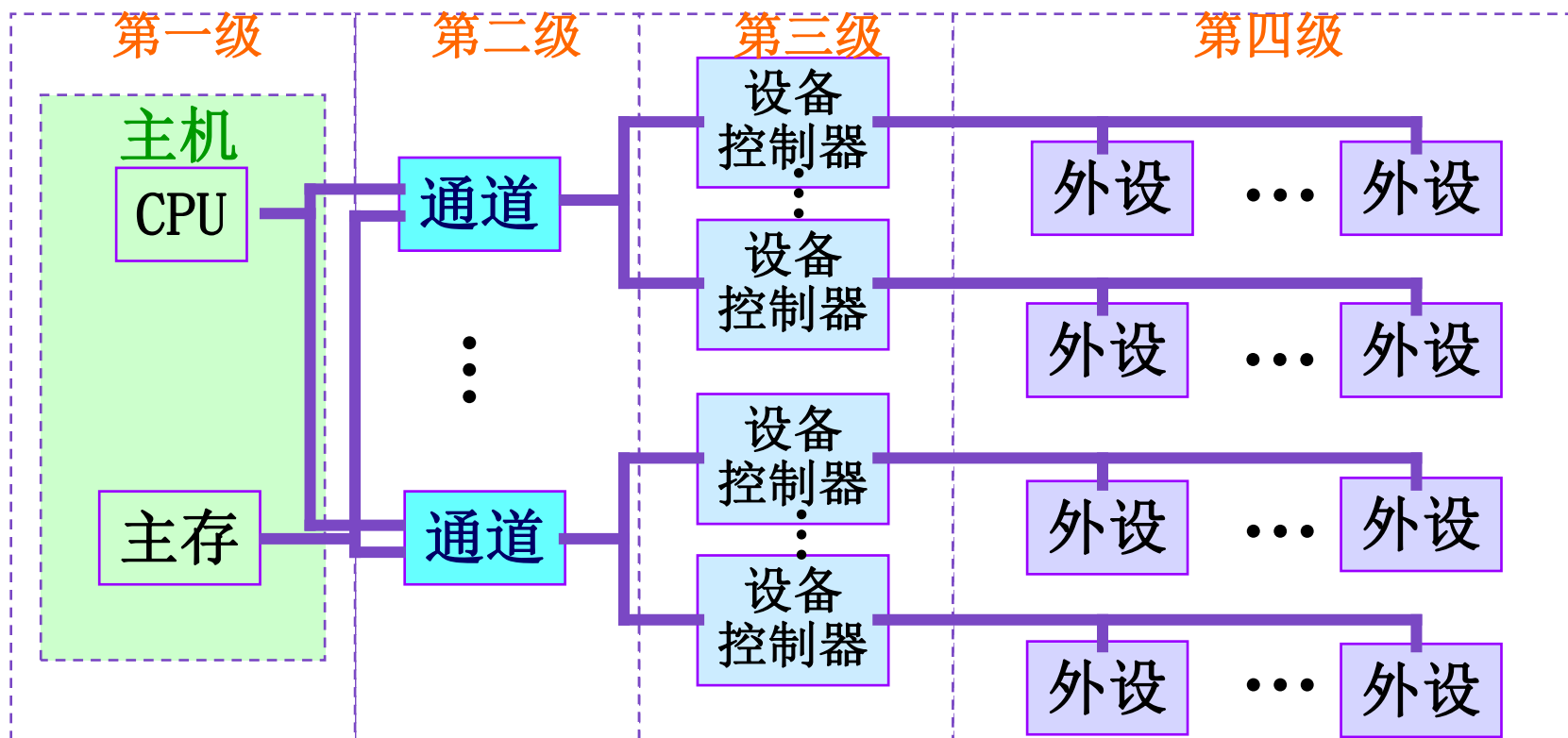
(1) DMA控制器是通过专门设计的硬件控制逻辑来实现对数据传送的控制；而通道则是一个具有特殊功能的处理器，它具有自己的指令和程序，通过执行通道程序来实现对数据传送的控制，故通道具有更强的独立处理数据输入输出的功能。

