

# 第6章 AT89C52单片机的 定时器/计数器

信息科学与工程学院自动化系

# 本章重点及要求

## 6.1 定时器/计数器的结构

## 6.2 定时器/计数器T0和T1的控制

## 6.3 定时器/计数器T0和T1的工作方式及应用

## 6.4 定时器/计数器T2

# 定时方式与计数方式的主要区别

**共同点：**由计数电路对脉冲个数进行计数，  
可按**递增**或**递减**方式计数。

## 不同点：

### 1、定时方式

定时器由数字电路中的计数电路构成，通过记录高精度晶振脉冲（**固定周期**）信号的个数，输出准确的时间间隔。

### 2、计数方式

计数电路记录外设提供的具有一定**随机性**的脉冲信号时，它主要反映脉冲的个数（进而获知外设的某种状态），常又称为计数器。

**定时：**对周期已知的脉冲信号计数

**计数：**对外部事件计数，对周期未知的外来脉冲信号计数

# 定时/计数器的应用

定时器/计数器的应用非常广泛，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系列单片机既有的两个16位定时计数器的工作原理及其应用技术。

# 6.1 定时器/计数器的结构

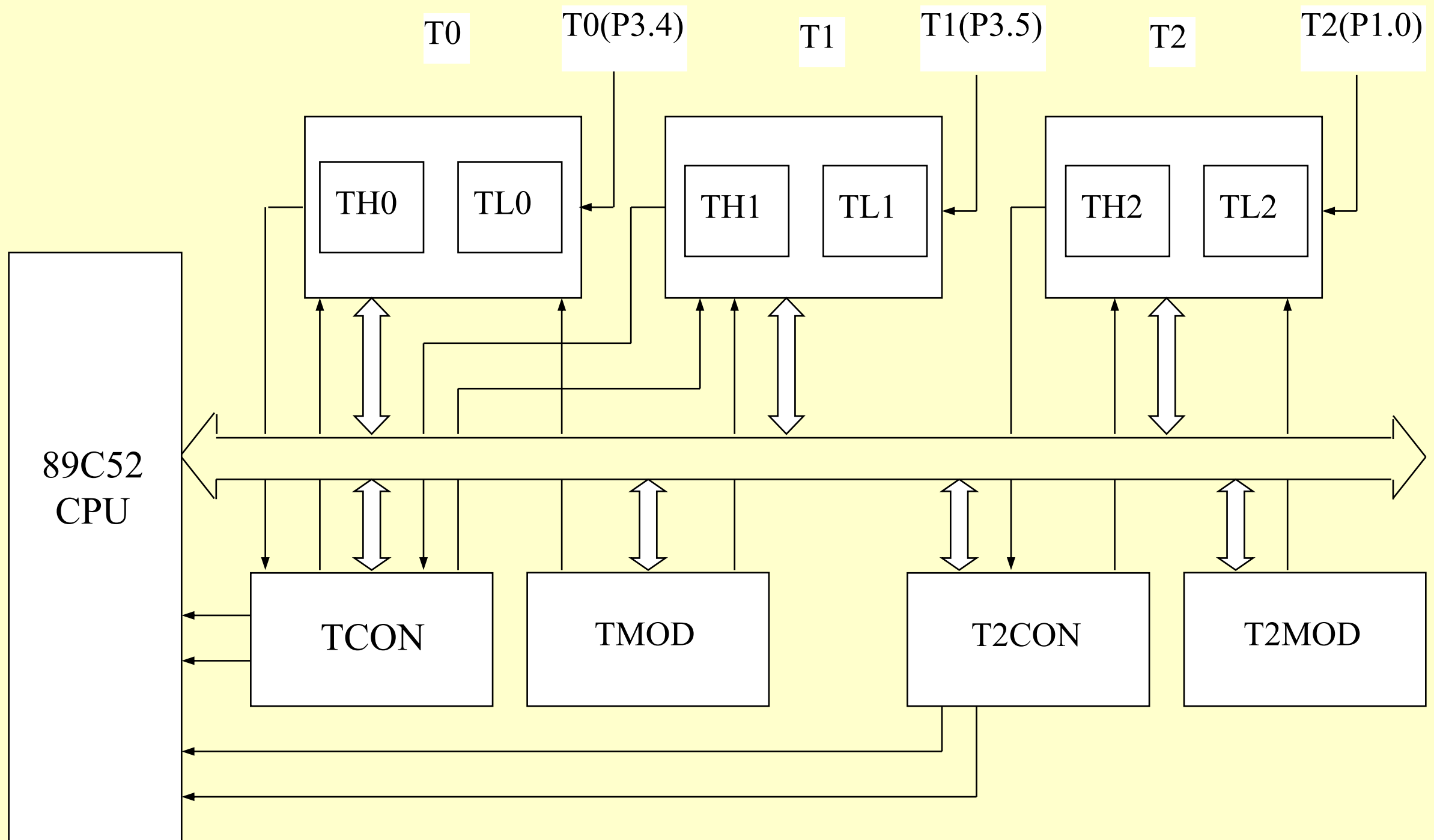


图6-1 定时器/计数器的结构

# 6.2 定时器/计数器T0和T1的控制

T/C0和T/C1都是可编程的16位加1计数器

## 结构组成:

❑ T/C0由两个8位的TH0 (8CH) 和TL0 (8AH) 组成

❑ T/C1由TH1 (8DH) 和TL1 (8BH) 组成

## 2种工作模式:

❑ 计数器工作模式;

