



合肥工业大学

HEFEI UNIVERSITY OF TECHNOLOGY

计算机组成原理

第五章 输入输出系统

阙夏

计算机与信息学院

2019/10/31



第五章 输入输出系统

5.1 节

概 述

5.2 节

外 部 设 备

5.3 节

I/O 接 口

5.4 节

程序查询方式

5.5 节

程序中断方式

5.6 节

DMA方式



- (一) I/O系统基本概念
- (二) 外部设备
 - 1. 输入设备：键盘、鼠标等
 - 2. 输出设备：显示器、打印机 等
 - 3. 辅助存储器：硬盘存储器、磁盘阵列、光盘存储器等
- (三) I/O接口 (I/O控制器)
 - 1. I/O接口的功能和基本结构
 - 2. I/O端口及其编址
- (四) I/O方式
 - 1. 程序查询方式
 - 2. 程序中断方式
 - 中断的基本概念；
 - 中断响应过程；
 - 中断处理过程；
 - 多重中断和中断屏蔽的概念。
 - 3. DMA方式
 - DMA控制器的组成；DMA传送过程。
 - 4. 通道方式



- **定义：**是计算机系统中的**主机**与**外部**进行通信的系统。
- **组成：**它由**外部设备**和**I/O控制系统（接口、总线和管理软件）**两部分组成，是计算机系统的重要组成部分。
- **外部设备：**包括**I/O设备**和**辅助存储设备**（如如磁盘存储器、磁带存储器、光盘存储器等）。
- 从某种意义上也可以把磁盘、磁带和光盘等设备看成一种I/O设备，所以**I/O设备**与**外部设备**这两个名词经常是通用的。
- 在计算机系统中，通常把**CPU**和**主存**之外的部分称为**I/O系统**。
- **I/O系统特点**是：**异步性**、**实时性**和**设备无关性**。



一、输入输出系统的发展概况（四阶段）

1. 早期阶段 分散连接 （程序查询方式）

CPU 和 I/O **串行** 工作，I/O通过CPU与主机信息交换。

2. 接口模块和DMA（Direct Memory Access） 阶段

总线连接 { 中断方式
DMA 方式

CPU 和 I/O **并行** 工作，I/O通过接口与主机信息交换 。



3. 具有通道(Channel)结构的阶段

通道：一种专门用于输入输出操作的部件，是一种具有特殊功能的处理器。CPU启动通道后，可以继续运行自己的程序，而通道则同时执行通道程序，控制I/O设备进行直接数据传送。

4. 具有 I/O 处理机的阶段

I/O处理机 (Peripheral Processor) 方式 比通道方式具有更强的独立性和专用性。IOP有自己的**指令系统**，可以通过对I/O处理机的编程，实现独立于CPU的I/O操作。



二、输入输出系统的组成

1. I/O 软件

主要任务：

- 将用户编制的程序（数据）输入到主机内；
- 将运算结果输送给用户；
- 实现输入输出系统与主机工作的协调等。

(1) I/O 指令 CPU 指令的一部分

操作码

命令码

设备码

操作码：I/O指令与其它指令的判别码

命令码：I/O设备具体操作

设备（地址）码：用于从多台I/O设备中选择某台设备



(2) 通道指令 (Channel Control Word)

通道自身的指令；

指出数据组的首地址、传送字数或末地址、操作命令；

如 IBM/370 通道指令为 64 位。

2. I/O 硬件

设备	I/O 接口	(总线型)	
设备	设备控制器	通道	(通道型)



三、I/O 与主机的联系方式

1. I/O 编址方式

(1) **统一编址**: 指把I/O地址 (I/O端口地址) 当作存储器的单元进行分配。

在这种方式下CPU不需设置专门的I/O指令, 用统一的访问存储器的取数、存数指令就可访问I/O端口。

优点: 1) 使CPU访问I/O的操作更灵活、更方便,
2) 使端口有较大的编址空间。

缺点: 1) 使端口占用了存储器地址, 使内存容量变小。
2) 执行速度较慢。



(2) 不统一编址（单独编址方式）：

指I/O端口地址与存储器地址无关，另行单独编址。

在这种方式下，CPU需要设置专门的I/O指令访问端口。

优点：程序编制清晰、便于理解。

缺点：1) 输入输出指令少；
2) 增加控制的复杂性。



2. 设备寻址

每台设备有一个设备号，I/O指令设备码字段指出设备号，用**设备选择电路**识别设备是否被选中。

3. 传送方式

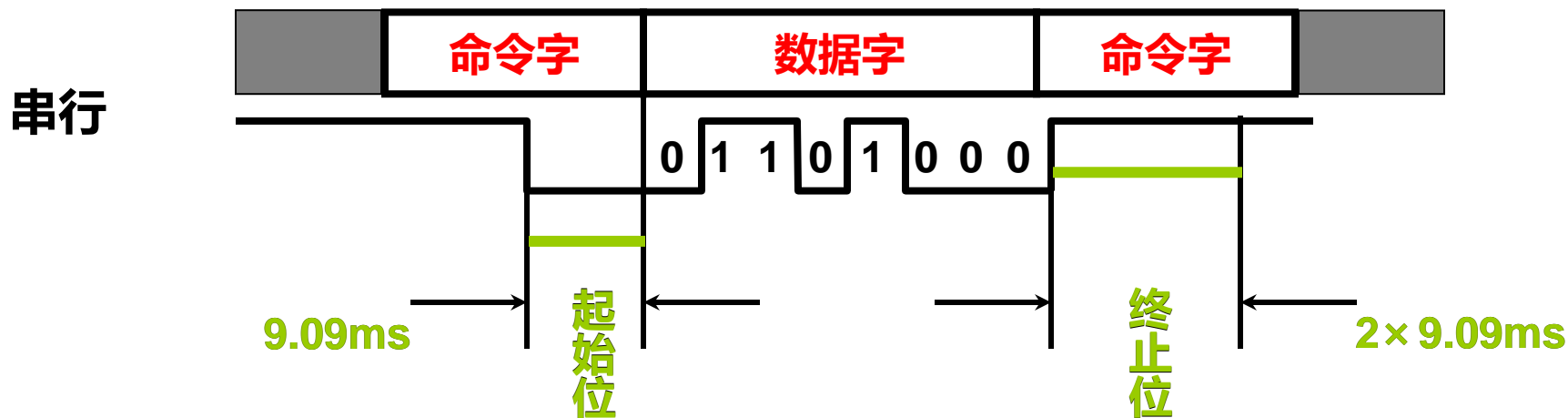
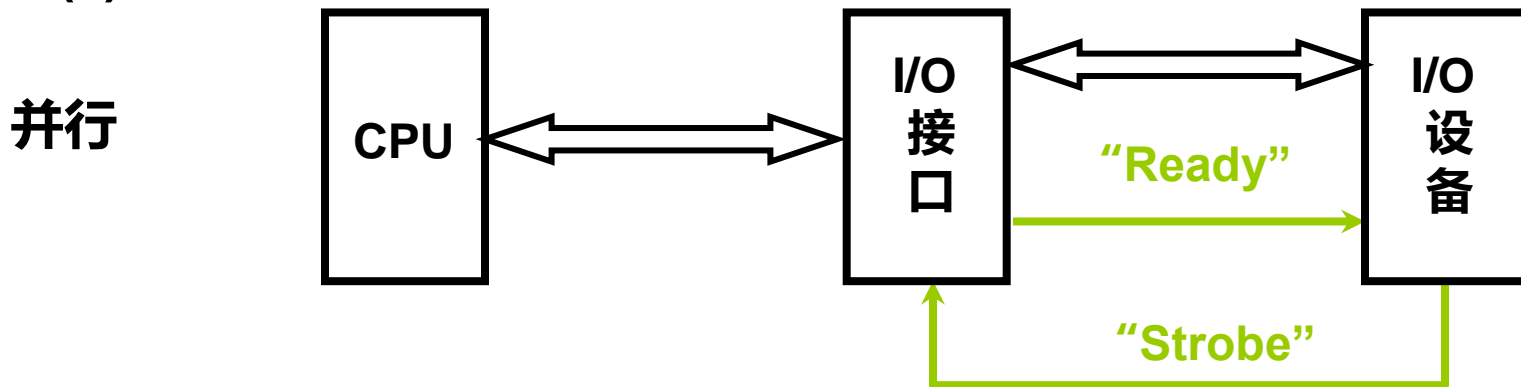
- (1) 串行：在主机与接口之间完全按照并行方式实现数据传输，但在接口与I/O设备之间是按照串行方式实现数据传输。（在串行接口中必须设置具有移位功能的数据缓冲器，实现数据格式的串 - 并转换。）
(适用于低速的串行I/O设备及信息远距离传送设备。)
- (2) 并行：在主机与接口、接口与I/O设备之间是按照并行方式完成信息传输。（当I/O设备本身按照并行方式工作，主机与I/O设备间距离较近，选用并口）



4. 联络方式

(1) 立即响应 (速度缓慢的I/O)

(2) 异步工作采用应答信号 (I/O设备与主机工作速度不匹配)

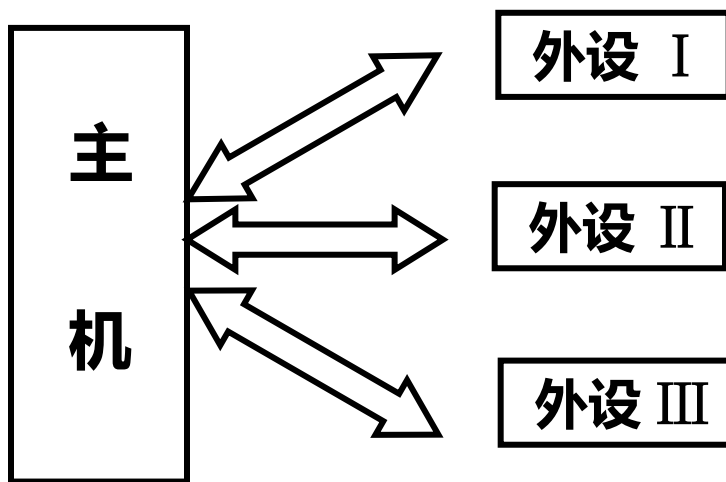


(3) 同步工作采用同步时标



5. I/O 与主机的连接方式

(1) 辐射式连接



每台设备都配有一套

控制线路和一组信号线

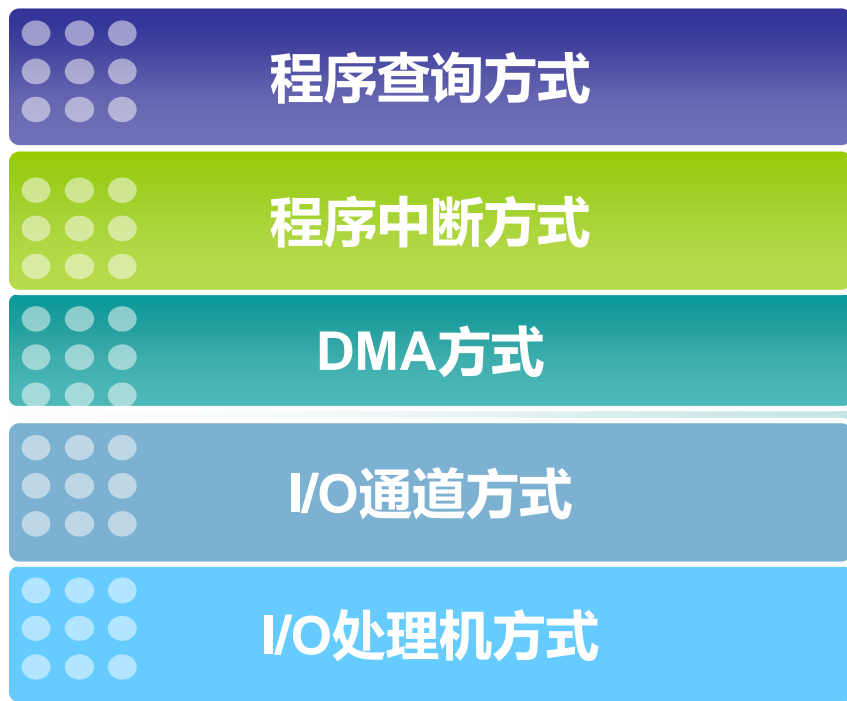
不便于增删设备

(2) 总线连接

便于增删设备



四、I/O 与主机信息传送的控制方式



外设与主机的信息
传送和控制方式



1. 程序查询方式

- 基本原理：用程序实现主机与外设之间的信息交换，即CPU直接利用I/O指令编程实现输入输出。
 - 通常方法是在用户程序中安排一段**输入输出服务程序**，当需要进行I/O操作时，直接控制I/O设备进行工作。
 - 开始时需要对设备所处的**工作状态进行查询**，所以称程序查询方式。
 - **程序每执行一条I/O指令（或传送指令），只能传送一个数据。每次都要查询设备所处的状态，只有当设备准备好后才能传送，否则主程序作循环等待。**
 - CPU与I/O设备处于**串行**工作方式，CPU必须等待I/O设备完成操作后才能继续执行其它程序。因此，此方式数据的**传送速度很慢，CPU的利用率较低**。
 - Note：计算机带有多台I/O设备时，应根据各设备的重要程度对它们进行**优先级排队**。

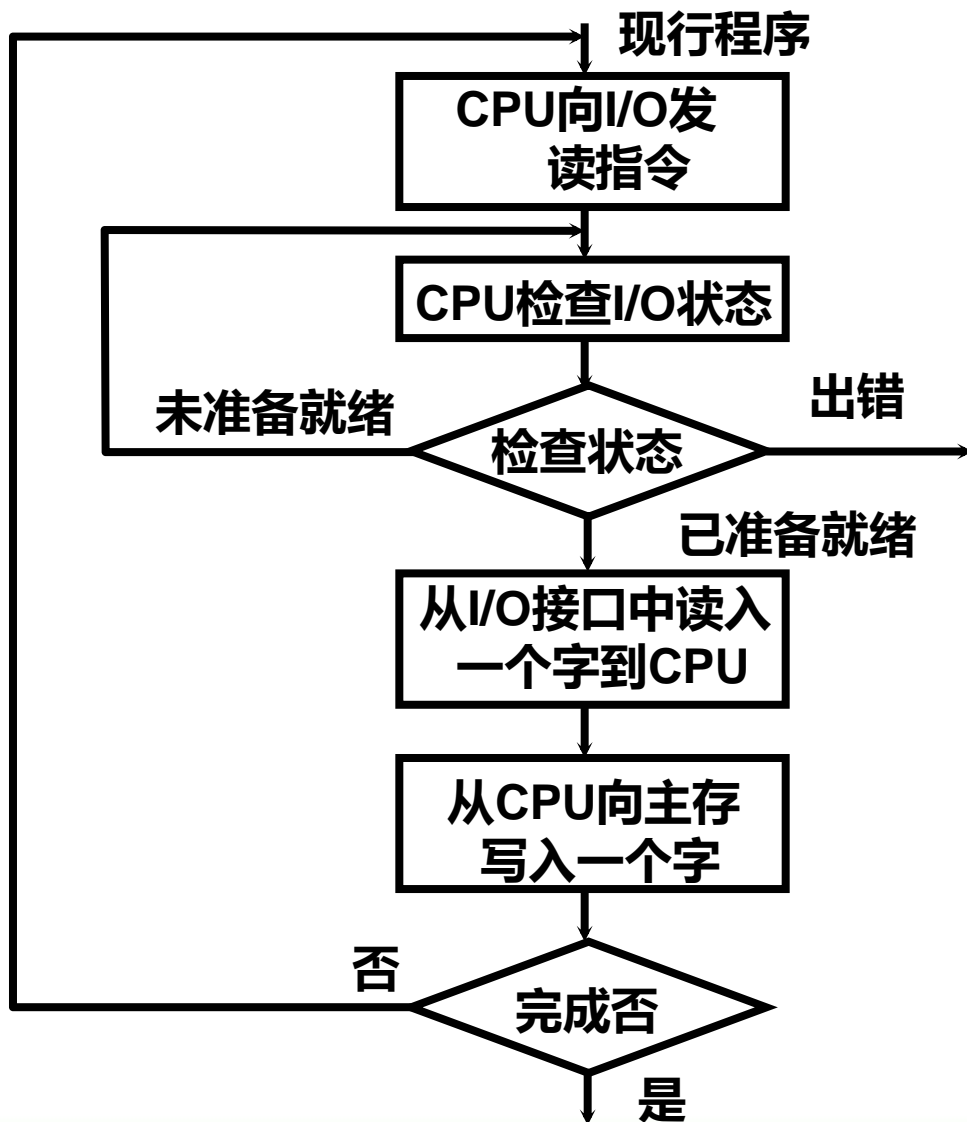
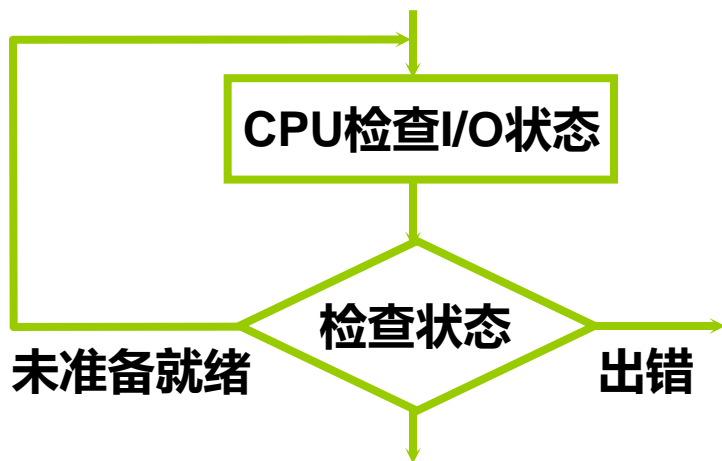


5.1 概述

1. 程序查询方式

CPU 和 I/O 串行工作

踏步等待





1. 程序查询方式

特点:

- ① 硬件结构简单;
- ② CPU能控制何时对何设备进行输入输出操作;
- ③ 数据输入输出要经过CPU;
- ④ 外设和CPU处于异步工作状态;
- ⑤ 用于连接低速外设（终端和打印机）和单片机系统;



2. 程序中断方式

- 中断（Interrupt Request）：CPU在执行程序的过程中，当出现**异常情况**或**特殊请求**处理时，CPU暂停当前正在执行的程序，转而去执行更为紧急的中断服务程序，并能在中断服务程序处理结束后，自动返回被暂停的程序继续执行。
- 中断系统：用于实现中断的**软件**和**硬件**组成的系统
- 一般适用于**随机**出现的服务，且一旦提出要求应立即实现。这种方式节省了CPU的时间，但硬件结构对复杂一些。



2. 程序中中断方式

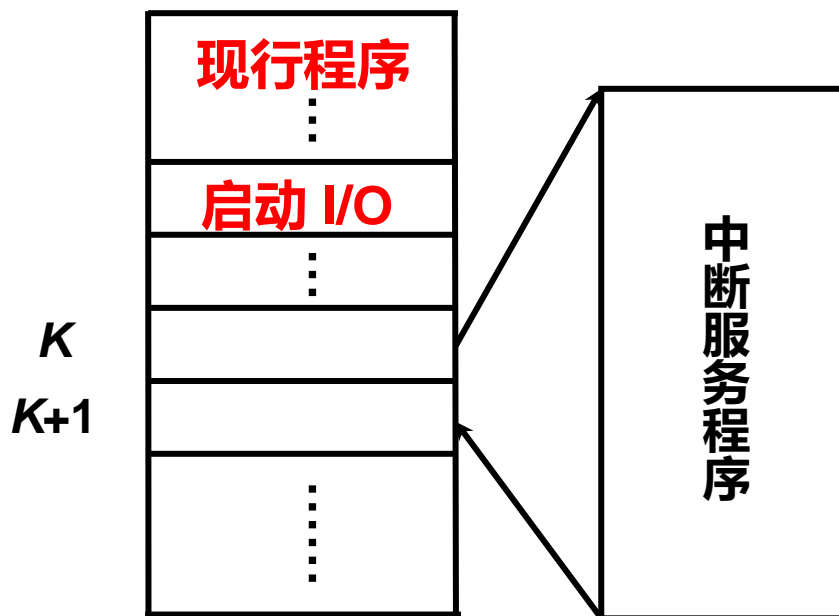
I/O 工作

{ 自身准备
与主机交换信息

CPU 不查询

CPU 暂停现行程序

CPU 和 I/O 并行工作



没有踏步等待现象

中断现行程序



2. 程序中中断方式

- 特点：
 - ① CPU与外部设备可以并行工作
 - ② 能够处理异常事件
 - ③ 数据的输入输出要经过CPU
 - ④ 一般用来连接低速设备（键盘、鼠标）



3. DMA 方式

- 直接存储器存取(Direct Memory Access, DMA)方式：一种直接依靠**硬件**在主存与设备之间进行数据传送，在数据传送期间**不需要CPU**程序干预的与I/O设备交换数据的工作方式。
- 在这种方式中，**DMA控制器**从CPU手中暂时**接管**对总线的控制，数据交换不必经过CPU而直接通过数据总线在内存和I/O设备之间直接进行。
- **DMA方式**几乎是目前所有计算机系统都要具备的一种重要工作方式。