(2) DMA控制器通常只能控制一台或少数几台同类设备；而一个通道则可以同时控制许多台同类或不同类的设备。

2. 通道的功能

从逻辑结构上说，通道控制方式具有4级连接：

主机→通道→设备控制器→外设。



通道具有以下功能：

- 接受CPU的I/O指令，按指令要求与指定外设进行联系；
- 从主存取出属于该通道程序的通道指令，经译码后向设备控制器和设备发送各种命令。
- 实施主存和外设间的数据传送。
- 从外设获得设备的状态信息，形成并保存通道本身的状态信息，并送到主存指定单元供CPU访问。
- 将外设的中断请求和通道本身的中断请求按次序及时报告CPU。

3. 设备控制器的功能

- 从通道接收控制信号，控制外设完成相应操作。
- 向通道反馈外设的状态
- 将外设的各种不同信号转换为通道能识别的标准信号。

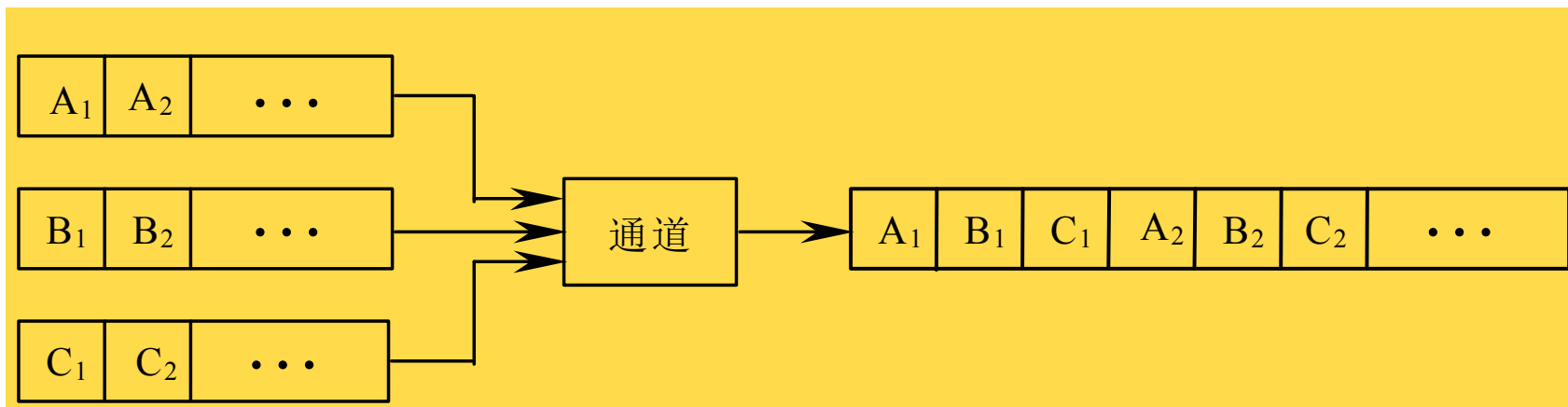
4. 通道的类型与结构

按照通道独立于主机的程度，可分为结合型通道和独立型通道两种类型。

按照输入输出信息的传送方式，可分为字节多路通道、选择通道和数组多路通道。

(1) 字节多路通道

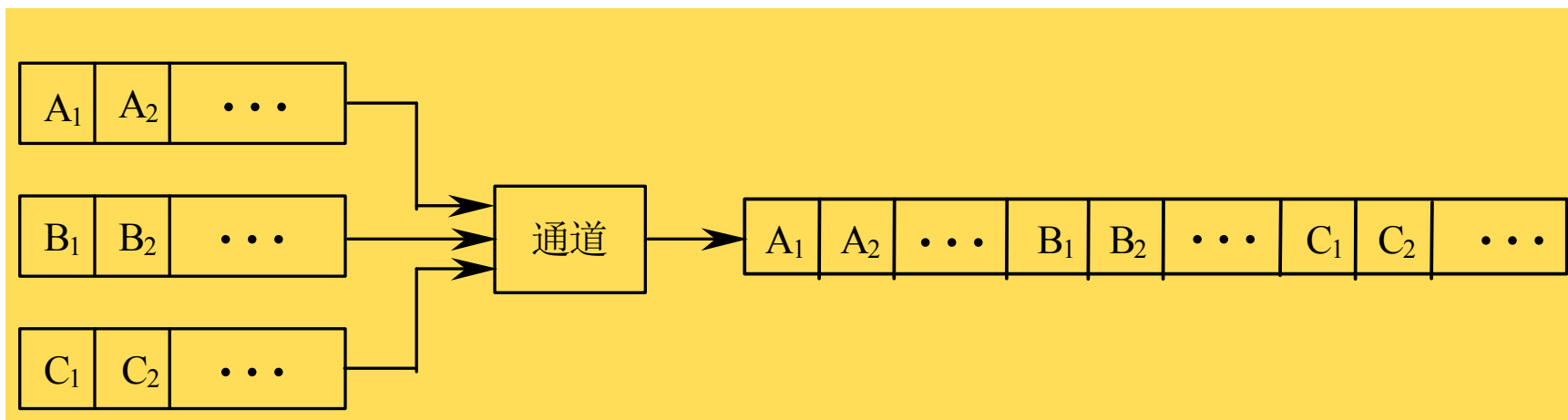
物理上可连接多个外设，且在一段时间内**以字节交叉方式轮流进行传送**多个外设的数据。字节多路型通道的数据传输率等于所连接外设的速度之和。



特点：用于连接多个慢速和中速设备。由于这些设备两次数据传输之间的间隔较长，利用这个间隔时间为其他设备服务，可提高IOP的利用率。

(2) 选择通道

物理上可连接多个设备，但这些设备不能同时工作，同一时间内只能为一个设备服务。选择通道的数据传输率等于所连接外设中速度最高的外设速度。