❑ 定时器工作模式。

## 特殊功能寄存器

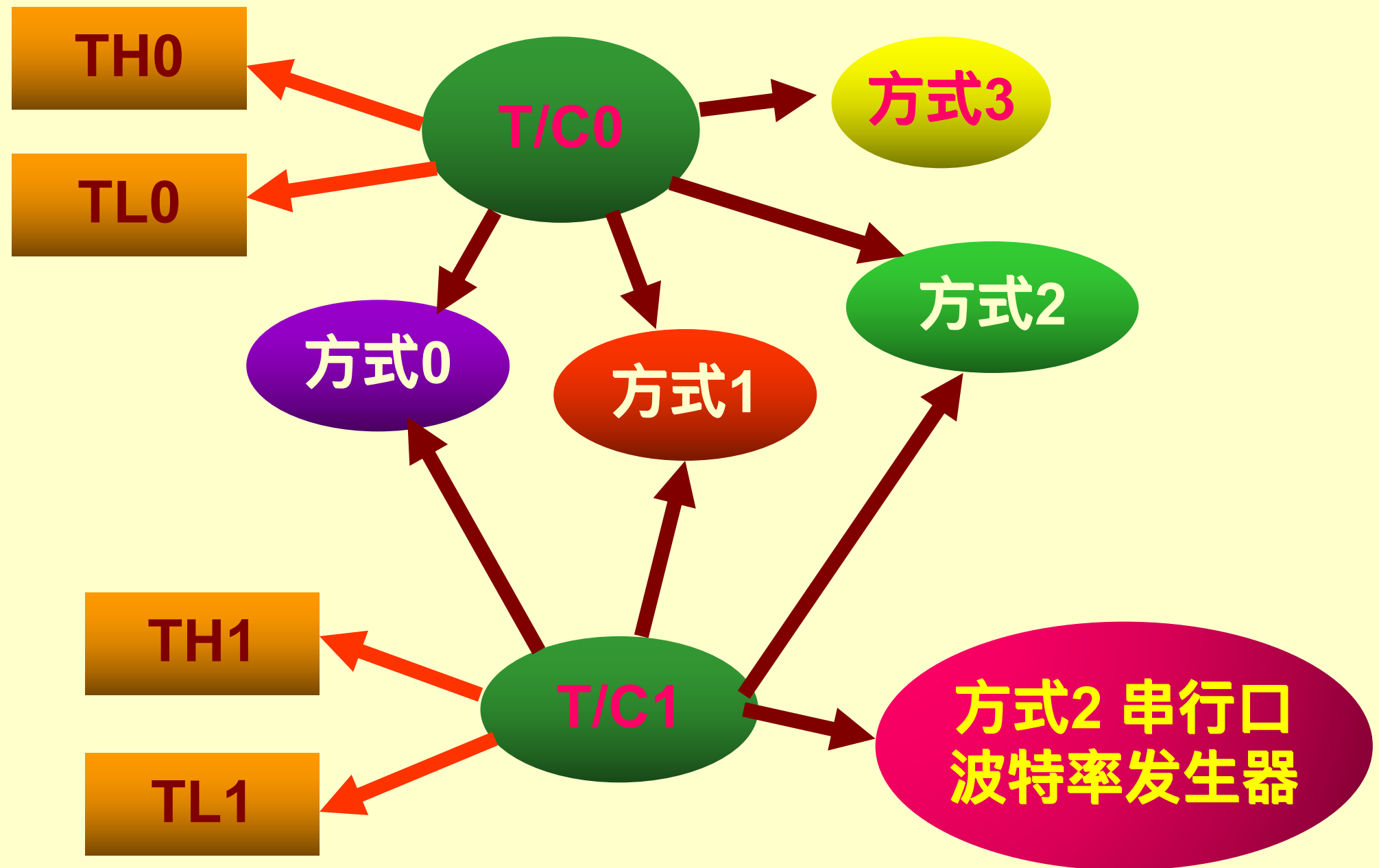
TMOD(Timer Mode Register): 选择T0、T1的工作模式和工作方式。