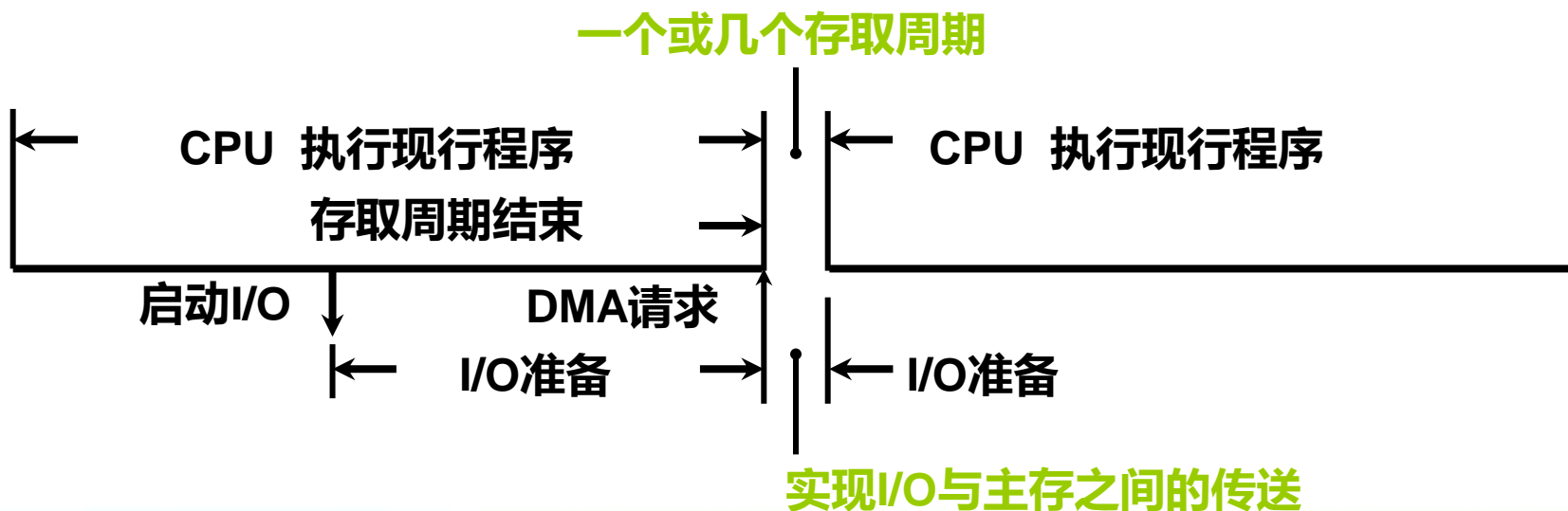
3. DMA 方式

主存和 I/O 之间有一条直接数据通道

不中断现行程序

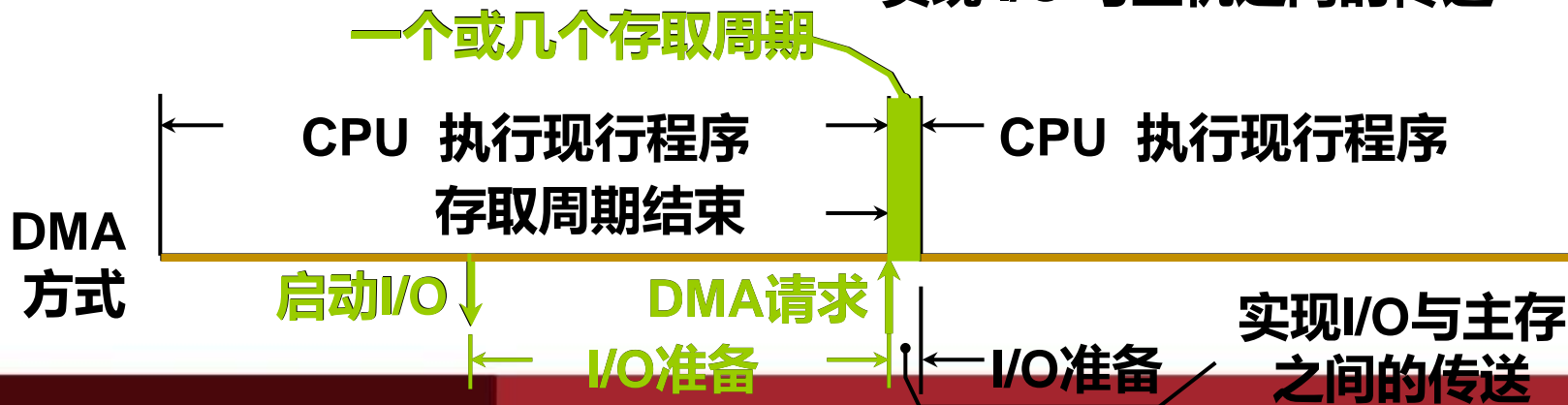
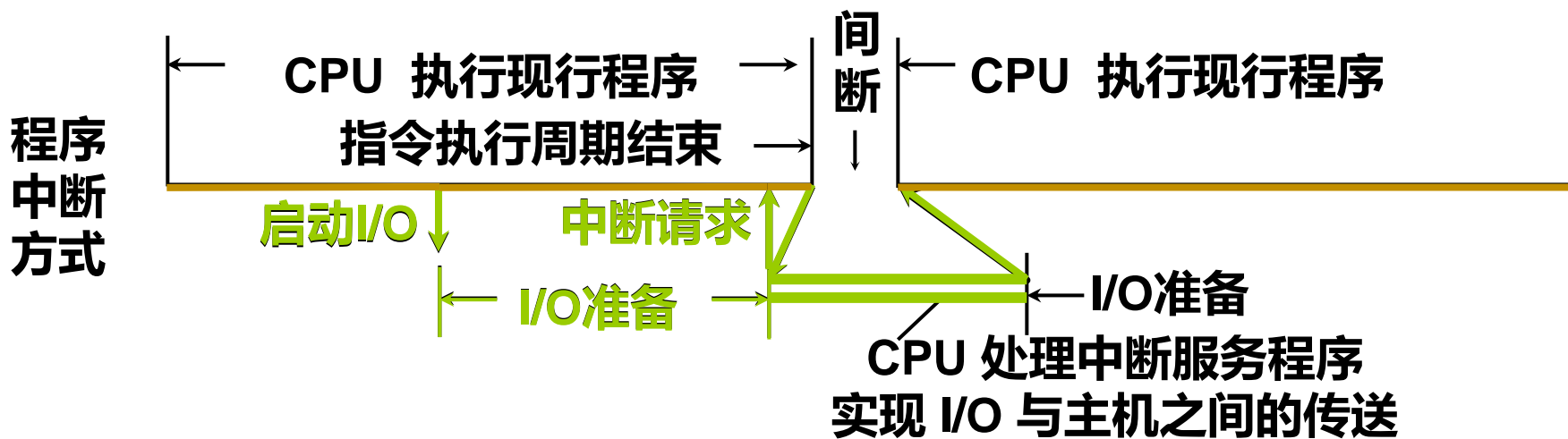
周期挪用（周期窃取）

CPU 和 I/O 并行工作





三种方式的 CPU 工作效率比较



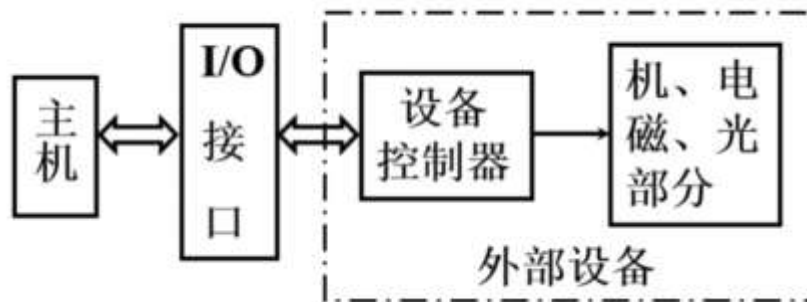


[例子]假设幼儿园一个阿姨带10个孩子，要给每个孩子分4块水果糖。假设孩子们都应把4块糖都吃完，那么她采用什么方法呢？

- **方法1：** 她先给孩子甲一块糖，盯着甲吃完，然后再给第二块，等吃完第二块又给第三块，吃完第三块又给第四块。接着给孩子乙，其过程与孩子甲完全一样。依次类推，直至到第10个孩子发完四块糖。看来这种方法效率太低，重要之点还在于孩子们吃糖时她一直在守候，什么事也不能干。(程序查询方式)
- **方法2：** 每人发一块糖各自去吃，并约定谁吃完后就向她举手报告，再发第二块。看来这种新方法提高了工作效率，而且在未接到孩子们吃完糖的报告以前，她还可以腾出时间给孩子们改作业。但是这种方法还可以改进。(程序中断方式)
- **方法3：** 进行批处理：每人拿4块糖各自去吃，吃完4块糖后再向她报告。显然这种方法工作效率大大提高，她可以腾出更多的时间改作业。(直接内存访问(DMA)方式)
- **方法4：** 权力下放，把发糖的事交给另一个人分管，只是必要时她才过问一下。(通道方式)



一、概述



外部设备大致分三类

1. 人机交互设备

键盘 鼠标 打印机 显示器

2. 计算机信息驻留设备

磁盘 光盘 磁带

3. 机 - 机通信设备

调制解调器等



二、输入设备

1. 键盘

按键

判断哪个键按下

将此键翻译成 ASCII 码（编码键盘法）

2. 鼠标

机械式、光电式

3. 触摸屏









三、输出设备

1. 显示器

(1) 字符显示

字符发生器

(2) 图形显示

主观图像 (矢量图)

(3) 图像显示

客观图像 (像素图)

2. 打印机

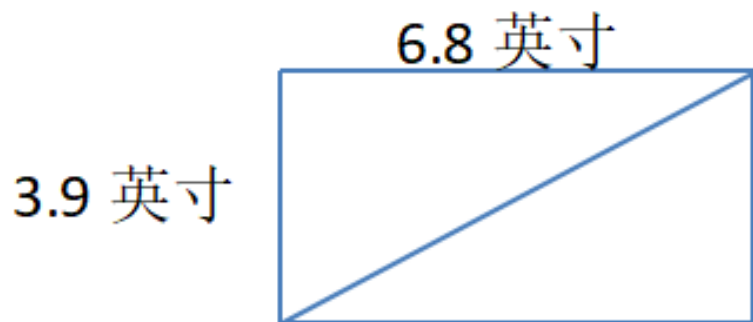
(1) 击打式

点阵式 (逐字、逐行)

(2) 非击打式

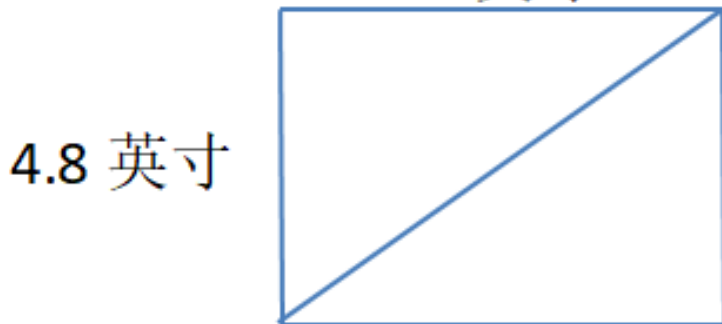
喷墨 (逐字)

激光 (逐页)

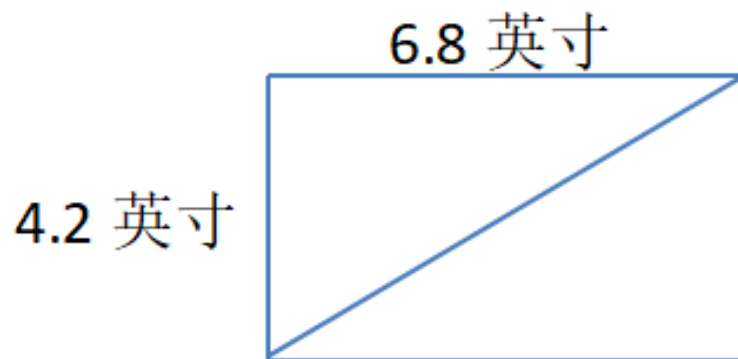


16:9 面积 26.5

6.4 英寸



4:3 面积 30.7



16:10 面积 28.6



4:3

2.39:1

16:9

16:10



四、其他

1. A/D、D/A 模拟/数字（数字/模拟）转换器

2. 终端 由键盘和显示器组成

完成显示控制与存储、 键盘管理及通信控制

3. 汉字处理 汉字输入、汉字存储、汉字输出

五、多媒体技术

1. 什么是多媒体（Multimedia）

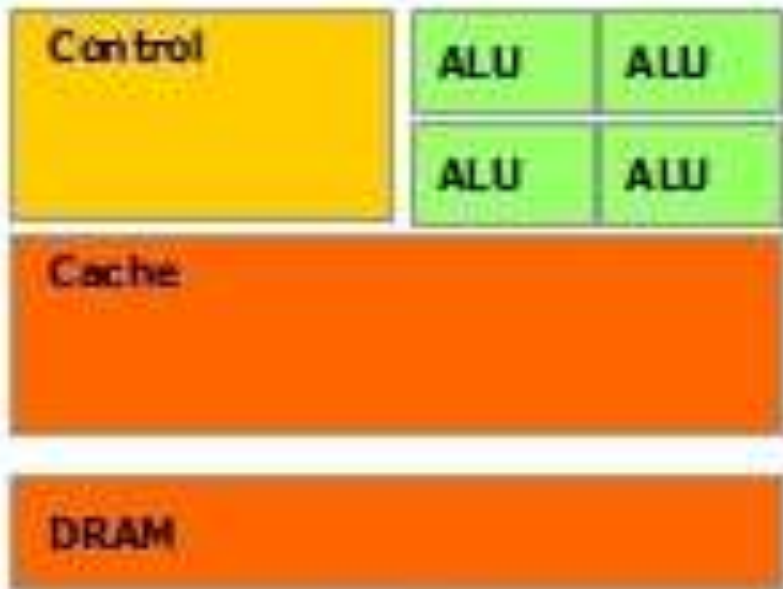
2. 多媒体计算机的关键技术 压缩和解压缩、专用芯片、大容量存储



绿色的是计算单元

红色的是存储单元

黄色的是控制单元



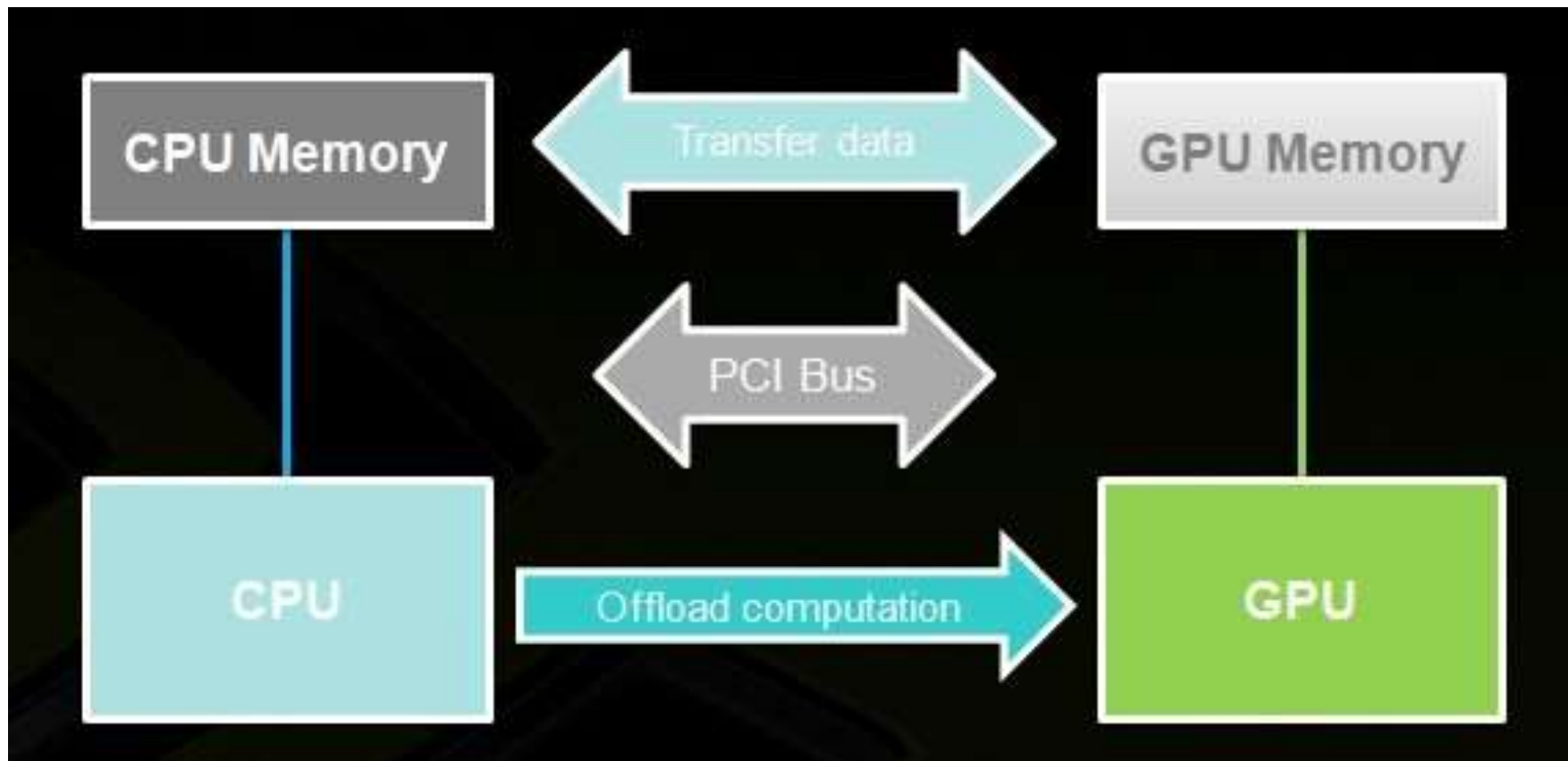
CPU



GPU

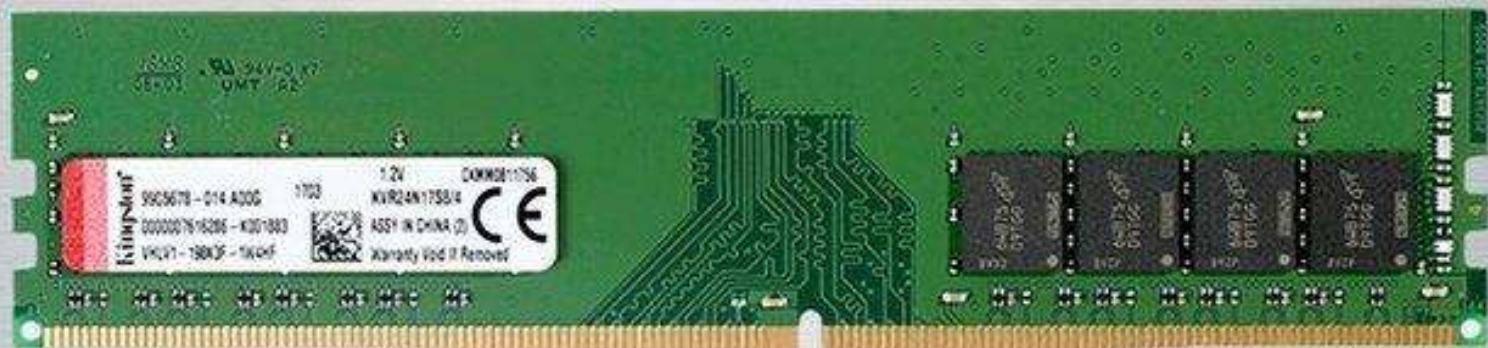


主存与显存





笔记本内存



台式机内存



显存

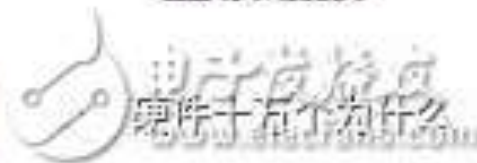
处理器
散热片

处理器
风扇



主板连接

GDDR5规格 显存





独立显卡



显示芯片



云从科技



台式机显卡



内存条



内存条



2017.12

NVIDIA TITAN V

售价:2999USD

显卡核心:GV100@1200Mhz

Boost频率: 1455Mhz

Tensor Core: 640

流处理器/光栅/纹理: 5120 / 96 / 320

接口速率: PCIE 3.0 x16

显存配置 12GB 3072BIT HBM2 @848Mhz 651.3GB/s {显存容量/位宽/种类/带宽}

DX版本: DirectX 12.0 FL:12-1 注: HBM2等效频率X2





2018.6

NVIDIA TITAN V CEO

售价：? USD

显卡核心：GV100@1200Mhz

Boost频率：1455Mhz

Tensor Core: 640

流处理器/光栅/纹理： 5120 / 128 / 320 接口速率： PCIE 3.0 x16

显存配置 32GB 4096BIT HBM2 @848Mhz 868.4GB/s {显存容量/位宽/种类/带宽}

DX版本： DirectX 12.0 FL:12-1 注： HBM2等效频率X2



2018.12

NVIDIA TITAN RTX

售价：2499USD

显卡核心：TU102@1350Mhz

Boost频率：1770Mhz

Tensor Core: 576 RT core: 72

流处理器/光栅/纹理： 4608 / 96 / 288 接口速率： PCIE 3.0 x16

显存配置 24GB 384BIT GDDR6 @1750Mhz 672GB/s {显存容量/位宽/种类/带宽}

DX版本：DirectX 12.0 FL:12-1



B站：Morisummers



英伟达历代旗舰单芯卡发展史
(PCI-E篇)

149 9-2



英伟达历代旗舰单芯卡发展史
(AGP篇)

227 7-14



AMD Radeon VEGA12核心
测试孤岛危机3

720 6-11



那些消逝在历史长河中的显卡
品牌

1153 2018-6-1



用11年前的老显卡运行GTA5
会是什么效果？

8.3万 2018-2-4



一、概述

1、I/O接口(Input/Output Interface)定义:

- I/O 接口是**主机**与**外部设备**进行信息交换的**纽带**。
- 主机通过I/O 接口与外部设备进行**数据交换**。
- 绝大部分I/O 接口电路都是**可编程**的。
- I/O 接口电路包括：**插头 / 插座的形式、通讯规程和电器特性**等。



2、为什么要设置接口？

- 实现设备（地址）的选择
- 实现数据缓冲达到速度匹配
- 实现数据串 - 并格式转换
- 实现电平转换
- 传送控制命令
- 反映设备的状态（“忙”、“就绪”、“中断请求”）



3、接口 (Interface) 和端口 (Port) 不同:

- **接口**: 指一种为在主机和外设之间传送信息而设置的**硬件线路**, 也可以指两个软件之间的共同**逻辑边界**。
- **端口**: 接口电路中的一些**寄存器**, 这些寄存器分别用来存放数据信息、控制信息和状态信息, 相应的就是**数据端口**、**控制端口**和**状态端口**。
- 若干个端口加上相应的**控制逻辑**才能组成接口。CPU通过输入指令, 从**端口**读入信息, 通过输出指令, 可将信息**写入**到端口中。



