



微机原理和接口技术

第十一讲 中断系统1



提 纲

1. 中断系统概述

2. 8051微控制器的中断系统

3. 中断处理过程

4. 中断程序设计

5. IO端口扩展外部中断源

提 纲

1. 中断系统概述



中断系统概述

中断：微控制器执行程序过程中，由于内部或者外部的某种原因，要求MCU中止正在运行的程序，转去执行另外一段处理程序，待处理结束后，再回来继续执行被中止了的原程序。这种程序在执行过程中，由于外界的原因而被中间打断的情况称为“中断”。

中断是硬件改变**CPU**程序运行方向的一种技术，既和硬件有关，也和软件有关。先进的中断系统能提高MCU实时处理外界异步事件的能力。

- **主程序（调用程序）：**原来运行的程序；
- **中断服务程序：**中断之后执行的处理程序；
- **断点：**主程序被中断的位置（地址）；
- **中断源：**引起中断的原因或能发出中断申请的来源；
- **中断请求：**中断源要求服务的请求。



中断系统概述

中断程序与子程序：

- 调用中断服务程序的过程类似于程序设计中调用子程序的过程。在执行程序的过程中，由于中断源发出中断请求，MCU响应后转去执行一段中断服务程序，相当于在中断发生时刻调用一个子程序。
- 但是子程序的调用是程序预先设计安排好的；而中断源发出中断请求是随机的，所以中断服务程序的调用是无法预知的，中断服务程序的调用过程是计算机系统硬件自动完成的。



中断系统概述

1. 分时操作

利用中断可以很好协调快速CPU与慢速外设互相配合高效地工作。CPU启动速度较慢的外设后，外设要经过比较长的时间完成一个任务，完成任务后需要CPU服务时，向CPU请求中断。CPU可命令多个外设同时工作，并在发生中断时及时为外设提供服务。

例如：外设ADC转换结束或有按键按下时向CPU请求中断，CPU中断正在执行的主程序转去执行中断服务程序，读取AD转换结果或扫描按键得到键值，中断处理完成后CPU返回主程序继续执行原程序，外设继续工作。

