



# 微机原理与接口技术

## §10 定时器/计数器

主讲人：佘青山

Homepage: <https://faculty.hdu.edu.cn/zdhxy/sqs/main.htm>

Email: [qsshe@hdu.edu.cn](mailto:qsshe@hdu.edu.cn)

Mob: 13758167196

Office: 第二教研楼南楼308室

2024年11月28日

**定时器/计数器是微控制器内部最基本的功能模块之一，运用该模块可以方便地实现微机系统测量与控制过程所需要的定时、计数等功能，是微机测控系统的重要组成部分。**

本章分为11个教学单元，主要包括定时器/计数器概述，8051 MCU定时器/计数器的组成结构与控制方法，定时器/计数器的工作方式，初始化，短、中、长定时的实现，以及定时器/计数器的定时、计数、脉冲宽度测量、外部中断扩展以及实时时钟设计等应用。

### 1. 定时器计数器概述

### 2. 定时器计数器工作原理

2.1 定时器计数器的结构与控制

2.2 定时器计数器的工作方式

2.3 定时器计数器的初始化

2.4 计数器的飞读

2.5 定时的实现方法

### 3. 定时/计数器的应用

3.1 定时方式的应用

3.2 计数方式的应用

3.3 脉冲宽度的测量 (自学)

3.4 扩展外部中断 (自学)

3.5 实时时钟的设计

定时器/计数器的**核心**是计数器（**Counter**）。计数器是能够对输入脉冲信号的**跳变沿进行检测**并能进行**加法或减法计数**的电路模块。



独立的定时器/计数器芯片，大多采用减法计数方式，如Z80CTC、8253、8254等。

MCU中的定时器/计数器模块是**可编程的**，通过编程可以使其工作在**定时模式**（用作定时器）或**计数模式**（用作计数器）。

### 1. 定时功能

- **软件定时**：通过执行一段程序（如延时程序）实现定时。其特点是无需额外的硬件，但需要消耗CPU的时间资源。
- **硬件定时**：用定时器/计数器进行定时。若计数器的计数脉冲是已知频率的脉冲信号，则通过对定时器/计数器的编程，可以实现准确的定时，其特点是无需消耗CPU的时间资源。

### 2. 计数功能

利用定时器/计数器对外部脉冲计数和统计。如记录一定时间内的外部脉冲的个数，实现该脉冲频率的测量；又如统计生产流水线上工件的数量等。

在实际的微机测控系统中，通常需要**定时采集输入信号或定时输出控制信号**，也需要对**外部脉冲信号和外界事件进行计数**。

#### 1. 定时器的应用

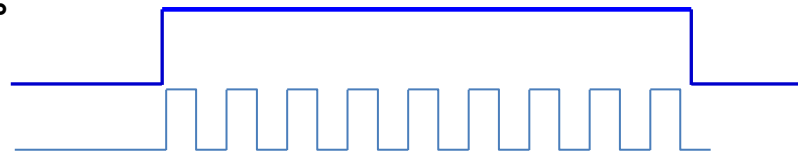
**实现硬件的准确定时；不占用CPU时间，应用广泛。**

- 如10ms进行一次小车轨迹测量，定时输出舵机控制信号，定时进行通信或显示等等。
- 可产生时间基准，如20ms、50ms、1s等，利用1s时基可设计实时时钟。

#### 2. 计数器的应用

**实现对外部脉冲或事件的计数。**

- 利用定时器/计数器，可以实现外部脉冲频率、周期的测量；
- 通过对外部事件的记录，使系统做出处理和控制的决策。如外部事件数达到一定量时，发出相应的控制信号等。



### 1. 定时器计数器概述

### 2. 定时器计数器工作原理

#### 2.1 定时器计数器的结构与控制

#### 2.2 定时器计数器的工作方式

#### 2.3 定时器计数器的初始化

#### 2.4 计数器的飞读

#### 2.5 定时的实现方法

### 3. 定时/计数器的应用

#### 3.1 定时方式的应用

#### 3.2 计数方式的应用

#### 3.3 脉冲宽度的测量

#### 3.4 扩展外部中断

#### 3.5 实时时钟的设计



8051微控制器有2个**可编程**的**16位**定时器/计数器T0和T1，采用的是**加法**计数器。

**所谓可编程是指可以通过程序，对其进行相关内容的设置：**

- 选择**定时**或**计数**模式、以及具体工作方式；（*干嘛？*）
- 预置定时或计数的**初值**；（*计多少？*）
- 设置定时器/计数器溢出时，是否允许**中断**；（*计完如何报告？*）
- 控制T0、T1**启动**或**停止**工作。（*何时开停？*）



## 10.2.1 定时器/计数器的结构

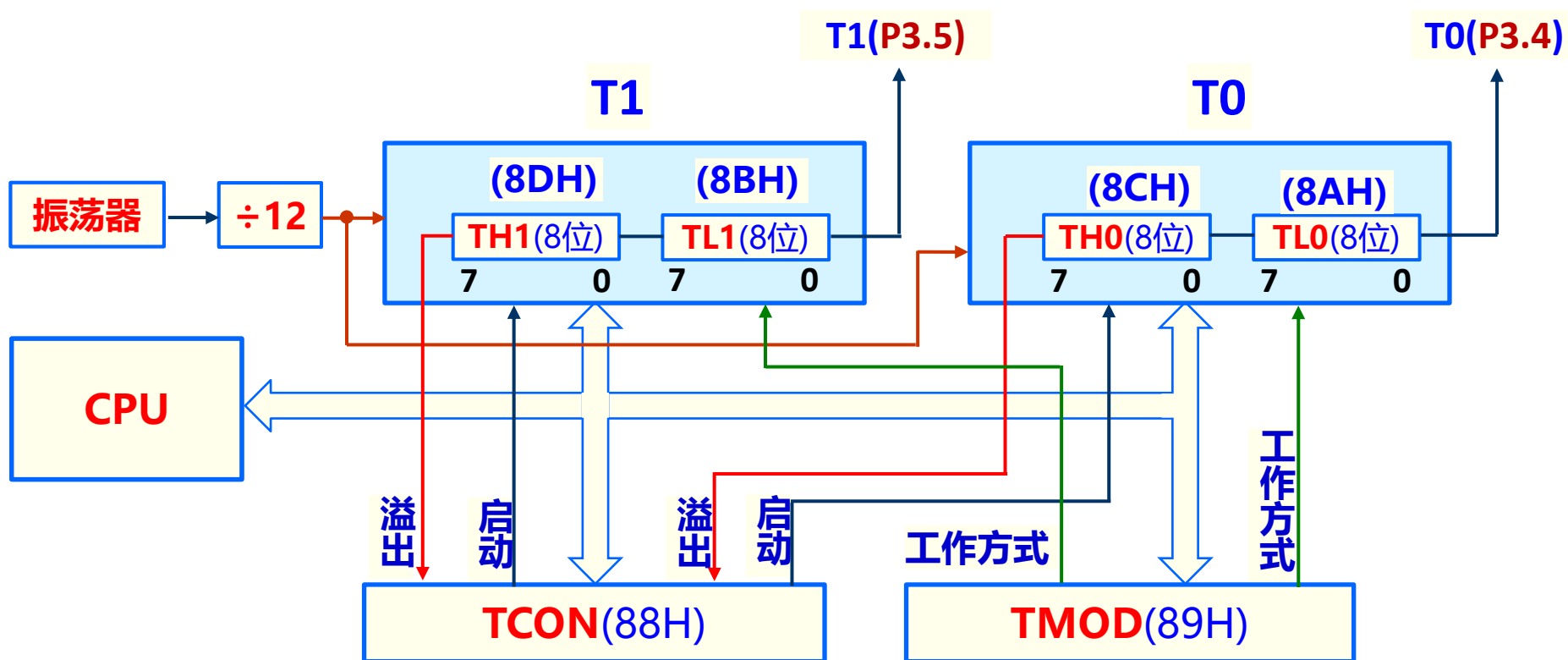
### § 10.2 定时/计数器工作原理

组成结构：6个特殊功能寄存器、内部总线、2个引脚。

2个级联的8位**加法计数器**组成的T0，对应2个8位寄存器**TH0**、**TL0**；

2个级联的8位**加法计数器**组成的T1，对应2个8位寄存器**TH1**、**TL1**；

**方式寄存器TMOD**和**控制寄存器TCON**：对T0、T1进行设置和控制等。



## 10.2.2 定时器/计数器的控制

### § 10.2 定时/计数器工作原理

#### 1. 方式寄存器TMOD: 字节地址为89H, 不可位寻址

位 符 号	7	6	5	4	3	2	1	0
英文 注释	GATE	$C/\bar{T}$	M1	M0	GATE	$C/\bar{T}$	M1	M0
	Gate	Counter/ Timer	Mode bit 1	Mode bit 0	Gate	Counter/ Timer	Mode bit 1	Mode bit 0

定时/计数器T1                      定时/计数器T0

- **$C/\bar{T}$** : 功能选择位。当 $C/\bar{T}=0$ 时, 选择定时模式; 当 $C/\bar{T}=1$ 时, 选择计数模式。
- **GATE**: 门控位。与 $INTi$ 信号相结合, 可以实现外部脉冲高电平宽度的测量。将在“脉冲宽度测量”中进行介绍。
- **M1、M0**: 工作方式 (0~3) 选择位。将结合工作方式予以介绍。

## 2. 控制寄存器TCON：字节地址为88H，可位寻址。

位 符 号	7	6	5	4	3	2	1	0
	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、TF1：T0、T1溢出标志位也称为中断标志位