二、接口的功能和组成

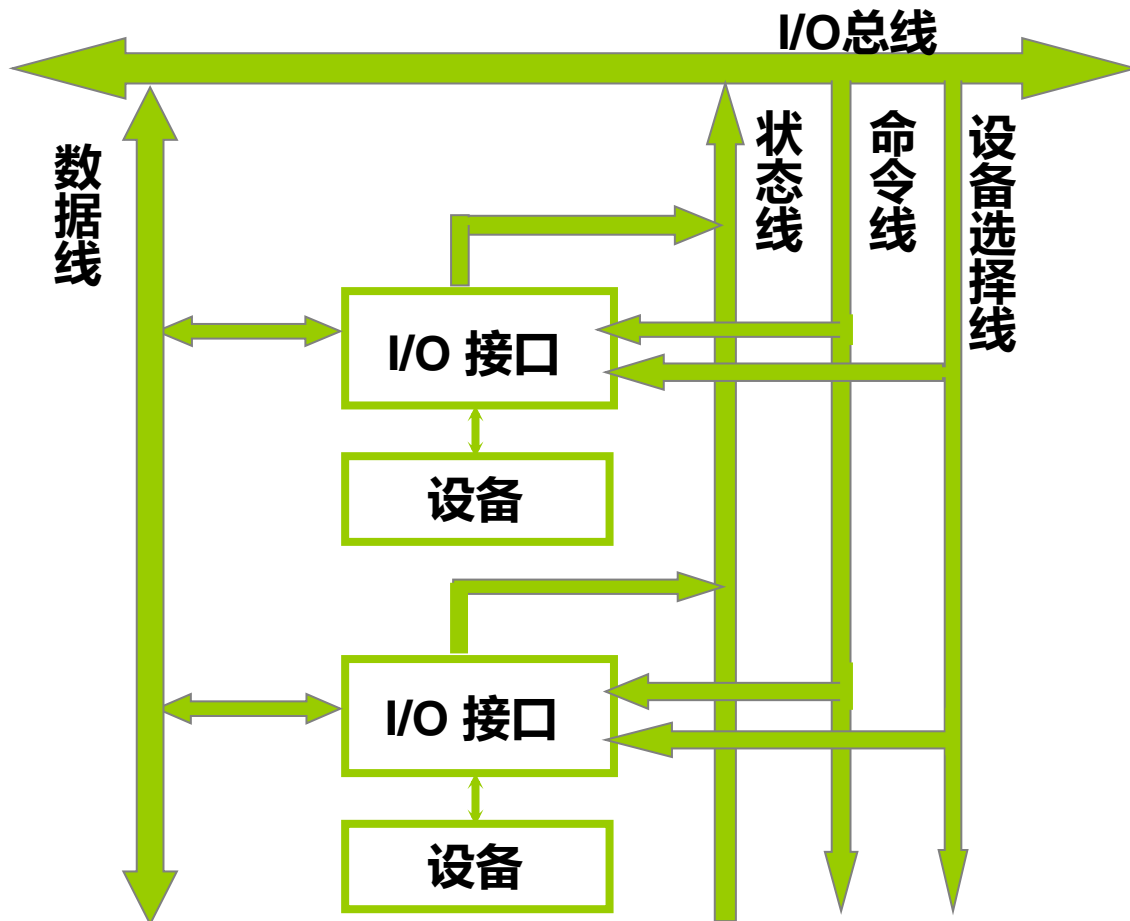
1. 总线连接方式的 I/O 接口电路

(1) 设备选择线

(2) 数据线

(3) 命令线

(4) 状态线





2. 接口的功能和组成

功能

选址功能

传送命令的功能

传送数据的功能

反映设备状态的功能

完成触发器 D

工作触发器 B

中断请求触发器 INTR

屏蔽触发器 MASK

组成

设备选择电路

命令寄存器 命令译码器

数据缓冲寄存器

设备状态标记



3. I/O 接口的基本组成





三、接口类型

1. 按数据 **传送方式** 分类

并行接口 Intel 8255

串行接口 Intel 8251

2. 按功能 **选择的灵活性** 分类

可编程接口 Intel 8255、 Intel 8251

不可编程接口 Intel 8212

3. 按 **通用性** 分类

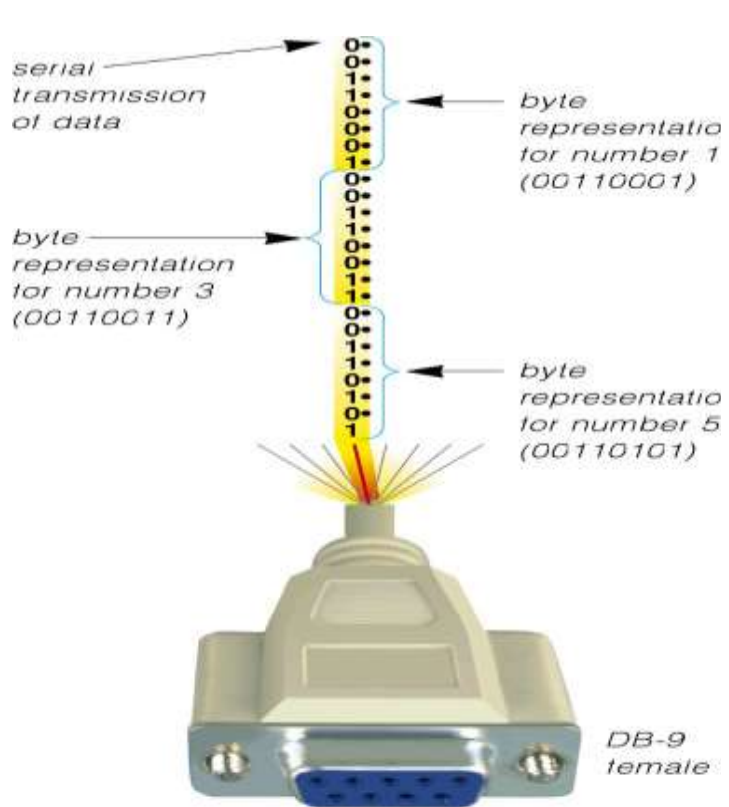
通用接口 Intel 8255、 Intel 8251

专用接口 Intel 8279、 Intel 8275

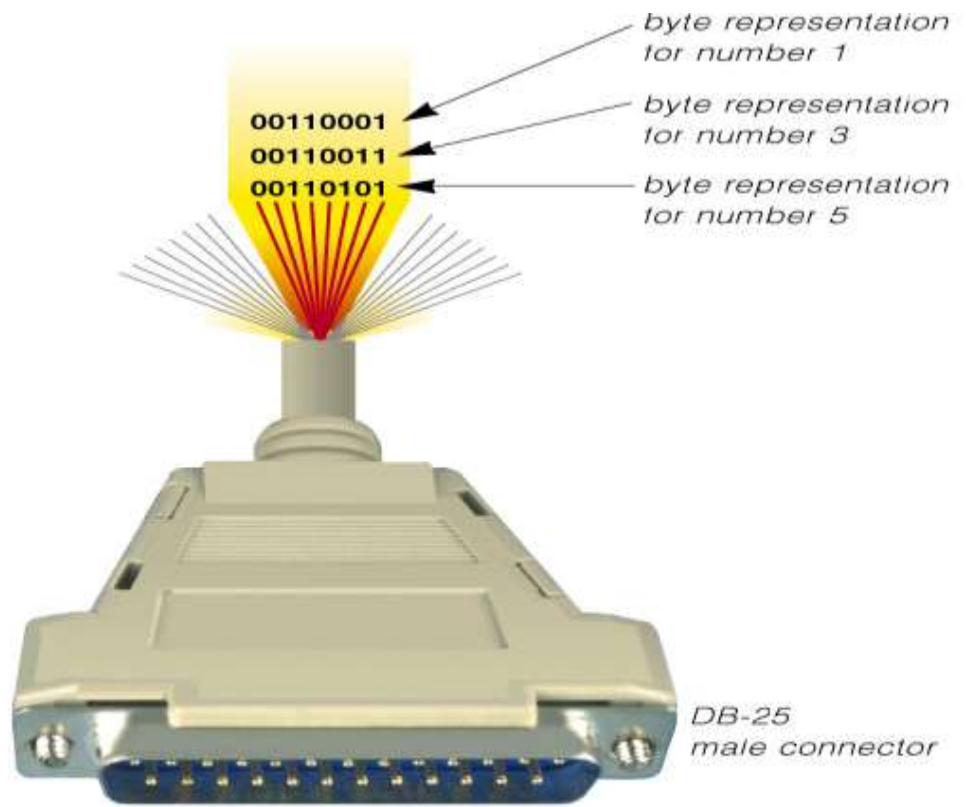
4. 按数据传送的 **控制方式** 分类

中断接口 Intel 8259

DMA 接口 Intel 8257



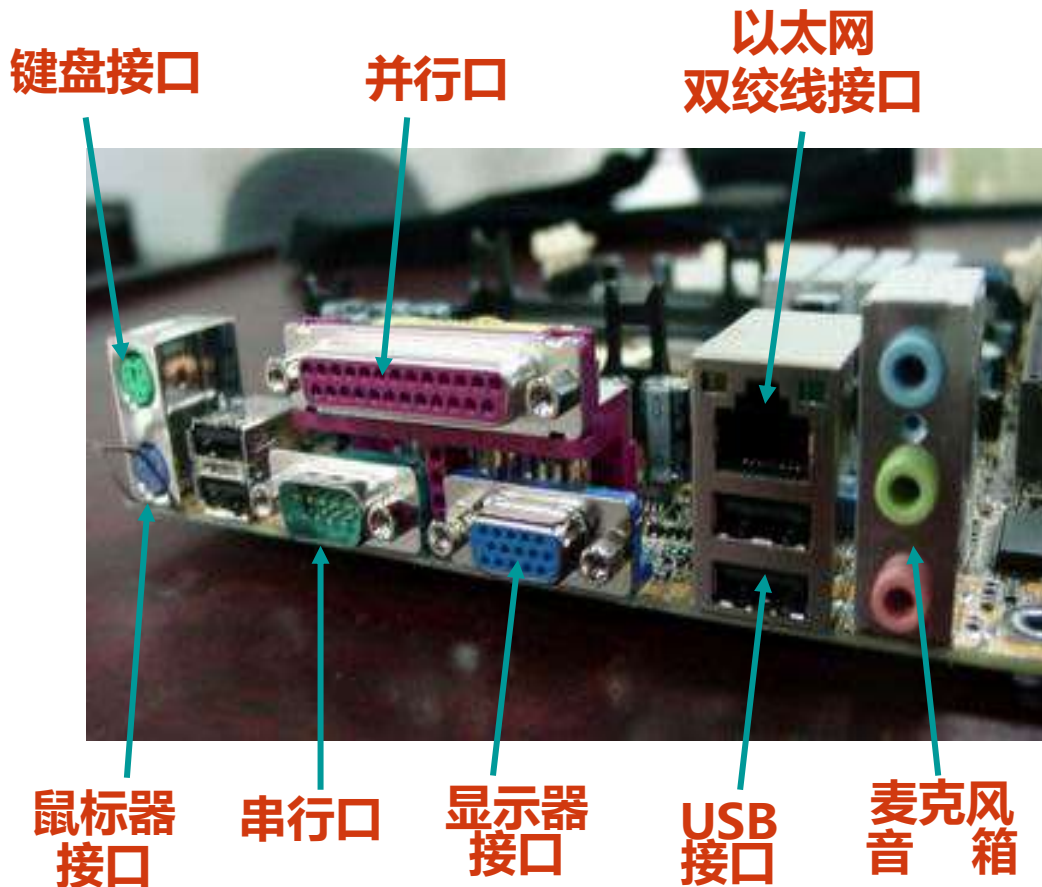
串行口



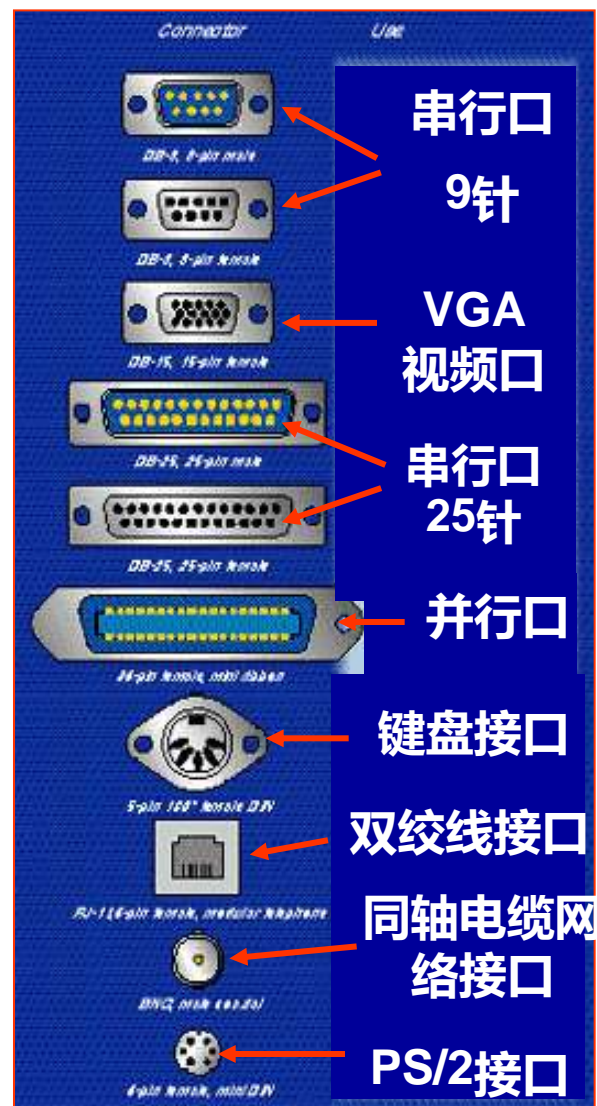
并行口



I/O设备接口



(安装在主板上的I/O设备接口)