2. 实时处理

在微机系统中，依靠中断技术能实现实时控制。即控制对象发出实时操作请求时，通过中断可以使CPU快速地作出响应。

3. 故障处理

如发生电源掉电，通信故障等，可以向CPU请求中断，以便及时作出处理。



中断系统概述

发出中断请求的内部功能模块或外部设备来源一般统称为“中断源”。

1. 外部设备

单片机的输入/输出设备，如A/D、打印机、按键等，可通过接口电路向CPU申请中断。

2. 内部设备

定时器定时时间到或计数个数到请求中断；串行口发送完一帧或接收到一帧数据的中断请求。

3. 故障源

如掉电故障、硬件故障、运算错误、程序运行故障等请求中断，使得CPU能够以中断方式及时处理发生的故障。

4. 控制对象

微控制器的控制对象，如电压、温度等检测量超过上下限时，继电器、开关动作时，向CPU请求中断。

中断系统概述

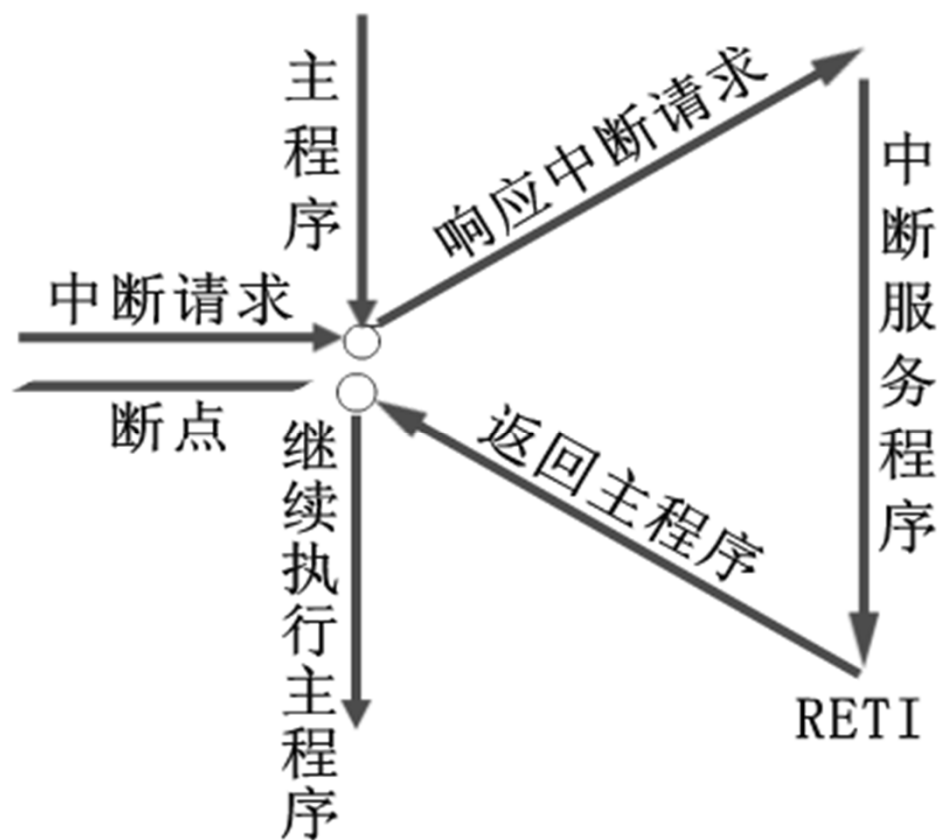
1. 中断的允许和禁止

即关中断和开中断，根据需要能够用指令控制中断的开放和关闭。只有在开中断情况下，CPU才能响应中断源的请求。

2. 中断响应和返回

- **保护断点：** CPU在现行指令执行完毕、转到中断程序前，把断点处的PC值（即下一条指令的地址）压入堆栈（由硬件自动完成）。
- **保护现场：** 用户在编写中断服务程序时，须对中断程序中用到的工作寄存器和SFR等内容进行保护。
- **恢复现场：** 在中断返回前，恢复保护的内容。
- **恢复断点：** 中断服务程序的最后一条指令必须为中断返回指令RETI，其功能是自动恢复断点地址送到PC，使CPU返回到断点处继续执行主程序。

