



# 第5章 单片机的定时器





## 5.1 定时/计数器

- ★ 定时计数器结构和工作原理
- ★ 定时计数器的控制寄存器
- ★ 定时器的四种工作方式
- ★ 定时计数器的应用编程





### 5.1.1 8051定时/计数器结构和工作原理

★ 51系列单片机基本型片内有二个十六位定时/计数器：定时器0(T0)和定时器1(T1)；增强型片内有三个十六位定时/计数器：T0、T1、T2。

★ 定时/计数器实际上是16位加1计数器。

T0由2个8位特殊功能寄存器TH0和TL0构成，

T1由2个8位特殊功能寄存TH1和TL1构成。

★每个定时器都可由软件设置为定时工作方式或计数工作方式。

★T0和T1受特殊功能寄存器TMOD和TCON控制。





### 1. 定时工作方式

- ★ 设置为定时工作方式时，定时器计数的脉冲是由51单片机片内振荡器经12分频后产生的。
- ★ 每经过一个机器周期定时器(T0或T1)的数值加1直至计数满产生溢出。





## 2. 计数工作方式

- ★ 设置为计数工作方式时，通过引脚T0 (P3. 4) 和 T1 (P3. 5) 对外部脉冲信号计数。
- ★ 当输入脉冲信号产生由1至0的下降沿时，定时器的值加1，在每个机器周期CPU采样T0和T1的输入电平。若前一个机器周期采样值为高，下一个机器周期采样值为低，则计数器加1。
- ★ 由于检测一个1至0的跳变需要二个机器周期，故最高计数频率为振荡频率的二十四分之一。
- ★ 虽然对输入信号的占空比无特殊要求，但为了确保某个电平在变化之前至少被采样一次，要求电平保持时间至少是一个完整的机器周期。





## 5.1.2 定时计数器的控制寄存器

★定时器共有两个控制寄存器：

定时器控制TCON（88H）

定时器工作模式寄存器TMOD(89H)

### 1. 工作模式寄存器TMOD(89H)

TMOD用于控制T0和T1的操作模式。其各位的定义如下：

定时器T1

定时器T0

TMOD	D7	D6	D5	D4	D3	D2	D1	D0
(89H)	GATE	C/T	M1	M0	GATE	C/T	M1	M0





### ★ GATE: 门控信号

**GATE=0**, **TR<sub>x</sub>=1**时即可启动定时器工作 ;

**GATE=1**, 除**TR<sub>x</sub>=1** 外,还需**INT<sub>x</sub>=1**才可启动定时器工作。

### ★ $\overline{C/T}$ : 定时器/计数器选择位

$\overline{C/T}=1$ , 为计数器方式;

$\overline{C/T}=0$ , 为定时器方式。

### ★ M1 M0 工作模式选择位

**M1M0=00** 工作方式0 (13位方式) 。

**M1M0=01** 工作方式1 (16位方式) 。

**M1M0=10** 工作方式2 (8位自动再装入方式) 。

**M1M0=11** 工作方式3 (T0为2个8位方式) 。





## 2. 控制寄存器TCON(88H)

TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
T1 请求 有/无	T1 工作 启/停	T0 请求 有/无	T0 工作 启/停	INT1 请求 有/无	INT1 方式 下沿/ 低 电平	INT0 请求 有/无	INT0 方式 下沿/低 电平