特点：适合于连接高速外设，信息以成组方式高速传输；数据宽度可变，通道的效率不是很高。

(3) 数组多路通道

是前两种方式的结合，可连接多台高速外设，允许多台高速外设并行工作，数据传送的基本单位是数组。当某设备进行数据传送时，通道只为该设备服务；当设备在执行辅助操作时，通道暂时断开与该设备的连接，去为其他设备服务。数组多路通道的数据传输率与所连接的外设速率无关，完全取决于主存的读写周期。

特点：既保留了选择方式高速传输数据的优点，又充分利用了控制操作的时间间隔为其他设备服务，交替传输数据，提高了系统效率。

§ 7.7 总线技术

7.7.1 总线概述

1. 总线

能为系统中多个部件 **分时、共享** 的一组传输线及相关逻辑。它是构成计算机系统的互联机构。

总线系统的组成：

- 总线： 一组信息传输线；
- 总线接口： 三态门驱动器
缓冲寄存器；
- 总线协议及总线控制器： 维护总线
进行争用总线裁决。

2. 总线分类

按总线连接的部件分： 芯片内总线/系统总线/外总线

按传送方向分： 单向/ 双向

按数据传送格式分： 并行/ 串行

注： 总线的数据宽度——可同时传送的二进制位数。

按时序控制方式分： 同步/异步/准同步

3. 总线标准化

总线的标准化便于不同厂家之间设备的互连和互换，也为设计和组成实际的计算机应用系统提供了良好的环境。

常见的系统总线标准有：

PCI总线 Peripheral Component Interconnect

PC总线（8位）

ISA总线（16位，AT总线）

Industry Standard Architecture

.....