TCON(Timer Control Register): 控制T0、T1的启动和停止计数,  
同时包含了T0、T1的状态。

TH<sub>x</sub> 存放计数初值的高8位

TL<sub>x</sub> 存放计数初值的低8位

## 4种工作方式(方式0~方式3)



T/C有四种工作方式，对应四种电路结构。其中，方式0—2都是一样的，方式3不一样。

## 6.2 定时器/计数器的控制

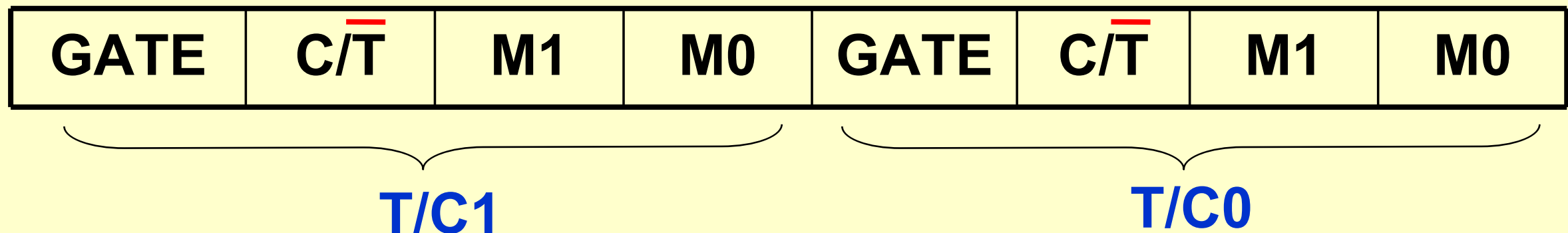
定时器/计数器的工作由TCON和TMOD控制

由软件把控制字写入TCON和TMOD，用来设置T/C0和T/C1的工作方式和控制功能。

当系统复位时，TCON和TMOD所有位都被清0。

### 一、工作方式寄存器TMOD (89H) (不可位寻址)

TMOD用于控制T/C0和T/C1的工作模式，其各位的定义格式如下：



其中，低4位用于T/C0，高4位用于T/C1。各位功能如下：



8位分为两组，高4位控制T1，低4位控制T0。

## (1) GATE——门控位

**0:** 以TRi (i=0,1) 来启动定时器/计数器运行。

**1:** 用外中断引脚( $\overline{INT0}$ 或 $\overline{INT1}$ )上的高电平和TRi来启动定时器/计数器运行。

## (2) M1、M0——工作方式选择位

M1	M0	工 作 方 式
0	0	方式0，13位定时器/计数器。
0	1	方式1，16位定时器/计数器。
1	0	方式2，8位常数自动重新装载
1	1	方式3，仅适用于T0，T0分成两个8位计数器，T1停止计数。

## (3) C/ $\overline{T}$ ——计数器模式和定时器模式选择位

## GATE 门控位（控制启动方式）

- GATE = 0时，T/C的启动只要用软件使TCON中的TR0或TR1为1，而不管/INT0（或/INT1）的电平是高还是低；（一个启动条件）
- GATE = 1时，要用软件使TR0或TR1为1，同时外部中断/INT0(或/INT1) 引脚为高电平，才能启动T/C工作，（两个启动条件），也就是T/C的启动受双重控制。

注意：TMOD不能位寻址，只能由字节设置T/C的工作方式，  
低半字节设定T/C0，高半字节设定T/C1。

## 二、控制寄存器 TCON (88H)

控制寄存器 TCON除可**字节寻址**外，各位还可**位寻址**。  
各位定义及格式如下：

8FH	8EH	8DH	8CH	8BH	8AH	89H	88H
<b>TF1</b>	<b>TR1</b>	<b>TF0</b>	<b>TR0</b>	<b>IE1</b>	<b>IT1</b>	<b>IE0</b>	<b>IT0</b>

1) TF1 (TCON.7) T/C1的溢出标志位。

当T/C1被允许计数后，T/C1从初值开始加1计数，回零时由高位产生溢出，由硬件自动置TF1为“1”，并向CPU请求中断，当CPU响应并进入中断服务程序后，TF1又被**硬件自动清0**。TF1也可以由程序查询和清0。

2) TF0 (TCON.5) T/C0溢出标志位。其功能同TF1。

3) TR1 (TCON.6) T/C1运行控制位。

由软件置位或复位。当GATE (TMOD.7) 为0时,

**?** TR1 = 1时, 启动T/C1计数;

**?** TR1 = 0时, 停止T/C1计数。

4) TR0 (TCON.4) T/C0运行控制位。

其功能同TR1, 只是GATE 为TMOD.3。

5) IE1、IT1、IE0、IT0 (TCON.3~TCON.0)

外部中断 $\overline{\text{INT1}}$ 、 $\overline{\text{INT0}}$ 的中断标志位和申请信号的  
触发方式控制位。前面第五章讲过。

## 6.3 定时器/计数器T0和T1的工作方式

🔊 TMOD中控制位C/ $\bar{T}$ 设置定时或计数功能

🔊 M1、M0位的设置选择四种工作方式，

方式0、方式1、方式2和方式3

❓ 方式0(T0、T1)

❓ 方式1(T0、T1)

❓ 方式2(T0、T1)

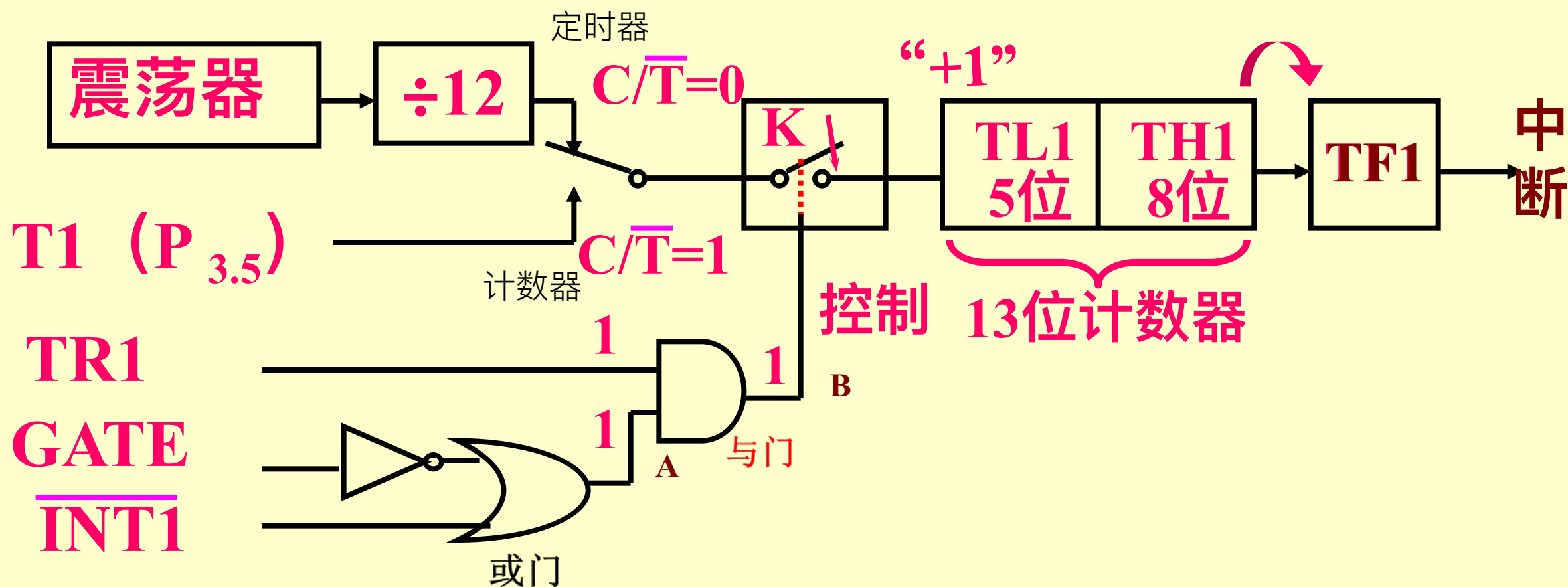
❓ 方式3(T0)