★TF1: T1溢出中断请求标志。

TF1=1, T1有溢出中断请求。

TF1=0, T1无溢出中断请求。

★TR1: T1运行控制位。

TR1=1, 启动T1工作。

TR1=0, 停止T1工作。

★TF0: T0溢出中断请求标志。

TF0=1, T0有溢出中断请求。

TF0=0, T0无溢出中断请求。

★TR0: T0运行控制位。

TR0=1, 启动T0工作。

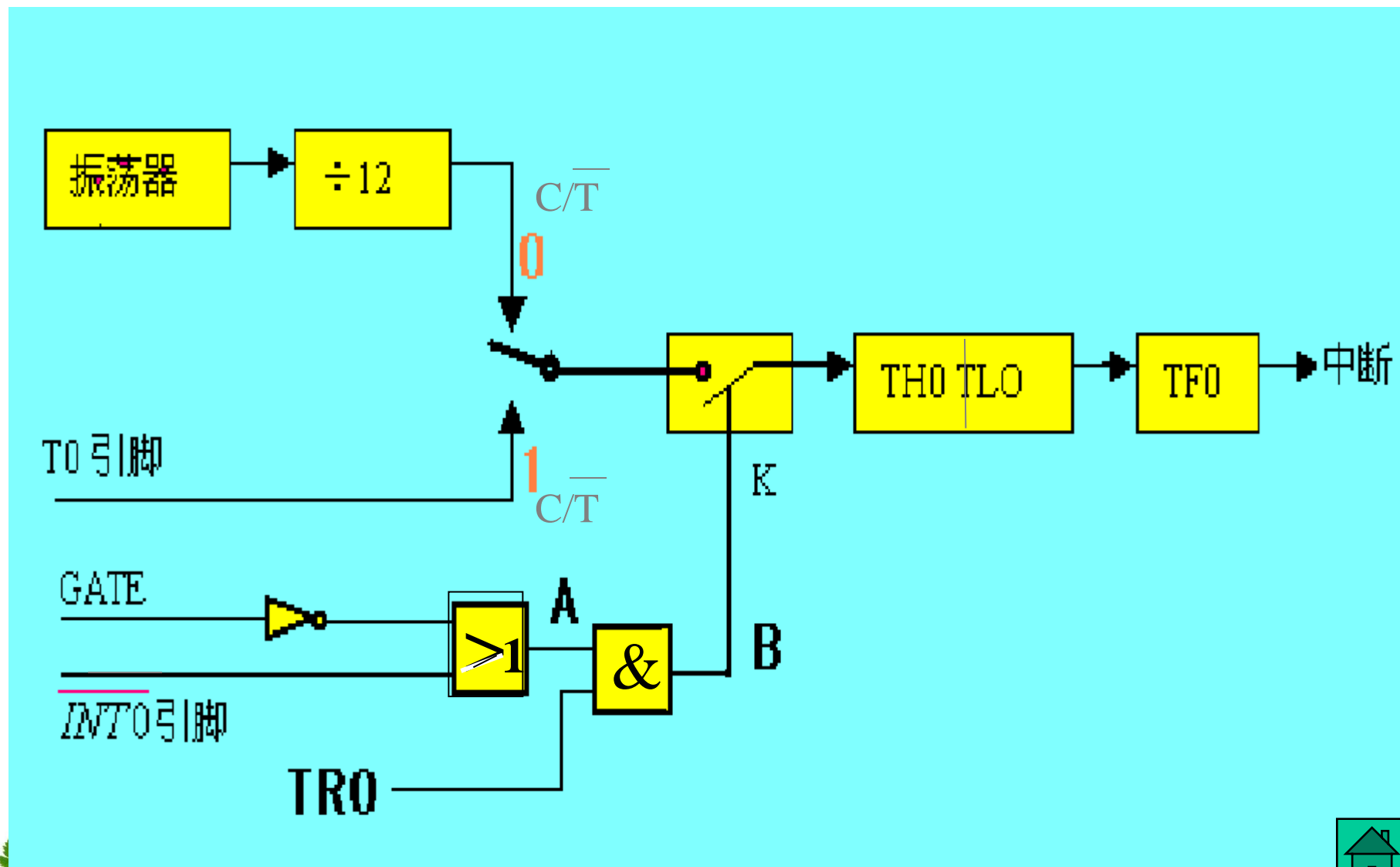
TR0=0, 停止T0工作。







# 定时计数器的内部逻辑电路结构





## 5.1.3 定时器的四种工作方式

对TMOD寄存器的M1、M0位的设置，可选择四种工作方式，即方式0、方式1、方式2和方式3。下面用THX、TLX（X=1或0）表示TH1 TL1； TH0 TL0。

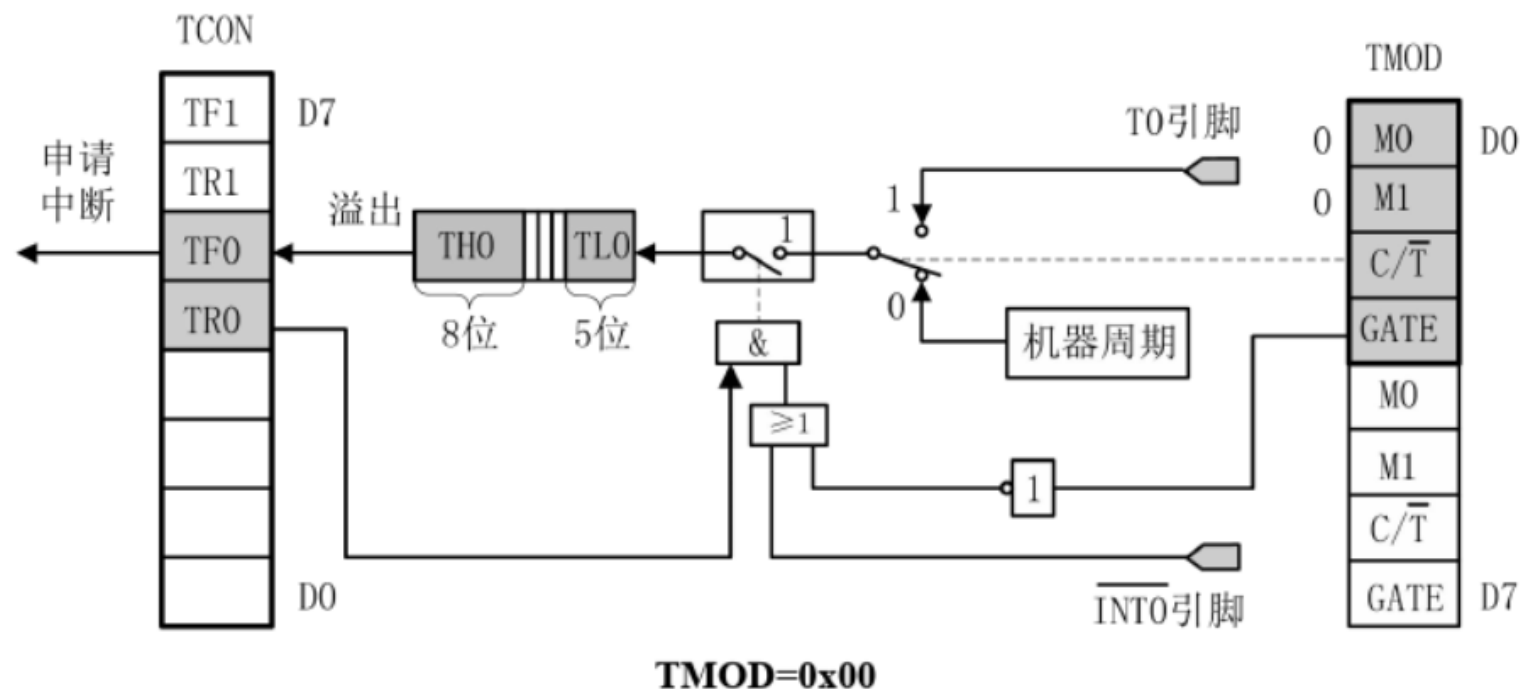
### 1.方式 0

★定时器(T0或T1)工作于13位定时、计数方式。用于计数方式时最大计数值为  $2^{13} = 8192$  个脉冲用于定时工作时，定时时间为： $t = (2^{13} - T0\text{初值}) \times \text{时钟周期} \times 12$