中断系统概述





中断系统概述

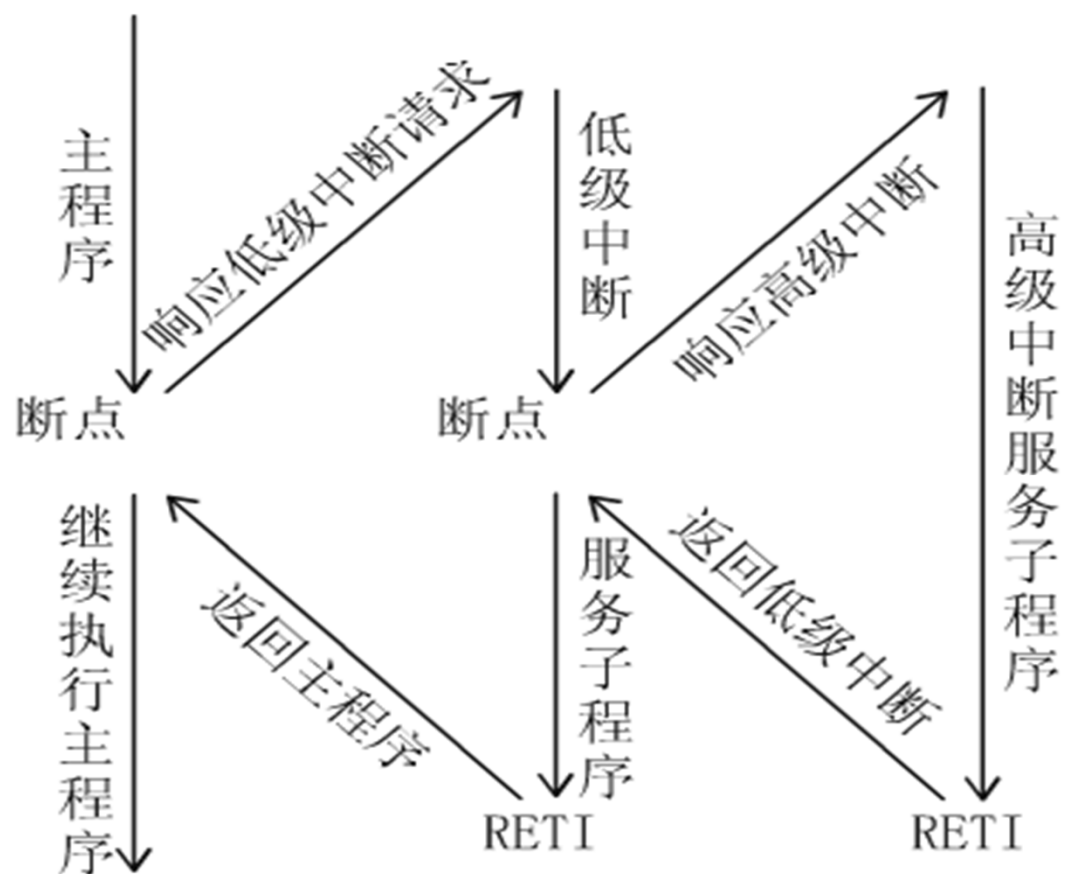
3. 中断优先级与中断嵌套

➤ **中断优先级：**当有多个中断源同时请求中断时，CPU会根据各中断源的优先级别，首先响应优先级高的中断请求，当该中断程序执行完毕返回主程序后，再响应优先权较低的中断源，实现中断优先级的控制。这个过程是MCU的中断系统自动完成的。

➤ **中断嵌套：**当CPU正在执行低级的中断服务程序时，若有高级中断源申请中断，则能够停下低级中断服务程序转去执行高级中断源的服务程序，实现中断嵌套，并能逐级正确返回。