7.7.2 系统总线

系统总线包括：

- 数据总线 (Data Bus)
- 地址总线 (Address Bus)
- 控制总线 (Control Bus)
- 电源线

数据总线：实现数据传送，一般为双向传送。

地址总线：单向，只有掌握总线控制权的部件
(如CPU、DMA、IOP) 向其发送信号。

控制总线：传送各种控制/状态信号

控制总线的组成情况体现了不同总线的特点。

3. 总线标准化

总线的标准化便于不同厂家之间设备的互连和互换，也为设计和组成实际的计算机应用系统提供了良好的环境。

常见的外总线标准有:

RS-232C 、 RS485等 一 串行总线。例如串口

IEEE1284标准 一一并口。例如打印机的并口

IEEE488 ——智能仪器互连的并行异步总线

USB(通用串行总线) 一串口的替代产品，
即插即用并支持热插拔

IEEE-1394 --并口的替代产品（高速）

7.7.2 总线系统需解决的几个主要问题

1. 总线仲裁

为了解决多个主设备同时竞争总线控制权的问题，必须具有总线仲裁部件。它通过采用优先级策略或公平策略，选择其中一个主设备作为总线的下一次主方，接管总线控制权。

按照总线仲裁电路的位置不同，分为：

集中式仲裁：有一个中央仲裁器，它受理所有功能模块的总线请求，按优先原则或公平原则。

分布式仲裁：不需要中央仲裁器，每个功能模块都有自己的仲裁号和仲裁器。

7.7.2 总线系统需解决的几个主要问题

1. 总线仲裁 （是核心问题之一）

为了解决多个主设备同时竞争总线控制权的问题，必须具有总线仲裁部件。它通过采用优先级策略或公平策略，选择其中一个主设备作为总线的下一次主方，接管总线控制权。

在一次总线传输中，数据的传输涉及到两个部件，其中由一方控制总线、发出总线地址和读写命令，这一方称为**主设备**，而另一方称为**从设备**。

7.7.2 总线系统需解决的几个主要问题

2. 总线定时 （是核心问题之一）

为了同步主、从双方的操作，必须制订定时协议。

1) 同步定时协议

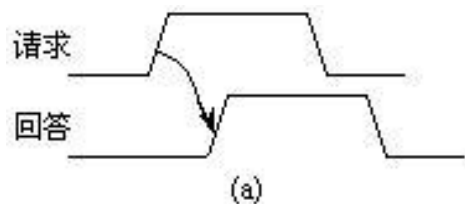
事件出现在总线上的时刻由总线时钟信号来确定，总线周期的长度是固定的。

2) 异步定时协议

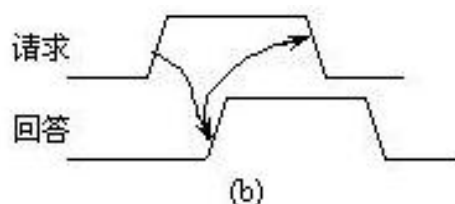
依靠传送双方的“握手”信号来实现定时控制, 无公用的时钟和固定的时间间隔。

特点：时间利用率高，具有很强的灵活性，但相应的控制较复杂。