# (1) 方式0 M1M0=00

$TL4 \sim 0$

13位的定时计数器，由TH的8位和TL的低5位组成

以T1为例：



启动控制

$$B = TR1 \cdot A = TR1 \cdot (\overline{INT1} + \overline{GATE})$$

# 启动控制

$$B = TR1 \bullet A = TR1 \bullet (\overline{INT1} + \overline{GATE})$$

GATE	TR <sub>x</sub>	$\overline{INT_x}$	启动情况
0	0	X	停止
0	1	X	启动定时/计数
1	0	X	停止
1	1	┐	启动定时/计数
1	1	┘	停止

其中： 1 表示高电平，0表示低电平，X表示任意状态

**GATE=0**时，仅由TR1来控制开关，是否启动计数；

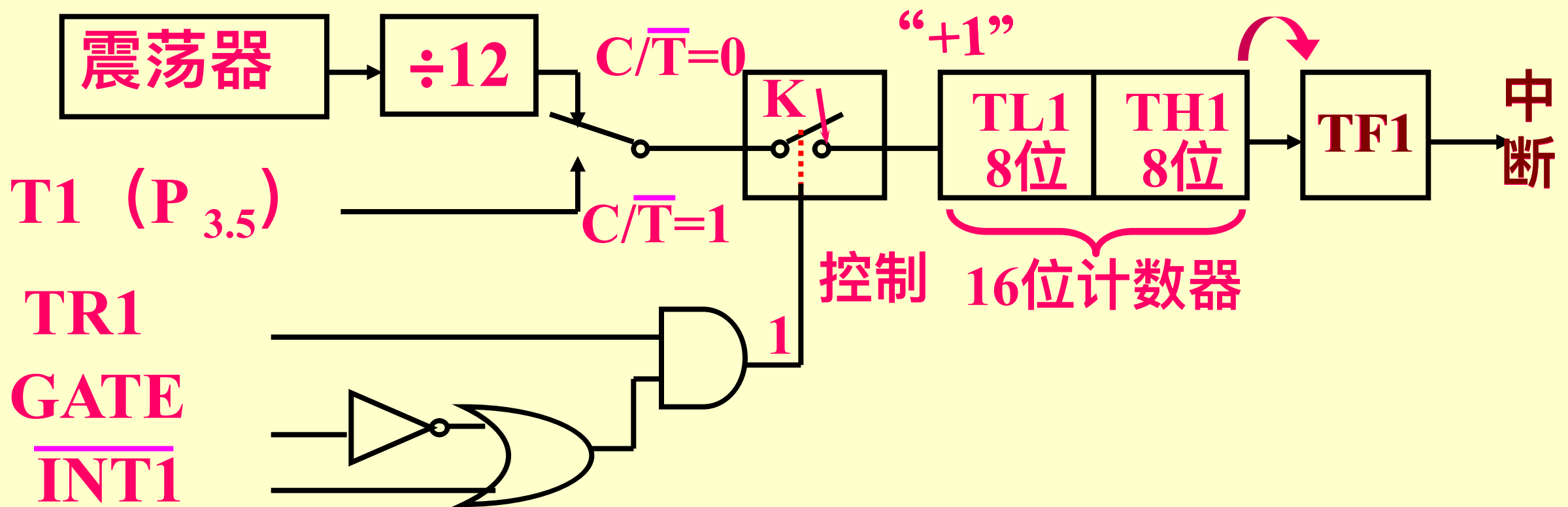
**GATE=1**时，由外中断引脚信号控制或门的输出，此时控制与门的开启由外中断引脚信号和TR1共同控制。当TR1=1时，外中断引脚信号的高电平启动计数，外中断引脚信号的低电平停止计数。

实用：利用GATE位测正脉冲的宽度

## (2) 方式1 M1M0=01

16位的定时计数器，由TH的8位和TL的8位组成

以T1为例：



方式1的结构几乎与方式0完全一样，唯一的差别：方式1中的TH1(TH0)和TL1 (TL0) 均是8位的，构成16位计数器。



## 定时器(方式1)应用程序举例：

例1： 要求对T0产生100mS定时进行初始化。  
(晶振=6MHz)

分析： 已知 $f_{osc} = 6\text{MHz}$  则：

(机器周期)  $1T_m = 12T_c = 12/6\text{MHz} = 2\mu\text{S}$

$$100\text{mS} \div 2 \mu\text{S} = 50000$$

16位定时器最大数值为：

$$2^{16} = 65536 (=0\text{FFFFH}+1)$$

故选择方式1工作可以满足要求。

计算初值：  $65536 -$

$$50000 = 15536 = 3\text{CB0H}$$

由分析得知：T0选择方式1，初值=3CB0H

定时器方式								
寄存器TMOD	GATE	C / $\overline{T}$	M1	M0	GATE	C / $\overline{T}$	M1	M0
	X	X	X	X	0	0	0	1