如果新的中断请求的优先级与正在处理的中断是同级别或低一级，则CPU暂时不响应这个新中断申请，直到正在处理的中断服务程序执行完毕，才会给予响应。

中断系统概述



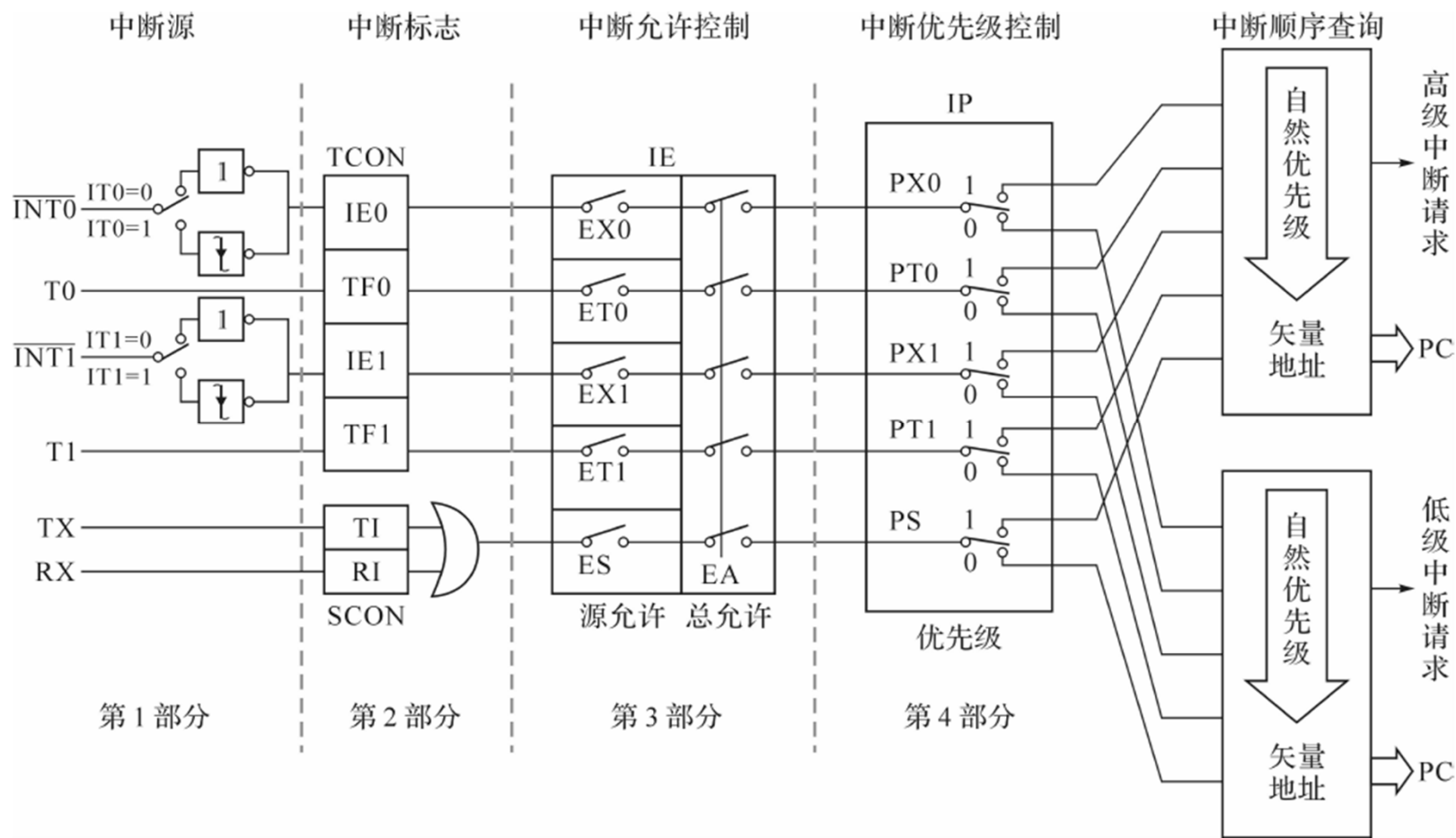
提 纲

2. 8051微控制器的中断系统

51的中断系统

1.中断系统结构图

由5个中断源、4个SFR（TCON、SCON、IE、IP）、中断查询逻辑电路等组成。





51的中断系统

2.中断源

8051微控制器有5个中断源。分别是外部中断、定时器/计数器中断和串行口中断。

(1)外部中断

- 两个外部中断源：INT0和INT1，外部中断请求信号分别从引脚INT0 (P3.2) 和INT1 (P3.3) 引入。外设如按键、掉电检测电路信号等可以请求外部中断。
- 二种中断触发方式：低电平触发和下降沿触发，可通过编程进行选择。

(2)定时中断

- T0、T1溢出中断：当T0、T1定时时间到或发生溢出时，向CPU请求中断。

(3)串行口中断

当串行口发送完一个字节数据或接收到一个字节数据时，产生中断请求。

51的中断系统

3.中断入口

5个中断源的中断入口地址是固定的。当CPU响应某中断源的中断请求后，硬件自动将断点地址压入堆栈保护，并将此中断源的中断入口地址赋给PC，使CPU执行该中断的中断服务程序。

中断源	入口地址
外部中断0 (INT0)	0003H
定时器/计数器0 (T0)	000BH
外部中断1 (INT1)	0013H
定时器/计数器1 (T1)	001BH
串行口中断 (TX或RX)	0023H

5个中断源的6个中断请求信号，分别为：INT0、T0、INT1、T1、TX、RX，其中TX和RX是串行口中断的发送中断请求和接收中断请求，这两个请求共用一个中断源。

51的中断系统

8051微控制器中与中断系统有关的SFR有TCON、SCON、IE、IP

1.定时器/计数器控制寄存器TCON (Timer Control)