5.4 程序查询方式

5.4.1、程序查询方式的流程

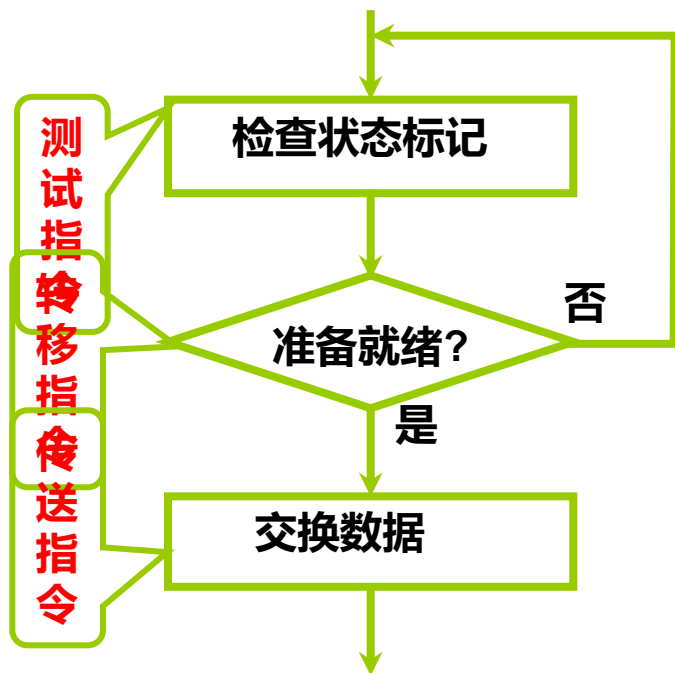
5.4.2 程序查询方式的接口电路



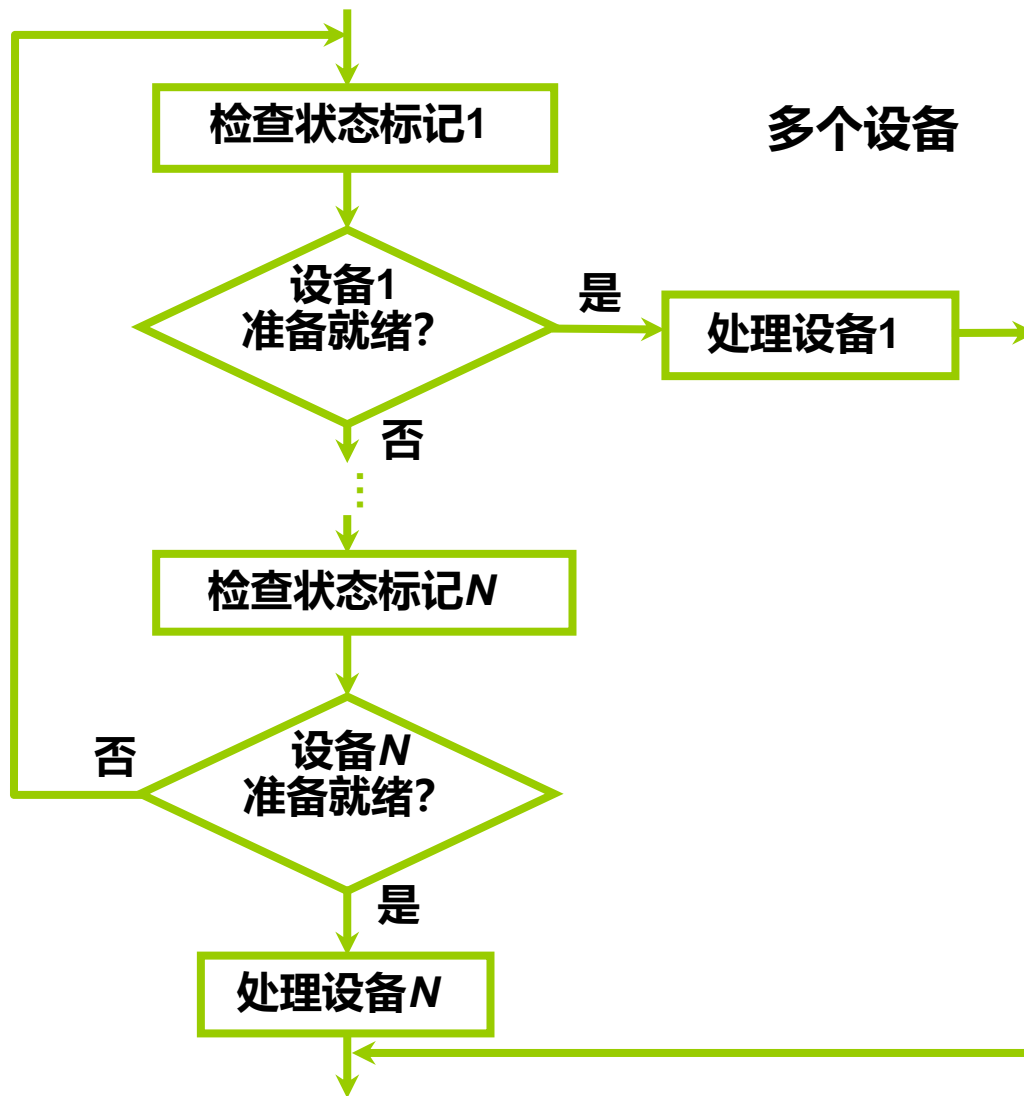
一、程序查询流程

1. 查询流程

单个设备

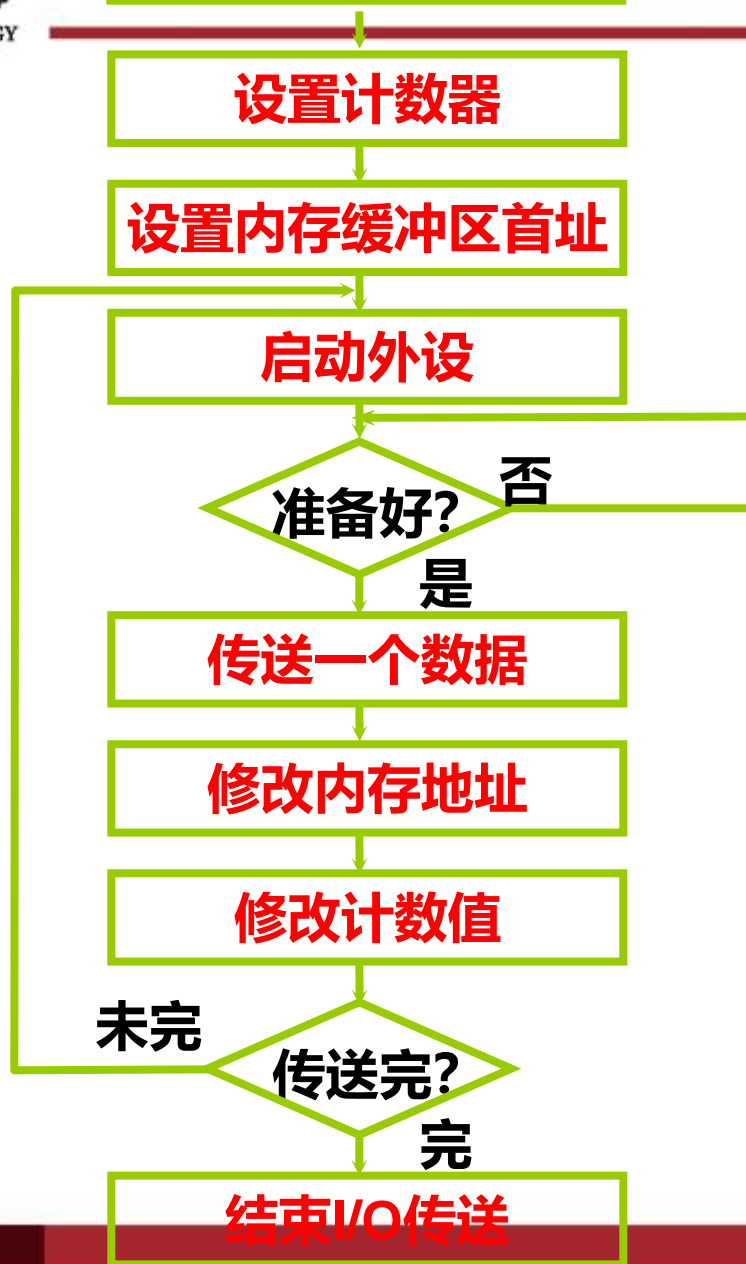


多个设备





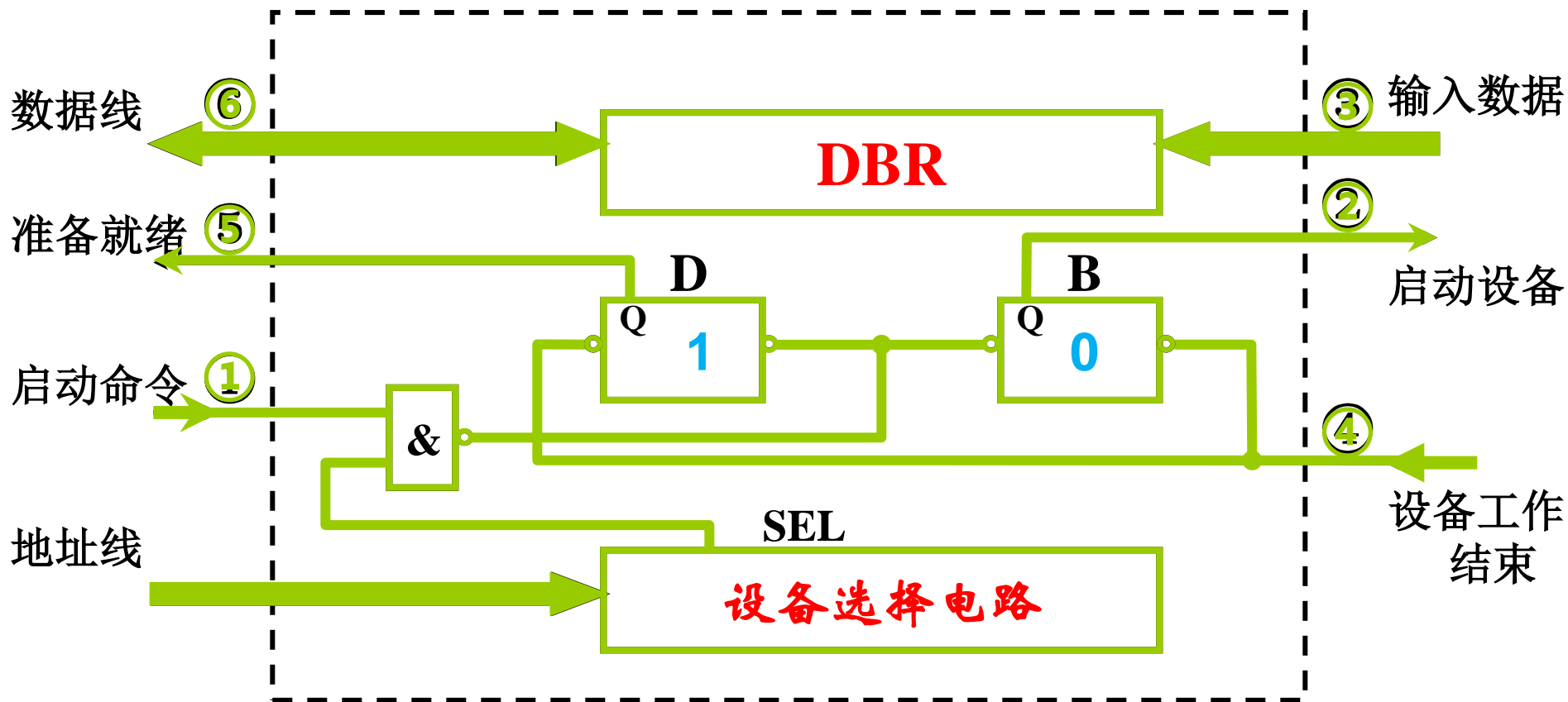
2. 程序流程





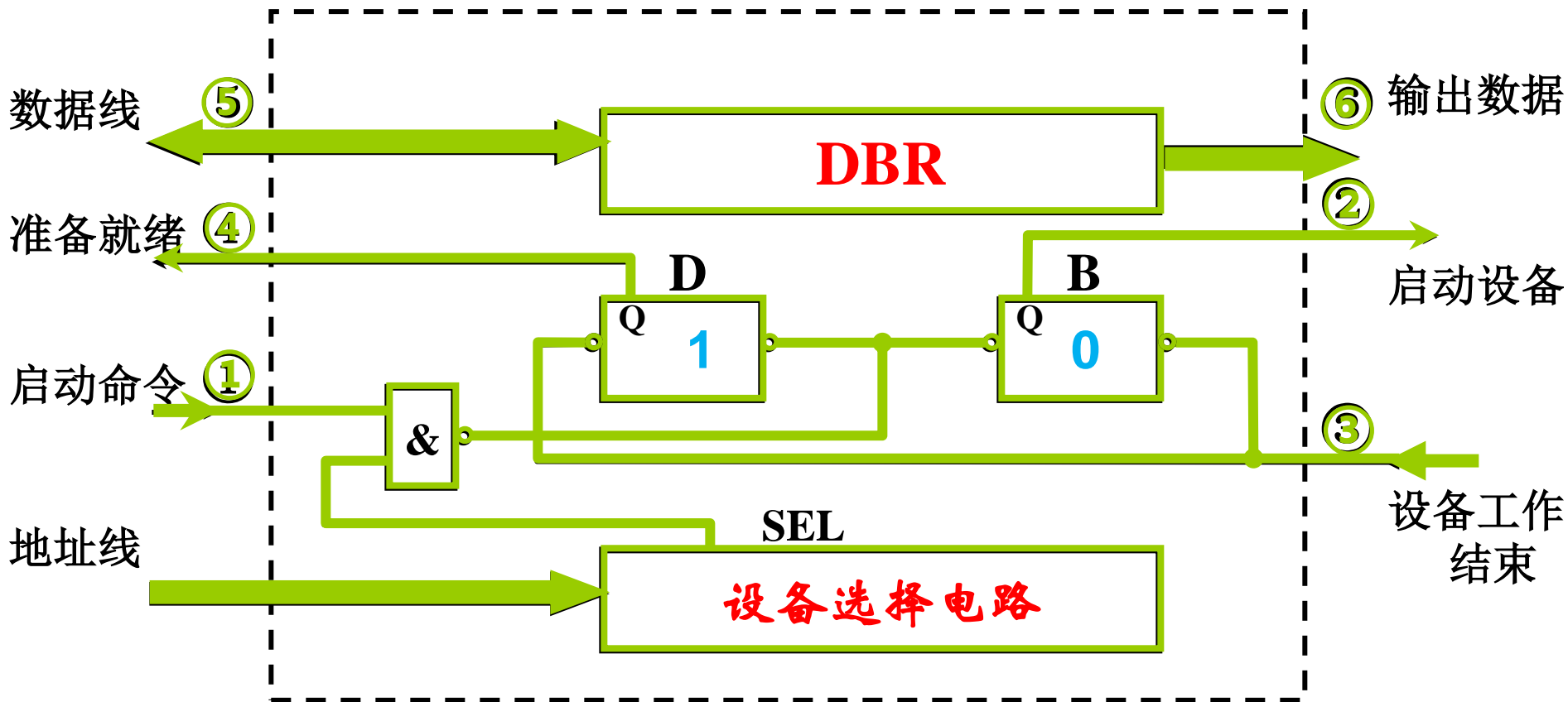
一、程序查询方式的接口电路

以输入为例





以输出为例





5.5 程序中断方式

5.5.1 中断的概念

5.5.2 I/O中断的产生

5.5.3 程序中断的接口电路

5.5.4 I/O中断处理过程

5.5.5 中断服务程序流程



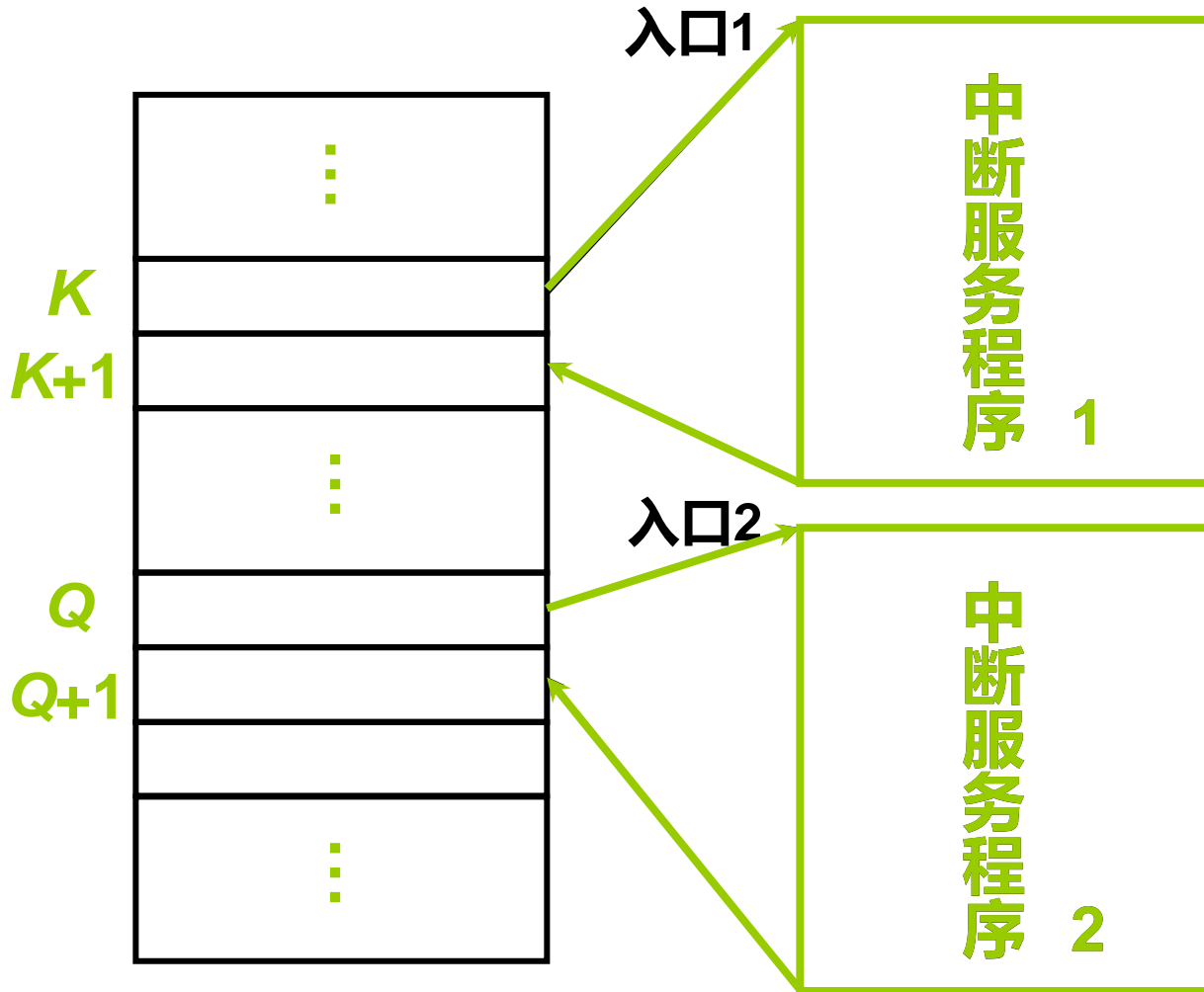
一、中断 (Interrupt) 的概念

是指在计算机执行期间，系统内发生任何**非寻常**的或**非预期**的**急需处理**事件，使得CPU**暂时中断**当前正在执行的程序而转去执行相应的事件处理程序。待处理完毕后又**返回**原来被中断处继续执行执行的过程。

中断的种类和方式很多。



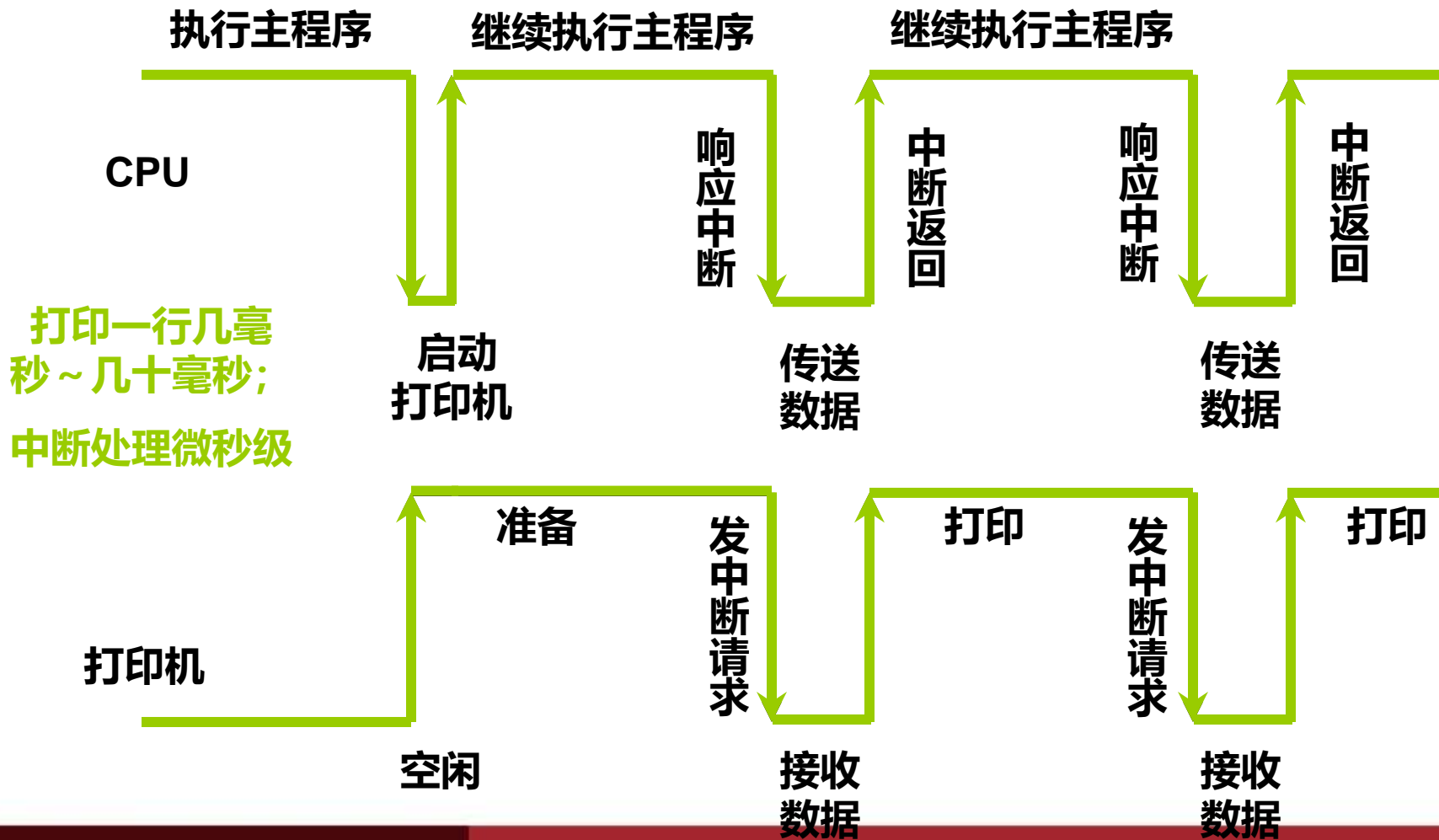
5.5.1 中断的概念





5.5.1 中断的概念

实现CPU与I/O设备并行工作 (以打印机为例)





一、中断的产生

设置中断的必要性：

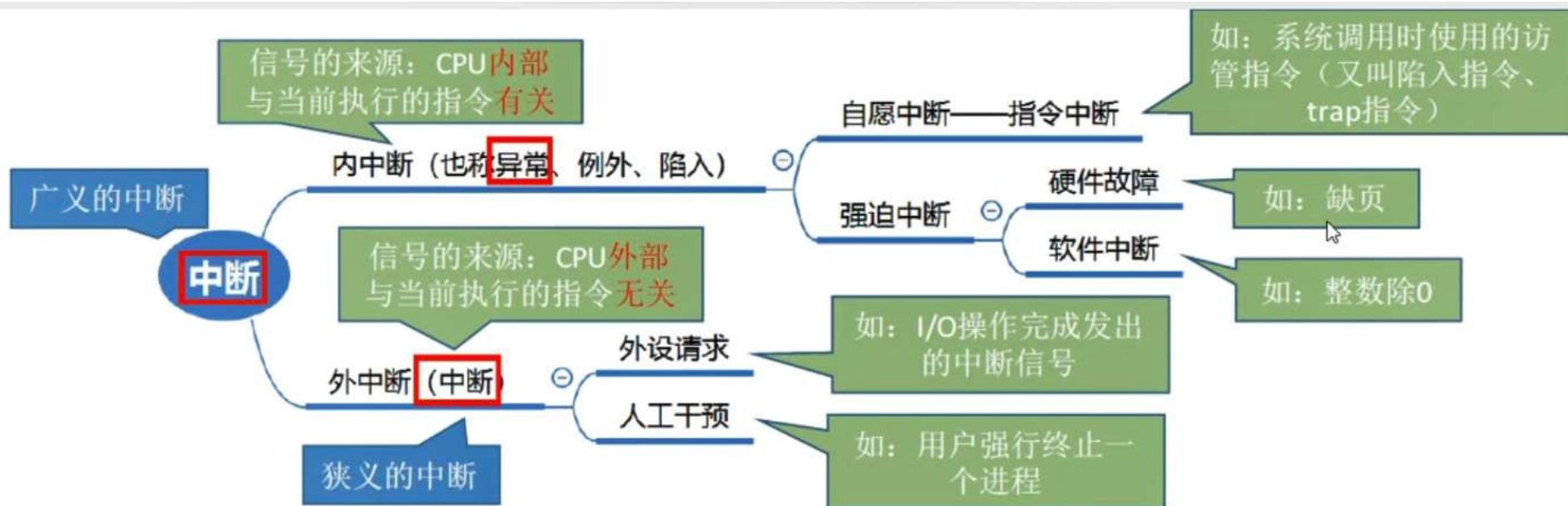
1. **分时操作**：采用中断技术能够使CPU与外围设备并行工作，实现分时操作，大大提高了CPU的利用率。
2. **实时处理**：实时控制是单片机系统一个重要的应用领域，它要求CPU对工作现场的各种参数和状态的随机变化能够快速响应并加以及时处理，而这些功能只有采用中断技术才能得以实现。
3. **故障处理**：计算机在运行过程中常常会突然发生故障，如电源掉电、内存出错及程序故障等。



二、中断的分类

1. 按中断来源分：

- (1) 内中断：发生在**主机内部**的中断称为内中断
- (2) 外中断：由**主机外部**事件引起的中断称为外中断





二、中断的分类

1. 按中断来源分：

(1) **内中断**：发生在**主机内部**的中断称为内中断

① **强迫中断**是在CPU没有事先预料的情况下发生的，此时CPU不得不停止现行的工作。产生的原因有**硬件故障**和**软件出错**等

- **硬件故障**包括部件中集成电路芯片、元件、器件、印刷线路版、导线及焊点引起的故障，电源电压的下降也属于硬件故障。
- **软件出错**包括指令出错、程序出错、地址出错、数据出错等。

② **自愿中断**是出于计算机系统管理的需要而自愿地进入中断。

- 计算机系统为了方便用户调试软件、检查程序、调用外部设备，设置了自中断指令和进管指令。CPU执行程序时遇到这类指令就进入中断。在中断过程中调出相应的管理程序。自愿中断是可以预料的。即如果程序重复执行，断点的位置不改变。

(2) **外中断**：由**主机外部**事件引起的中断称为外中断，外中断是**强迫中断**。



2. 按中断服务程序入口地址的获取方式分：

- (1) **向量中断**：外部设备在提出中断请求的同时，通过**硬件**自动形成中断服务程序**入口地址**。CPU响应中断后，将根据向量地址直接转入相应中断服务程序。
- (2) **非向量中断**：非向量中断在产生中断的同时不能直接提供中断服务程序入口地址，提供一个统一函数的入口地址，跳转到**统一的函数**，此统一的函数通过判别寄存器标识位和优先级关系进行中断处理。



5.5.2 I/O中断的产生

8086的中断向量表

用户定义 224个	Type 225	CS	03FFH	
		IP	03FCH	
		:		
	Type 32	CS	0083H	
		IP	0080H	
	Type 31	CS	007FH	
		IP	007CH	
		:		
	Type 5	CS		
		IP	0014H	
	Type 4	CS	0013H	
		IP	0010H	溢出中断
	Type 3	CS	000FH	
		IP	000CH	断点中断
	Type 8	CS	000FH	
		IP	0008H	非屏蔽中断
	Type 1	CS	0007H	
		IP	0004H	单步中断



3. 按是否可屏蔽分：

CPU根据需要可以禁止响应某些中断源的中断请求。

- 可屏蔽中断：CPU可以禁止响应的中断。
- 不可屏蔽中断：CPU必须响应的中断。

4. 按引发中断事件分：

- 硬件中断-由外设引发的，大部分可屏蔽；
- 软件中断-由程序调用发生的，不可屏蔽。



一、 程序中断方式的接口电路

1. 配置中断请求触发器和中断屏蔽触发器

(1) 中断请求触发器

- 当中断发送引起中断的事件时，先将它保存在设备控制器的中断请求触发器INTR中，即将“中断请求触发器”置“1”。
- 当中断请求触发器为“1”时，向CPU发出“中断请求”信号。
- 每个中断源有一个**中断请求触发器**。全机的多个中断请求触发器构成中断请求寄存器，其内容称为**中断字**或**中断码（号）**。CPU进行中断处理时，根据中断字确定中断源，转入相应的服务程序。



(2) 中断屏蔽:

为了便于控制中断请求信号的产生, 也为了利用屏蔽码改变中断处理的有限级别, 当产生中断请求后, 用程序方式**有选择地**封锁部分中断, 而允许其余**部分中断**仍得到响应, 称为中断屏蔽。

- 实现方法是为每个中断源设置一个中断屏蔽触发器MASK来屏蔽该设备的中断请求。即用程序方式将该触发器**置“1”**, 则对应的设备**中断被封锁**, 若将其**置“0”**, 才允许该设备的中断请求得到**响应**。



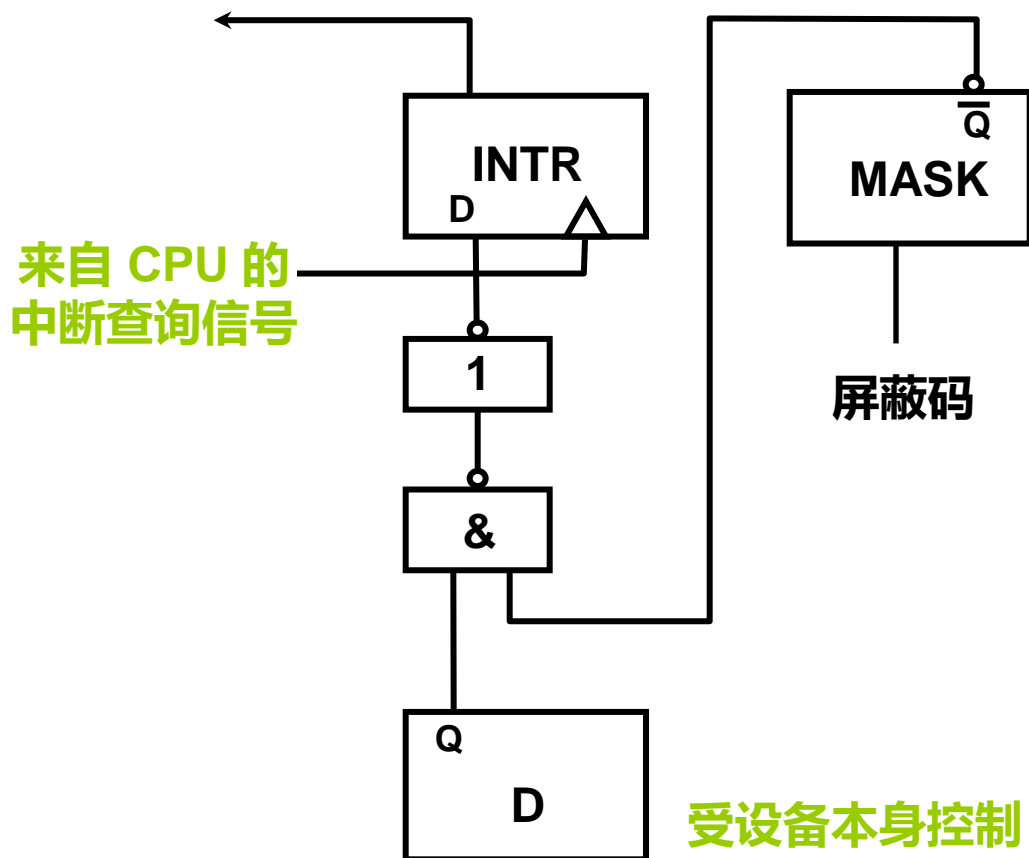
5.5.3 程序中中断方式的接口电路

- 各设备的中断屏蔽触发器组成**中断屏蔽寄存器**。中断屏蔽寄存器通常是一个与系统中的中断请求寄存器同样位数的专用寄存器。CPU可以根据需要给中断屏蔽寄存器写入屏蔽字（二进制代码），代码的每一位可屏蔽一种中断源。如当中断屏蔽寄存器的某位为“1”时，表示该位所对应的中断被屏蔽。
- 有些中断请求是**不可屏蔽**的。也就是说，不管中断系统是否开中断，这些中断源的中断请求一旦提出，CPU必须立即响应。例如，电源掉电就是不可屏蔽中断。所以，中断又分为**可屏蔽中断**和**非屏蔽中断**。非屏蔽中断具有最高优先权。