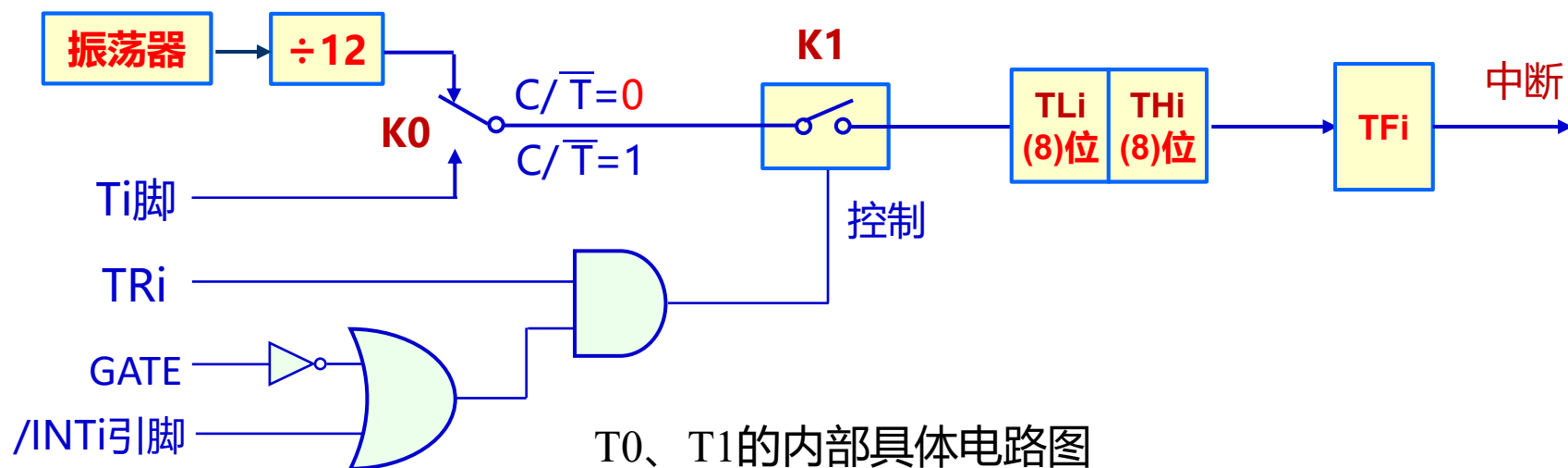
T0、T1溢出时，相应位置“1”。在中断方式时，MCU响应中断后由硬件自动清“0”。在查询方式时，需要由程序清“0”。

## □ TR0、TR1：T0、T1的运行(启停)控制位

- GATE=0时：TR0/TR1=1，启动T0/T1计数；TR0/TR1=0，停止T0/T1计数。
- GATE=1时，由TR1和INT1引脚的电平同时控制T1的启停。

### 1. 定时模式

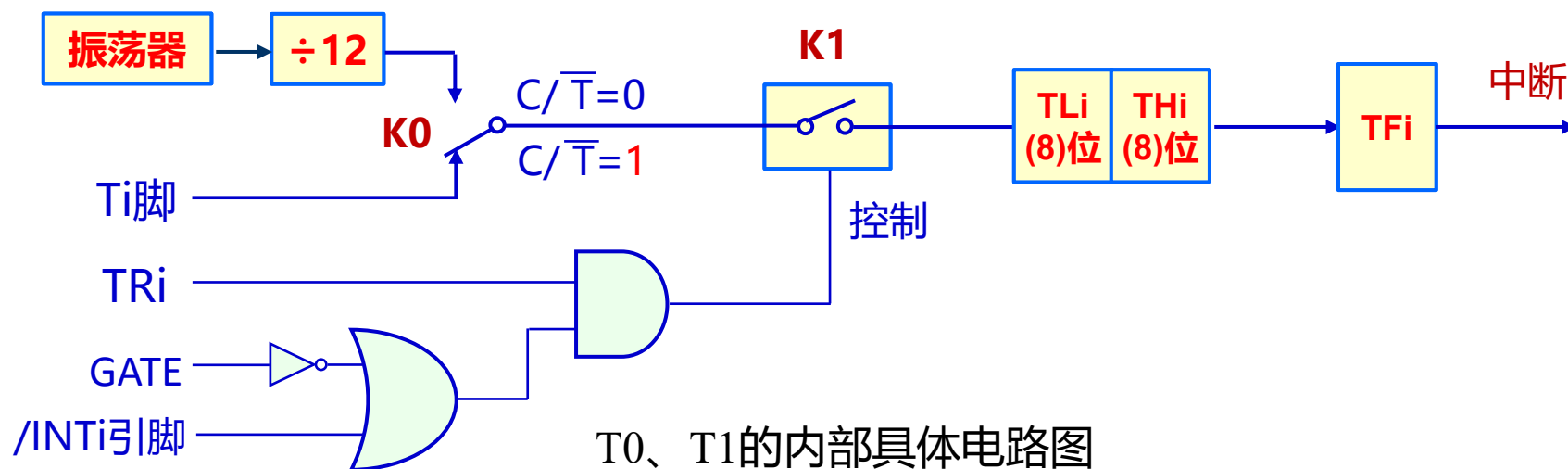
当 $C/\overline{T}=0$ 时，选择定时模式。此时 $T_i$  ( $i=0,1$ )的输入脉冲是内部振荡频率的12分频（即机器周期），即每个机器周期计数器加1。



定时模式时，首先设置 $T_0$ 、 $T_1$ 寄存器的计数初值，再启动定时器工作（使 $K_1$ 闭合），当计数器不断加1到溢出时，定时器中断标志 $TF_i$  ( $i=0$ 或 $1$ )置位，表示定时器溢出（定时时间到）。

### 2. 计数模式

当 $C/\overline{T}=1$ 时，选择计数模式。计数器累计连接到T0、T1引脚上的外部脉冲，每输入一个脉冲计数器加1。



对于计数模式，通常设置T0、T1寄存器的计数初值为0，再启动定时器工作（使K1闭合），计数器累计外部脉冲，可以实现外部脉冲频率的测量。

### 2. 计数模式

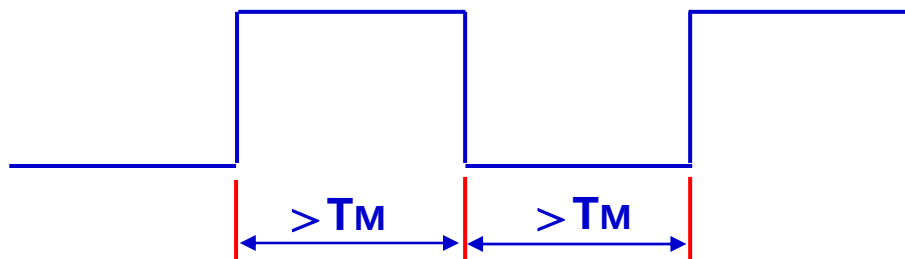
能够测量任何频率的脉冲吗?



#### 对外部脉冲的要求

**外部脉冲的采样过程：**计数模式时，计数器每个机器周期采样一次T0（或T1）引脚状态，当检测到一个**下降沿**时计数器加1。（**下降沿：**前一个机器周期的采样值为高电平，当前机器周期采样值为低电平）

**能够测量的最高频率？** 脉冲的采样需要**高电平**和**低电平**宽度均应**大于一个机器周期 $T_M$** ，因此计数器能够记录的外部脉冲的**最高频率**是 $1/(2T_M) =$ 系统晶振频率的 $1/24$ ，并要求外部脉冲的电平应与微控制器的电平相匹配。



### 1. 定时器计数器概述

### 2. 定时器计数器工作原理

#### 2.1 定时器计数器的结构与控制

#### 2.2 定时器计数器的工作方式

#### 2.3 定时器计数器的初始化

#### 2.4 计数器的飞读

#### 2.5 定时的实现方法

### 3. 定时/计数器的应用

#### 3.1 定时方式的应用

#### 3.2 计数方式的应用

#### 3.3 脉冲宽度的测量 (自学)

#### 3.4 扩展外部中断 (自学)

#### 3.5 实时时钟的设计



## 10.2.3 定时器/计数器的工作方式

### § 10.2 定时/计数器工作原理

**工作方式的选择：**有4种工作方式，用**TMOD**的M1、M0进行选择。

位 符 号	7	6	5	4	3	2	1	0
	GATE	C/ $\bar{T}$	M1	M0	GATE	C/ $\bar{T}$	M1	M0
英文 注释	Gate	Counter/ Timer	Mode bit 1	Mode bit 0	Gate	Counter/ Timer	Mode bit 1	Mode bit 0

定时/计数器T1

定时/计数器T0

□对普通的定时和计数方式，要令  
**GATE=0;**

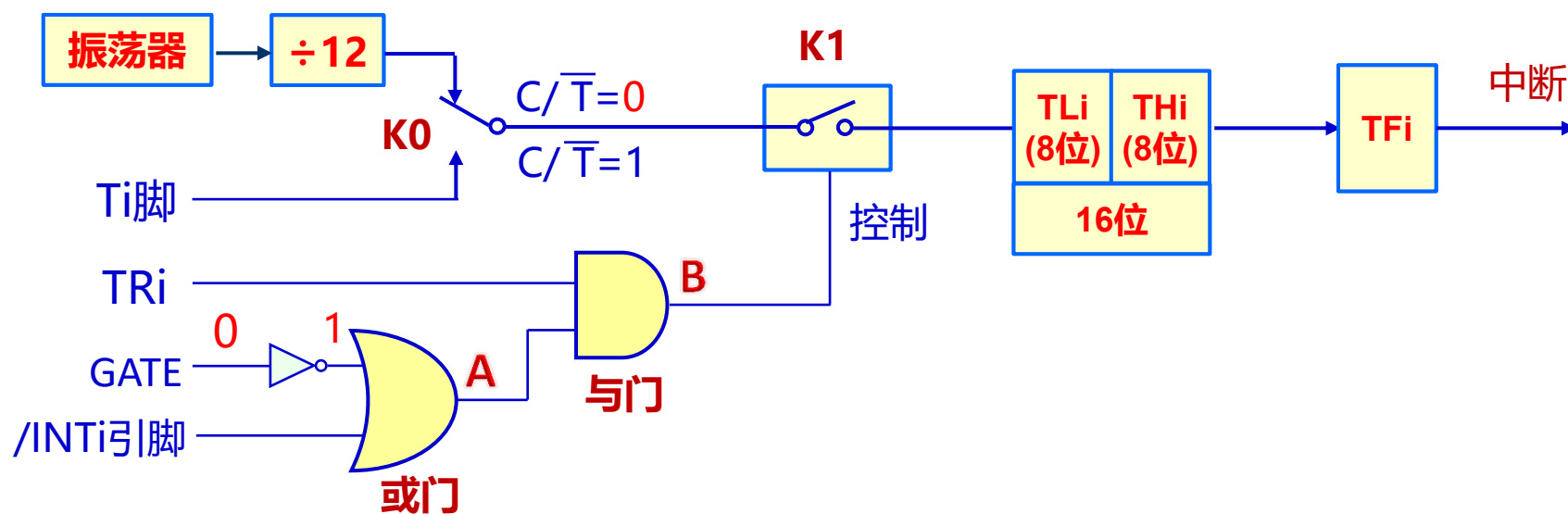
□测量外部脉冲高电平宽度时，要  
令**GATE=1。**

M1	M0	工作方式	功 能
0	0	方式0	13位计数器
0	1	<b>方式1</b>	<b>16位计数器</b>
1	0	<b>方式2</b>	<b>计数初值自动重载8位计数器</b>
1	1	方式3	定时器0分成两个8位计数器， 定时器1停止计数。

由于方式0和方式3很少应用，因此主要介绍**方式1**和**方式2**。

## 1. 方式1

当相应的M1M0设置为0 1时，Ti设定为工作方式1。THi和TLi构成16位的加1计数器，因此称为**16位计数方式**，其最大计数值为 $2^{16}=65536$ 。