★在这种模式下，只用13位，其中THX8位，TLX5位，**TLX的高3位末用**。

★当TLX的**低5位溢出**时向THX进位,而THX溢出时硬件置位TF0，并申请中断。

★ 定时、计数溢出否可查询TF0是否置位，如果开中断则产生溢出中断。





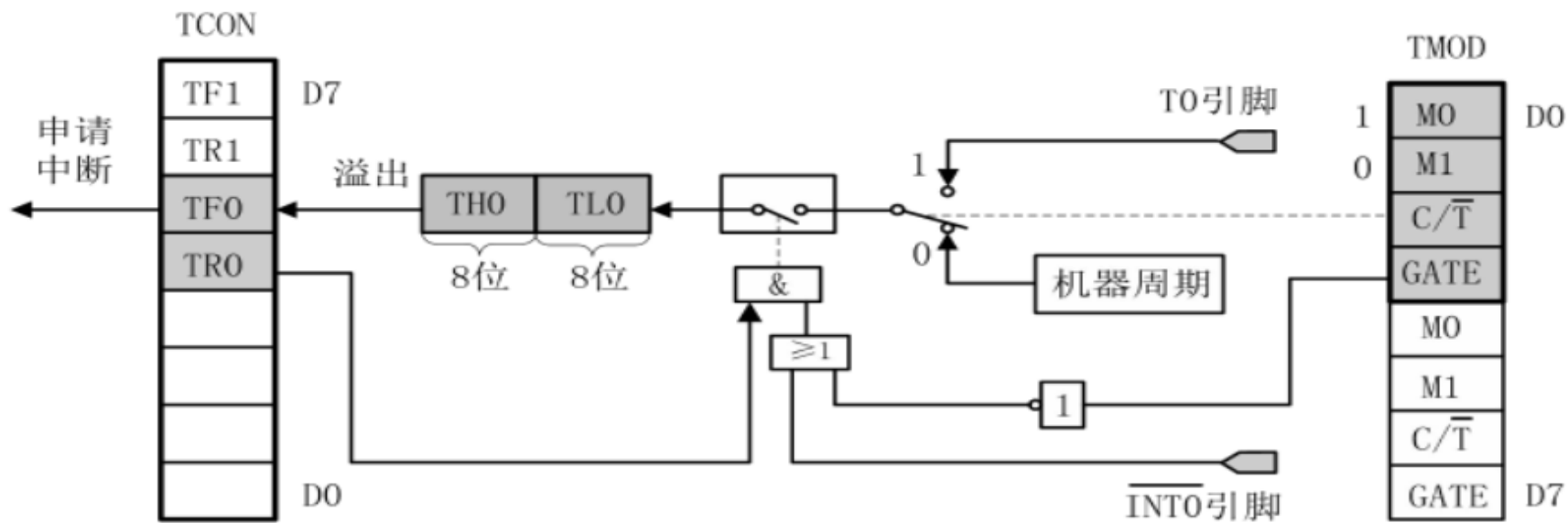
## 2.方式1

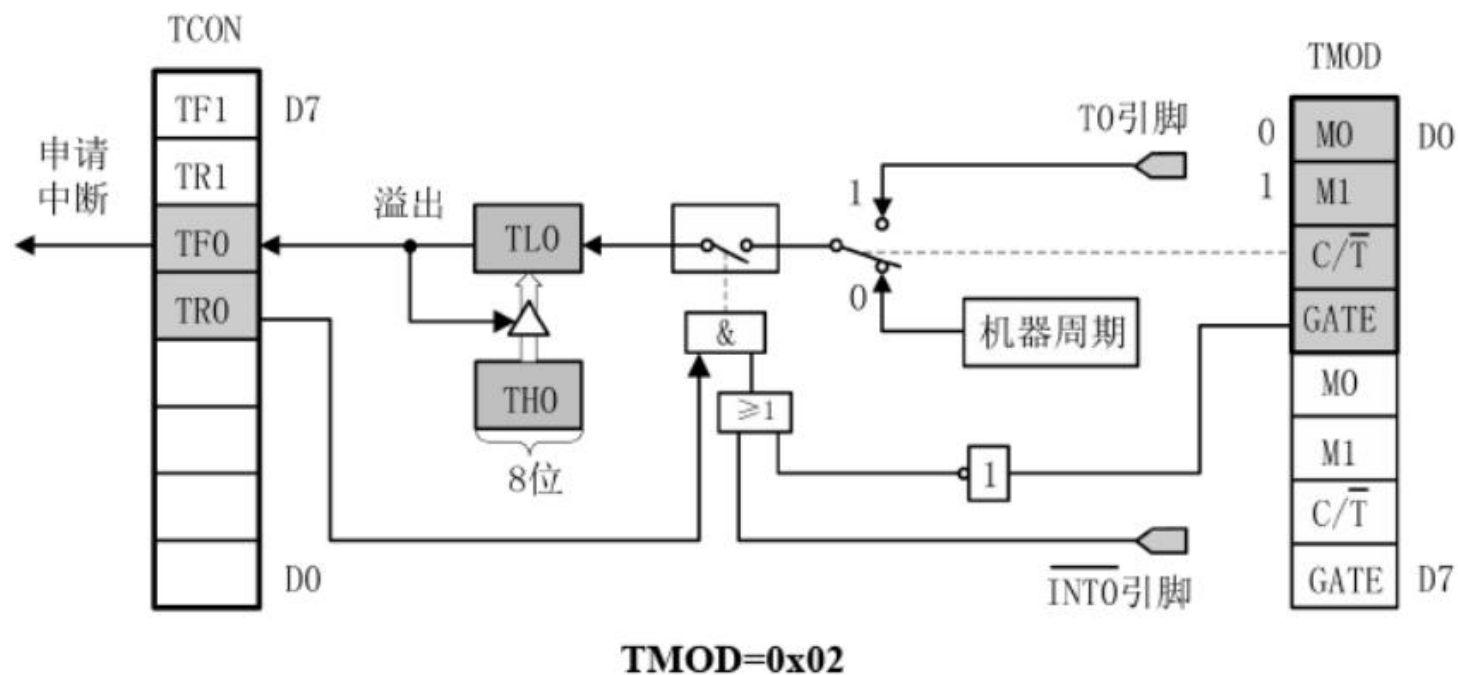
当TMOD中M1M0=01时，定时计数器工作在方式1。

★该模式是一个16位定时 / 计数方式。寄存器TH0和TL0是以全16位参与操作，计数方式时最大计数  $2^{16}=65536$ (个外部脉冲) 用于定时工作方式时，定时时间为：