(3) 接口电路中几个信号间的关系

中断请求



INTR
中断请求触发器

INTR = 1 有请求

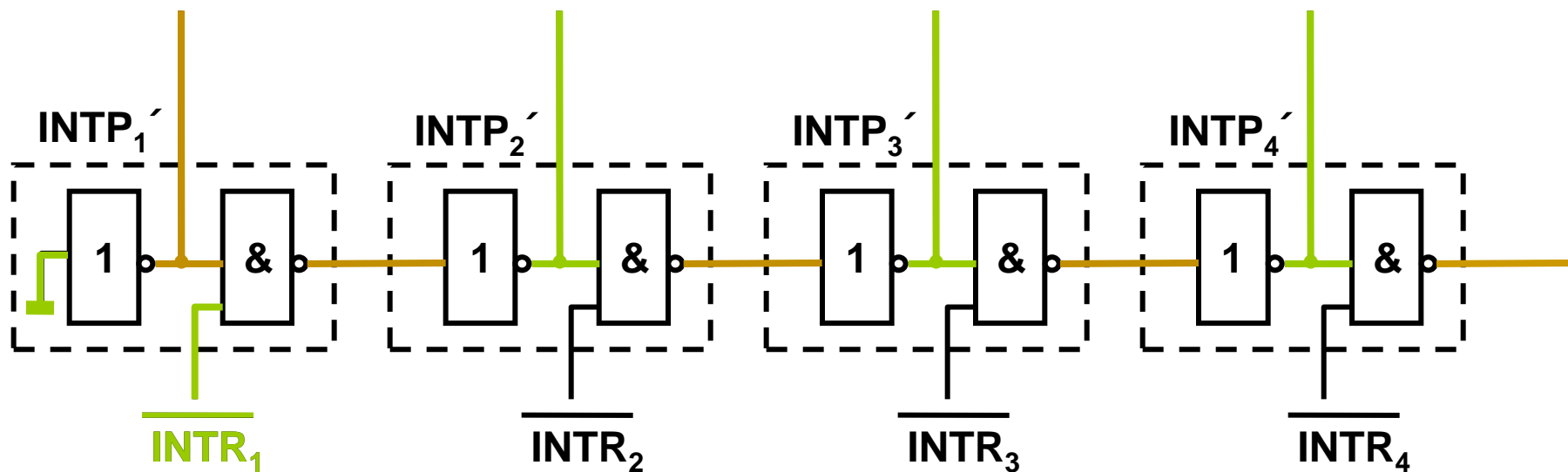
MASK
中断屏蔽触发器

MASK = 1 被屏蔽

D 完成触发器

2. 排队器

排队 { 硬件 在 CPU 内、在接口电路中（链式排队器）
 软件 详见第八章

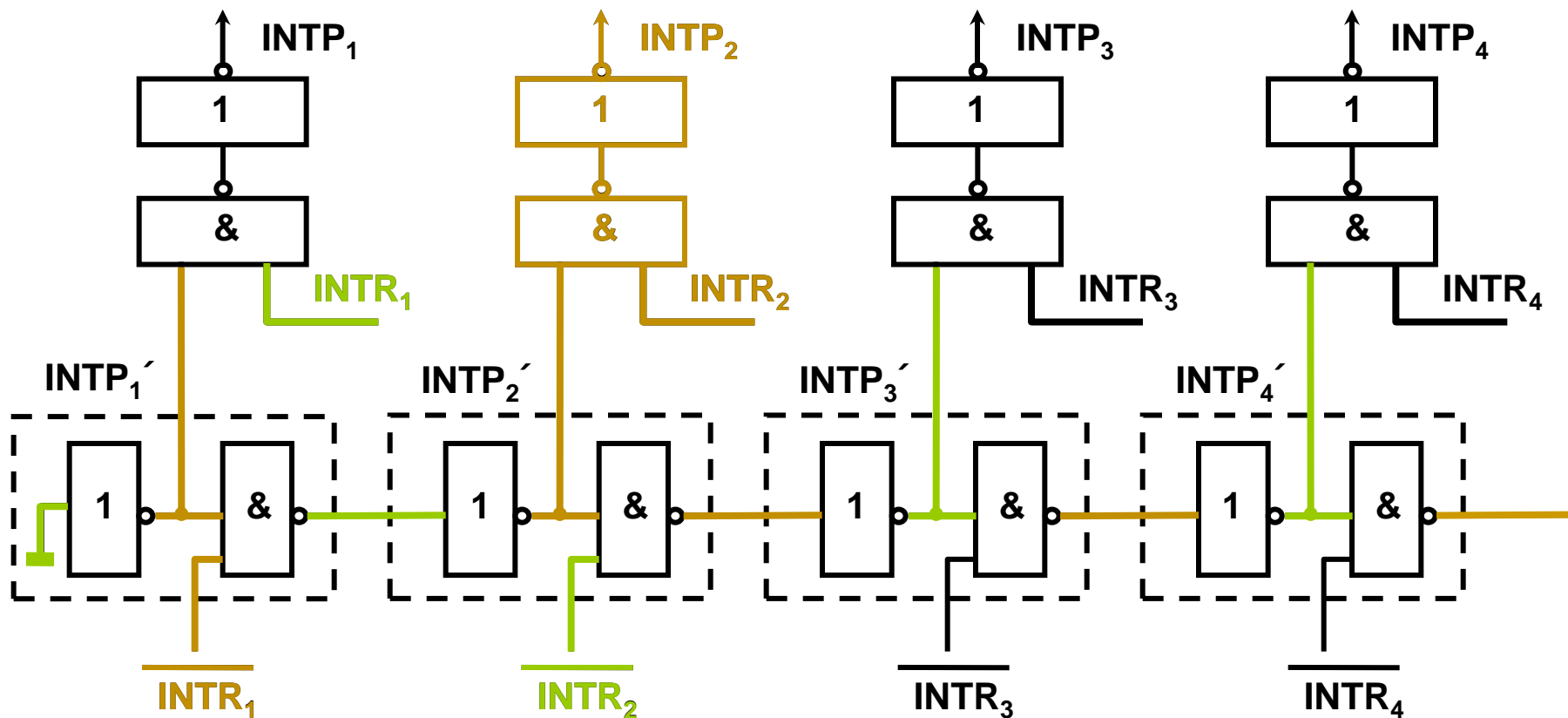


设备 1[#]、2[#]、3[#]、4[#] 优先级按 **降序排列**

$\text{INTR}_i = 1$ 有请求 即 $\overline{\text{INTR}_i} = 0$

5.5.3 程序中断方式的接口电路

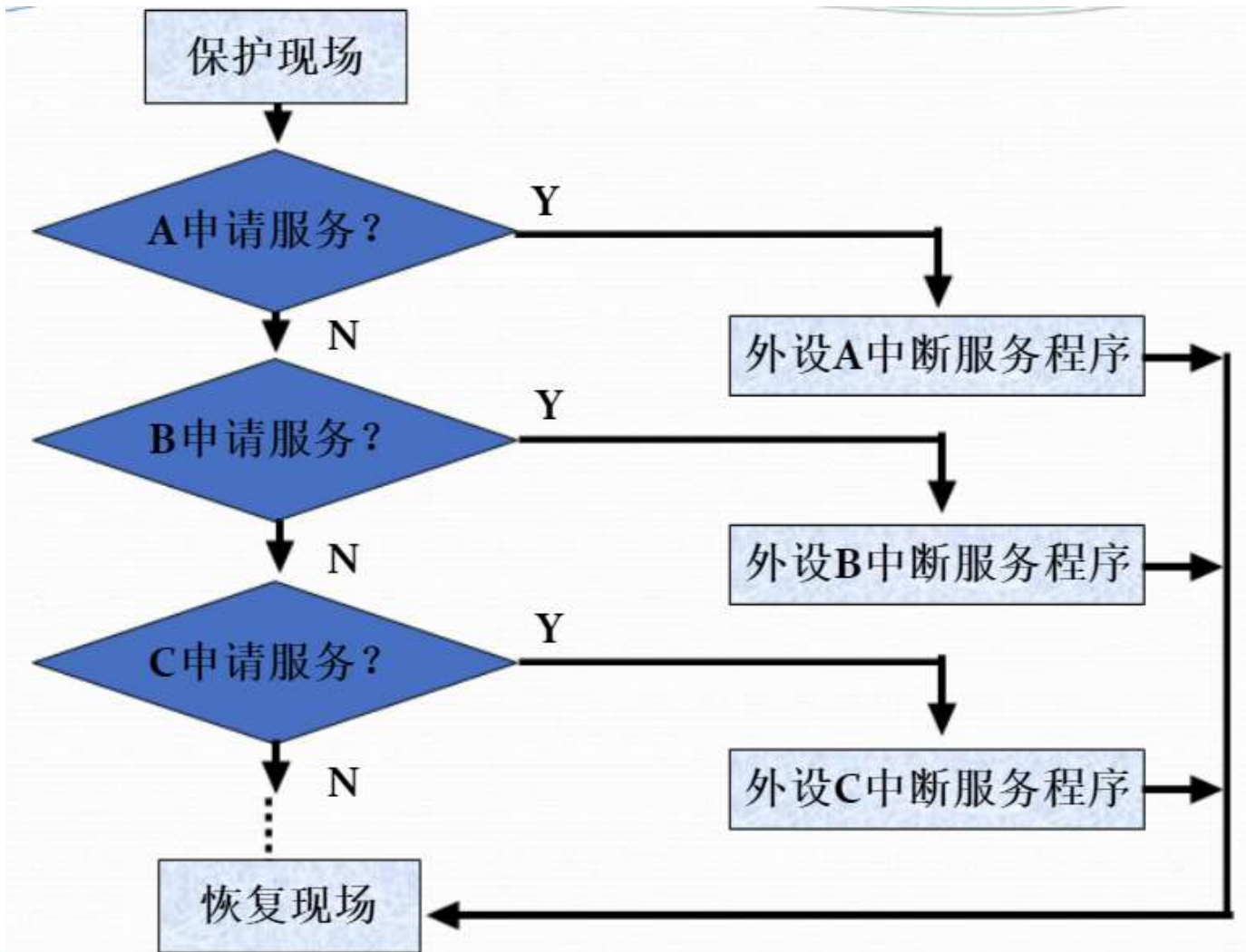
链式排队器





5.5.3 程序中断方式的接口电路

使用软件查询法解决中断优先级只需少量硬件电路。





3. 中断向量地址形成部件

入口地址 {	由软件产生	详见第八章
	硬件向量法	由 硬件 产生 向量地址
		再由 向量地址 找到 入口地址

区分几个概念：

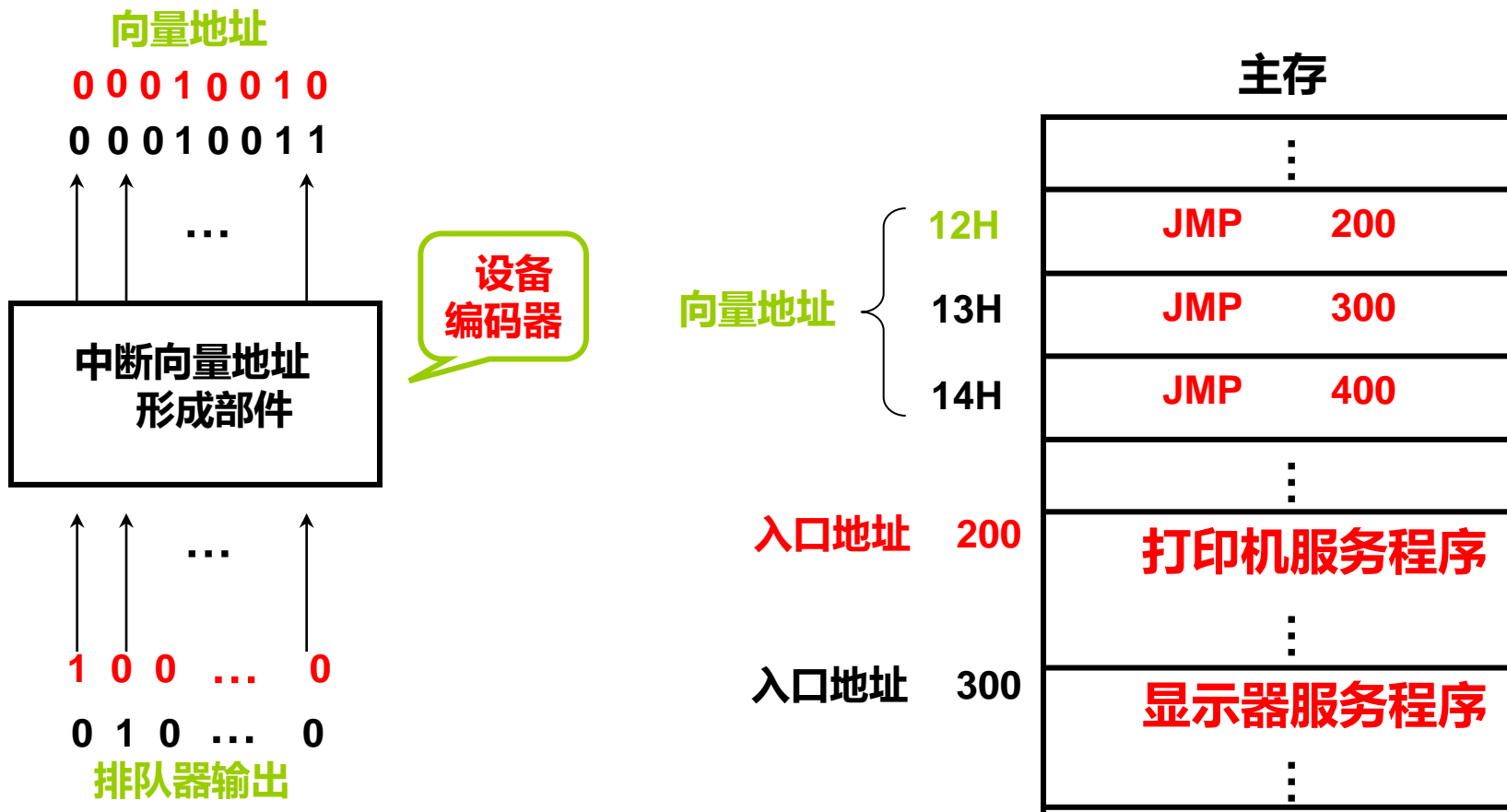
中断（号）码：系统分配给每个中断源的代号，以便识别和处理。

中断向量：中断服务程序的入口地址（跳转指令）。

向量地址：保存中断向量的主存地址。

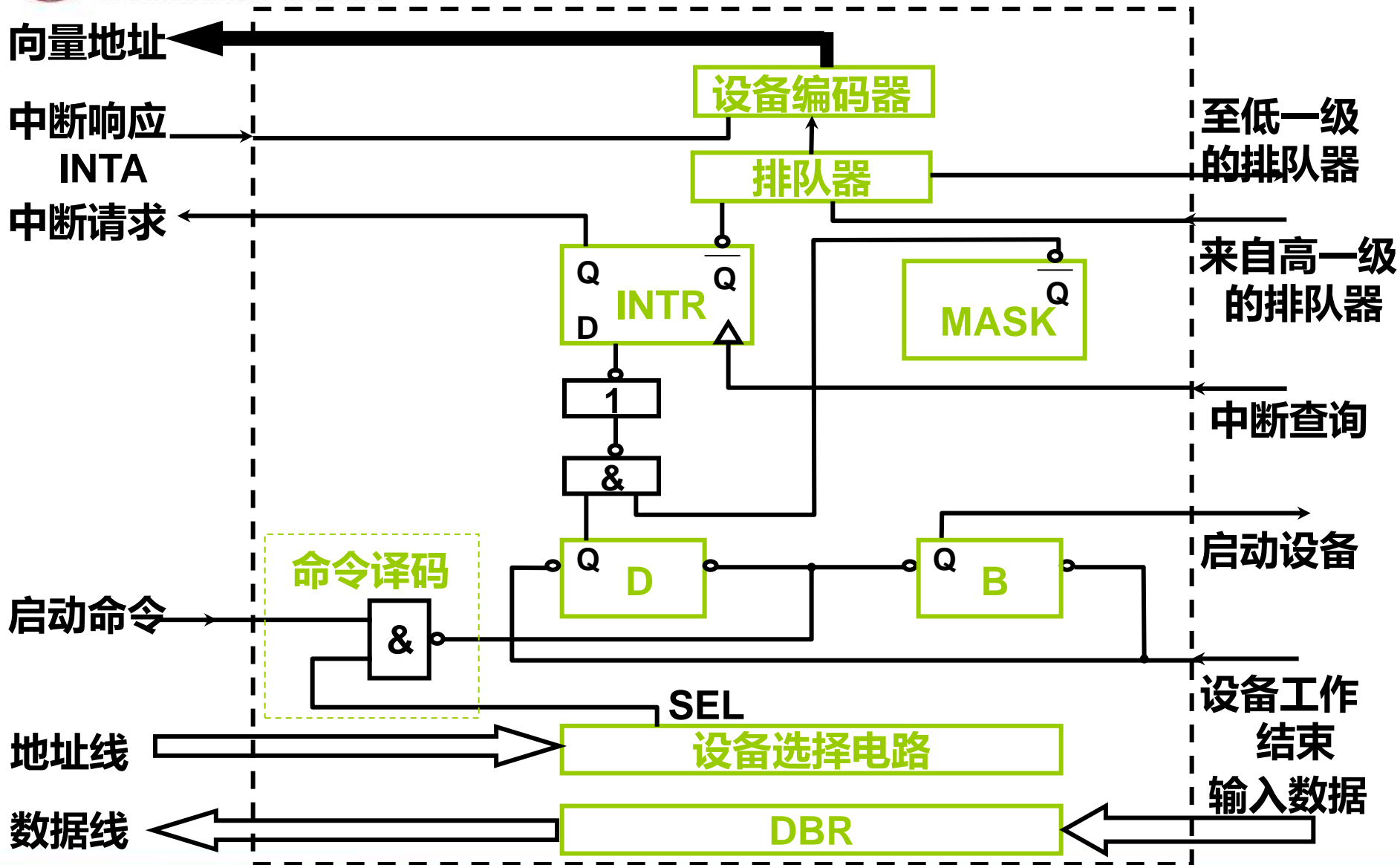


5.5.3 程序中断方式的接口电路





4. 程序中中断方式接口电路的基本组成





四、I/O 中断处理过程

1. CPU 响应中断的条件和时间

(1) 条件

中断允许触发器 **EINT = 1**

用 **开中断** 指令置 **"1"** EINT

用 **关中断** 指令置 **"0"** EINT 或硬件 **自动复位**

(2) 时间

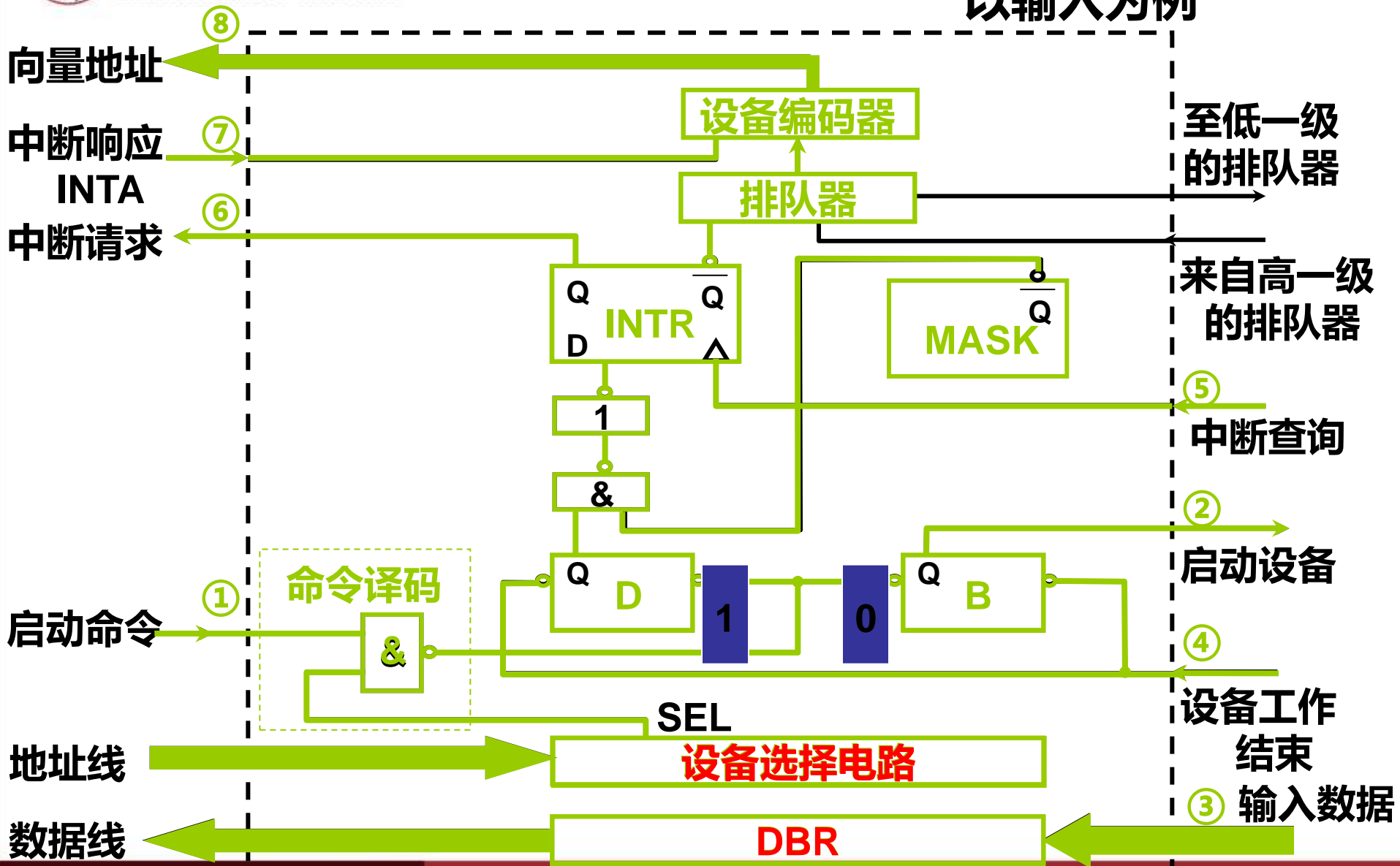
当 $D = 1$ (随机) 且 $MASK = 0$ 时

在每条指令执行阶段的结束前

CPU 发 中断查询信号 (将 INTR 置 "1")



以输入为例





1. 中断服务程序的流程

机器指令系统中没有的指令，它是CPU在中断周期内由硬件自动完成的一条指令。

(1) 保护现场

{ 程序断点的保护
寄存器内容的保护

中断隐指令完成

进栈指令

(2) 中断服务

对不同的 I/O 设备具有不同内容的设备服务

(3) 恢复现场

出栈指令

(4) 中断返回

中断返回指令

2. 单重中断和多重中断

单重 中断

不允许中断 现行的 **中断服务程序**

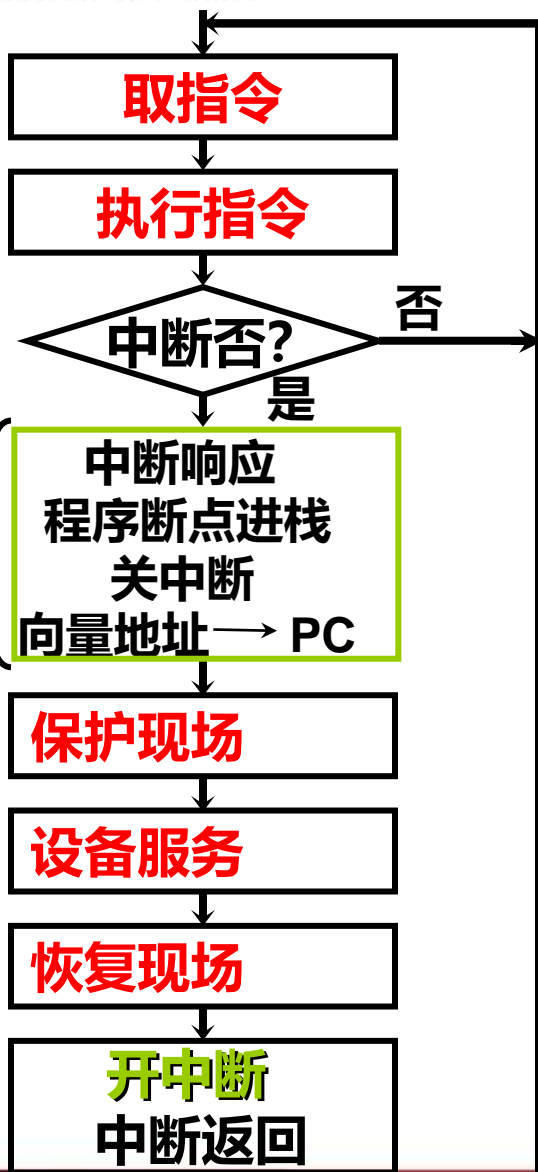
多重 中断
(中断嵌套)

允许级别更高 的中断源

中断 现行的 **中断服务程序**



单重

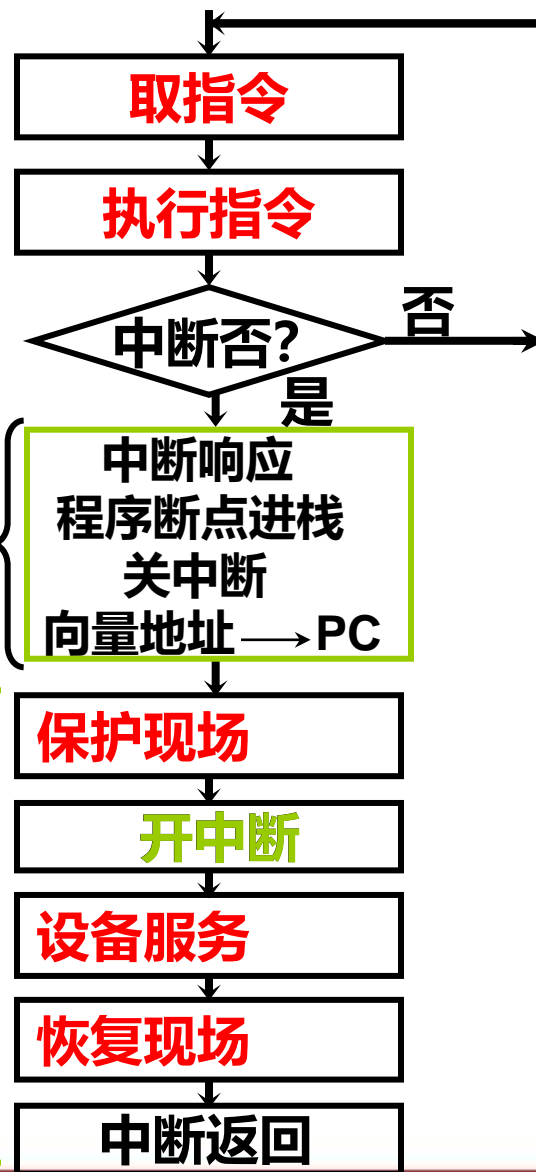


中断隐指令

中断周期

中断服务程序

多重



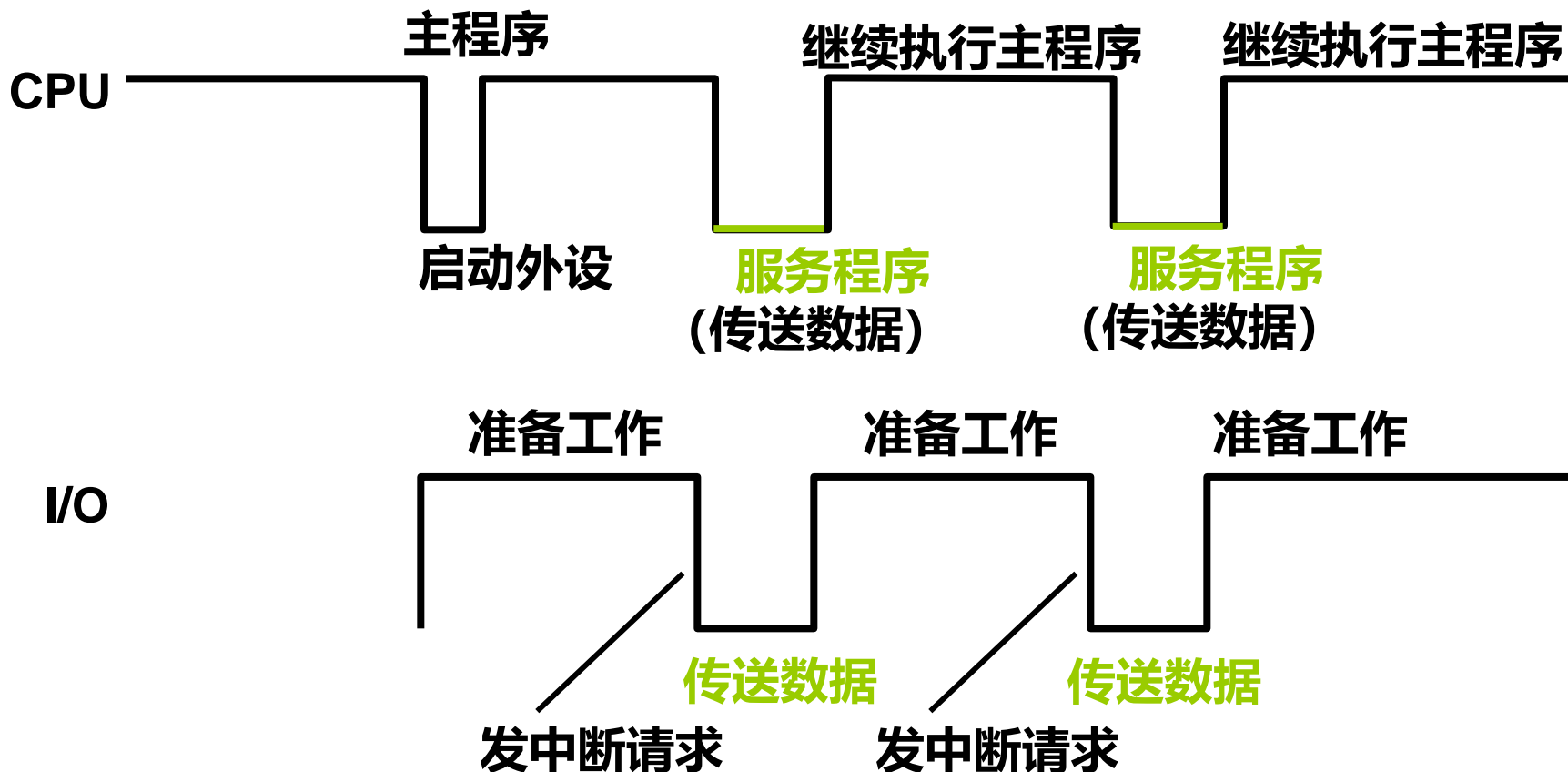
中断隐指令

中断周期

中断服务程序



主程序和服务程序抢占 CPU 示意



宏观 上 CPU 和 I/O 并行 工作

微观 上 CPU 中断现行程序 为 I/O 服务



中断过程包含五个阶段：

- 中断请求；
- 判断优先级（排队）；
- 中断响应（中断周期操作）；
- 中断处理；
- 中断返回。



多核CPU的中断处理和单核有很大不同。多核的各处理器核心之间需要通过中断方式进行通信，所以CPU芯片内部既有各处理器核心的**本地中断控制器**，又有负责仲裁各核之间中断分配的**全局中断控制器**。