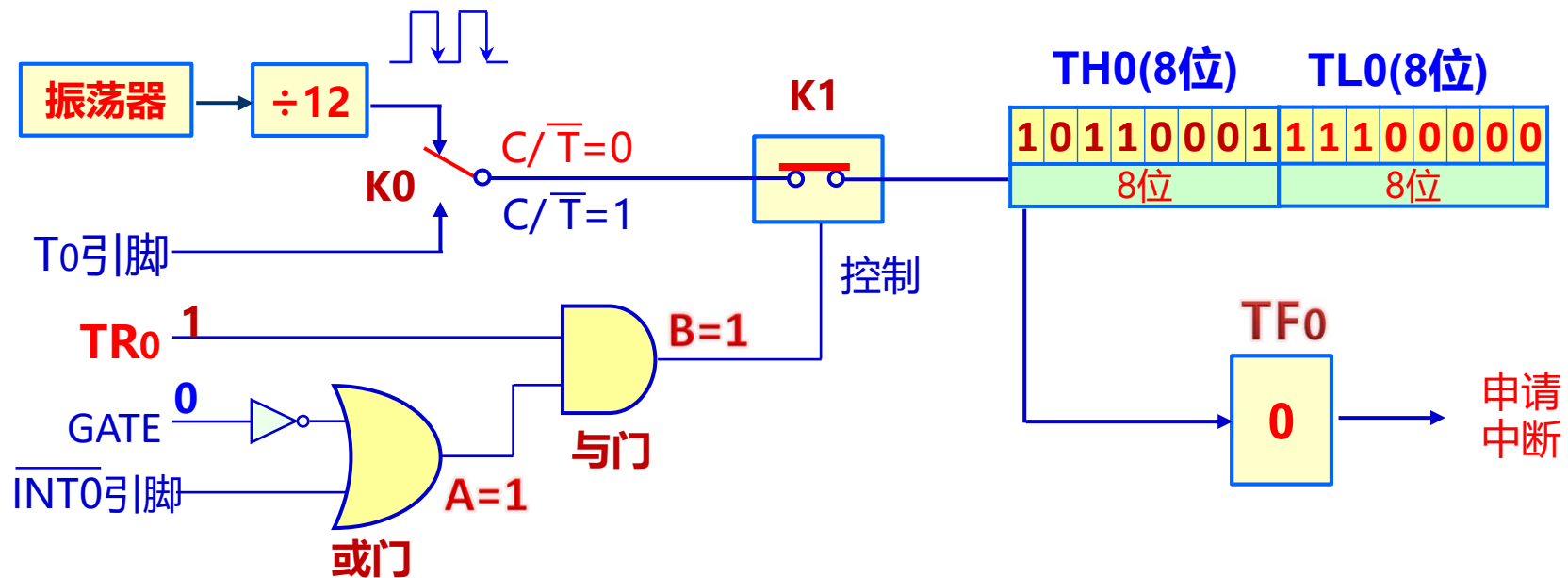
□ 令GATE=0，或门输出A=1，与门输出B=TRi，则**定时器的启停受控于TRi**。

□ **TRi=1**，K1闭合，**开始**定时或计数；**TRi=0**，K1断开，**结束**定时或计数。

## 10.2.3 定时器/计数器的工作方式

### § 10.2 定时/计数器工作原理

#### 举例



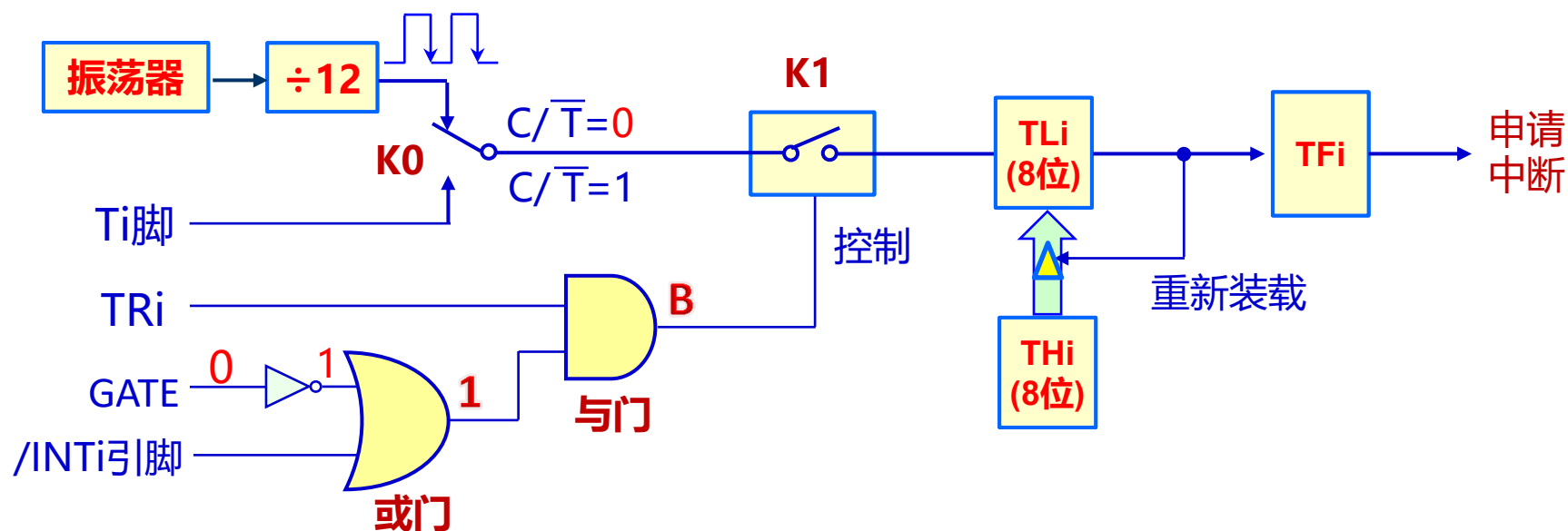
- 设微控制器的振荡频率为**12MHz** (机器周期 $T_M = 1\mu s$ );
- 定时**20ms**的初值 $= 65536 - 20000 = 45536 = 0B1E0H$ , 分别将**0B1H**、**0E0H**, 写入TH0、TL0;
- 当TR0置**1**时, 计数器开始工作。

#### 1. 方式1

- 当16位计数器从设置初值不断加1到**溢出**（变为全0）时，表示定时时间到或计数脉冲达到设定个数，溢出标志位TFi变为“1”。如果Ti**中断允许**，则会向CPU发出中断请求。
- 为使Ti重新从设置初值开始计数，在Ti的中断服务程序或溢出后的处理程序中，**必须首先要给THi、TLi重装载初值**。
- 由于中断响应需要一定时间并存在中断响应时间的随机性，因此**定时时间有一定误差**（误差大小与系统程序的设计、中断的优先级别等有关）。

## 2. 方式2

- 当M1M0设置为1 0时，Ti设定为工作方式2。该方式把TLi配置成一个独立工作的8位加1计数器，THi为重装载初值寄存器，因此称为**8位初值重装载方式**，其最大计数值为 $2^8=256$ 。
- 当TLi产生溢出时，一方面使溢出标志TFi置1，同时把THi中的8位数据重新装入TLi中，**方式2不存在定时误差**。





MCS-51微控制器定时/计数器采用的如下哪一种？

☒ A 加法器结构

☐ B 减法器结构

提交



定时器/计数器T0、T1工作在计数方式时，能够计数任意频率的外部脉冲。

☐ A 对

☒ B 错

提交

MCS-51的定时器与计数器主要区别：前者计数脉冲来自内部晶振，后者计数脉冲来自端口T0或T1。

☒ A 对

☐ B 错

提交

定时器/计数器工作在定时方式时，其加1计数器的计数脉冲周期为：

- ☐ A 振荡频率
- ☐ B 指令周期
- ☒ C 机器周期
- ☐ D 状态周期

提交

### 1. 定时器计数器概述

### 2. 定时器计数器工作原理

#### 2.1 定时器计数器的结构与控制

#### 2.2 定时器计数器的工作方式

#### 2.3 定时器计数器的初始化

#### 2.4 计数器的飞读

#### 2.5 定时的实现方法

### 3. 定时/计数器的应用

#### 3.1 定时方式的应用

#### 3.2 计数方式的应用

#### 3.3 脉冲宽度的测量 (自学)

#### 3.4 扩展外部中断 (自学)

#### 3.5 实时时钟的设计

## 10.2.4 定时器/计数器的初始化

### § 10.2 定时/计数器工作原理

#### 1. 初始化步骤

位 符 号	7	6	5	4	3	2	1	0
英文 注释	GATE	C/ $\bar{T}$	M1	M0	GATE	C/ $\bar{T}$	M1	M0
	Gate	Counter/ Timer	Mode bit 1	Mode bit 0	Gate	Counter/ Timer	Mode bit 1	Mode bit 0

定时/计数器T1                      定时/计数器T0

- 1) **确定工作方式**，即给方式寄存器TMOD赋值。
- 2) **预置定时初值或计数初值**，将初值写入TH0、TL0或TH1、TL1中。
- 3) **中断设置（给IE赋值）**，允许或禁止定时器/计数器的中断。
- 4) **启动定时器/计数器**，令TCON中的TR0或TR1为“1”。

#### 2. 定时/计数初值的确定

对于加1计数器，设置的计数初值应是需要定时（计数）值相对于定时器/计数器**最大计数值M的补码**。

最大计数值M取决于计数器的位数L， **$M=2^L$** 。

方式1时L=16，最大计数值 $M=2^{16}=65536$ ；

方式2时L=8，最大计数值 $M=2^8=256$ 。

#### 2. 定时/计数初值的确定

设晶振频率为12MHz，则机器周期为1 $\mu$ s。

##### (1) 对于方式1: $M=65536$

若定时初值  $X=0$ ，则定时时间为 65536 $\mu$ s。若需要定时  $t$  为 20ms=20000 $\mu$ s，则定时计数初值  $X$  应为： $\text{最长定时时间} - \text{需要定时时间} = 65536 - 20000 = 45536$ 。

加1计数器在该初值基础上，计数20000个机器周期脉冲，就发生溢出，表示20ms定时时间到。

##### (2) 对于方式2: $M=256$

若定时初值  $X=0$ ，则定时时间为256 $\mu$ s。若需要定时  $t$  为200 $\mu$ s，则定时初值  $X$  应为： $\text{最长定时时间} - \text{需要定时时间} = 256 - 200 = 56$ 。