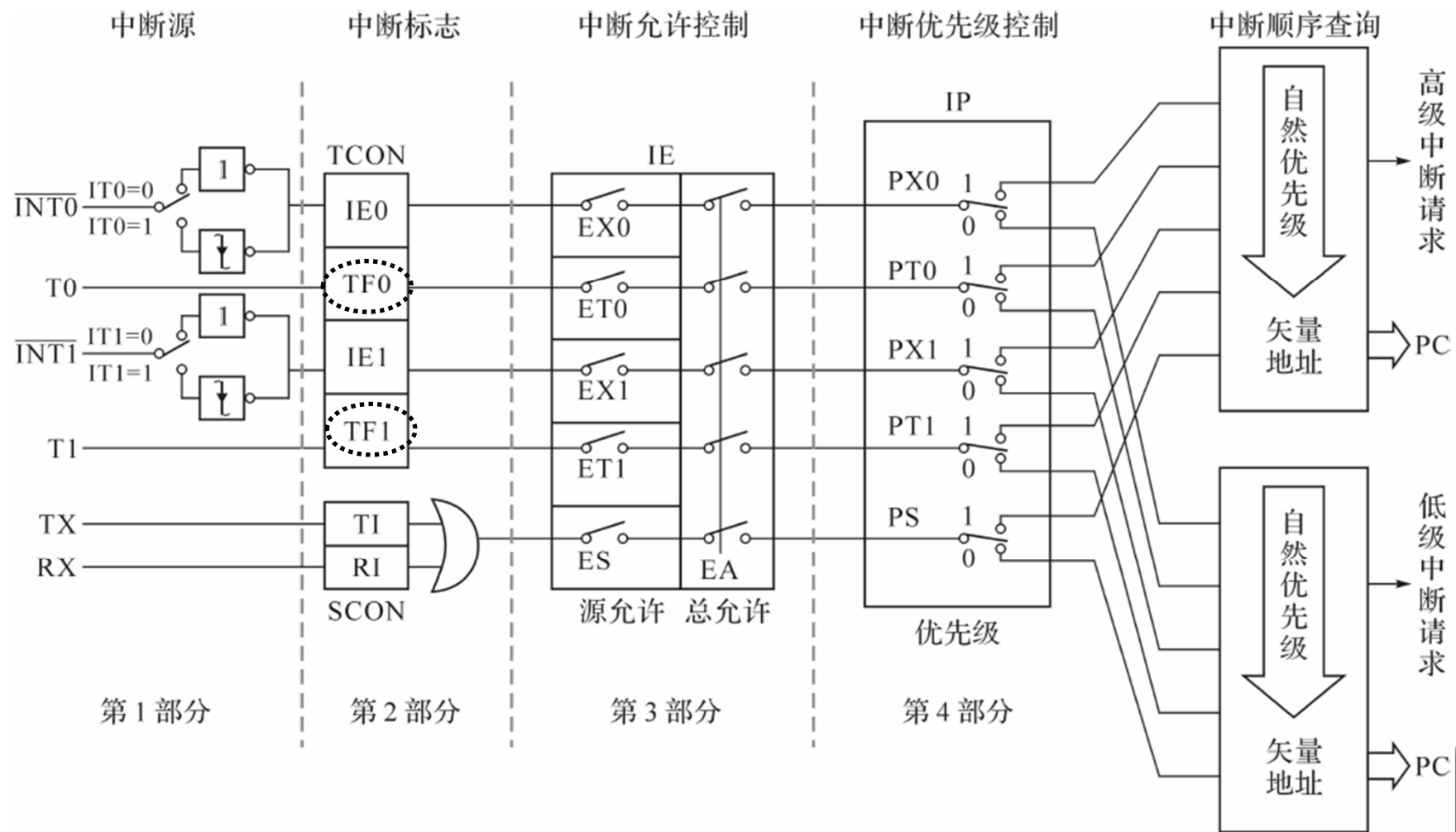
TCON的字节地址为88H，是可以位寻址的SFR。

	7	6	5	4	3	2	1	0
位符号	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
英文注释	Timer1 Overflow	Timer1 Run	Timer0 Overflow	Timer0 Run	Interrupt External 1 flag	Interrupt 1 Type control bit	Interrupt External 0 flag	Interrupt 0 Type control bit

- **TF0**: T0溢出标志，溢出时由硬件置1，并请求中断，CPU响应后，由硬件自动将TF0清0；查询方式时，要用软件清0。
- **TF1**: T1溢出标志，溢出时由硬件置1，并请求中断，CPU响应后，由硬件自动将TF1清0；查询方式时，要用软件清0。

51的中断系统

TF0 和 TF1 在哪儿?