现今的多核处理器在中断处理和中断控制方面主要使用的是**APIC**（Advanced Programmable Interrupt Controllers），即**高级编程中断控制器**。它是基于中断控制器两个基础功能单元——本地单元以及I/O单元的分布式体系结构。在多核系统中，多个本地和I/O APIC单元能够作为一个整体通过APIC总线互相操作。

APIC的功能有：

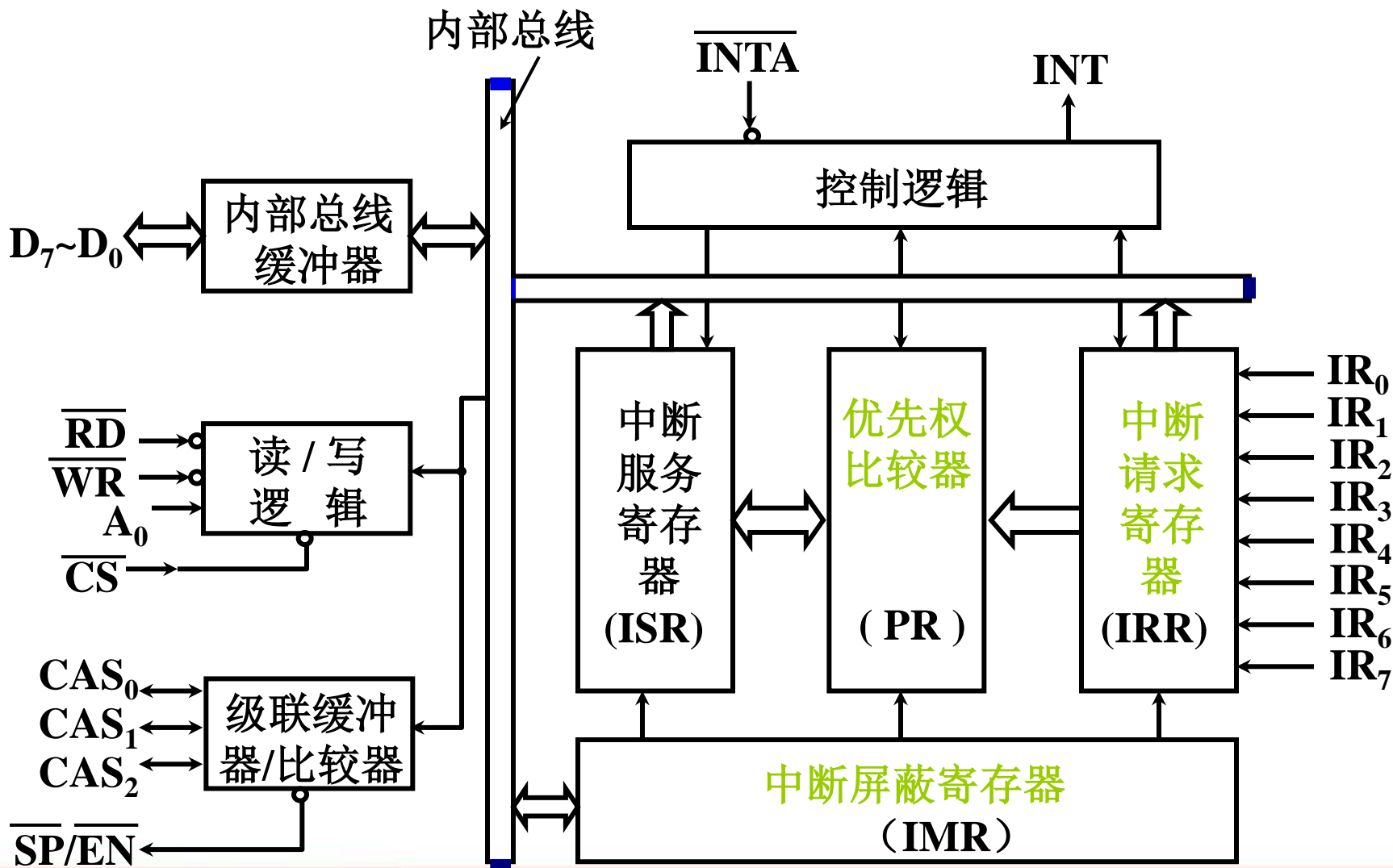
1. 接受来自处理器中断引脚的内部或外部I/O APIC的中断，然后将这些中断发送给处理器核心进行处理；
2. 在多核处理器系统中，接收和发送核内中断消息；

对于外部设备发出的中断请求，由全局中断控制器接收请求并决定交给CPU的哪一个核心处理。也可针对APIC编程，让所有的中断都被一个固定的CPU处理。



5.5 程序中中断方式

程序中中断接口芯片 8259A 的内部结构





5.5 程序中中断方式

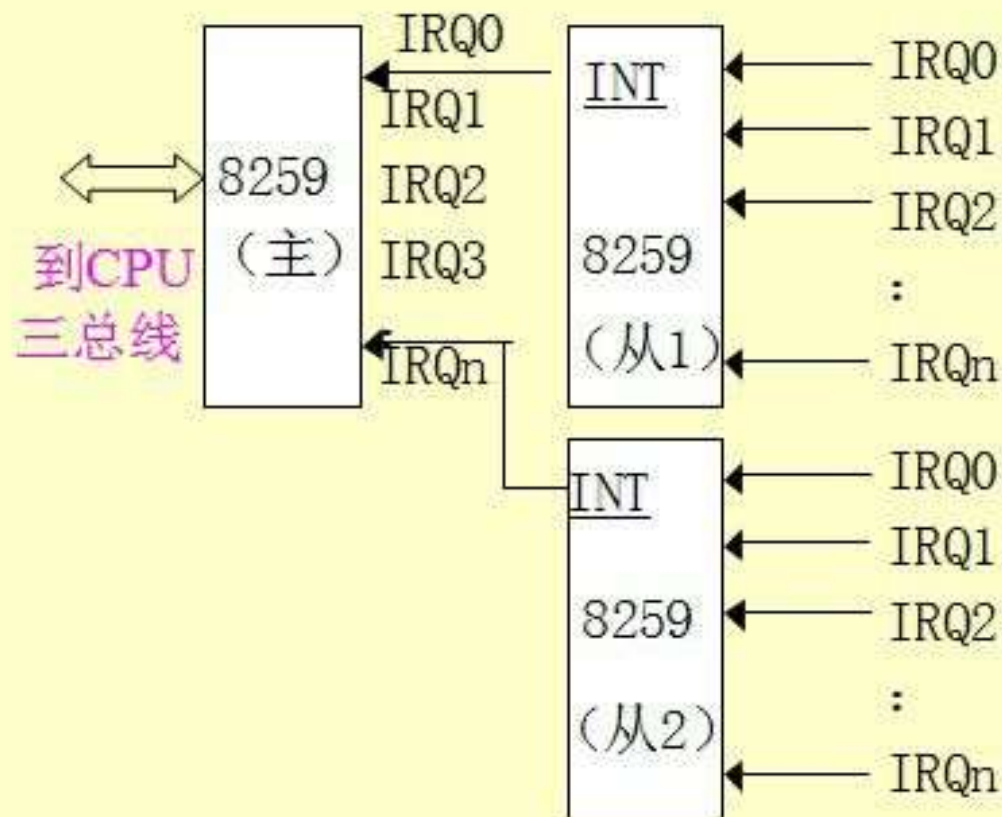


图6-18 8259A非缓冲方式下级连示意图



5.5 DMA方式

5.5.1 DMA方式的特点

5.5.2 DMA接口的功能和组成

5.5.3 DMA的工作过程

5.5.4 I/O DMA接口的类型



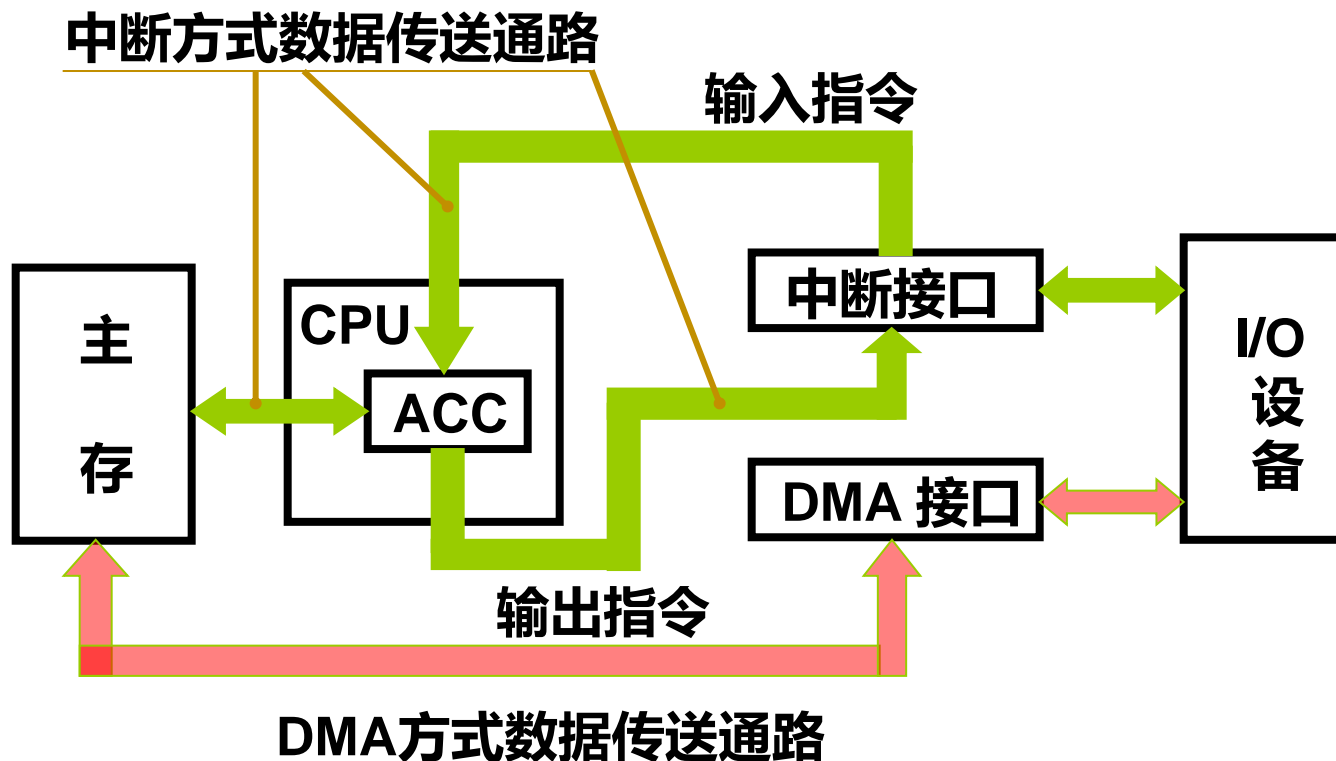
一、DMA 方式的特点

- 特点：主机与I/O并行工作，主存和I/O之间有一条**直接数据通路**，CPU启动I/O后，不必查询I/O是否准备好，当I/O准备就绪后，发出DMA请求，此时CPU**不直接参与**I/O和主存间的信息交换，只是把外部总线的使用权暂时交付DMA，仍然可以完成自身内部的操作，所以不必中断现行程序，只需暂停一个或几个**存取周期**访存，CPU效率更高。



5.6.1 DMA方式的特点

1. DMA 和程序中断两种方式的数据通路





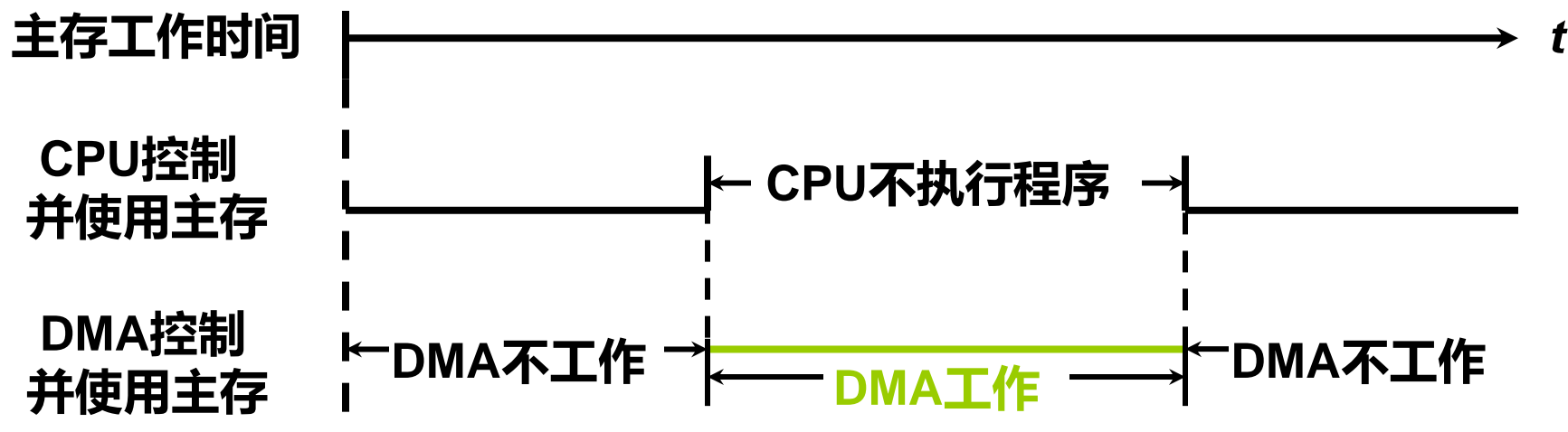
2. DMA与主存交换数据三种工作方式

(1) 停止 CPU 访问主存

优点：控制简单

缺点：CPU 处于不工作状态或保持状态

未充分发挥 CPU 对主存的利用率



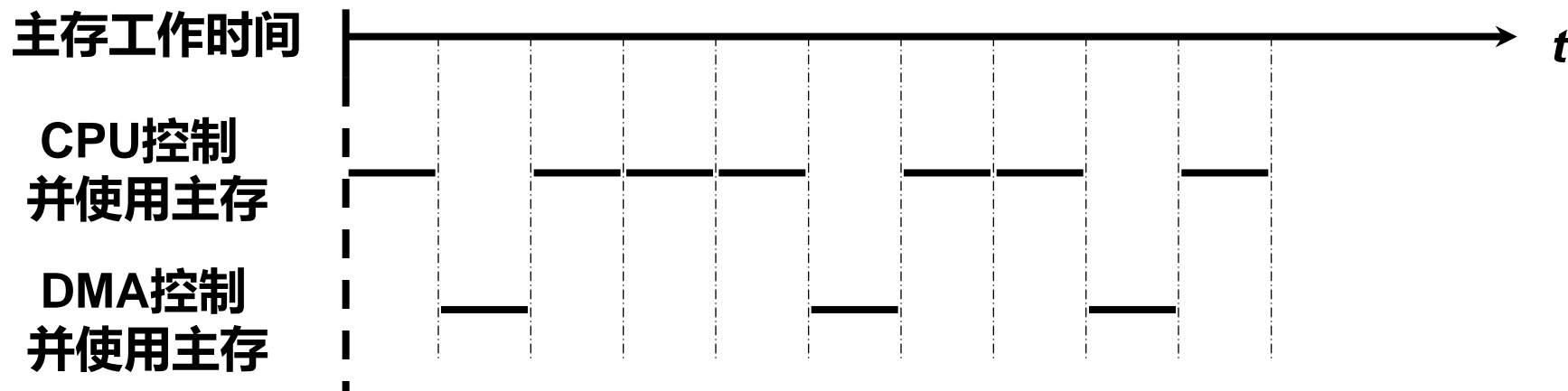


(2) 周期挪用 (或周期窃取)

DMA 访问主存有三种可能情况:

- CPU 此时不访存, DMA直接访问主存
- CPU 正在访存, 等存取周期结束CPU才能让出总线
- CPU 与 DMA **同时请求访存**

此时 **CPU 将总线控制权让给 DMA**



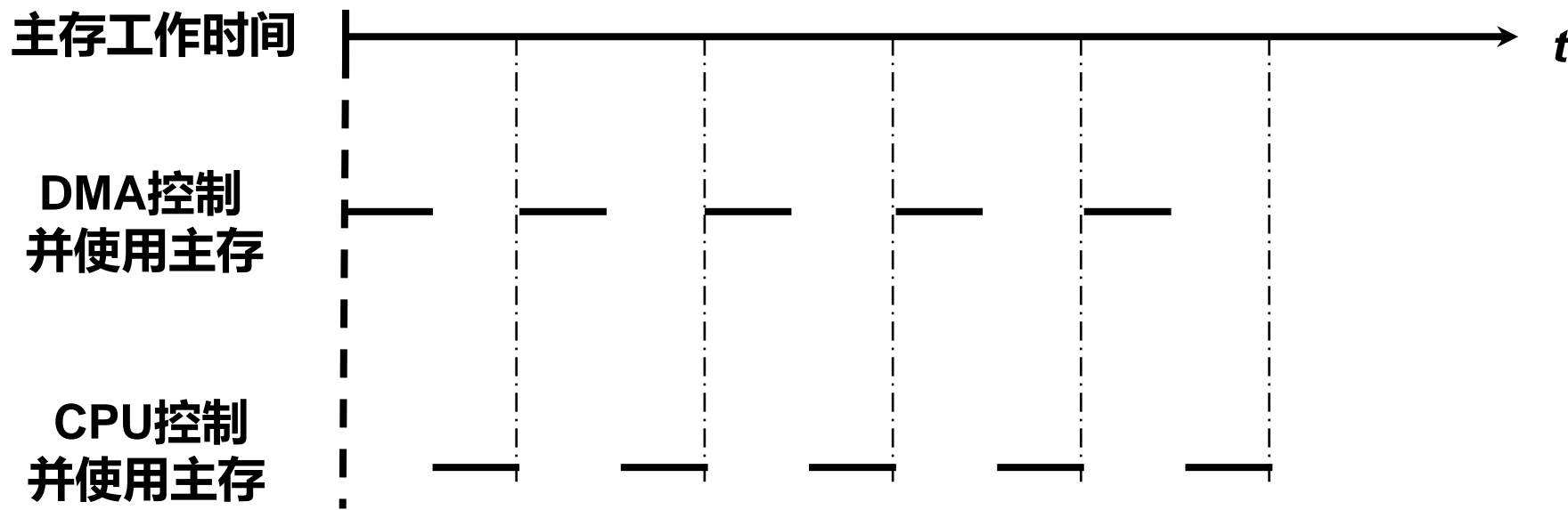
此方法适用于I/O设备读/写周期**大于**存取周期的情况。



(3) DMA 与 CPU 交替访问

适用于CPU工作周期**大于**主存取取周期的情况。

CPU **工作周期** $\left\{ \begin{array}{l} C_1 \text{ 专供 DMA 访存} \\ C_2 \text{ 专供 CPU 访存} \end{array} \right.$
↓
所有指令执行过程中的一个基准时间



不需要**建立和归还**总线使用权



一、DMA 接口的功能和组成

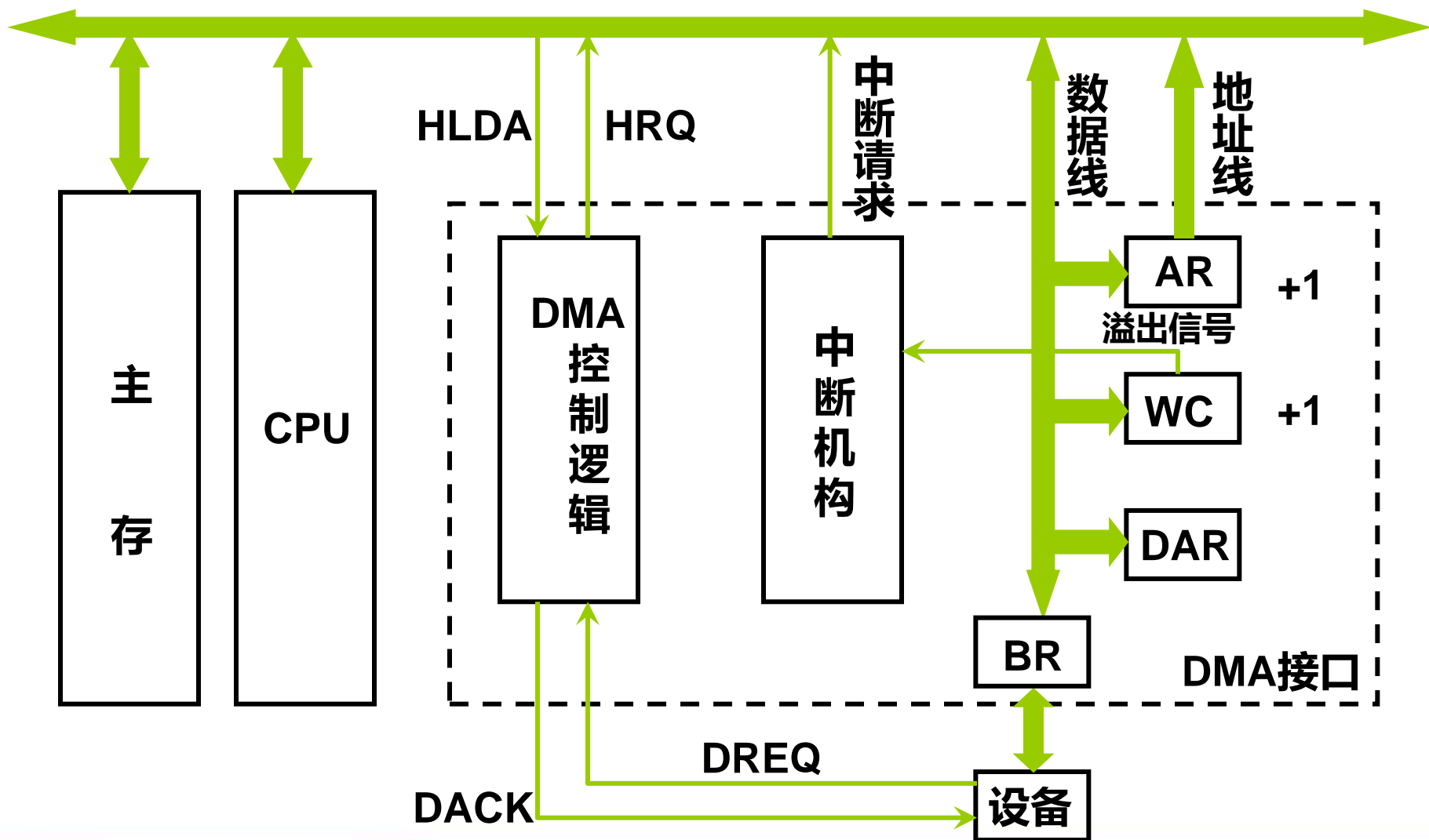
1. DMA 接口功能

- (1) 向 CPU **申请** DMA 传送
- (2) 处理总线 **控制权的转交**
- (3) **管理** 系统总线、**控制** 数据传送
- (4) **确定** 数据传送的 **首地址和长度**
修正 传送过程中的数据地址和数据长度
- (5) DMA 传送结束时，**给出操作完成信号**