$$t=(2^{16}-T0初值) \times \text{时钟周期} \times 12$$

★ 16寄存器(THX和TLX) 中THX提供高8位、TLX提供低8位计数初值







## 4. 方式3

当TMOD中M1M0=11时，定时器工作在方式3。

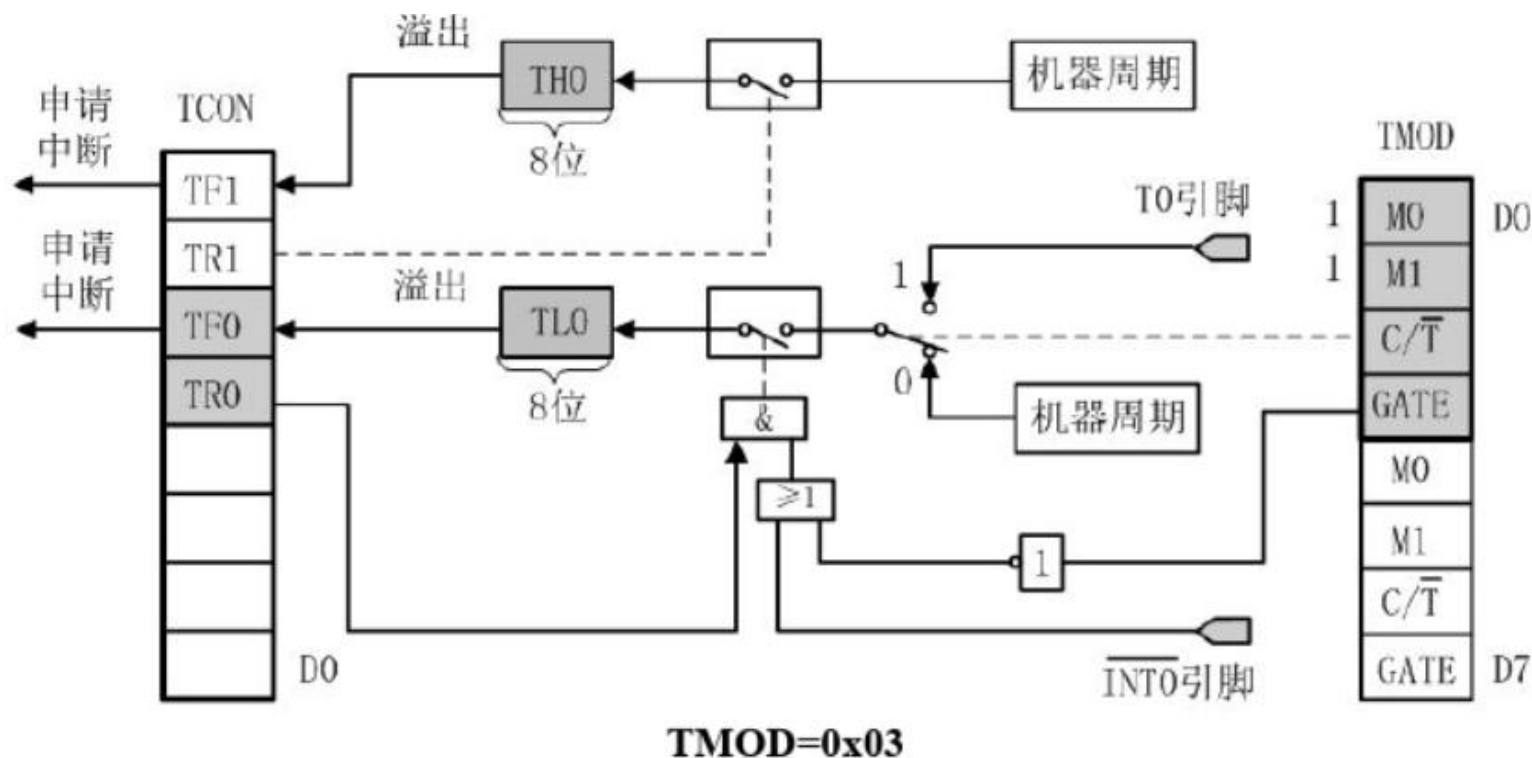
★若将T0设置为模式3，TL0和TH0被分成为两个互相独立的8位计数器。

★ TL0可工作为定时方式或计数方式。

TH0只可用作定时功能，  
占用定时器T1的控制位TR1  
和T1的中断标志位TF1。

★定时器T1无模式3，可工  
作于方式0、1、2，但不能  
使用中断方式。

★ 只有将T1用做串行口的波  
特率发生器时，T0才工作在方  
式3，以便增加一个定时器。





## 5.1.4 定时计数器的应用编程

### 5.1.4.1 定时器的计数初值C的计算和装入

如前所述，8xx51定时器/计数器不同工作方式的模值不同，由于采用加1计数，因此计数初值应为负值，计算机中用有符号数采用补码表示。

计数初值（C）的求法如下。

#### ★计数方式：

计数初值  $C = \text{模} - X$ （其中X为要计的脉冲个数）

#### ★定时方式：

计数初值  $C = [t / MC] \text{补} = \text{模} - t / MC$

其中t为欲定时时间，MC为8xx51的机器周MC=12/fosc

当采用12MHZ晶振时，MC=1μs ；

当采用6MHZ晶振时，MC=2μs。





**例** 要计**100**个脉冲的计数初值

**方式0（13位方式）：**

$C = (-64H) \text{ 补} = 2000H - 64H = 1F9CH = 0001\ 1111\ 1001\ 1100B$

把13位中的高八位1111 1100B装入TH0，而把13位中的低五位xxx1 1100B装入TL0。

**MOV TH0, #0FCH; MOV TL0, #1CH;** (xxx用“0”填入)

**方式1(16位方式)：**

$C = (-64H) \text{ 补} = 10000H - 64H = FF9CH$  用指令装入计数初值：

**MOV TH0, #0FFH; MOV TL0, #9CH**

**方式2（8位自动再装入方式）**

$C = (-64H) \text{ 补} = 100H - 64H = 9CH$  初值既要装入TH0，也要装入TL0。

**MOV TH0, #9CH; MOV TL0, #9CH**





### 5.1.4.2 定时计数器的初始化编程

定时计数器的初始化编程步骤：

- 1) 根据定时时间要求或计数要求计算计数器初值；
- 2) 工作方式控制字送TMOD寄存器；
- 3) 送计数初值的高八位和低八位到THx和TLx寄存器中；
- 4) 启动定时（或计数），即将TRX置位。