51的中断系统

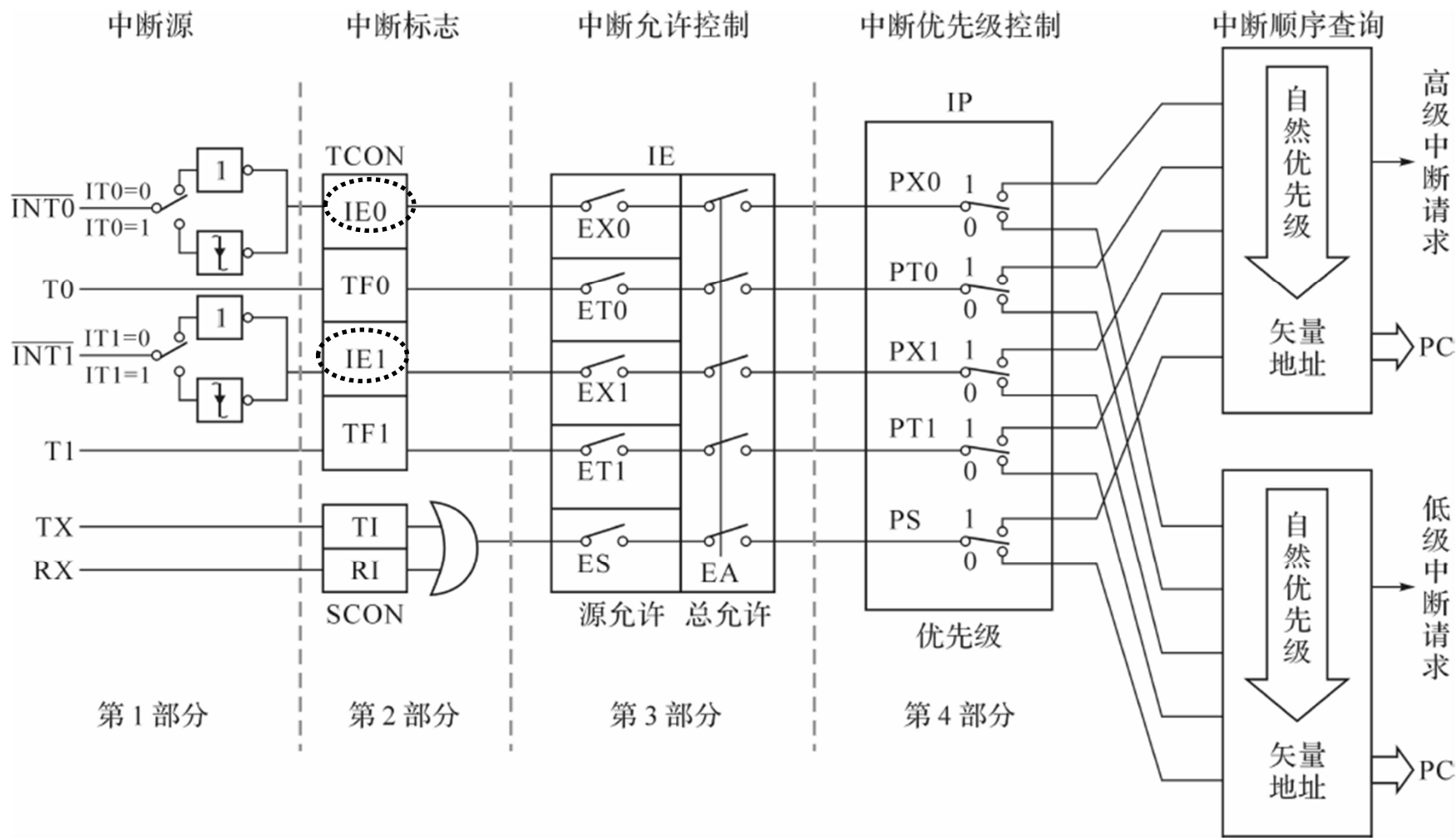
1.定时器/计数器控制寄存器TCON (Timer Control)

- **IE0**: INT0中断标志, 发生INT0中断时, 硬件置IE0为1, 并向CPU请求中断。
- **IE1**: INT1中断标志, 发生INT1中断时, 硬件置IE1为1, 并向CPU请求中断。

	7	6	5	4	3	2	1	0
位符号	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
英文注释	Timer1 Overflow	Timer1 Run	Timer0 Overflow	Timer0 Run	Interrupt External 1 flag	Interrupt 1 Type control bit	Interrupt External 0 flag	Interrupt 0 Type control bit

51的中断系统

IE0 和 IE1 在哪儿？



51的中断系统

1.定时器/计数器控制寄存器TCON (Timer Control)

	7	6	5	4	3	2	1	0
位符号	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
英文注释	Timer1 Overflow	Timer1 Run	Timer0 Overflow	Timer0 Run	Interrupt External 1 flag	Interrupt 1 Type control bit	Interrupt External 0 flag	Interrupt 0 Type control bit

- **IT0**: 外部中断0的触发方式选择位, 设置为“0”时, 表示选择低电平触发; 设置为“1”时, 表示选择下降沿触发 (边沿触发)。
- **IT1**: 外部中断1的触发方式选择位, 设置为“0”时, 表示选择低电平触发; 设置为“1”时, 表示选择下降沿触发 (边沿触发)。

实际使用时, 常采用下降沿触发方式, 低电平触发很少使用。



51的中断系统

1.定时器/计数器控制寄存器TCON (Timer Control)