加1计数器在该初值基础上，计数200个机器周期脉冲，就发生溢出，表示200 $\mu$ s定时时间到。

#### 2. 定时/计数初值的确定

设定时初值为 $X$ ，则定时时间 $t = (M - X) \times 1\mu s$ ；

对于晶振频率为 $f$ 的MCU系统，定时时间 $t$ 为：

$$t = (M - X) \times \frac{12}{f} = (M - X) \times \text{机器周期} (\mu s)$$

如果要求定时的时间为 $t$ ，则根据上式可以得到定时初值 $X$ ：

$$X = M - \frac{tf}{12}$$

或

$$X = M - t / T_M \quad (T_M: \text{机器周期})$$



**例6-1：**某8051微控制器采用的晶振频率为6MHz，请用T1产生1ms的定时，试计算定时初值。

**解：**因为 $f_{osc}=6\text{MHz}$ ，则机器周期为 $2\mu\text{s}$ 。

计数器需要累计的脉冲个数为：

$$1000\mu\text{s}/2\mu\text{s}=500$$

由于方式2的最大计数值是256，因此无法实现。

采用方式1，定时初值 $X=65536-500=65036=0\text{FE}0\text{CH}$ ，

即向TH1、TL1设置 $0\text{FEH}$ 、 $0\text{CH}$ 。

**例6-2：** 设某8051微控制器系统的晶振频率为12MHz，若要T1定时50ms，试确定定时初值。(自学)

**解：**

已知 $f=12\text{MHz}$ ，则机器周期为1 $\mu\text{s}$ 。

计数器需要累计的脉冲个数为50000个；

选用工作方式1，定时初值：

$$X = 65536 - 50000 = 15536 = 3CB0H,$$

即向TH1、TL1设置3CH、B0H。

**例6-3：**设晶振频率为12MHz，若要T1产生250μs的定时信号，试选择工作方式、确定定时初值。

**解：**计数脉冲即机器周期是1μs；

则计数器需要累计的脉冲个数为250个。

选择工作方式2，定时初值：

$$X = 256 - 250 = 6$$

即向TL1、TH1设置初值为6。

**对于计数方式，通常将计数初值设置为0，则经过一定时间（如1s）后，TH0、TL0或TH1、TL1的内容即为这段时间内记录的脉冲数。**

不论工作在何种方式下，8051微控制器定时器/计数器的计数初值仅在初始化时设置即可。

☐ A 对

☒ B 错

提交

8051微控制器T0用作计数器，采用工作方式2，则其最大的计数值为256。

☒ A 对

☐ B 错

提交

### 1. 定时器计数器概述

### 2. 定时器计数器工作原理

#### 2.1 定时器计数器的结构与控制

#### 2.2 定时器计数器的工作方式

#### 2.3 定时器计数器的初始化

#### 2.4 计数器的飞读

#### 2.5 定时的实现方法

### 3. 定时/计数器的应用

#### 3.1 定时方式的应用

#### 3.2 计数方式的应用

#### 3.3 脉冲宽度的测量 (自学)

#### 3.4 扩展外部中断 (自学)

#### 3.5 实时时钟的设计

要获得16位加1计数器的值，需要分别对高8位和低8位两个寄存器（THi和TLi）进行读取。若读高8位寄存器时，恰逢低8位寄存器溢出，则读取的低8位寄存器的值就会出现**计数误差（或称为“相位误差”）**。

**例：**当**16位计数器**中的值为**00FFH**时

- 若先读高8位**THi**，得到00H，再读低8位**TLi**时，TLi已发生进位溢出，即此时16位计数器的值变成了0100H，所以读到TLi的值为00H，这样读取的**16位计数结果变为0000H**。
- 若先读低8位**TLi**，得到FFH，再读高8位**THi**时，TLi已发生进位溢出，计数器的值变成了0100H，所以读到THi的值为01H，因此得到的**16位计数结果变为01FFH**。

这两种结果都是不正确的



**解决方案：采用飞读方式。**

先读取THi值，再读TLi值，然后再重新读取一遍THi。若两次THi值相同，表示读取的结果正确；若不相同，则再重复上述过程。

**“飞读”子程序：**

**汇编程序（计数值存入R0、R1）：**

```
RDT0:  MOV  A, TH0          ;读入TH0到A
        MOV  R1, TL0        ;读入TL0到R1
        CJNE A, TH0, RDT0    ;将第一次读取的TH0与当前的TH0比较，不相等重读
        MOV  R0, A          ;若相同，则保存TH0到R0
        RET
```

### 1. 定时器计数器概述

### 2. 定时器计数器工作原理

#### 2.1 定时器计数器的结构与控制

#### 2.2 定时器计数器的工作方式

#### 2.3 定时器计数器的初始化

#### 2.4 计数器的飞读

#### 2.5 定时的实现方法

### 3. 定时/计数器的应用

#### 3.1 定时方式的应用

#### 3.2 计数方式的应用

#### 3.3 脉冲宽度的测量 (自学)

#### 3.4 扩展外部中断 (自学)

#### 3.5 实时时钟的设计

#### (1) 软件定时

软件定时可以实现8051MCU的最短定时时间，以及长时间的定时。

- 对于只有几个机器周期的定时，通常采用若干个NOP指令来实现；
- 对于较长的定时，可以采用延时子程序实现。

#### (2) 定时器定时

- 定时器**方式2**（8位）能够自动重装载定时初值，适合用于产生256个机器周期内的定时；
- 定时器**方式1**（16位）适合用于定时时间大于256小于65536个机器周期的定时。
- **更长时间的定时**，将定时器的硬件定时和软件计数结合起来。

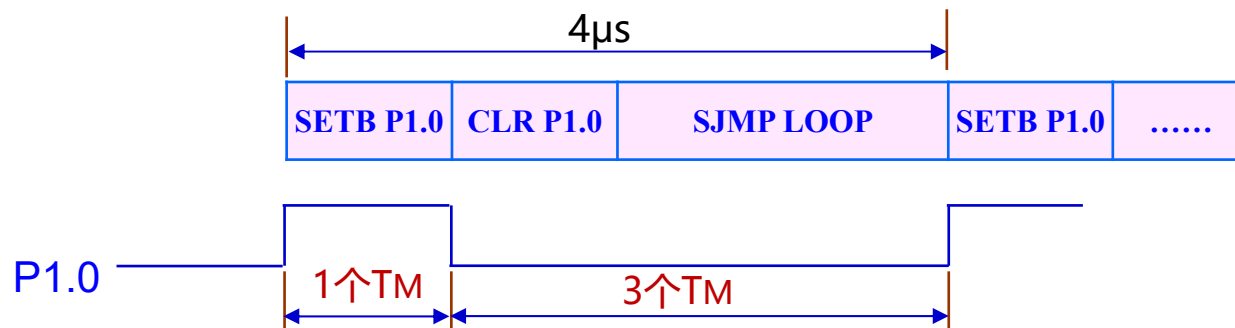
#### 产生不同定时时间的方法

定时时间 ( $\mu\text{s}$ )	方法
几个 $\mu\text{s}$ -较长定时	软件编写
256 ( <b>max</b> )	8位定时方式
65536 ( <b>max</b> )	16位定时方式
较长定时	16位定时方式及软件计数

表中假设系统晶振为12MHz, 则机器周期为1 $\mu\text{s}$

- ❑ 软件延时需要CPU运行程序, 因此要占用CPU的时间资源;
- ❑ 定时器不需要占用CPU时间, 只有在定时器/计数器溢出时, 才向CPU请求中断。定时准确、灵活性强, 能有效提高微控制器的性能。

**例6-4：**编写一个脉冲波形产生程序，在引脚P1.0产生最高频率的周期脉冲波。能产生的最高频率是多少？该周期脉冲波形的占空比是多少？



**汇编程序：**

```
ORG 0000H  
LOOP: SETB P1.0 ;1个TM  
      CLR P1.0 ;1个TM  
      SJMP LOOP ;2个TM  
      END
```

**分析:**

在每个循环中，高电平时间为 $1\mu\text{s}$ ，低电平时间为 $3\mu\text{s}$ ，周期是 $4\mu\text{s}$ ，占空比为？

- ☒ A  $1/4$
- ☐ B  $1/3$
- ☐ C  $1/2$
- ☐ D  $1$

提交

**分析:**

P1.0引脚上输出的脉冲频率为?

- ☒ A 250KHz
- ☐ B 250Hz
- ☐ C 500KHz
- ☐ D 500Hz

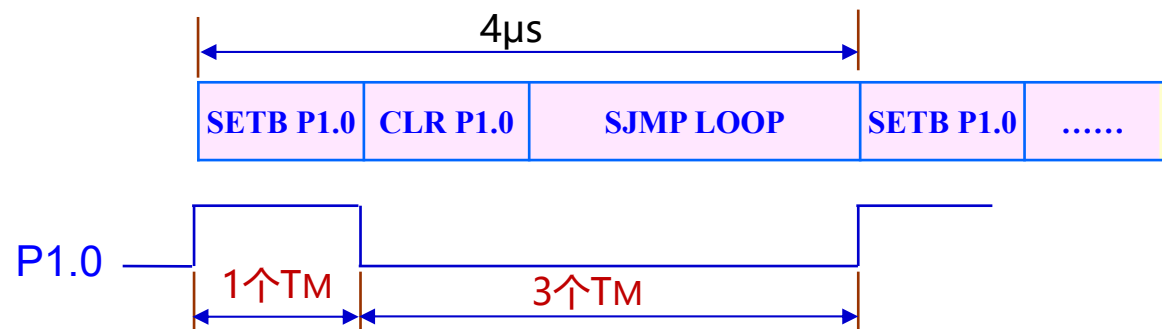
提交

**例6-4:** 编写一个脉冲波形产生程序，在引脚P1.0产生最高频率的周期脉冲波。能产生的最高频率是多少？该周期脉冲波形的占空比是多少？（自习）

**汇编程序:**

```

ORG 0000H
LOOP: SETB P1.0 ;1个 $T_M$ 
      CLR P1.0  ;1个 $T_M$ 
      SJMP LOOP ;2个 $T_M$ 
END
    
```



**分析:**

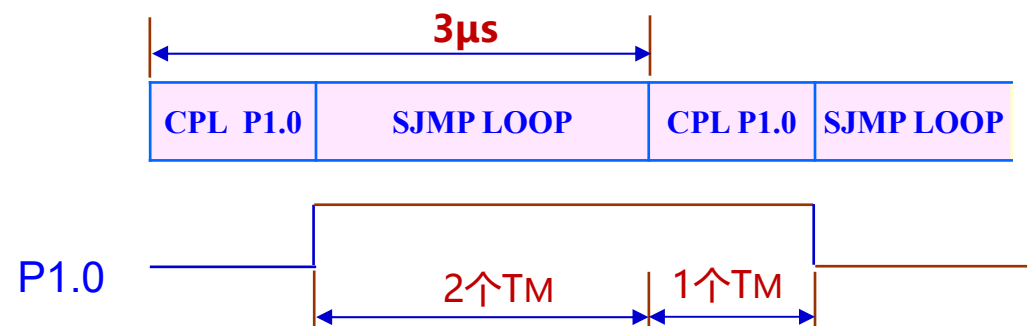
- 在每个循环中，高电平时间为1 $\mu s$ ，低电平时间为3 $\mu s$ ，周期是4 $\mu s$ ，占空比为1/4=25%。因此P1.0引脚上输出的脉冲频率为250KHz(1/4 $\mu s$ )。
- 在指令SETB P1.0之后增加2条NOP指令可以使输出波形变为方波（占空比=50%），周期变为6 $\mu s$ ，频率变为169.7KHz（1/6 $\mu s$ ）。