如果工作于中断方式，需要置位EA（中断总开关）及ETx（允许定时/计数器中断）。并编中断服务程序。



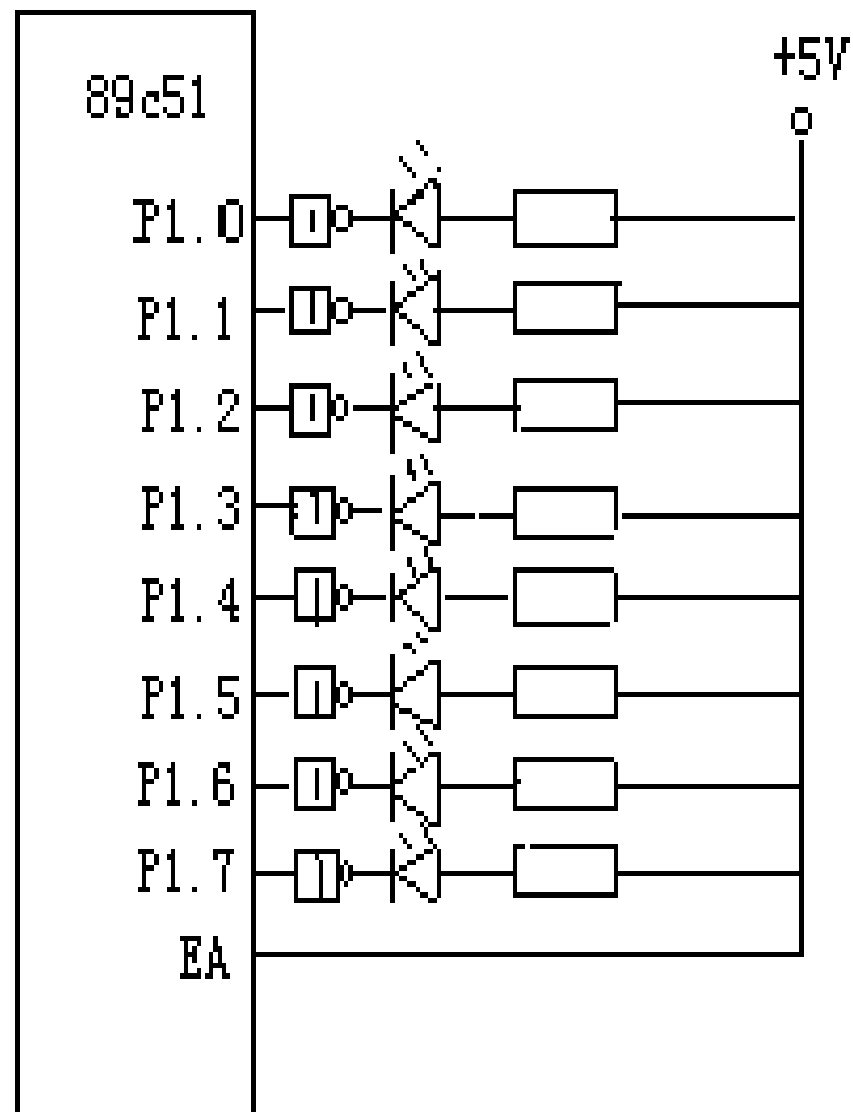




## 5.1.4.3 应用编程举例

**例5-1** 如图7-2所示，

**P1**中接有八个发光二极管，  
编程使八个管轮流点亮，每  
个管亮**100ms**，设晶振为  
**6MHz**。





**分析** 利用T1完成100ms的定时，当P1口线输出“1”时，发光二极管亮，每隔100ms”1”左移一次，采用定时方式1，先计算计数初值：

$$MC=2\mu s$$

$$\text{计数值}=100\text{ms}/2\mu s_{\mu}=50000=\text{C350H}$$

$$\text{初值}=(\text{C350H})_{\text{补}}=10000\text{H}-\text{C350H}=\text{3CB0H}$$





①查询方式如下：

TMOD	D7	D6	D5	D4	D3	D2	D1	D0
(89H)	GATE	C/T	M1	M0	GATE	C/T	M1	M0

ORG 0030H

MOV A, #01H: 置第一个LED亮

NEXT: MOV P1, A

MOV TMOD, #10H ; T1工作于定时方式1

MOV TH1, #3CH

MOV TL1, #0B0H; 定时100ms

SETB TR1

AGAI: JBC TF1, SHI; 100ms到转SHI, 并清TF1

SJMP AGAI

SHI: RL A

SJMP NEXT





## ②中断方式

```
ORG 0000H
AJMP MAIN      ; 单片机从0000H开始执行
```

```
ORG 001BH
AJMP IV1       ; 转移到IV1
```

```
ORG 0030H      ; 主程序
```

```
MAIN: MOV A, #01H
      MOV P1, A      ; 置第一个LED亮
      MOV TMOD, #10H ; T1定时方式1
      MOV TH1, #3CH
      MOV TL1, #0B0H ; 定时100ms
      SETB TR1       ; 启动T1工作
      SETB ET1       ; 允许T1中断
      SETB EA        ; 允许总中断
WAIT: SJMP WAIT      ; 等待中断
```

TMOD	D7	D6	D5	D4	D3	D2	D1	D0
(89H)	GATE	C/T	M1	M0	GATE	C/T	M1	M0

```
IV1: RL  A      ; 中断程序, 左移一位
      MOV P1, A  ; 下一个二极管亮
      MOV TH1, #3CH
      MOV TL1, #0B0H ; 重装初值
      RETI       ; 中断返回
```

以上程序进入循环执行, 八个LED一直循环轮流点亮。





**例5-2** 在P1.7端接一个发光二极管LED，要求利用定时控制使LED亮一秒灭一秒周而复始，设 $f_{osc}=6\text{MHz}$ 。

**解：** 16位定时最大为  $2^{16} * 2\mu\text{s} = 131.072\text{ms}$ ，显然不能满足要求，可用以下两种方法解决。

**方法1：**采用T0产生周期为200ms脉冲，即P1.0每100ms取反一次作为T1的计数脉冲，T1对下降沿计数，因此T1计5个脉冲正好100ms。

T0采用方式1，

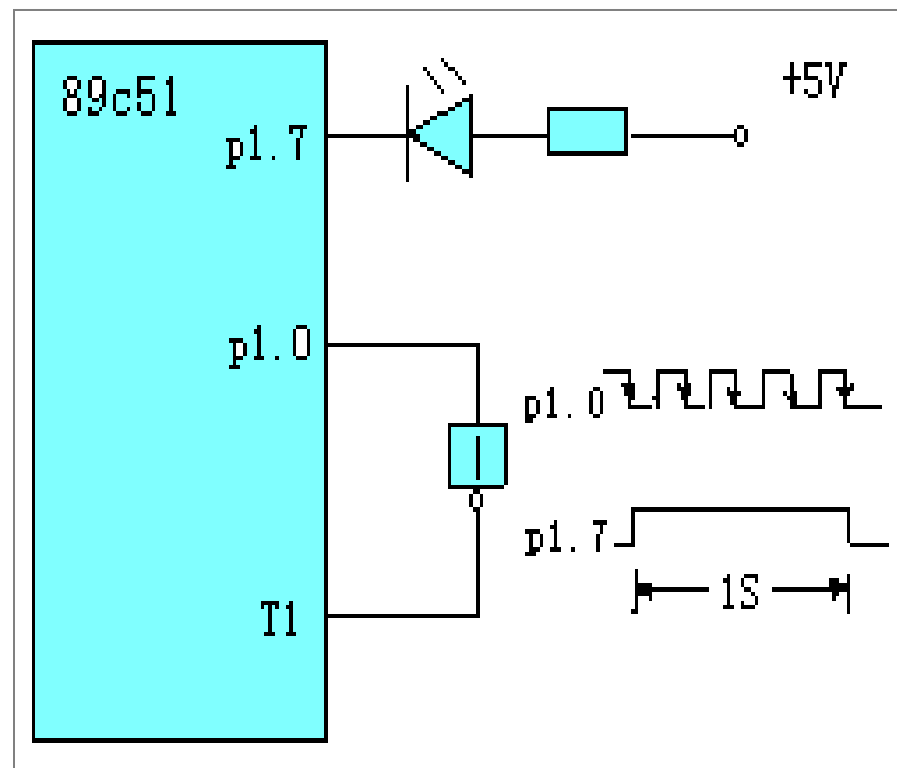
$X = 2^{16} - \frac{100 \times 10^3}{2}$  得  $X = 3\text{CB}0\text{H}$ ，