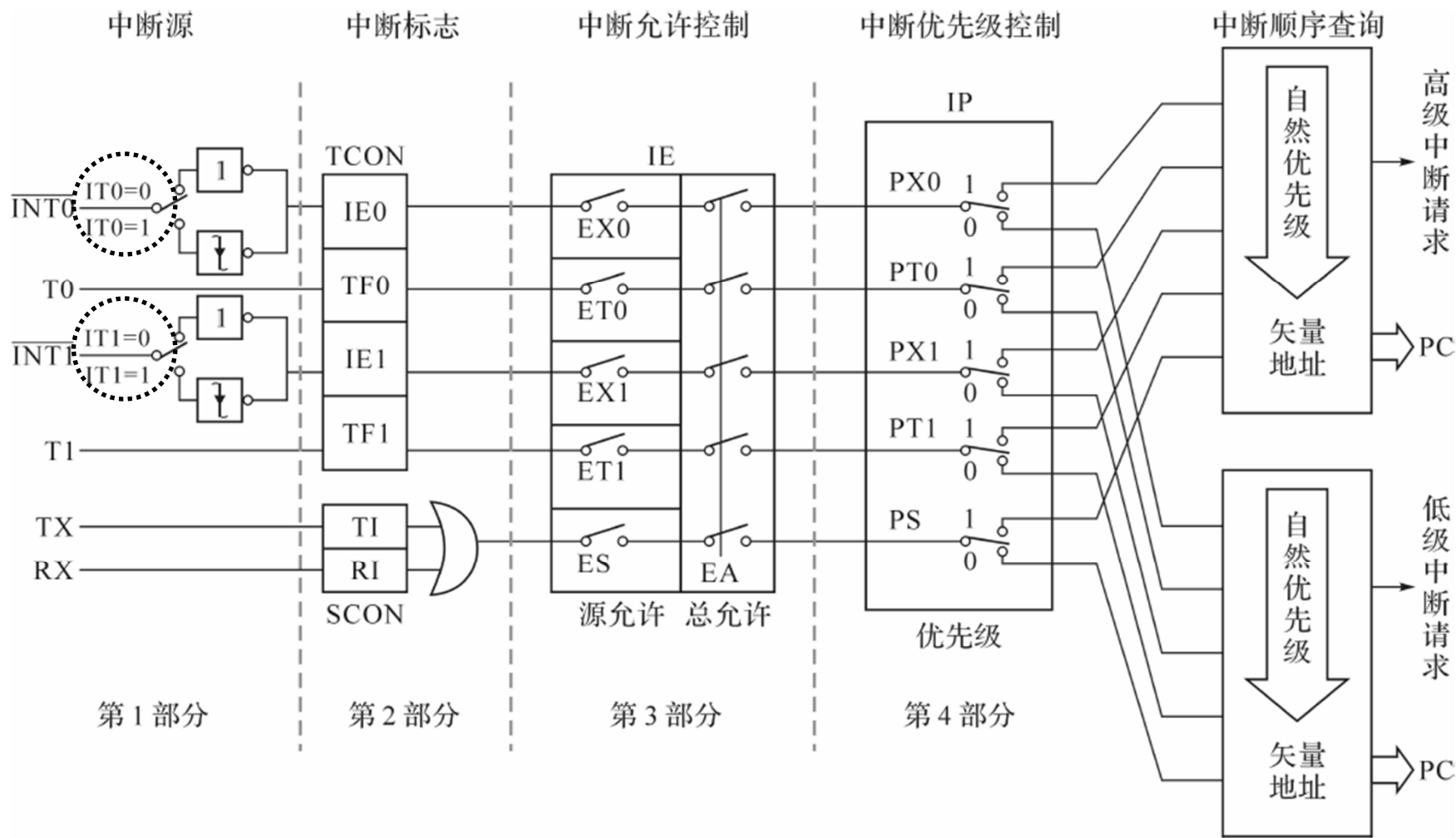
外部中断的检测过程:

CPU每个机器周期检测一次INT0、INT1引脚, 对中断请求信号的要求为:

- 对于下降沿触发方式: INT0、INT1引脚上中断请求信号的高、低电平至少应各保持一个机器周期;
- 对于低电平触发方式: INT0、INT1引脚上中断请求信号的低电平应保持到CPU响应中断为止。

51的中断系统

IT0 和 IT1 在哪儿?



51的中断系统

2.串行口控制寄存器SCON (Serial Control)