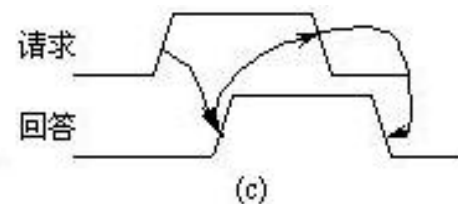
异步方式的三种应答方式：



不互锁



半互锁



全互锁

7.7.2 总线系统需解决的几个主要问题

3. 信息传输方式

- 1) 并行传送;
- 2) 串行传送;
- 3) 复用传送。

总线带宽：总线本身所能达到的最高传输速率。

7.7.2 总线系统需解决的几个主要问题

4. 当代流行的标准总线的追求

追求与结构、CPU、技术无关的开发标准。

其总线内部结构包含：

- ① 数据传送总线（由地址线、数据线、控制线组成）
- ② 仲裁总线
- ③ 中断和同步总线
- ④ 公用线（电源、地线、时钟、复位等信号线）

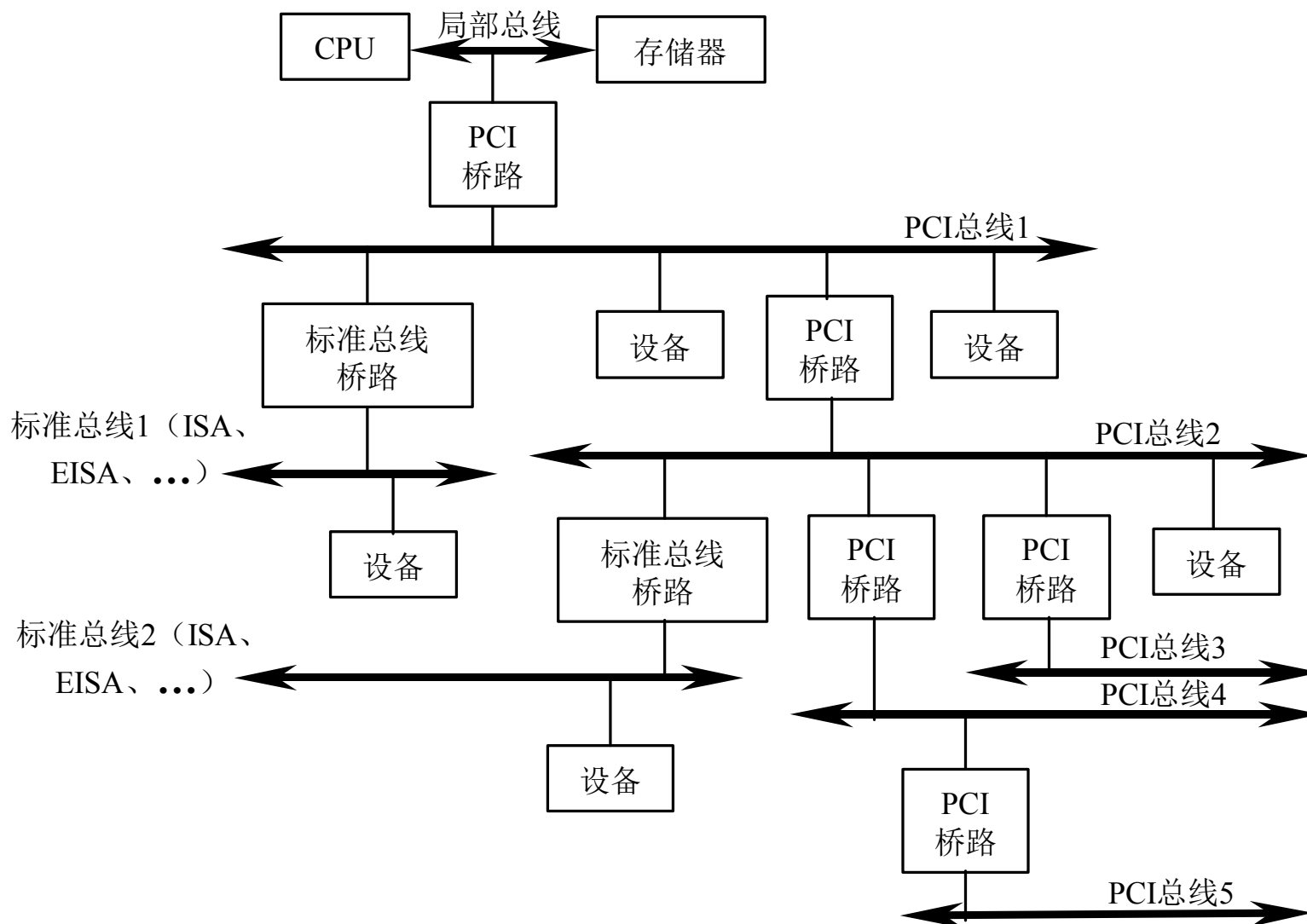
7.7.2 总线系统需解决的几个主要问题

4. 当代流行的标准总线的追求

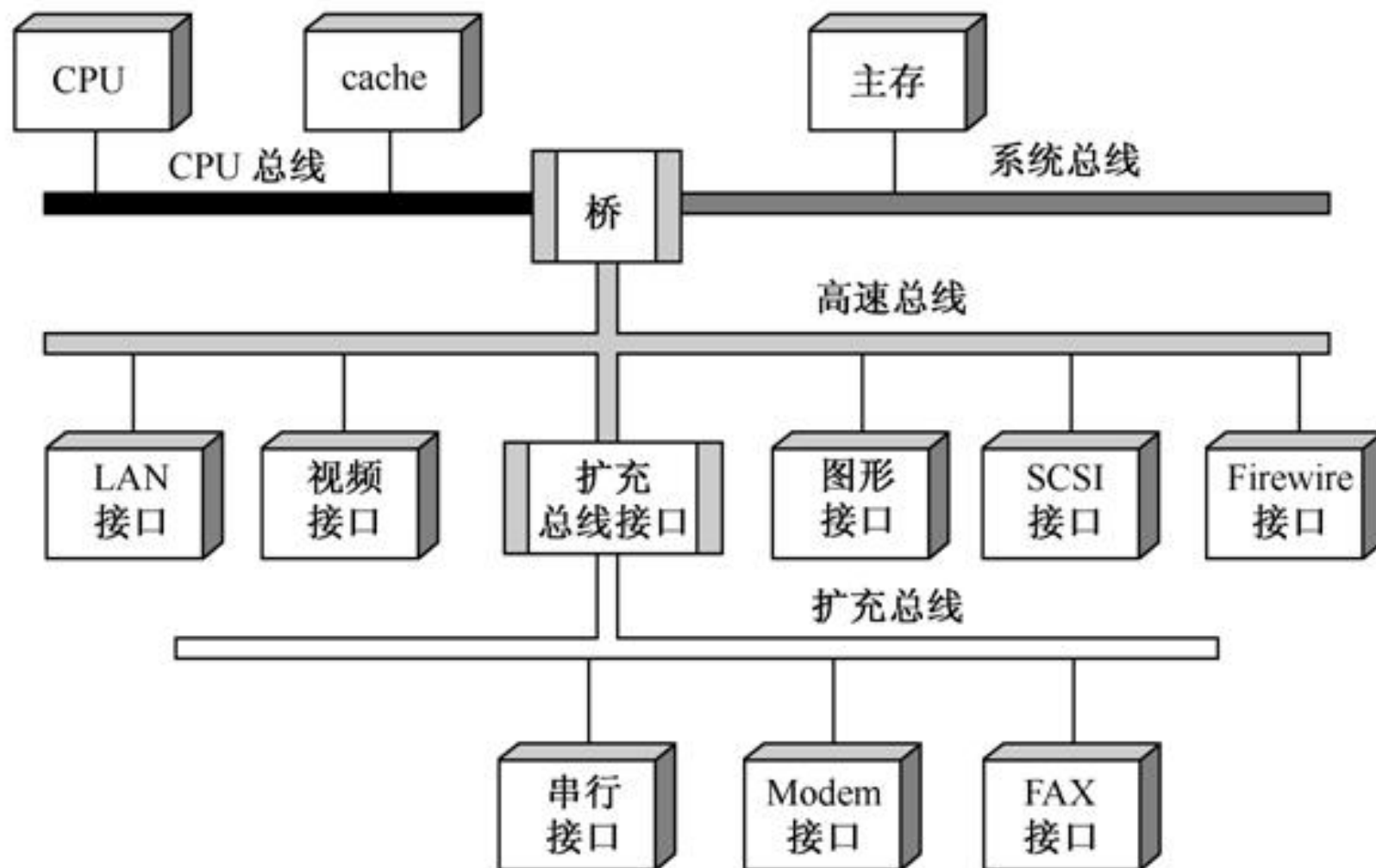
追求与结构、CPU、技术无关的开发标准。

PCI总线是当前实用的总线，是一个高带宽且与处理器无关的标准总线，又是重要的层次总线。它采用同步定时协议和集中式仲裁策略，并具有自动配置能力。PCI适合于低成本的小系统，因此在微型机系统中得到了广泛的应用。