初始化：MOV TMOD, #01H ;选 T0 方式 1

MOV TH0, #3CH ;赋初值高8位

MOV TL0, #0B0H ;赋初值低8位

SETB TR0 ;启动 T0定时

若需要定时器0产生中断还应当写如下语句：

SETB ET0 ;开T0中断允许

SETB EA ;开总中断允许

以及相应的中断服务程序。

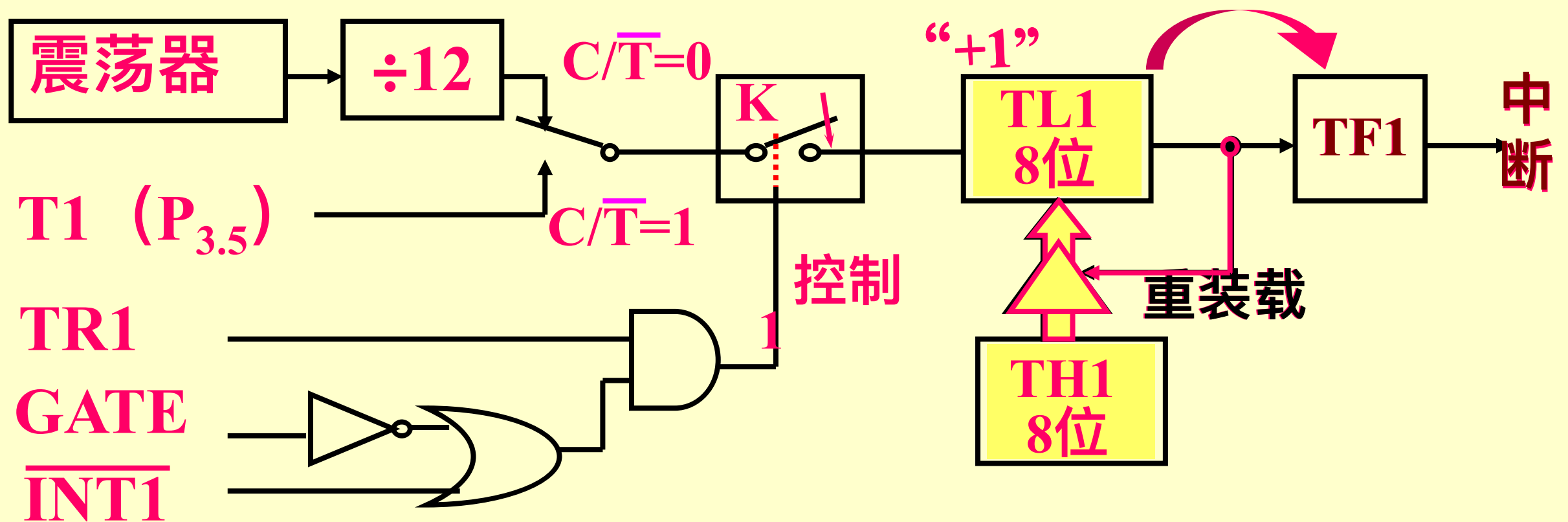
### (3) 方式2 M1M0=10

可自动重载的8位计数器

TL1 (TL0) 被定义为计数器

TH1 (TH0) 被定义为赋值寄存器

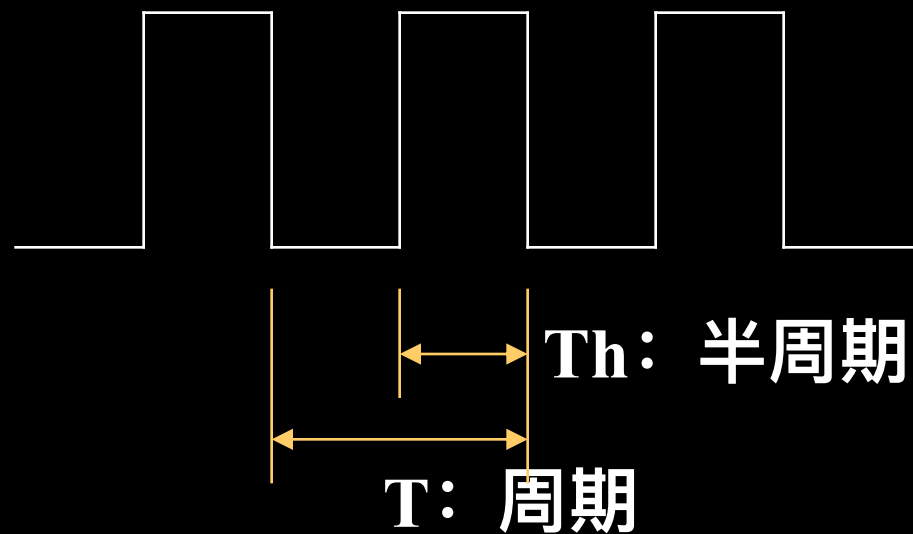
以T1为例:



编程时必须给TH1 (或TH0) 和TL1 (或TL0) 置入相同的初值。

## 定时器（方式2）应用程序举例：

- 例2:从P1.0 脚输出频率=1KHz方波。  
设：晶振=6MHz。利用T1定时中断。



分析：fosc = 6MHz

1机器周期 = 2  $\mu$ S

1KHz方波周期 = 1 ms

半个方波周期 = 500  $\mu$ S

500  $\mu$ S  $\div$  2  $\mu$ S = 250

若选择方式2 工作,8位定时器最大数值为：

$2^8 = 256 = 0FFH + 1$

可以满足要求。

计算初值：256 - 250 = 6



ORG 0000H

AJMP MAIN

ORG 001BH ; T1的中断矢量

CPL P1.0 ; 中断服务: P1.0取非

RETI ; 中断返回

ORG 1000H

MAIN: MOV TMOD, #20H ; 选T1方式2

MOV TH1, #6 ; 赋重装值

MOV TL1, #6 ; 赋初值

SETB ET1 ; 开T1中断

SETB EA ; 开总中断

SETB TR1 ; 启动T1

HERE: AJMP HERE ; 原地等待中断

END

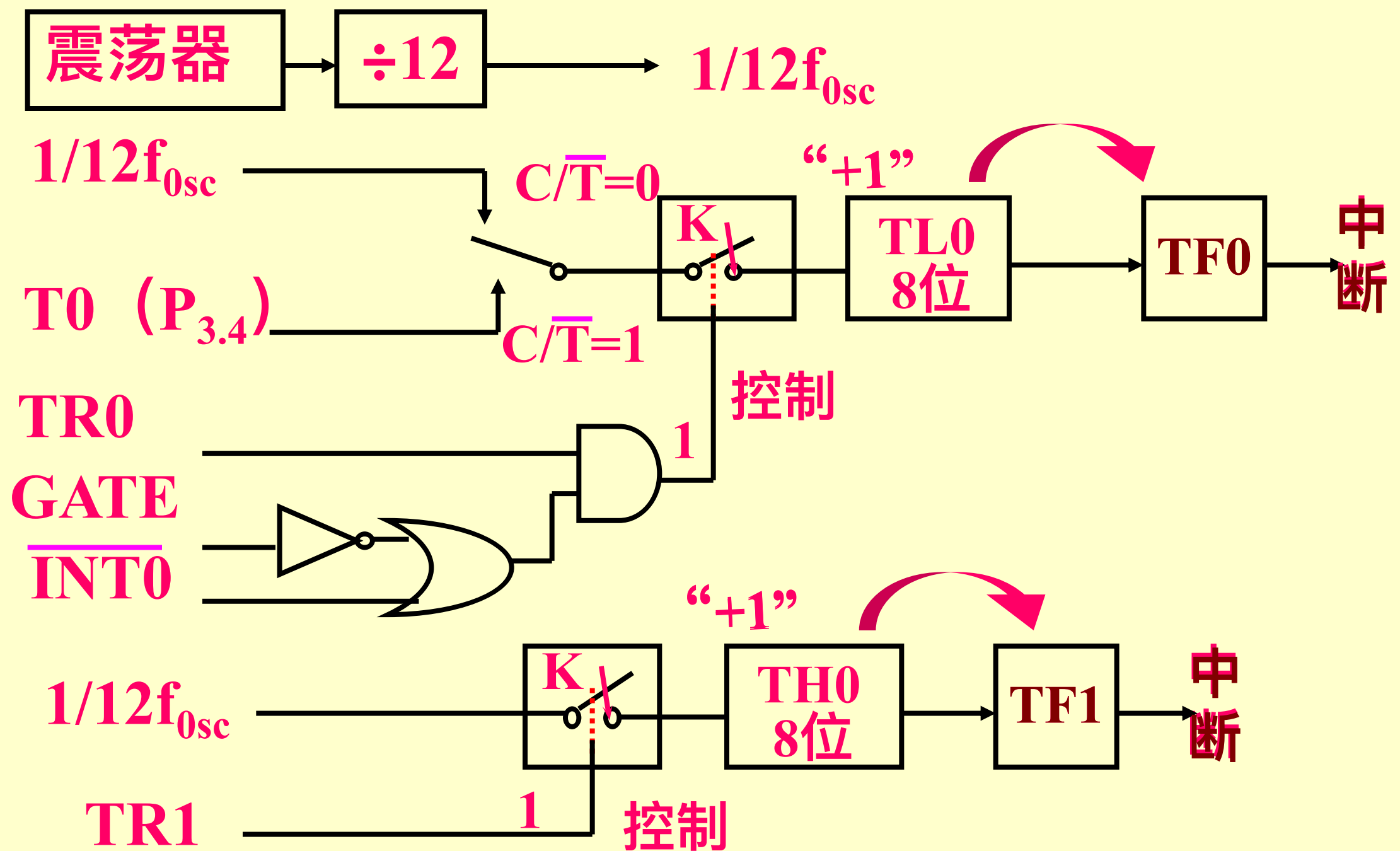
初始化

#### (4) 方式3 $M1M0=11$

T0被分成2个相互独立的8位计数器TL0、TH0

**TL0**使用自己本身的一些控制位C/ $\bar{T}$ 、GATE、TR0、TF0、 $\overline{INT0}$ ;

**TH0**只能做定时器，并使用T1的控制位TR1、TF1，同时占用T1的中断源，其启动/关闭仅受TR1控制。



*TH0借用了T1的TR1和TF1,因此控制了T1的中断  
此时T1只能用在一些不要中断的情况下*

在T/C0工作在方式3时，T/C1仍可设置为方式0~2。

由于TR1和TF1已被T/C0（TH0）占用，计数开关已被接通，此时仅用T/C1的C/ $\overline{T}$ 来切换其定时或计数工作方式就可使T/C1工作。

计数器（8位、13位或16位）回零溢出时，只能将输出送入串行口或用于不需要中断的场合。

一般情况下，当T/C1用作串行口波特率发生器时，T/C0才设置为工作方式3。

此时，常把T/C1设置为方式2用作波特率发生器。



### 3. 定时器/计数器的初始化

初始化一般有以下几个步骤：

- ① 确定工作方式，对方式寄存器TMOD赋值；
- ② 预置定时或计数初值，直接将其写入TH0、TL0或TH1、TL1中；
- ③ 根据需要，置位EA使CPU开放中断，同时置位ETx允许T/C中断，以开放或禁止定时器/计数器中断。IP设定中断优先级；
- ④ 给TCON送命令控制字，将TRi 赋值为“1”启动定时器/计数器。