SCON的字节地址为98H，可位寻址，用于串行口的操作管理，其中两位为串行口的中断标志RI和TI。

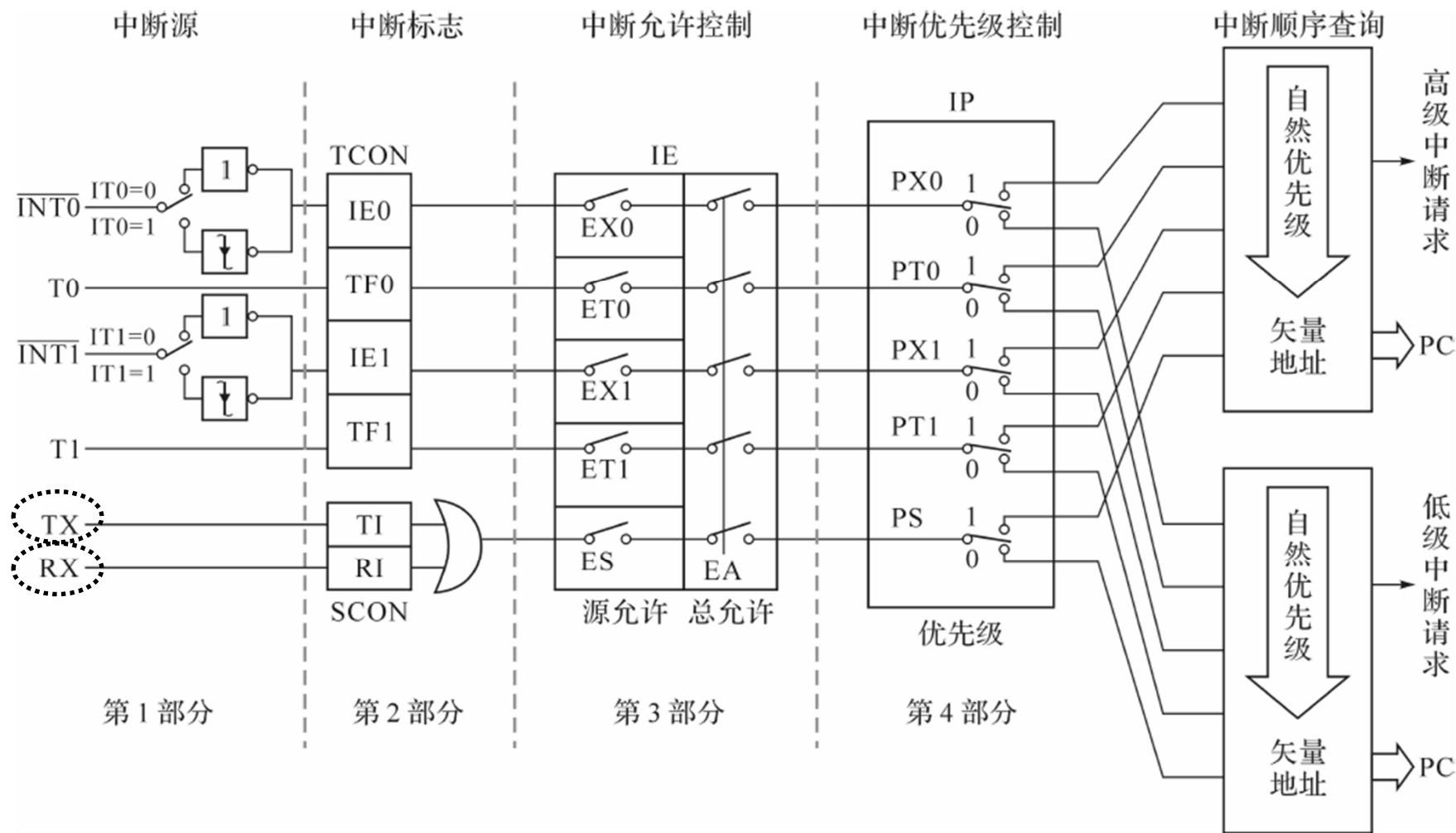
	7	6	5	4	3	2	1	0
位符号	SM0	SM1	SM2	REN	TB8	RB8	TI	RI
英文注释	Serial Mode bit0	Serial Mode bit1	Serial Mode bit2	Receive Enable	Transmit bit 8	Receive bit 8	Transmit Interrupt flag	Receive Interrupt flag

➤**TI**：发送中断标志。当串行口发送完一帧数据时，硬件自动将TI置位。

➤**RI**：接收中断标志。当串行口接收到一帧数据时，硬件自动将RI置位。

51的中断系统

TI 和 RI 在哪儿？





51的中断系统

关于中断标志的清除

- **T0、T1中断标志：**在中断工作模式下，TF0、TF1一直保持到CPU响应中断，并由硬件自动清0；如果采用查询方式，则此标志需要软件清0。
- **外部中断标志：**对于下降沿触发方式，IE0、IE1一直保持到CPU响应中断，并由硬件自动清除。如果是低电平触发方式，只有当中断引脚变为高电平时，才会消除。
- **串行口中断标志：**不论在中断方式还是查询方式，均必须通过软件清除。

Thank you!