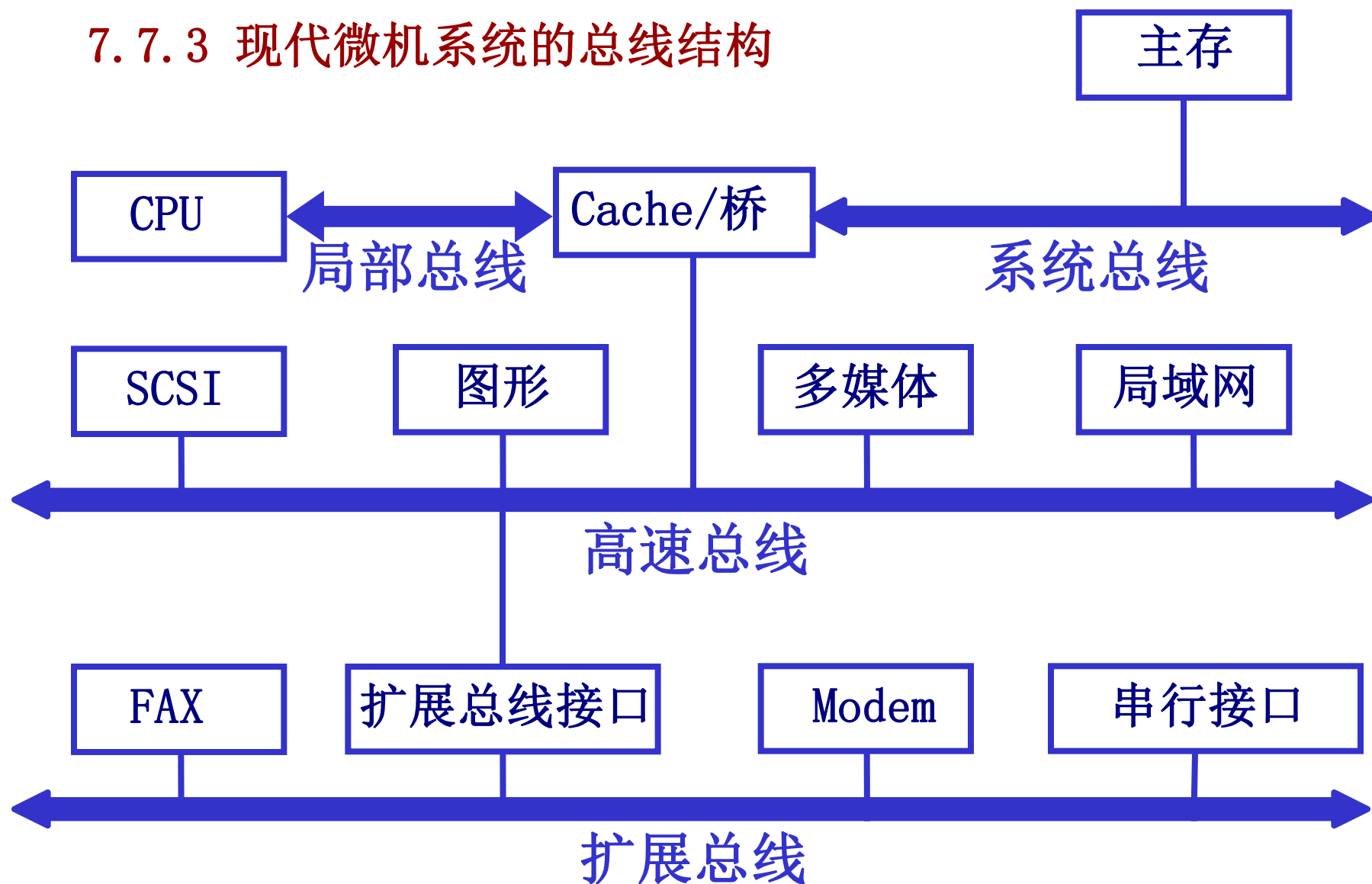
7.7.3 现代微机系统的总线结构



7.7.3 现代微机系统的总线结构



7.7.3 现代微机系统的总线结构