## 计数初值的设定:

T/C的计数器是在计数初值的基础上加1计数的，当计数器回“0”时自动产生溢出，置位TF<sub>x</sub>中断标志，向CPU 提出中断请求。

**最大计数值M：不同的工作方式M值不同**

**方式0：**  $M = 2^{13} = 8192$

**方式1：**  $M = 2^{16} = 65536$

**方式2、3：**  $M = 2^8 = 256$

## 计数/定时初值X的计算方法:

### 计数方式:

$$X = M - \text{计数值} \quad (X \text{ 即为计数值的补码数})$$

### 定时方式:

$$(M - X) \times T = \text{定时时间}$$

$$\therefore X = M - \text{定时值} / T$$

其中T为机器周期，时钟的12分频，

若晶振为6MHz，则 $T = 2\mu s$ ，

若晶振为12MHz，则 $T = 1\mu s$

**例1** 若单片机晶振为12MHz，  
要求产生500 $\mu$ s定时，试计算X的初值。

**解：**由于T = 1 $\mu$ s，产生500 $\mu$ s定时，  
需要“+1” 500次，定时器方能产生溢出。

采用方式0: (13位计数器)

$$X = 2^{13} - (500 \times 10^{-6} / 10^{-6})$$

$$= 8192 - 500 = 7692 = 1E0CH \stackrel{\text{暂时写成}}{=} \text{F00CH}$$

$$\text{F00CH} = \begin{array}{|c|c|c|c|c|c|c|c|} \hline 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ \hline \end{array} \begin{array}{|c|c|c|c|c|c|c|c|} \hline \text{D7} & \text{D6} & \text{D5} & & & & & \\ \hline 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ \hline \end{array}$$

THi                      不用                      TLi

采用方式1: (16位计数器)

$$X = 2^{16} - (500 \times 10^{-6} / 10^{-6}) = 65036 = \text{FE0CH}$$

**例2：**选用定时器1，且CPU的工作频率为12MHz，则工作在方式1时，定时50ms所需要的定时初值为多少？并写出初始化代码。

解：计数脉冲周期  $T=1\ \mu s$ ，

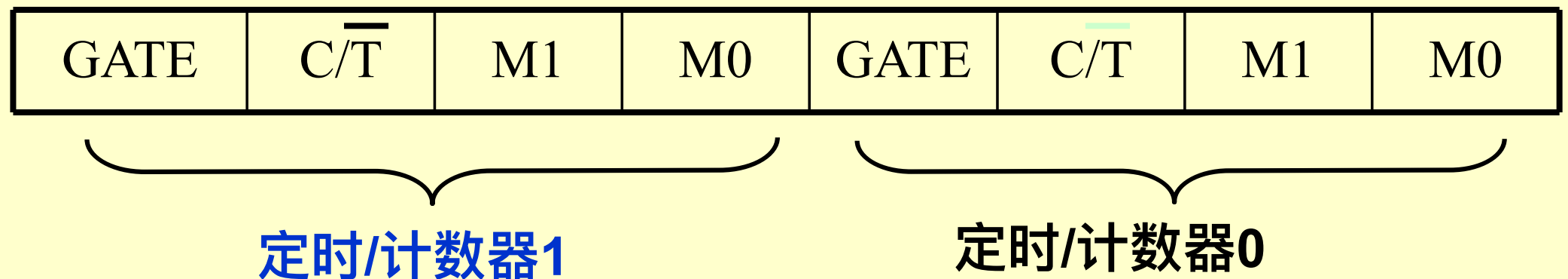
计数值= $50ms \times 10^3 / 1\ \mu s = 50000$ ， $M=2^{16}=65536$