**例6-5：**下段程序可在引脚P1.0产生最高频率的方波信号，请分析其最高频率和占空比是多少？（自习）

**汇编程序：**

```
ORG 0000H  
LOOP: CPL P1.0      ;1个 $T_M$   
      SJMP LOOP     ;2个 $T_M$   
END
```



**分析：**

高低电平的时间分别为 $3\mu s$ ，周期为 $6\mu s$ ，频率 $169.67\text{KHz}$  ( $1/6\mu s$ )，占空比为50%。

### 1. 定时器计数器概述

### 2. 定时器计数器工作原理

#### 2.1 定时器计数器的结构与控制

#### 2.2 定时器计数器的工作方式

#### 2.3 定时器计数器的初始化

#### 2.4 计数器的飞读

#### 2.5 定时的实现方法

### 3. 定时/计数器的应用

#### 3.1 定时方式的应用

#### 3.2 计数方式的应用

#### 3.3 脉冲宽度的测量 (自学)

#### 3.4 扩展外部中断 (自学)

#### 3.5 实时时钟的设计

## 10.3.1 定时方式的应用

## § 10.3 定时器/计数器应用

**例6-6：** 设8051微控制器采用的晶振频率为12MHz，利用定时器T1定时，使P1.0输出周期为2ms的方波。

**分析：** 要产生周期为2ms的方波需要每1ms改变一次P1.0的电平，故定时时间为1ms，应采用工作方式1。TMOD的方式控制字应为10H；计数脉冲周期为1μs，所以**定时初值**为  $X = 2^{16} - 1000 = 64536 = 0FC18H$

位 符 号	7	6	5	4	3	2	1	0
英文 注释	GATE	C/̄T	M1	M0	GATE	C/̄T	M1	M0
	Gate	Counter/ Timer	Mode bit 1	Mode bit 0	Gate	Counter/ Timer	Mode bit 1	Mode bit 0

### 1、汇编程序(查询方式):

```
MOV    TMOD,#10H    ;设置T1为方式1
SETB   TR1          ;启动T1定时
LOOP:  MOV    TH1,#0FCH
        MOV    TL1#18H    ;装入定时初值
        JNB    TF1,$      ;等待溢出，若TF1=0，则继续查询等待
        CPL    P1.0        ;若TF1=1，P1.0状态翻转，输出方波
        CLR    TF1        ;查询方式，TF1需要软件清0
        SJMP   LOOP      ;重复循环
```

## 10.3.1 定时方式的应用

### § 10.3 定时器/计数器应用

位 符 号	7	6	5	4	3	2	1	0
英文 注释	GATE	C/ $\bar{T}$	M1	M0	GATE	C/ $\bar{T}$	M1	M0
	Gate	Counter/ Timer	Mode bit 1	Mode bit 0	Gate	Counter/ Timer	Mode bit 1	Mode bit 0

定时/计数器T1

定时/计数器T0

### 2、汇编程序(中断方式):

#### (1) 主程序

```
ORG    0000H
LJMP    MAIN           ;跳转到主程序

ORG    001BH           ;T1中断入口
LJMP    T1SUB          ;转T1中断服务程序

ORG    0100H
MAIN:   MOV    TMOD, #10H    ;设置T1方式1
        MOV    TH1, #0FCH
        MOV    TL1, #18H    ;设置定时初值
        SETB   EA           ;CPU中断允许
        SETB   ET1          ;T1中断允许
        SETB   TR1          ;启动T1
        SJMP   $            ;模拟主程序
```

#### (2) 中断服务程序

```
T1SUB:  ORG    0800H
        MOV    TH1, #0FCH
        MOV    TL1, #18H    ;重载定时初值
        CPL    P1.0         ;P1.0状态翻转, 输出方波
        RETI                ;中断返回
```

### 总结：

- 采用查询方式的程序简单，但效率低。需要CPU不断查询溢出标志，没有提高CPU的效率；
- 中断方式的程序编写相对复杂，效率高（并行处理）。CPU对定时器/计数器初始化后，就由硬件进行定时，CPU只在定时时间到时，响应中断执行中断服务程序即可；
- 在实际应用程序中，通常采用中断方式设计定时程序。

### 1. 定时器计数器概述

### 2. 定时器计数器工作原理

#### 2.1 定时器计数器的结构与控制

#### 2.2 定时器计数器的工作方式

#### 2.3 定时器计数器的初始化

#### 2.4 计数器的飞读

#### 2.5 定时的实现方法

### 3. 定时/计数器的应用

#### 3.1 定时方式的应用

#### 3.2 计数方式的应用

#### 3.3 脉冲宽度的测量 (自学)

#### 3.4 扩展外部中断 (自学)

#### 3.5 实时时钟的设计

**例6-7：**设计一个微机系统，用于记录生产流水线上每天生产的工件箱数。每箱装100个工件，因此每次计数到100个工件时，该系统要向包装机发出打包命令（输出一个高脉冲信号），使包装机执行打包动作，并推出装满工件的箱子引入空箱子。

**分析：**首先通过电路设计将经过流水线上的工件转换为脉冲信号，一个工件输出一个脉冲，用定时器/计数器记录工件脉冲的个数。

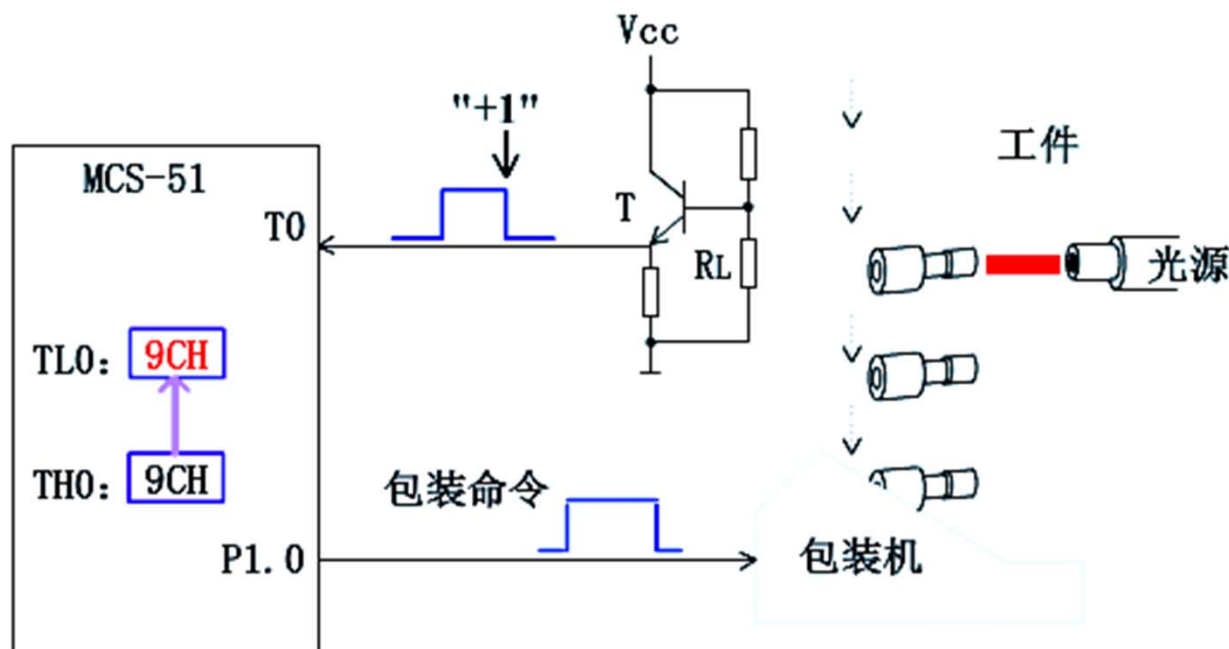
- 由于每100个工件为一箱，所以选择工作方式2，设置初值为 $256 - 100 = 156$  (9CH)，则累计100个工件脉冲后，溢出请求中断；
- 在中断程序中发出打包信号，并完成箱数的累计等。

## 10.3.2 计数方式的应用

### § 10.3 定时器/计数器应用

**外围电路设计：** 选用LED光源和光敏电阻RL作为流水线上工件的检测模块。

当有工件通过时，LED发出的光线受阻挡无法到达光敏电阻RL，其阻值很大，使三极管T导通输出高电平；当没有工件时，光敏电阻接收到LED光使RL变小，此时T截止而输出低电平。因此，每通过一个工件，T0端就会接收到一个正脉冲信号，由T0进行计数。





## 10.3.2 计数方式的应用

### § 10.3 定时器/计数器应用

**程序设计：**计数模式，工作方式2，TMOD方式控制字为06H。用P1.0输出打包机需要的包装命令（高脉冲）；用R5和R4作为每天生产工件箱数的计数器。

**汇编程序（主程序）：**

位 符 号	7	6	5	4	3	2	1	0
英文 注释	GATE	C/T	M1	M0	GATE	C/T	M1	M0
	Gate	Counter/ Timer	Mode bit 1	Mode bit 0	Gate	Counter/ Timer	Mode bit 1	Mode bit 0

```
ORG 0000H
SJMP MAIN

ORG 000BH                ;T0中断入口
LJMP COUNT              ;转向中断服务

MAIN: ORG 0040H
      CLR P1.0           ;P1.0置低电平
      MOV R5, #0
      MOV R4, #0          ;箱计数器清0
      MOV R2, #0          ;工件数清0
      MOV TMOD, #0000110B ; T0工作方式2设置
      MOV TH0, #9CH       ;设置重新装载系数
      MOV TL0, #9CH       ;设置计数初值
      SETB EA
      SETB ET0
      SETB TR0
      SJMP $
```