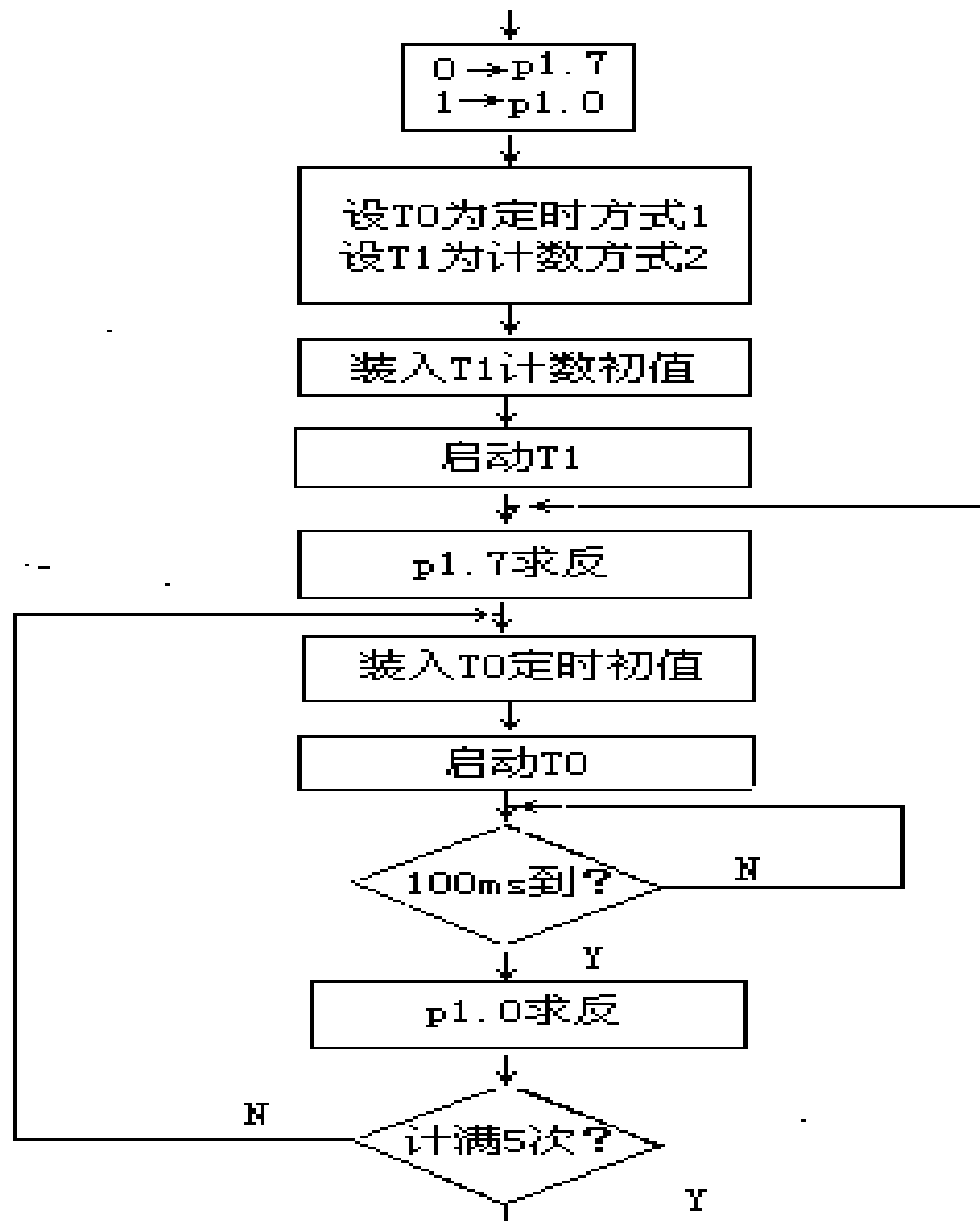
T1采用方式2，

计数初值  $X = 2^8 - 5 = \text{FBH}$

均采用查询方式，

流程图和程序如下：







```
ORG 0000H  
  
MAIN: CLR P1.7  
  
SETB P1.0  
  
MOV TMOD, #61H  
  
MOV TH1, #0FBH  
  
MOV TL1, #0FBH  
  
SETB TR1  
  
LOOP1: CPL P1.7
```

TMOD	D7	D6	D5	D4	D3	D2	D1	D0
(89H)	GATE	C/T	M1	M0	GATE	C/T	M1	M0

```
LOOP2: MOV TH0, #3CH  
  
MOV TL0, #0B0H  
  
SETB TR0  
  
LOOP3: JBC TF0, LOOP4  
  
SJMP LOOP3  
  
LOOP4: CPL P1.0  
  
JBC TF1, LOOP1  
  
AJMP LOOP2  
  
END
```

程序中用JBC指令对定时/计数溢出标志位进行检测，当标志位为1时跳转并清标志。





TMOD	D7	D6	D5	D4	D3	D2	D1	D0
(89H)	GATE	C/T	M1	M0	GATE	C/T	M1	M0

**方法2:** T0每隔100ms中断一次，

利用软件对T0的中断次数进行计数，中断10次即实现了1秒的定时。

**ORG 000BH ; T0中断程序**

**AJMP IP0**

**ORG 0030H ; 主程序开始**

**MAIN: CLR P1.7 ; T0定时100ms**

**MOV TMOD, #01H**

**MOV TH0, #3CH**

**MOV TL0, #0B0H**

**SETB ET0**

**SETB EA**



**MOV R4, #0AH; 中断10次计数**

**SETB TR0**

**SJMP \$ ; 等待中断**

**IP0: DJNZ R4, RET0;**

**MOV R4, #0AH**

**CPL P1.7**

**RET0: MOV TH0, #3CH**

**MOV TL0, #0B0H**

**SETB TR0**

**RETI**







## 10.8.4 定时/计数器的C语言编程

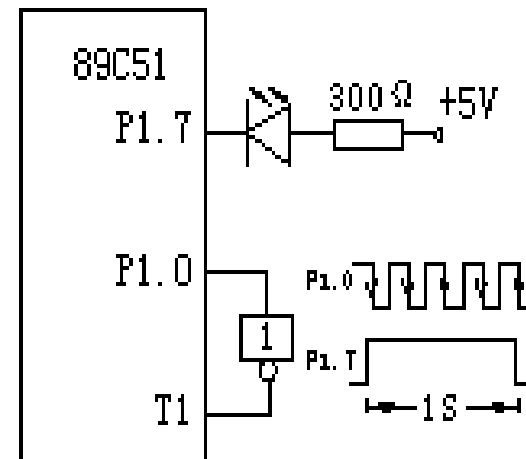
**例10-17** 在P1.7端接一个发光二极管LED，要求利用定时控制使LED亮一秒灭一秒周而复始，设 $f_{osc}=6\text{MHz}$ 。