DMA 记录待传递的总字数，每传递一个字寄存区的地址信息，当溢出信号溢出寄存区的最高位和进位时，表示待传递的所有数据传送完毕。

2. DMA 接口组成





5.6.2 DMA 接口的功能和组成

- DMAC (Direct Memory Access Controller) 是指直接内存访问控制器。
- 数据传送方式中，它是实现**地址的修改**与**传送字节数**技术的主要功能部件。



3. DMA 与主存交换数据的三种传输方式

(1) 单字节传输方式;

一次DMAC只传输一个字节，效率很低。

但是在传输过程中**CPU有机会获取对总线的控制器。**

(2) 数据块传输方式;

数据以数据块的方式进行传输。

只要DREQ启动就会连续地传送数据块。

一次请求传送一个数据块，效率高。

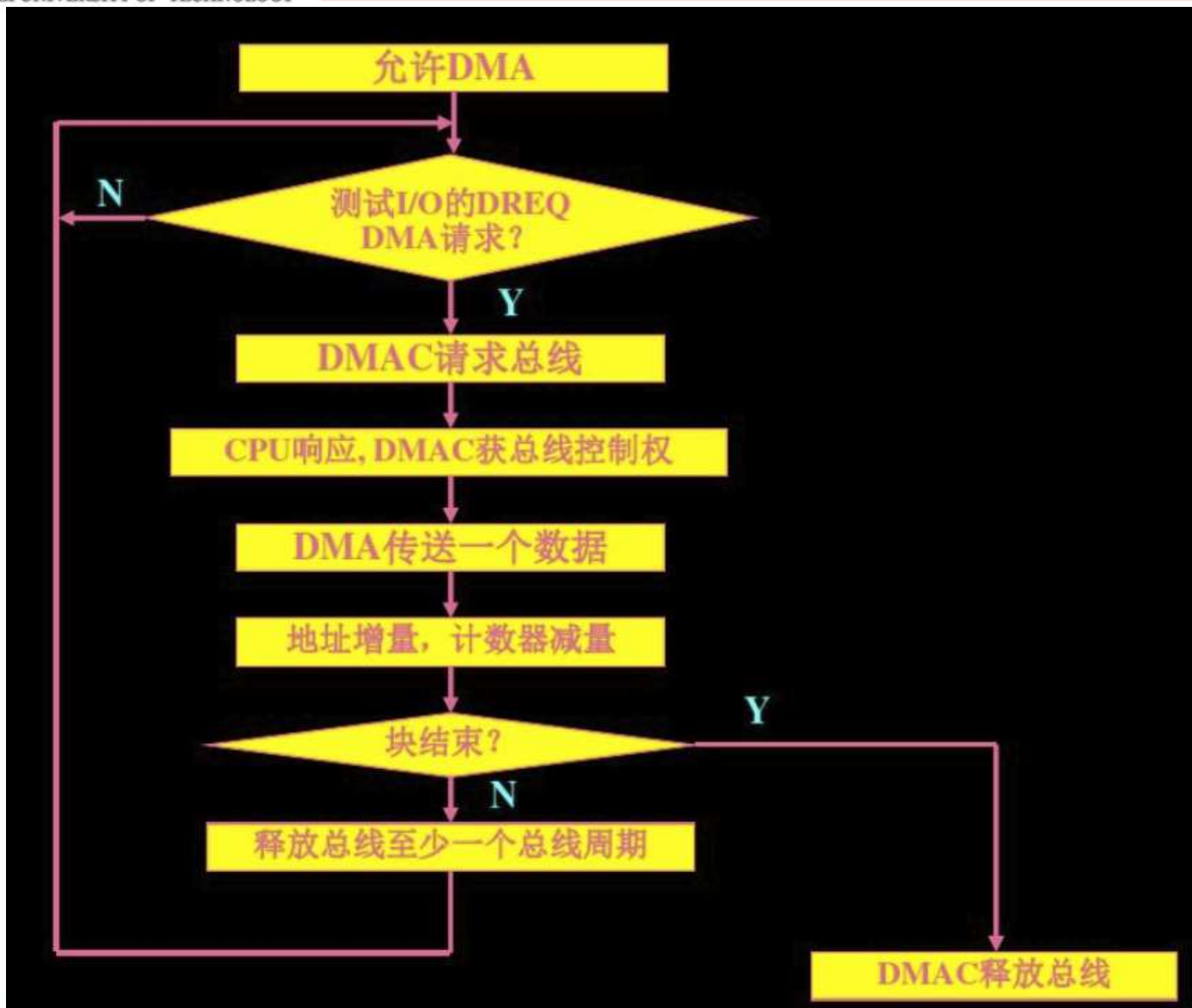
在数据的传送期间，CPU长时间无法控制总线。

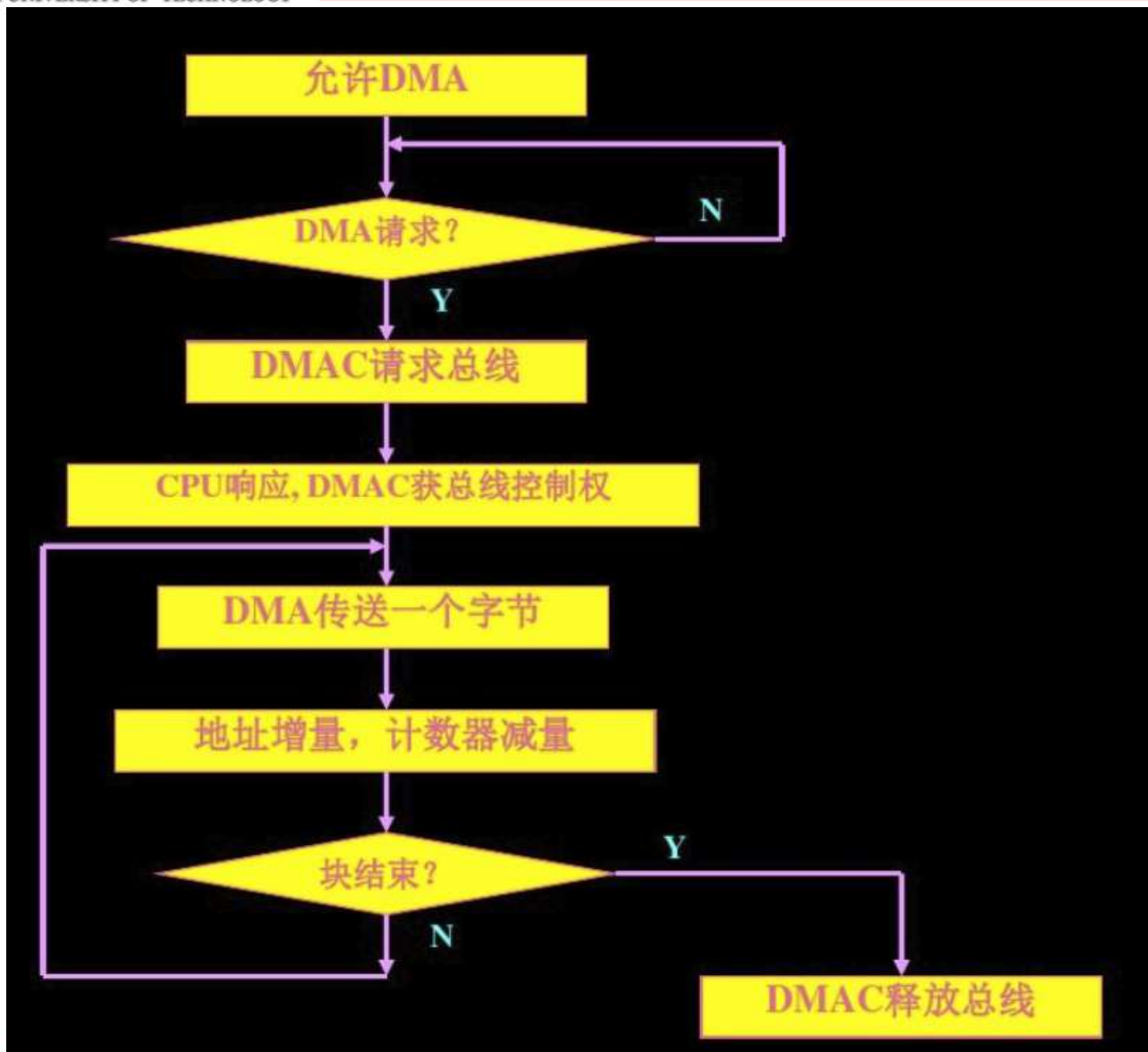
(3) 按需传输方式;

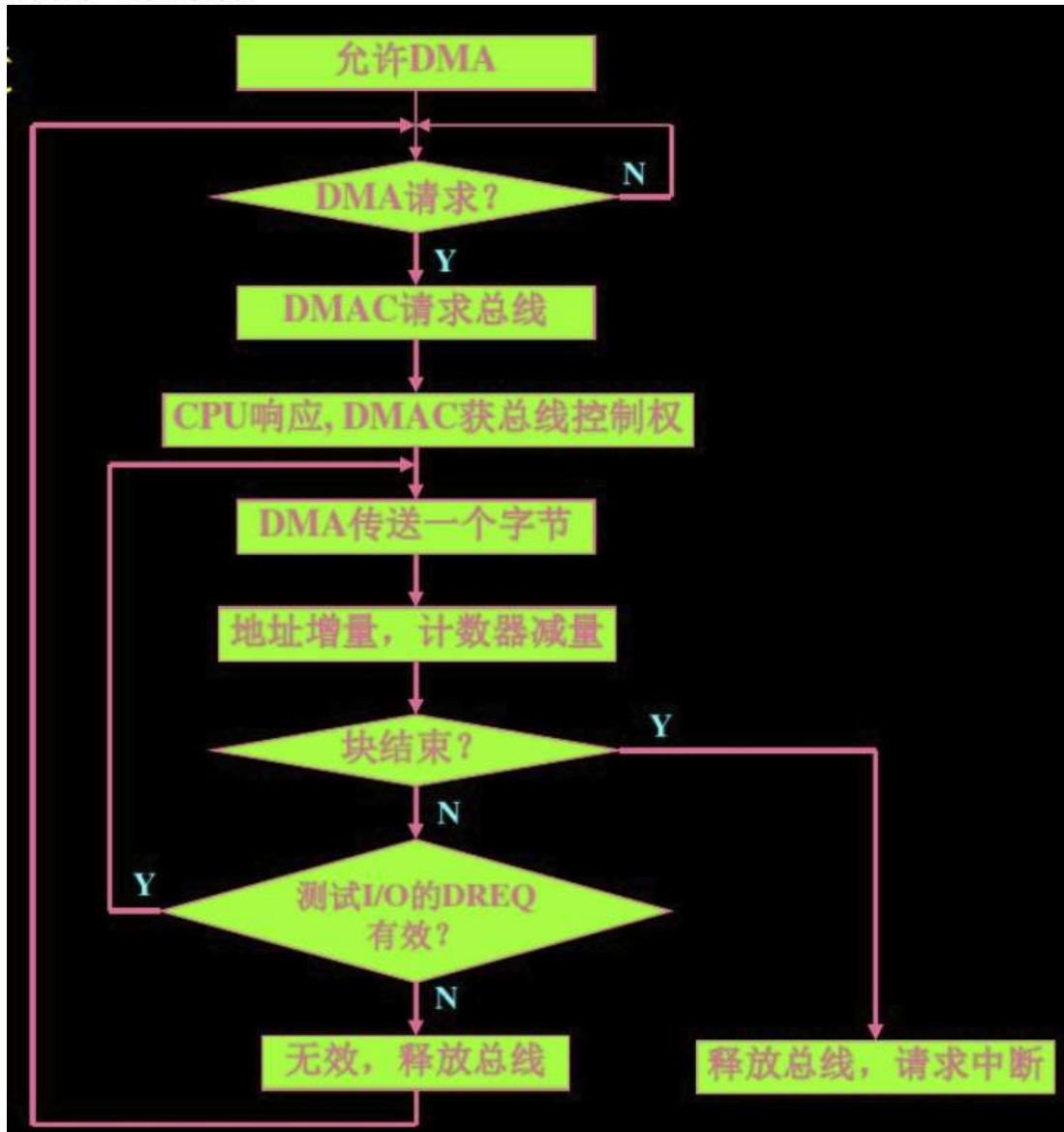
DREQ信号有效就连续传输数据，否则不能进行数据的传输。



单字节传输方式

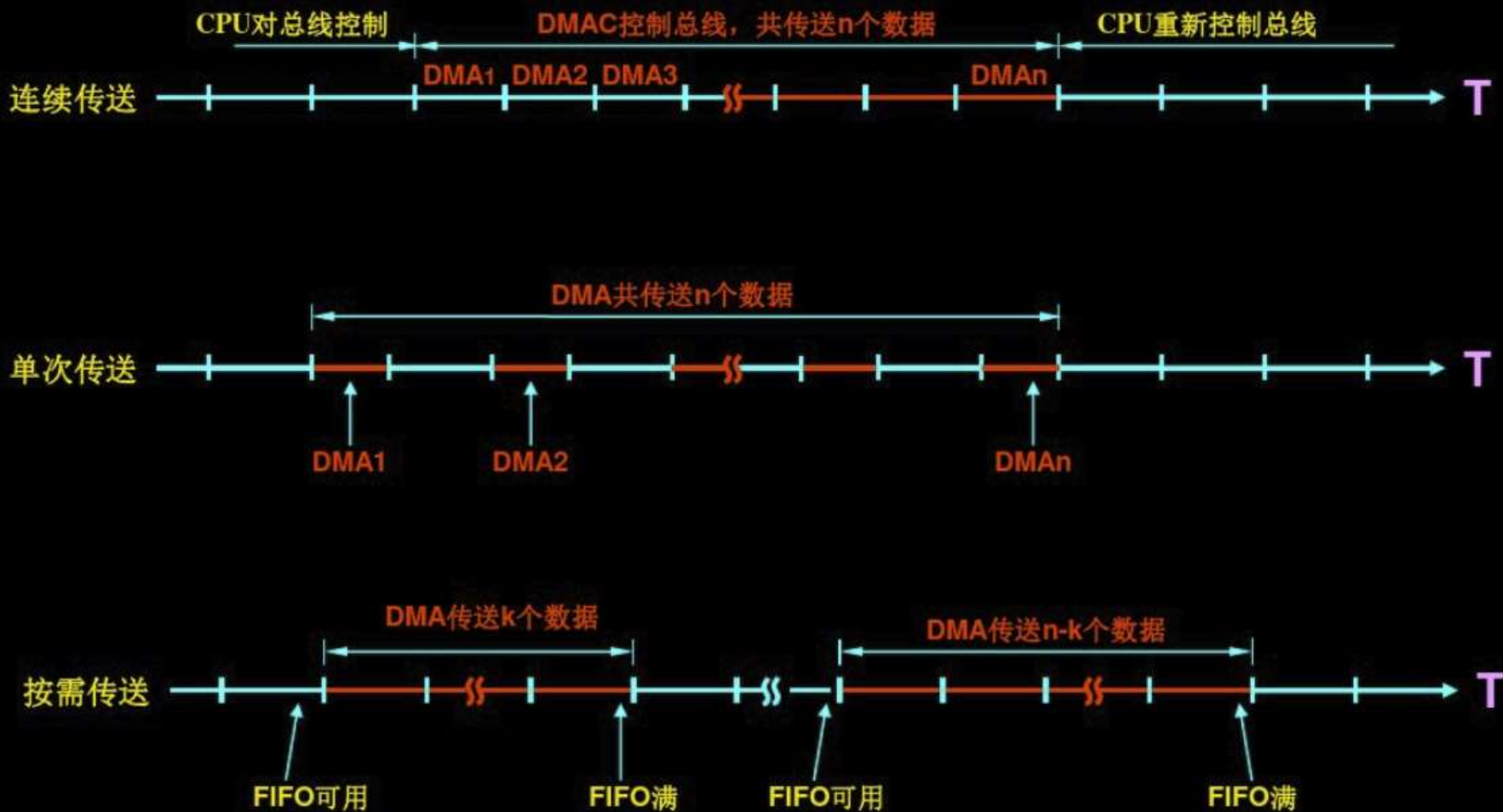








三种传输方式示意图



图例： ———— 一个总线周期



一、DMA 的工作过程

1. DMA 传送过程

预处理、数据传送、后处理

(1) 预处理

通过几条输入输出指令预置如下信息

- 通知 DMA 控制逻辑传送方向（入/出）
- 设备地址——→DMA 的 DAR
- 主存地址——→DMA 的 AR
- 传送字数——→DMA 的 WC



CPU (2) DMA 传送过程示意图

预处理:

主存起始地址 → DMA
设备地址 → DMA
传送数据个数 → DMA
启动设备



数据传送:

继续执行主程序
同时完成一批数据传送

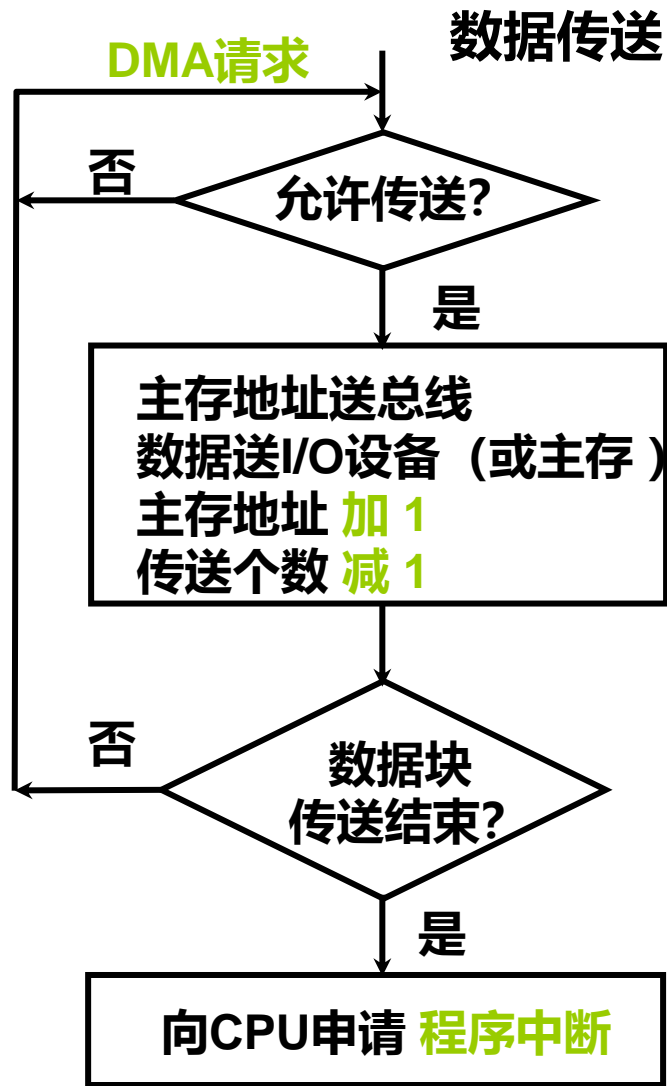


后处理:

中断服务程序
做 DMA 结束处理



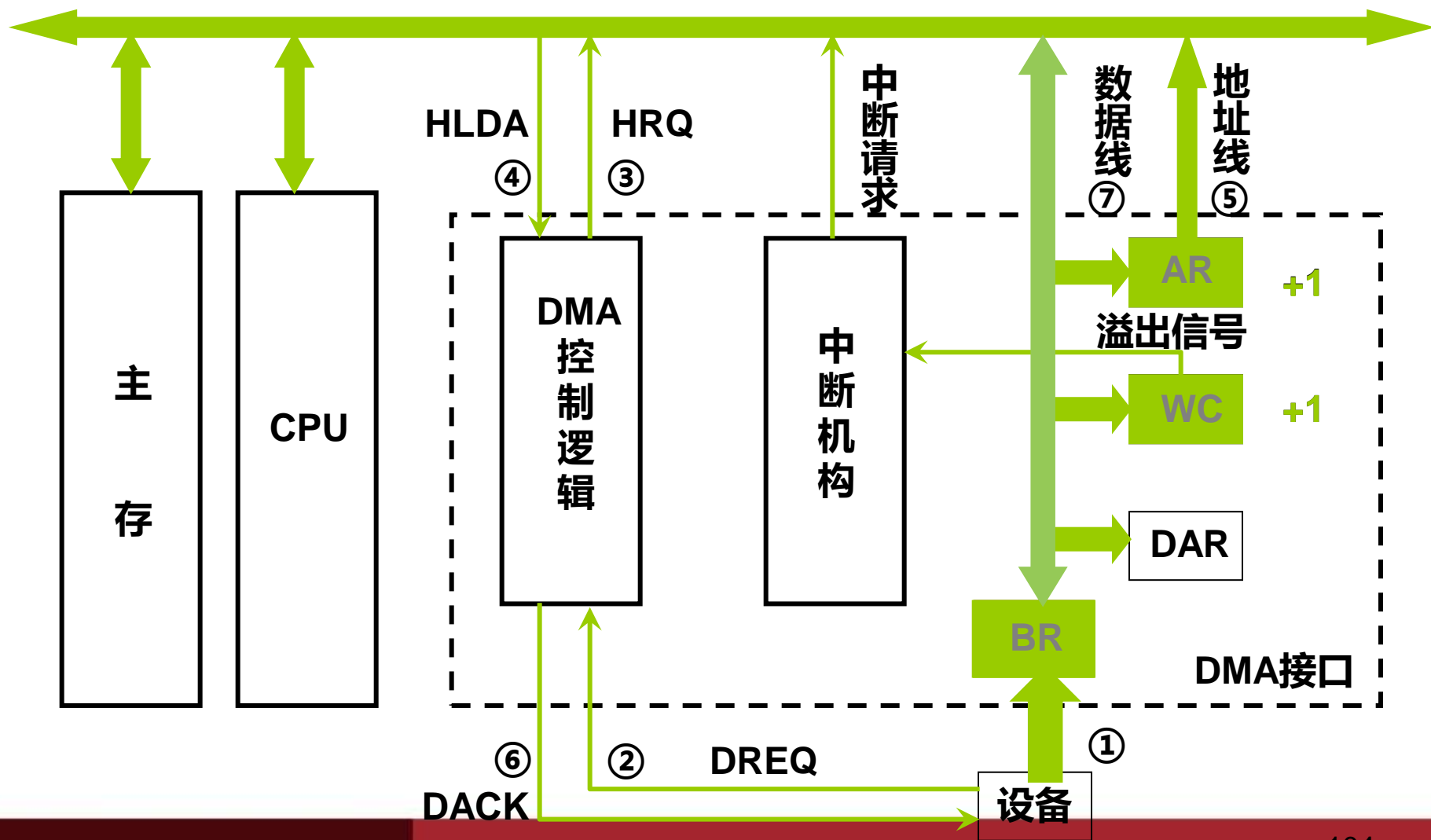
继续执行主程序





5.6.3 DMA 的工作过程

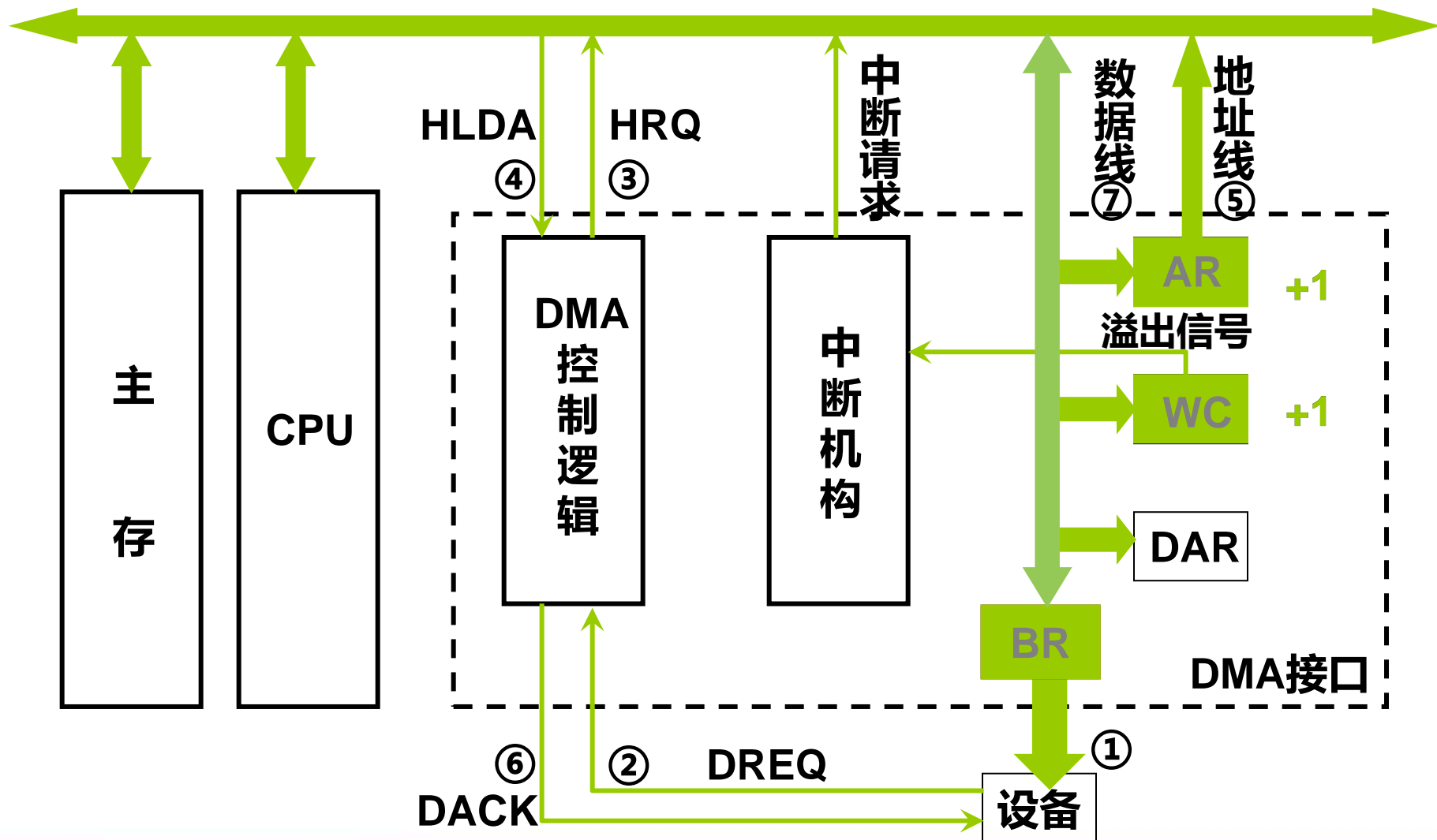
(3) 数据传送过程 (输入)





5.6.3 DMA 的工作过程

(4) 数据传送过程 (输出)





(5) 后处理

校验送入主存的数是否正确

是否继续用 DMA

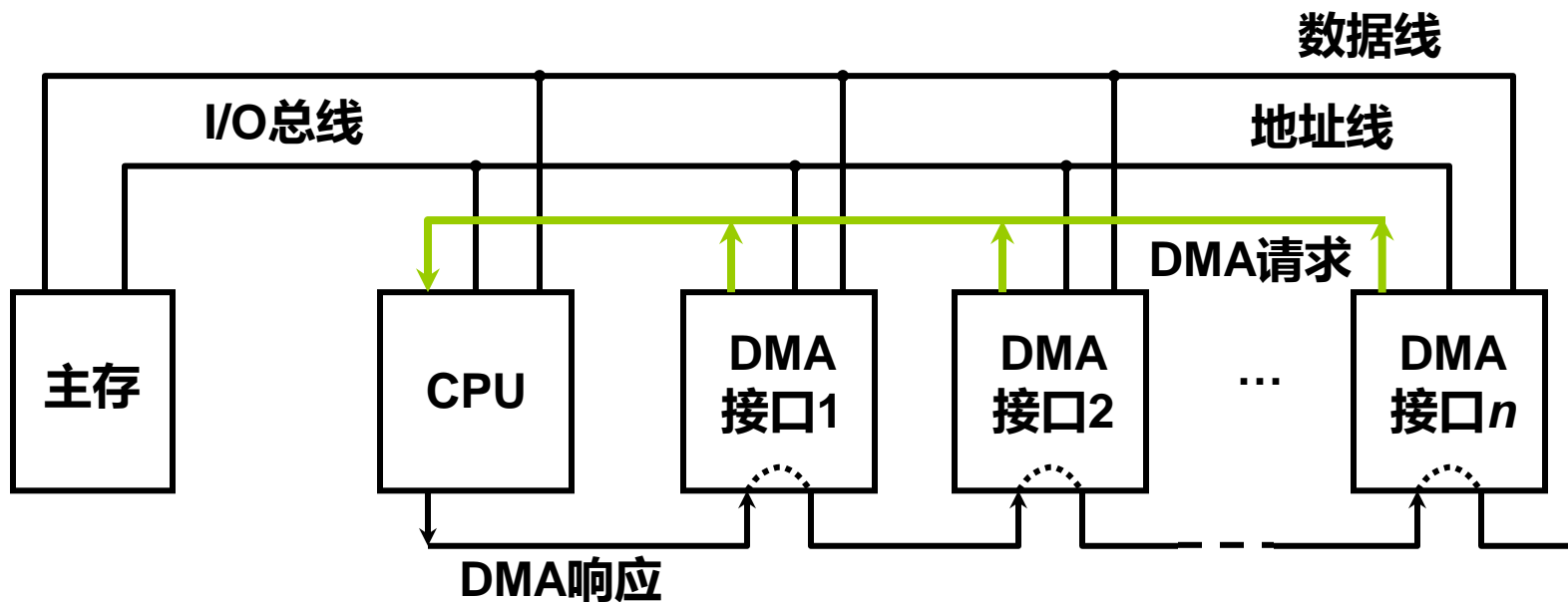
测试传送过程是否正确，错则转诊断程序

由中断服务程序完成



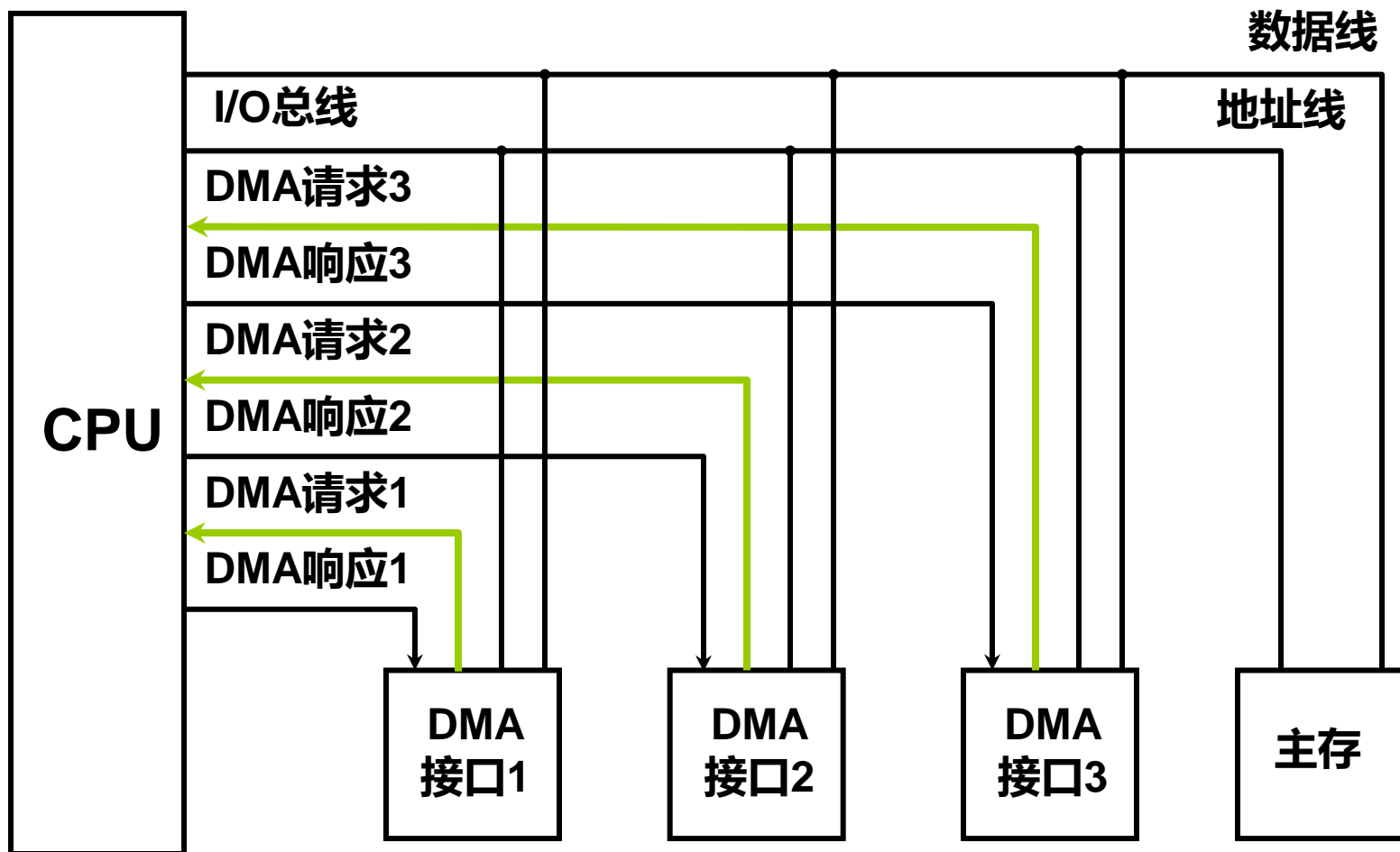
2. DMA 接口与系统的连接方式

(1) 具有公共请求线的 DMA 请求





(2) 独立的 DMA 请求





3. DMA 方式与程序中中断方式的比较

	中断方式	DMA 方式
(1) 数据传送	程序	硬件
(2) 响应时间	指令执行结束	存取周期结束
(3) 处理异常情况	能	不能
(4) 中断请求	传送数据	后处理
(5) 优先级	低	高

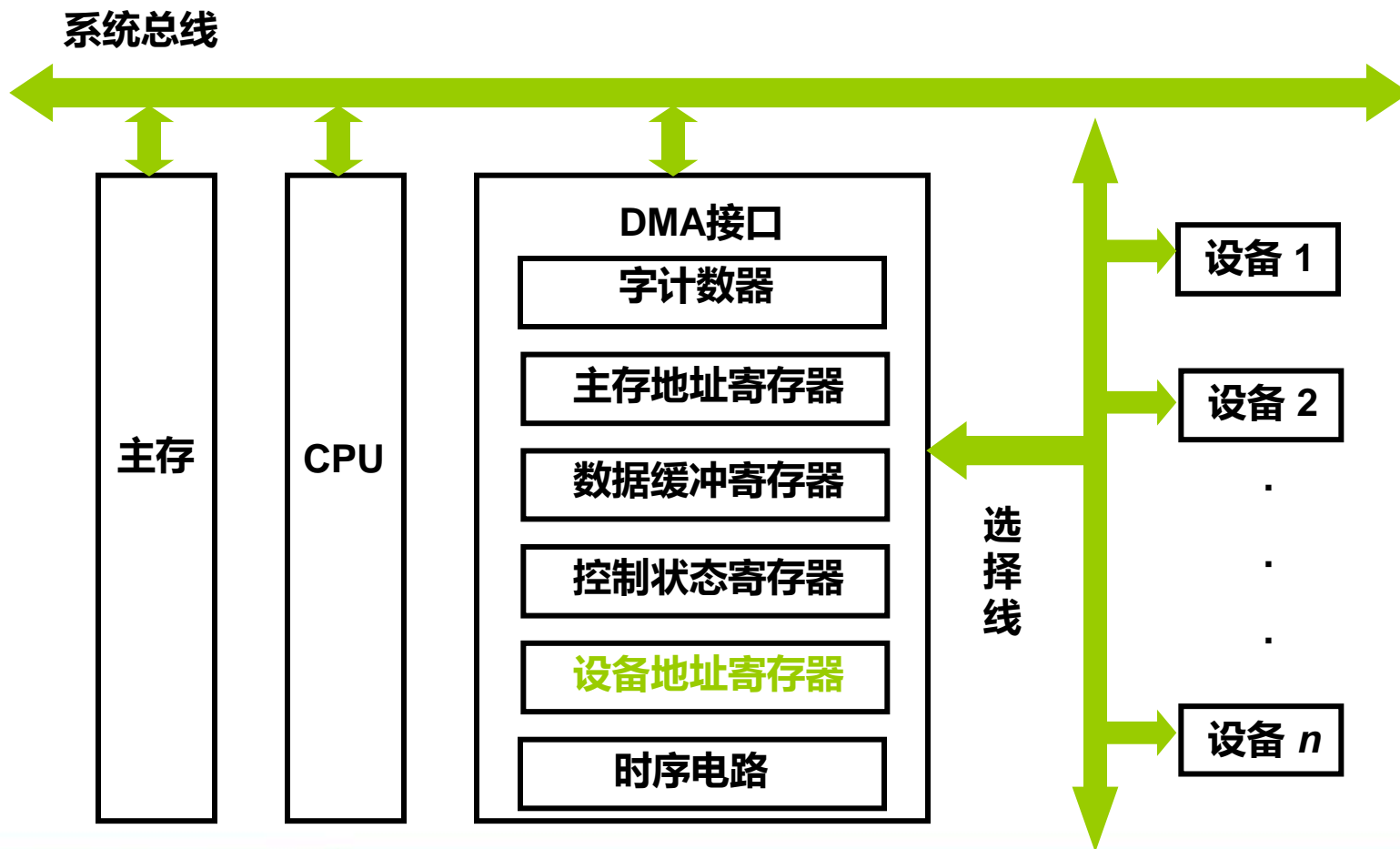


一、DMA 接口的类型

1. 选择型

在 **物理上** 连接 **多个** 设备

在 **逻辑上** 只允许连接 **一个** 设备



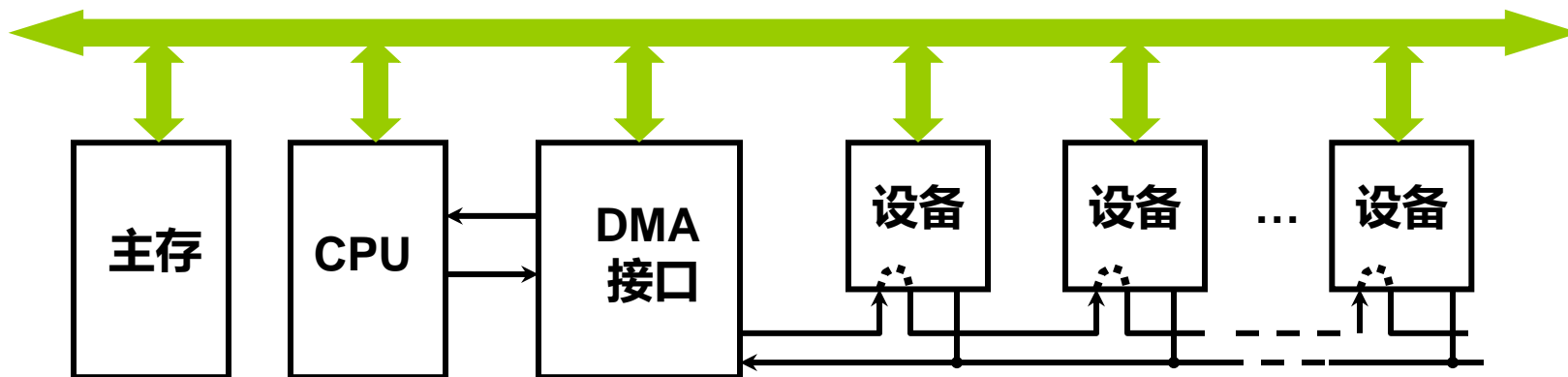


5.6 DMA 方式

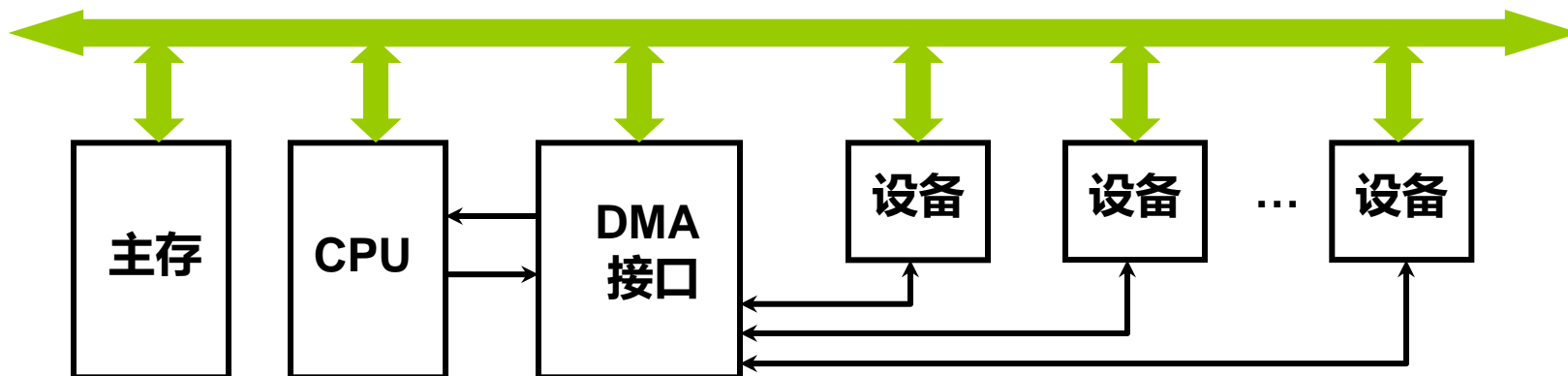
2. 多路型

在 **物理上** 连接 **多个** 设备
在 **逻辑上** 允许连接 **多个** 设备同时工作

链式

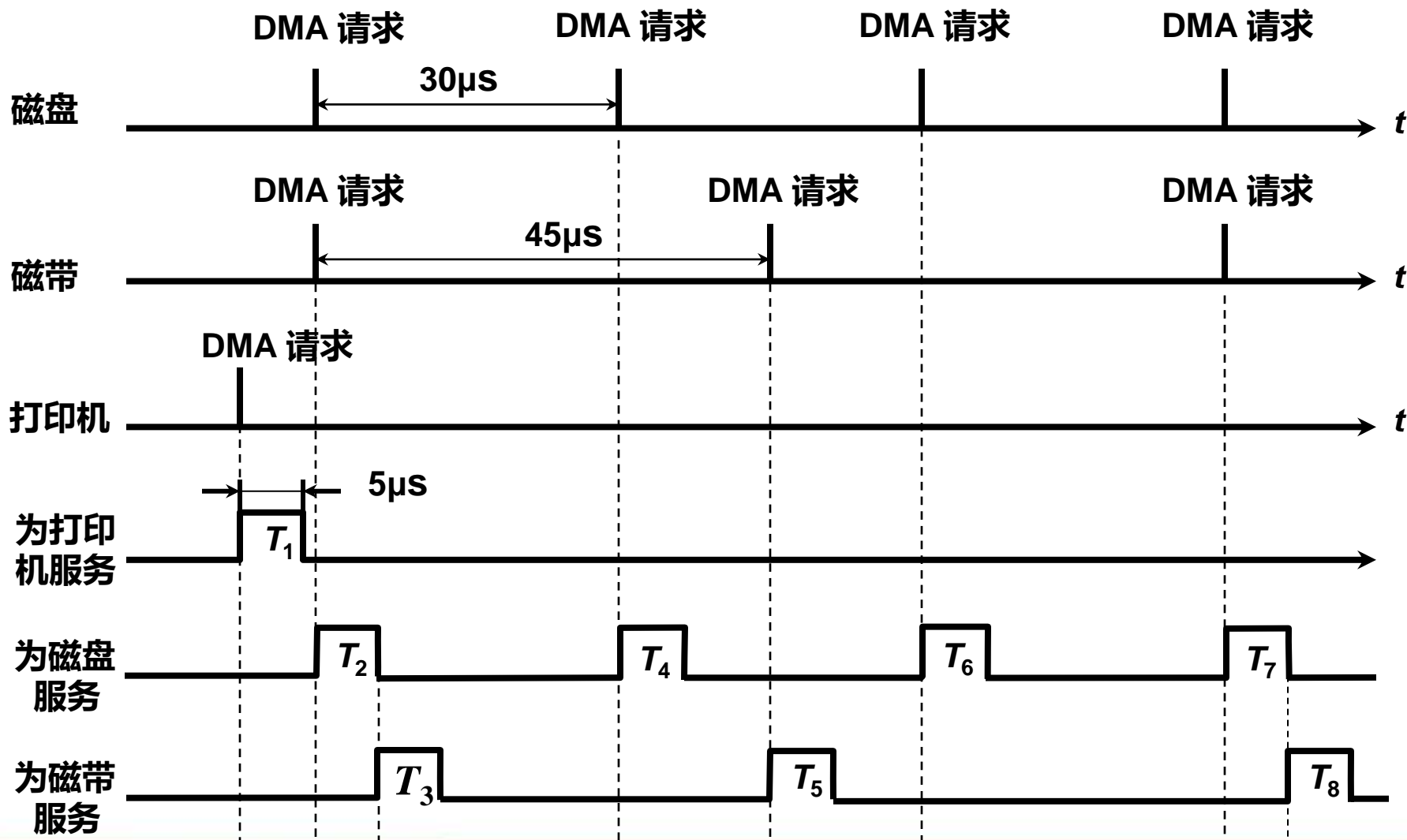


独立请求式





3. 多路型 DMA 接口的工作原理





- IDE硬盘、SATA硬盘等高速辅助存储器；
- 显示卡、PCIE、一部分网卡都是支持DMA的，USB 2.0/3.0设备也是支持DMA的，这是指在底层驱动支持。
- 采用数据块传送方式的外围设备，如行式打印机，激光打印机，卡片阅读机，部分绘图仪等。



合肥工业大学

HEFEI UNIVERSITY OF TECHNOLOGY

谢谢！



逻辑门	IEEE 推荐符号	我国部颁符号	输出表达式	标准符号
“与”门			$F = A \cdot B$	
“或”门			$F = A + B$	
“非”门			$F = \bar{A}$	
“与非”门			$F = \overline{A \cdot B}$	
“或非”门			$F = \overline{A + B}$	
“异或”门			$F = A \oplus B$ $= A\bar{B} + \bar{A}B$	
“与或非”门			$F = \overline{AB + CD}$	



名称	国标符号	曾用符号	国外流行符号
与			
或			
非			
与非			
或非			
与或非			
异或			
同或			