## 10.3.2 计数方式的应用

### § 10.3 定时器/计数器应用

汇编程序（中断程序）：

```
COUNT:      ORG      0800H           ;中断服务程序
             MOV      A, R4
             ADD      A, #1
             MOV      R4, A
             MOV      A, R5
             ADDC     A, #0
             MOV      R5, A           ;箱计数器加1
             SETB     P1.0           ;输出包装机打包信号
             MOV      R3, #10
DLY:         NOP
             DJNZ     R3, DLY         ;使得高脉冲有一定宽度
             CLR      P1.0           ;停止包装
             RETI
```

### 1. 定时器计数器概述

### 2. 定时器计数器工作原理

#### 2.1 定时器计数器的结构与控制

#### 2.2 定时器计数器的工作方式

#### 2.3 定时器计数器的初始化

#### 2.4 计数器的飞读

#### 2.5 定时的实现方法

### 3. 定时/计数器的应用

#### 3.1 定时方式的应用

#### 3.2 计数方式的应用

#### 3.3 脉冲宽度的测量 (自学)

#### 3.4 扩展外部中断 (自学)

#### 3.5 实时时钟的设计

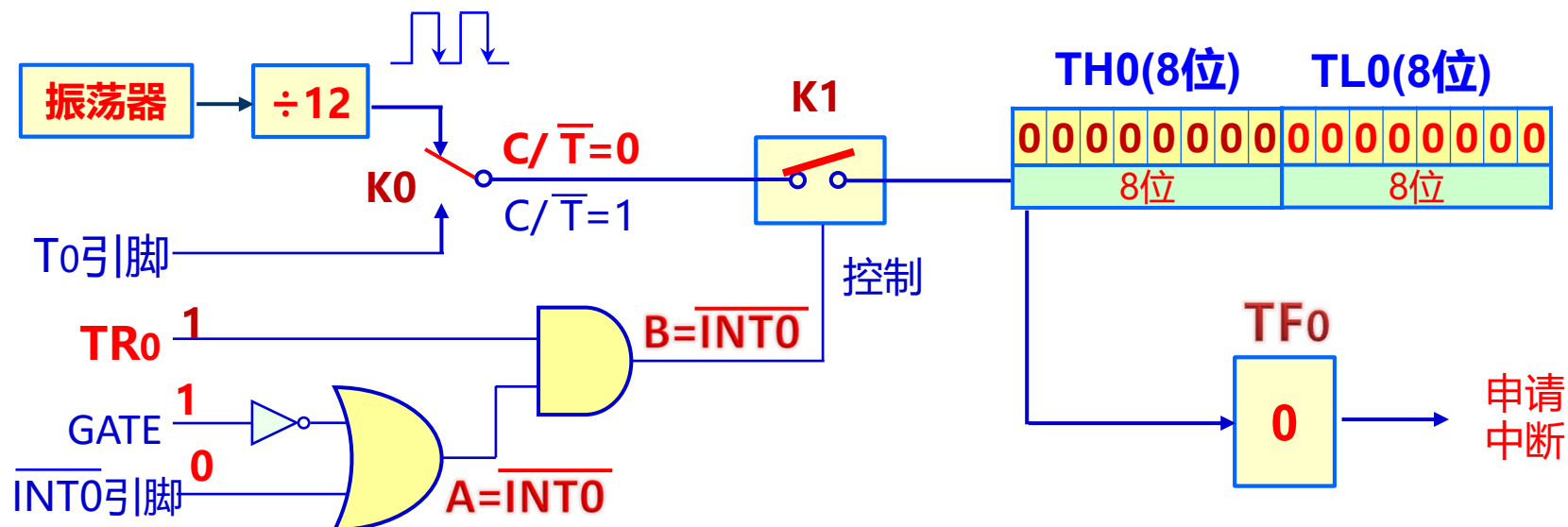
在实际应用中，通常需要测量脉冲的频率、周期或高、低电平的宽度。利用经典8051 MCU中的定时器/计数器结合外部中断引脚，可以实现脉冲高电平宽度的测量。（教材第10章介绍脉冲频率和周期的测量方法）。

脉冲高电平的测量，需要用到方式寄存器TMOD中的GATE信号，并要将被测脉冲信号连接到INT0或INT1引脚。

### 10.3.3 脉冲宽度的测量(自学)

### § 10.3 定时器/计数器应用

#### 定时器测量外部脉冲宽度



- 1、 $\text{GATE}=1$ ，则“ $A$ ” =  $\overline{\text{INT0}}$ ；当 $\text{TR0}=1$ ，“控制” =  $\overline{\text{INT0}}$
- 2、 $\overline{\text{INT0}}$ 引脚为“1”， $K1$ 闭合（开始计数）， $\overline{\text{INT0}}$ 引脚为“0”时， $K1$ 断开（停止计数）；则T0记录的机器周期数，即为INT0引脚的高电平宽度的时间。

#### 测量方法总结:

- 设置为定时模式 ( $C/\overline{T}=0$ ) , 工作方式1, 定时初值为0000H;
- 外部脉冲连接至INTi引脚;
- 设置GATE=1, 在INTi为低电平时, 令TRi=1; 这时, 定时器/计数器的启停受INTi引脚电平控制; 高电平开始计数, 低电平停止计数;
- 停止计数时, 读取Ti16位寄存器的内容, 即为INTi引脚高电平期间累计的机器周期数, 即高电平的宽度。

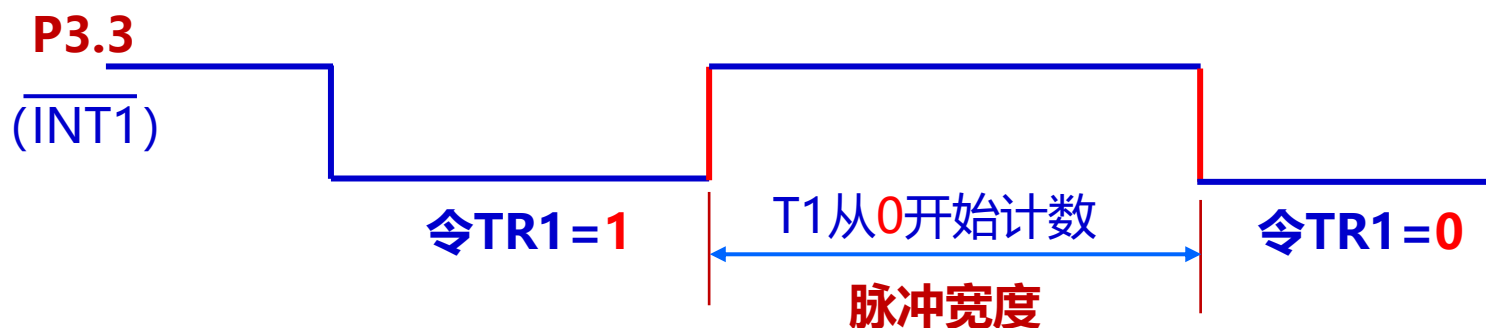
### 10.3.3 脉冲宽度的测量(自学)

#### § 10.3 定时器/计数器应用

**例6-8:** 利用T1, 测量INT1引脚上的高脉冲宽度。设晶振为12MHz, 高脉冲宽度小于65ms。

**分析:** 选择T1定时模式, 工作方式1, 定时初值设置为0。

首先在INT1低电平期间设置TR1=1 (为INT1控制T1的启停做准备), 当INT1引脚从0变1时, 就启动了T1工作(计数器从0开始累计), 当INT1引脚变低时停止计数。此时T1寄存器中的计数值即为正脉冲的宽度(机器周期数)。所能测量到的最大脉冲宽度为65535 $\mu$ s。



### 10.3.3 脉冲宽度的测量(自学)

### § 10.3 定时器/计数器应用

#### 汇编程序:

位 符 号	7	6	5	4	3	2	1	0
英文 注释	GATE	C/ $\bar{T}$	M1	M0	GATE	C/ $\bar{T}$	M1	M0
	Gate	Counter/ Timer	Mode bit 1	Mode bit 0	Gate	Counter/ Timer	Mode bit 1	Mode bit 0

```
MOV    TMOD,#90H ;设置方式1、GATE=1          定时/计数器T1      定时/计数器T0
CLR     TR1
MOV     TL1,#0
MOV     TH1,#0
JB      P3.3,$      ;判断INT1是否为低电平 (需要在低电平时, 令TR1=1)
SETB    TR1         ;在INT1=0时, 置TR1=1, 使计数器的启停受控于INT1
JNB     P3.3,$      ;等待INT1变高电平, 使计数器开始计, 开始脉宽测量
JB      P3.3,$      ;等INT1变为低电平, 使计数器停止计数, 结束脉宽测量
CLR     TR1         ;置TR1=0, 为下次测量做准备
MOV     A, TL1      ;读取16位计数器累计的机器周期数, 高低8位分别送B、A寄存器
MOV     B, TH1
SJMP    $           ;一次测量结束
```



### 1. 定时器计数器概述

### 2. 定时器计数器工作原理

#### 2.1 定时器计数器的结构与控制

#### 2.2 定时器计数器的工作方式

#### 2.3 定时器计数器的初始化

#### 2.4 计数器的飞读

#### 2.5 定时的实现方法

### 3. 定时/计数器的应用

#### 3.1 定时方式的应用

#### 3.2 计数方式的应用

#### 3.3 脉冲宽度的测量 (自学)

#### 3.4 扩展外部中断 (自学)

#### 3.5 实时时钟的设计

## 10.3.4 扩展外部中断(自学)

### 外部中断外展方法:

- 把T0、T1设置为计数方式，选用工作方式2，**设置初值为FFH**，允许中断。
- 外部中断源连接到T0、T1引脚，则在其引脚上发生**负跳变**时（产生一个**下降沿**），**定时器/计数器加1，就会产生溢出而向CPU请求中断。**
- 利用这个特性可以把**T0**（P3.4）和**T1**（P3.5）两个引脚扩展为外部中断请求引脚，中断的**触发条件**是**下降沿触发**，中断标志为T0、T1的溢出标志TF0和TF1。
- 其作用和功能与外部中断INT0、INT1完全相同。这样就把经典8051 MCU的外部中断源扩展到了**4个**。

## 10.3.4 扩展外部中断(自学)

## § 10.3 定时器/计数器应用

**例6-9：**利用定时器/计数器T0扩展外部中断。

**汇编程序：**

```
ORG 0000H
LJMP MAIN                ;转主程序
ORG 000BH                ;T0中断入口
LJMP T0INT               ;转T0中断服务

ORG 0100H
MAIN: MOV TMOD, #06H      ;计数模式, 工作方式2
      MOV TL0, #0FFH      ;设置初值
      MOV TH0, #0FFH      ;设置重装初值
      SETB EA             ;CPU中断开放
      SETB ET0            ;T0中断开放
      SET TR0             ;启动T0
      SJMP $              ;模拟主程序

ORG 2000H                ;中断程序
T0INT: ...
      RETI
```