**分析** T0定时100ms初值 $=100 \times 10^3 / 2 = 50000$ ，即初值为-50000。

T1计数5个脉冲工作于方式2，计数初值为-5，T0和T1均采用中断方式。程序如下：

```
#include <reg51.h>
sbit P1_0=P1^0;
sbit P1_7=P1^7;
timer0() interrupt 1 using 1 /* T0中断服务程序 */
{
    P1_0=! P1_0; /* 100ms到P1.0反相*/
    TH0=-50000/256; /* 重载计数初值 */
    TL0=-50000%256;
}
timer1() interrupt 3 using 2 /* T1中断服务程序 */
{
    P1_7=! P1_7; /* 1s到，灯改变状态 */
}
```

TMOD	D7	D6	D5	D4	D3	D2	D1	D0
(89H)	GATE	C/T	M1	M0	GATE	C/T	M1	M0





TMOD	D7	D6	D5	D4	D3	D2	D1	D0
(89H)	GATE	C/T	M1	M0	GATE	C/T	M1	M0

```
main () {  
    P1_7=0;          /* 置灯初始灭 */  
    P1_0=1;          /* 保证第一次反相便开始计数 */  
    TMOD=0x61;       /* T0方式 1 定时, T1方式 2 计数 */  
    TH0=-50000/256;  /* 预置计数初值 */  
    TL0=-50000%256;  
    TH1=-5;  
    TL1=-5;  
    IP=0x08;         /* 置优先级寄存器 */  
    EA=1; ET0=1; ET1=1; /* 开中断 */  
    TR0=1; TR1=1;    /* 启动定时/计数器 */  
    for(;;) {; }     /* 等待中断 */  
}
```





**例5-3** 有P3.4引脚(T0)输入一低频信号(其小于0.5kHz),要求P3.4每发生一次负跳变时, P1.0输出一个500us同步负脉冲,同时P1.1输出一个1ms的同步正脉冲。已知晶振频率为6MHZ。

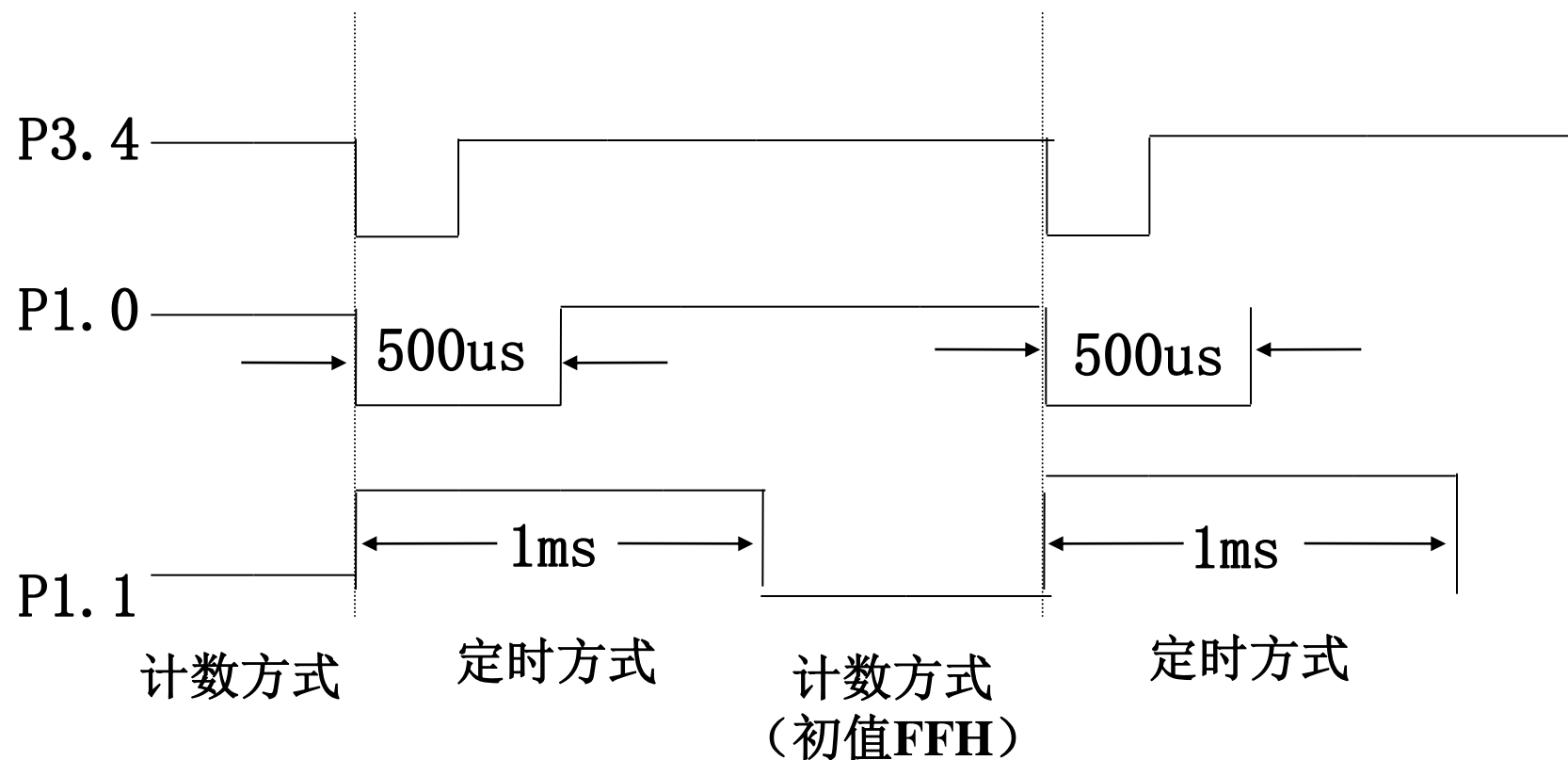


图7-5





解 按提意，设计方法如图7-5所示。

初态P1.1输出高电平(系统复位时实现),P1.1输出低电平，T0选方式2计数方式(计一个脉冲，初值为FFH)。当加在P3.4上的外部脉冲负跳变时，T0加1，计数器溢出，程序查询到TF0为1，改变T0为500μs定时工作方式，并使P1.0输出0，P1.1输出1。T0第一次定时500μs溢出后，P1.0恢复1，T0第二次定时500μs溢出后，P1.1恢复0，T0恢复外部脉冲计数。