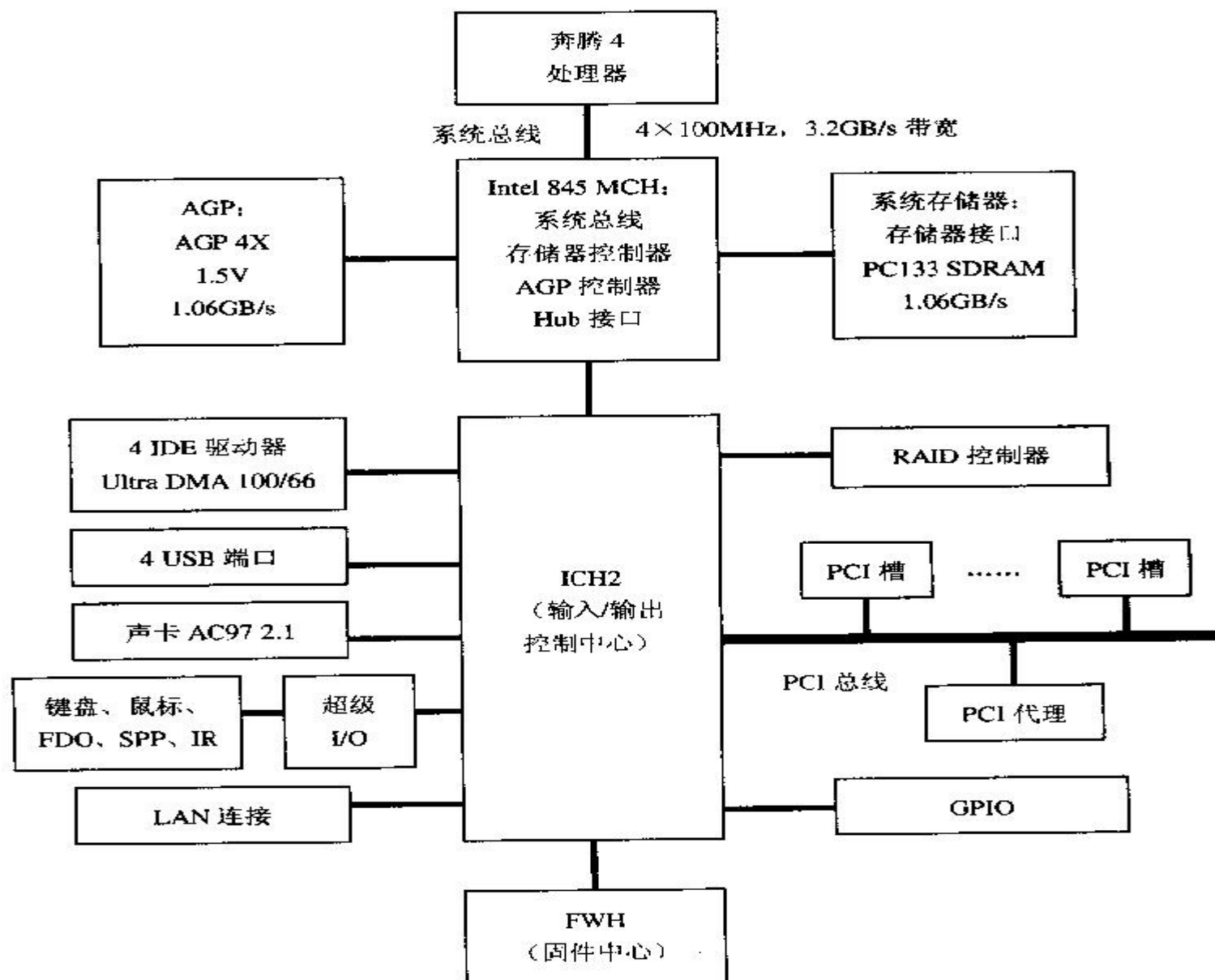


图 13.3 Intel 845 芯片组的主要特性

思考：P348 9--14, 20--23, 30

习题：P348 5, 6