位 符 号	7	6	5	4	3	2	1	0
英文 注释	GATE	C/ $\bar{T}$	M1	M0	GATE	C/ $\bar{T}$	M1	M0
	Gate	Counter/ Timer	Mode bit 1	Mode bit 0	Gate	Counter/ Timer	Mode bit 1	Mode bit 0

定时/计数器T1      定时/计数器T0

### 1. 定时器计数器概述

### 2. 定时器计数器工作原理

#### 2.1 定时器计数器的结构与控制

#### 2.2 定时器计数器的工作方式

#### 2.3 定时器计数器的初始化

#### 2.4 计数器的飞读

#### 2.5 定时的实现方法

### 3. 定时/计数器的应用

#### 3.1 定时方式的应用

#### 3.2 计数方式的应用

#### 3.3 脉冲宽度的测量

#### 3.4 扩展外部中断

#### 3.5 实时时钟的设计

#### 1. 如何获得1s定时

对于**16位的工作方式1**，其最长定时时间是65536个机器周期，**无法直接获得1s的定时**，而需要定时器的硬件定时和软件计数相结合来实现。

可以利用定时器产生**50ms定时**，设置定时模式、工作方式1，用一个软件计数器（如一个工作寄存器或存储单元）用于累计50ms的个数，该软件计数器初值设为0，当其从0累计20时，表示已有20个50ms，即1s定时时间到。

定时器的**50ms定时**，可以采用查询方式，也可以用中断方式，通常用中断方式。

## 2. 实时时钟的设计思路

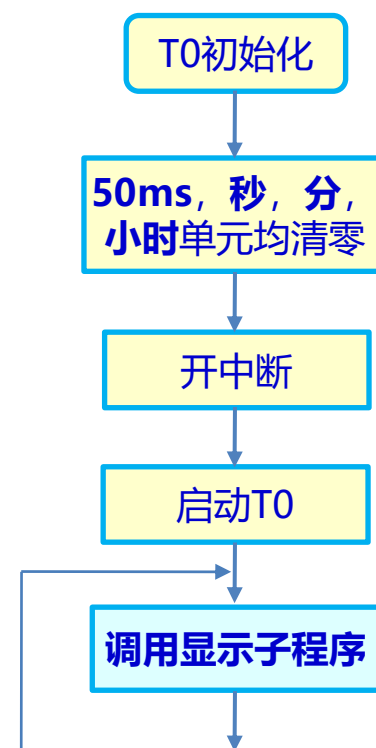
- 在内部RAM中设置4个单元如43H-40H分别作为软件计数器（存放50ms个数、秒、分、时）的存储单元。
- 在50ms中断服务程序中，50ms个数加1，累加到20时表示1s到，再把50ms个数清0，秒单元内容加1，满60秒时，秒单元清0，分钟单元加1，满60分钟时，分钟单元清0，时单元加1，满24小时时，小时单元清0。
- 重复上述定时和运算过程，即可实现不断运行的实时时钟。
- 如果在时分秒的存储单元中，预先存入当前实际的时分秒，则运行实时时钟程序，就可以设计出一个电子钟；软件计数器中的时分秒，也可以通过按键输入，即可实现用按键设置或修改时间。

43H	50mS
42H	S
41H	M
40H	H

**例6-10：**用定时器T0设计一个能够计算时、分、秒的时钟。设晶振频率为12MHz。

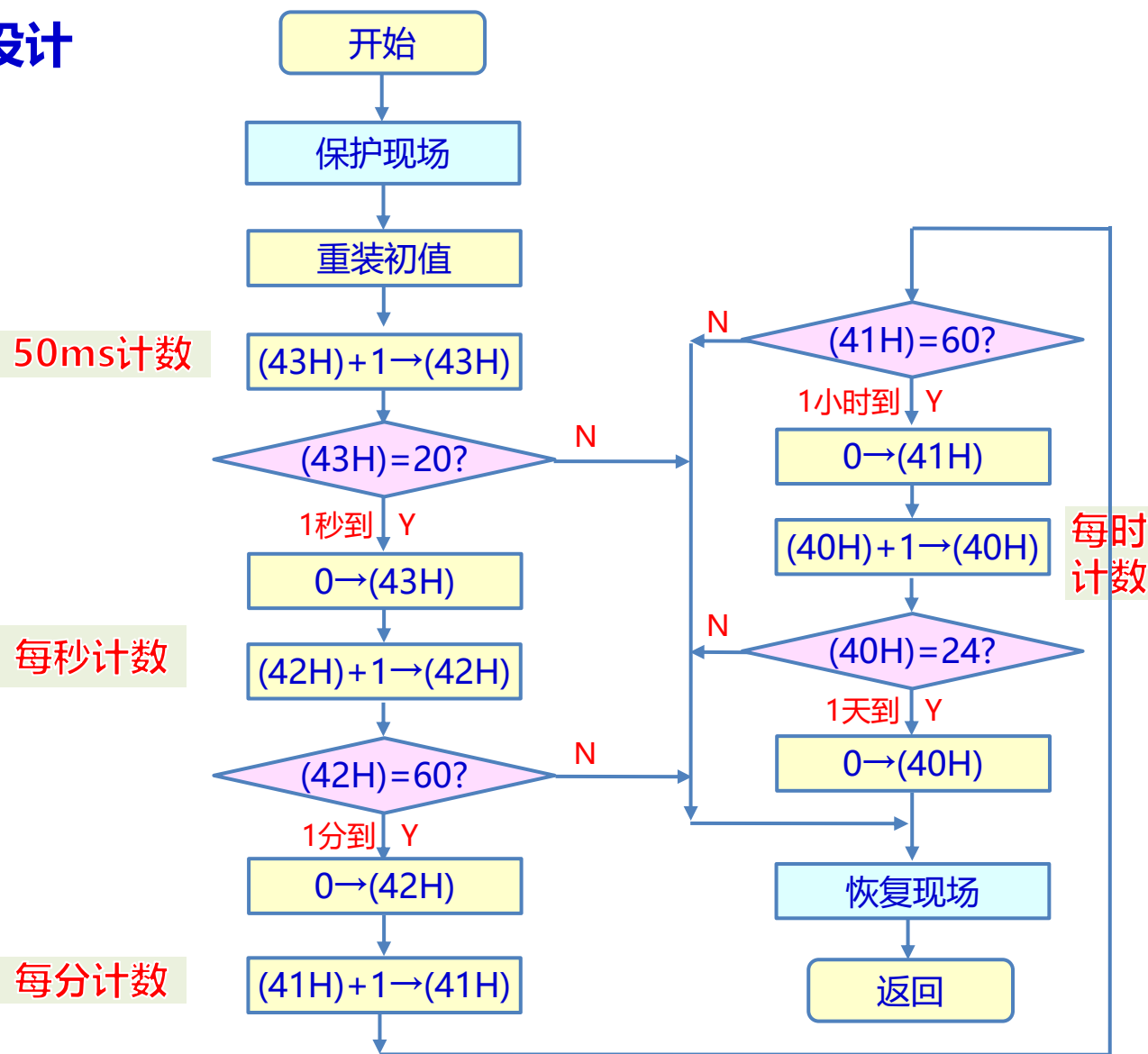
#### 1、主程序设计

- T0、中断和软件计数器初始化；
- 启动T0工作；
- 然后执行其他功能程序，本例中为反复调用显示子程序，显示实时时钟；同时**随时相应50ms定时中断**。



## 2、中断服务程序流程图设计

**中断服务程序：**实现50ms、秒、分、时的计时处理和实时时钟的更新。





### 3、显示预处理子程序设计

目的是将要显示的实际数据送到显示缓冲区中，由于数据类型不同，这部分的程序有较大的差异。

如何将两位的十六进制数分配到两个数码管中呢？

设要显示的十进制数为“58”，拆成“05”和“08”，分别送到2个显示缓冲单元 (52H、53H)中，其程序如下：

；方法1: 如果将十进制数58看作普通十六进制数3AH，通过除法实现

```
MOV    A, #3AH    ;十六进制数
MOV    B, #0AH
DIV    AB          ; A=05H, B=08H
MOV    52H, A
MOV    53H, B
```

；方法2: 如果看作压缩十进制数58H，通过与运算实现，与0FH做与运算得到低位数码管数值，与F0H做与运算后再高低位交换得到高位数码管数值

```
MOV    A, #58H    ;压缩十进制数
ANL    A, #0FH
MOV    53H, A
MOV    A, #58H
ANL    A, #0F0H
SWAP   A
MOV    52H, A
```

## 10.3.5 实时时钟设计

位 符 号	7	6	5	4	3	2	1	0
英文 注释	GATE	C/T	M1	M0	GATE	C/T	M1	M0
	Gate	Counter/ Timer	Mode bit 1	Mode bit 0	Gate	Counter/ Timer	Mode bit 1	Mode bit 0
	定时/计数器T1				定时/计数器T0			

### 汇编程序:

#### 主程序

```

ORG    0000H
SJMP   MAIN           ;转主程序
ORG    000BH
LJMP   T0INT          ;T0中断程序

      ORG    0030H
MAIN:  MOV     SP, #5FH
      MOV     TMOD, #01H      ;设T0为方式1
      CLR     A
      MOV     40H, A          ;“时”单元清“0”
      MOV     41H, A          ;“分”单元清“0”
      MOV     42H, A          ;“秒”单元清“0”
      MOV     43H, A          ;“50ms”单元清“0”
      SETB    ET0             ;T0中断允许
      SETB    EA             ;CPU中断允许
      MOV     TH0, #3CH       ;设置定时初值
      MOV     TL0, #0B0H
      SETB    TR0             ;启动T0
HERE:  LCALL   DISPLAY        ;循环调用显示程序（及其它程序）
      SJMP    HERE           ;（也可1s调用一次显示程序）
  
```

## 10.3.5 实时时钟设计

## § 10.3 定时器/计数器应用

### 中断服务程序

60H,24H均为  
压缩十进制数

```
TOINT:  PUSH    PSW                ;T0中断服务
        PUSH    ACC
        MOV     TH0, #3CH        ;重装定时初值
        MOV     TL0, #0B0H
        INC     43H              ;50ms个数+1
        MOV     A,43H
        CJNE    A, #20, RETURN   ;是否到1秒, 未到则返回
        MOV     43H, #00H       ;50ms个数清0

        MOV     A, #01H         ;秒数+1
        ADD     A, 42H
        DA      A               ;秒数进行十进制调整
        MOV     42H, A          ;保存修改后的秒数
        CJNE    A, #60H, RETURN ;是否到60秒, 未到则返回
        MOV     42H, #00H       ;计满60秒, 秒数清0

        MOV     A, #01H         ;分数+1
        ADD     A, 41H
        DA      A               ;分数进行十进制调整
        MOV     41H, A          ;保存修改后的分数
        CJNE    A, #60H, RETURN ;是否到60分, 未到则返回
        MOV     41H, #00H       ;满60分, 分数清0

        MOV     A, #01H         ;时数+1
        ADD     A, 40H
        DA      A               ;时数进行十进制调整
        MOV     40H, A
        CJNE    A, #24H, RETURN ;是否到24小时, 未到则返回
        MOV     40H, #00H       ;到24小时, 时数清0

RETURN:  POP     ACC             ;恢复现场
        POP     PSW
        RETI                   ;中断返回
        END
```