设定500μs的初始值为X，则：

$$(256 - X) \times 2 \times 10^{-6} = 500 \times 10^{-6}$$

解得 X=6





TMOD	D7	D6	D5	D4	D3	D2	D1	D0
(89H)	GATE	C/T	M1	M0	GATE	C/T	M1	M0

源程序如下：

**BEGIN:MOV TMOD,#6H ; 设T0为方式 2 外部计数**

**MOV TH0,#0FFH ; 计数一个脉冲**

**MOV TL0,#0FFH**

**CLR P1.1 ; P1.1初值为0**

**SETB TR0 ; 启动计数器**

**DELL: JBC TF0,RESP1 ; 检测外跳变信号**

**AJMP DELL**

**RESP1:CLR TR0**

**MOV TMOD,#02H; 重置T0为500ms定时**





```
MOV TH0,#06H      ; 重置定时初值
MOV TL0,#06H
SETB P1.1          ; P1.1置1
CLR P1.0            ; P1.0清0
SETB TR0           ; 启动定时计数器
DEL2: JBC TF0,RESP2 ; 检测第一次500us到否
      AJMP DEL2
RESP: SETB P1.0      ; P1.0恢复1
DEL3: JBC TF0,RESP3  ; 检测第二次500us到否
      AJMP DEL3
RESP3: CLR P1.1      ; P1.1复0
      CLR TR0
      AJMP BEGIN
```





TMOD	D7	D6	D5	D4	D3	D2	D1	D0
(89H)	GATE	C/T	M1	M0	GATE	C/T	M1	M0

## 5.1.4.4 门控位的应用

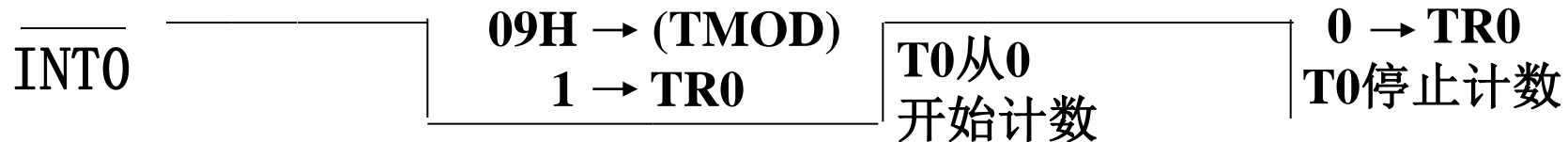
门控位GATE为1时， $TR_x=1$ ， $\overline{INT_x}=1$ 才能启动定时器。利用这个特性可以测量外部输入脉冲的宽度。

**例7-4** 利用T0门控位测试  $\overline{INT0}$  引脚上出现的正脉冲宽度，已知晶振频率为12MHz，将所测得值最高位存入片内71H单元，低位存入70H单元。

解：设外部脉冲 由(P3.2)输入，T0工作于定时方式 1

(16位计数)，GATE设为1。测试时，应在 $\overline{INT0}$  低电平时，设置TR0为1(16位计数)；当 $\overline{INT0}$  变为高电平时，就启动计数；再次变低时，停止计数。此计数值与机器周期的乘积即为被测正脉冲的宽度。因 $f_{osc}=12MHz$ ，机器周期为1us，测试过程如下。





源程序如下：

```
MOV TMOD, #09H ; 设T0为方式1
MOV TL0, #00H   ; 设计初值取最大值
MOV TH0, #00H
MOV R0, #70H
JB P3.2, $      ; 等P3.2( $\overline{\text{INT0}}$ )变低
SETB TR0        ; 启动T0准备工作
JNB P3.2, $     ; 等待P3.2( $\overline{\text{INT0}}$ )
JB P3.2, $      ; 等待P3.2( $\overline{\text{INT0}}$ )
CLR TR0         ; 停止计数
MOV @R0, TL0    ; 存放结果
INC R0
MOV @R0, TH0
SJMP $
```







### 5.1.5 小结

定时计数器应用非常广泛，如定时采样、时间测量、产生音响、作脉冲源、制作日历时钟、测量波形的频率和占空比、检测电机转速等。因此应很好掌握。

★ 51系列单片机既有两个16位的定时计数器，有四种不同的工作方式，归纳于下表：



方 式	方式 0 13位定时 计数方式	方式 1 16位定时 计数方式	方式 2 8位自动 重装方式	方式 3 T0两个8位 方式
模 值 (计数最大值)	<b><math>2^{13}=8192</math> =2000H</b>	<b><math>2^{16}=65536</math> =10000H</b>	<b><math>2^8=256</math> =100H</b>	<b><math>2^8=256</math> =100H</b>
计数初值  C的装入	高八位→TH 低五位→TL	高八位→TH 低八位→TL	八位↙ <sup>TH</sup> ↘ <sup>TL</sup>	同左
	每启动一次工作,需装入一次计数初值		第一次装入,启动工作后,每次TL回零后,不用程序装入,由TH自动装入到TL	同方式0、1
应用场合 (设f <sub>osc</sub> =12MHz)	用于 定时时间 <8.19ms , 计数脉冲 < 8192 个 场合	用于 定时时间 <65.5ms, 计数脉冲 < 65536 个 场合	定时、计数范围小, 不用重装时间常数, 多用于串行通信的波特率发生器	TL <sub>0</sub> 定时, 计数 占用TR <sub>0</sub> 、TF <sub>0</sub> ; TH0定时, 使用 T <sub>1</sub> 的TR <sub>1</sub> 、TF <sub>1</sub> 此时T <sub>1</sub> 作波特率 发生器



★ 定时和计数**实质都是对脉冲的计数**，只是被计脉冲的来源不同，定时方式的被计脉冲来源于时钟，计数方式的被计脉冲来源于外部，定时方式的计数初值和被计脉冲周期有关，计数方式的和被计脉冲的个数有关。

★ 无论定时还是计数，当计满规定的 脉冲个数产生溢出（计数初值寄存器回零），置位TF<sub>x</sub>，可以通过程序查询，如果允许中断，会产生中断。

★ 本章应重点掌握定时计数器的应用设计