### 任务:

1. 复习这章学习内容

2. 完成下面的作业

(1) 按实验指导书要求完成

《实验四 脉冲计数实验》

(2) 预习教材第九章 模拟接口技术, 9.1、9.2.1、9.3.1



作业:

P181: 2~6、8、9

THE END

THANK YOU

### 1. 定时器/计数器的功能

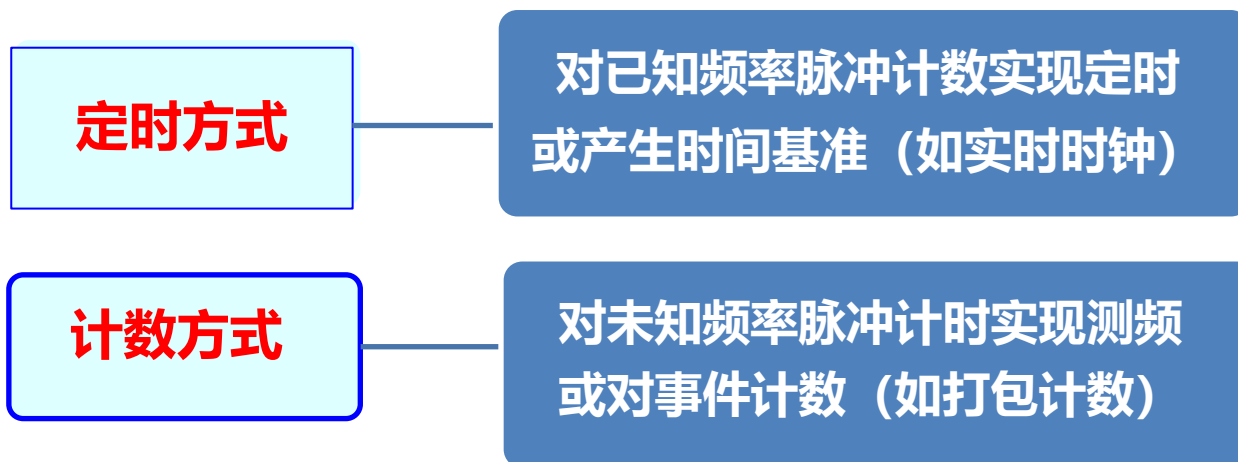
#### □ 定时器（模式）：实现硬件的准确定时

- 微机系统的定时检测和控制；
- 测量事件的时间间隔（如脉冲宽度）；
- 为串行端口提供波特率时钟信号。

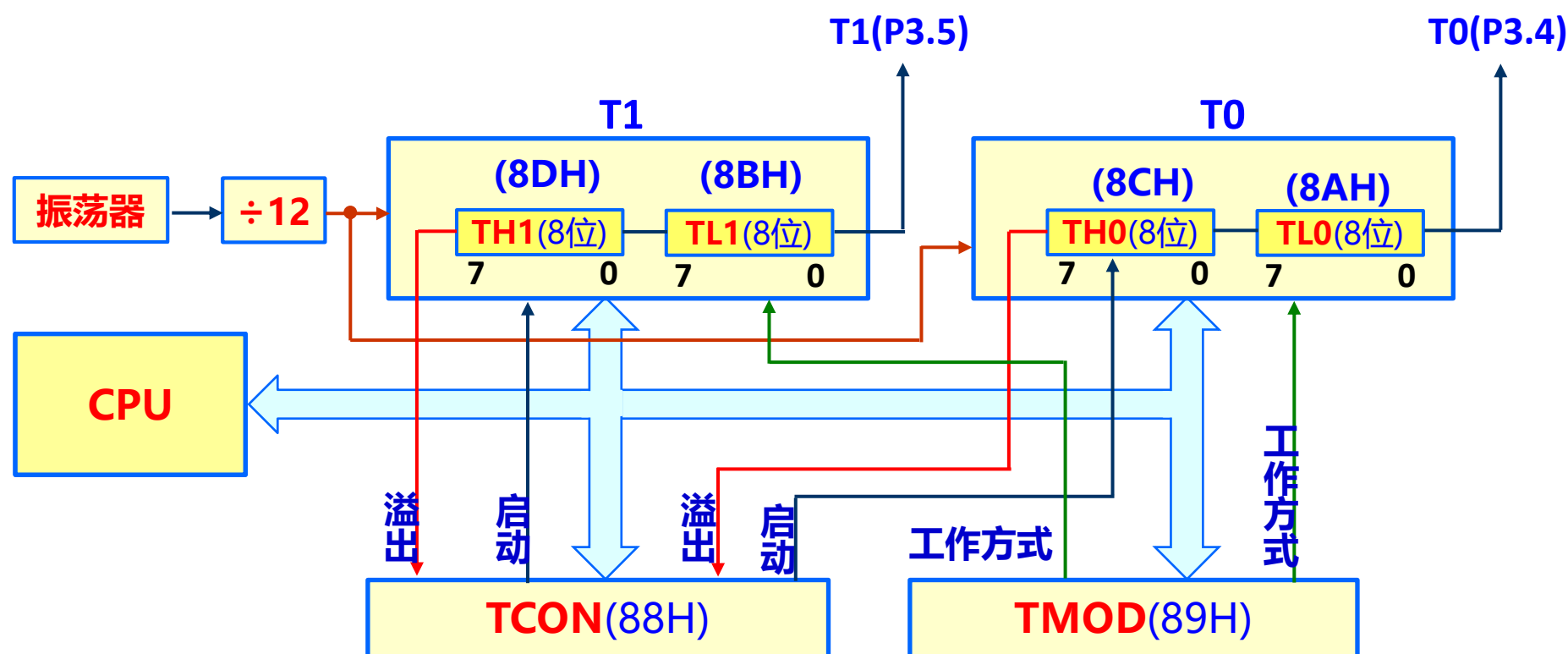
核心部件：**脉冲计数器**

基本原理：对输入脉冲**进行  
加减计数**

#### □ 计数器（模式）：测定外部脉冲/事件发生的次数

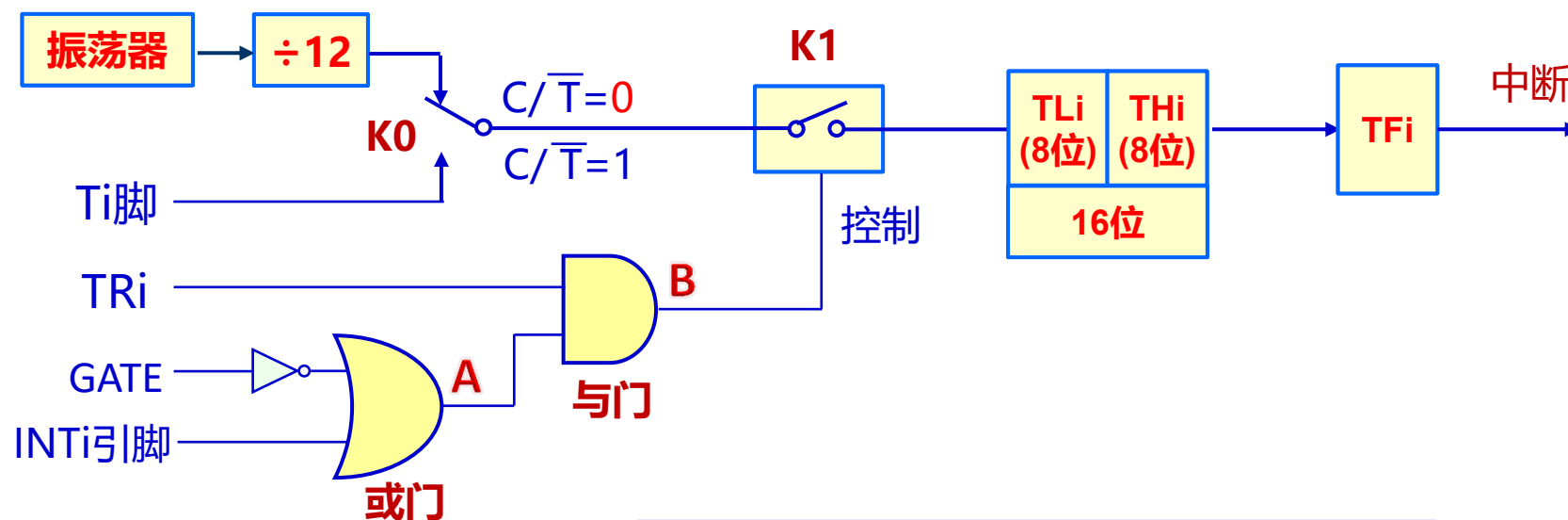


### 2. 定时器/计数器组成结构



### 3. 定时器/计数器工作方式

#### 方式1：16位定时/计数方式



- 最大计数值为  $2^{16}=65536$ ;
- 溢出时，必须立刻重装载初值;
- 存在一定的定时/计数误差。





### 4. 定时器/计数器的初始化

#### 1. 初始化步骤

- 确定工作方式，即设置方式寄存器TMOD。
- 预置定时初值/计数初值，并写入TH0、TL0或TH1、TL1。
- 中断设置（给IE赋值），允许或禁止定时器/计数器的中断。
- 启动定时器/计数器，令TCON中的TR0或TR1为“1”。

#### 2. 定时/计数初值的确定

- 取决于需要定时（计数）值相对于工作方式最大计数值的补码
- 方式1最大计数值：65536；方式2最大计数值：256

### 5. 定时器/计数器的应用

定时方式的应用	<ul style="list-style-type: none"><li>• 波形输出，如输出周期性方波</li><li>• 产生等时间间隔定时信号，定时执行某功能程序等</li></ul>
计数方式的应用	外部事件统计，如流水线上工件的计数； 外部脉冲计数，当脉冲频率未知时，应采用工作方式1
脉冲宽度测量	结合GATE，测量外部高脉冲的宽度
外部扩展中断	方式2，定时初值为0FFH时，T0或T1引脚的下降沿即产生溢出中断
实时时钟设计	以16位工作方式定时50ms为基准，结合软件计数器，计算得到秒、分、时，实现时钟功能