需设置的定时初值= $65536-50000=15536=3CB0H$

# 初始化程序

```
MOV TMOD,#00010000B ; T1为方式1定时模式
MOV TH1,#3CH
MOV TL1,#0B0H ; 送T1的50ms的定时初值
SETB TR1 ; 启动T1定时器
```

模式寄存器TMOD



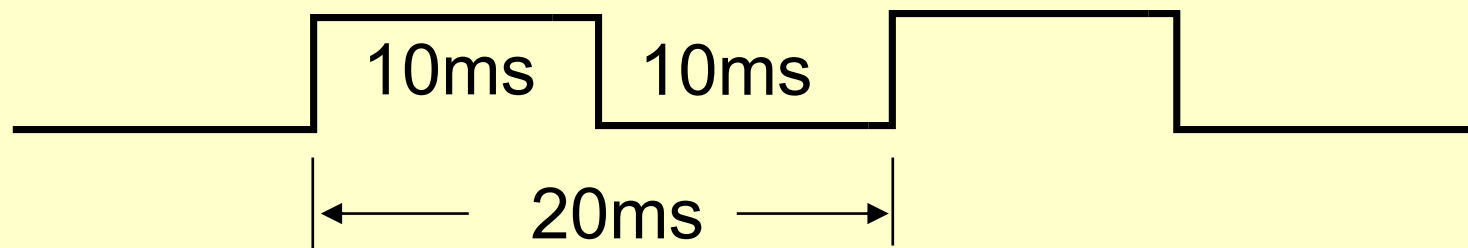
## 4. 定时器/计数器应用举例

### (1) 定时器应用

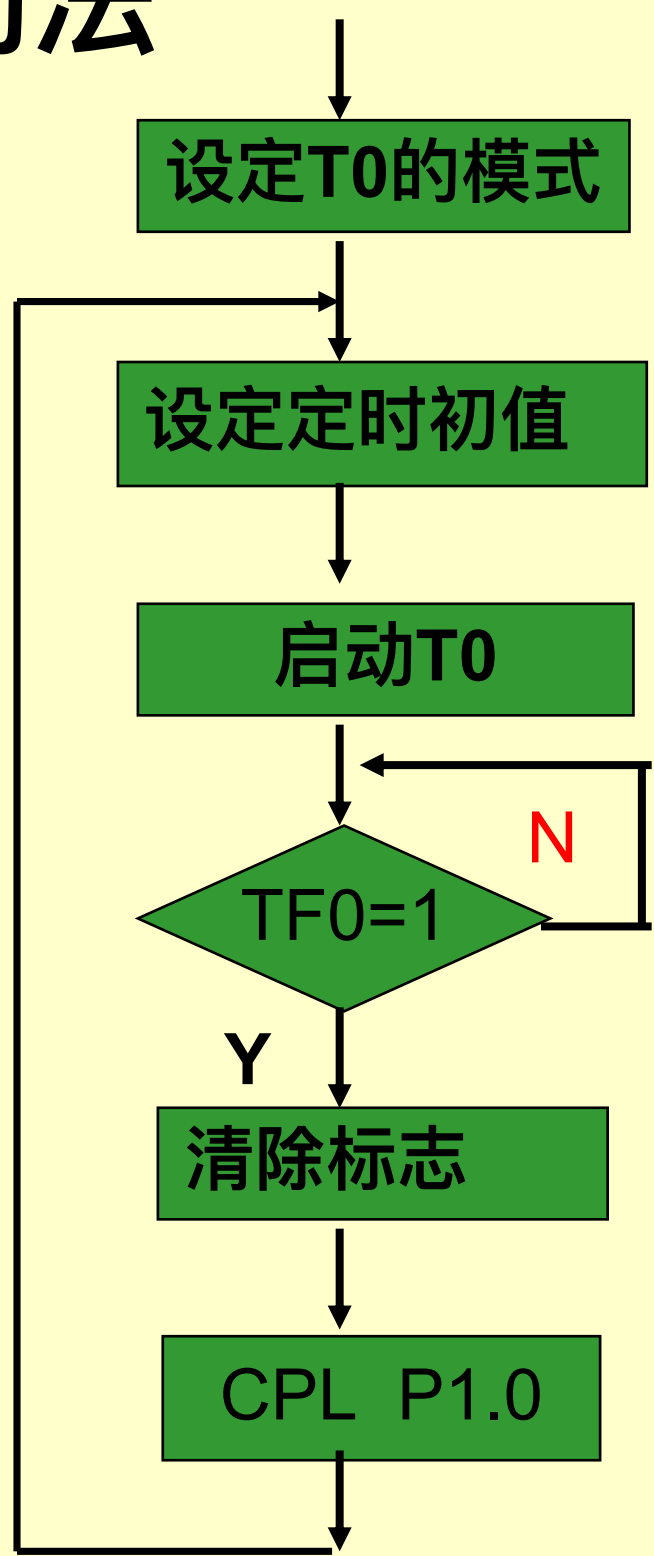
**例6-3：**利用单片机的P1.0输出一个频率为50Hz的对称方波。（若使用T0方式1，且CPU的时钟频率为12MHz，确定工作方式、模式和定时初值）

**解：**每秒产生50个方波，每个方波的周期=20ms。

计算出10ms所对应的定时初值为D8F0H。



# 查询法



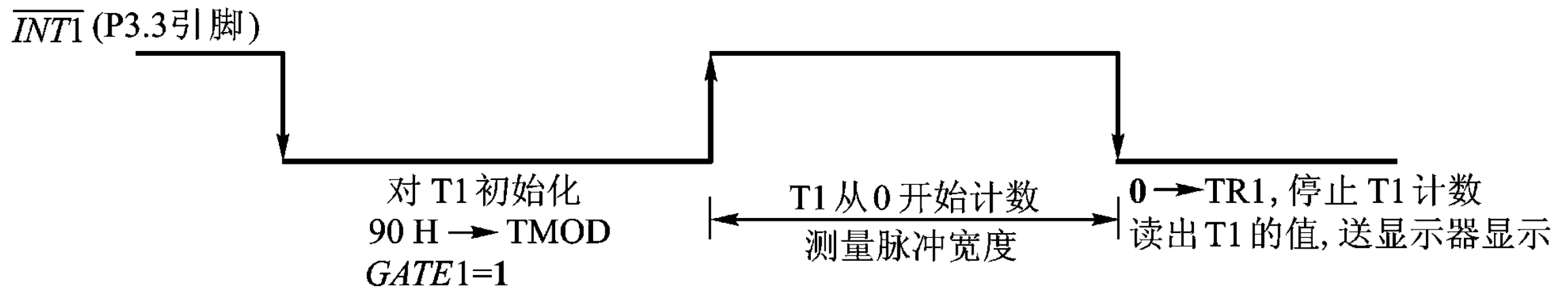
```
ORG 1000H
STRAT:MOV TMOD,#01H;
LOOP: MOV TH0,# 0D8H
      MOV TL0,# 0F0H
      SETB TR0
      JNB TF0, $
      CLR TF0
      CPL P1.0
      SJMP LOOP
END
```

GATE	C/ $\overline{T}$	M1	M0	GATE	C/ $\overline{T}$	M1	M0
------	-------------------	----	----	------	-------------------	----	----

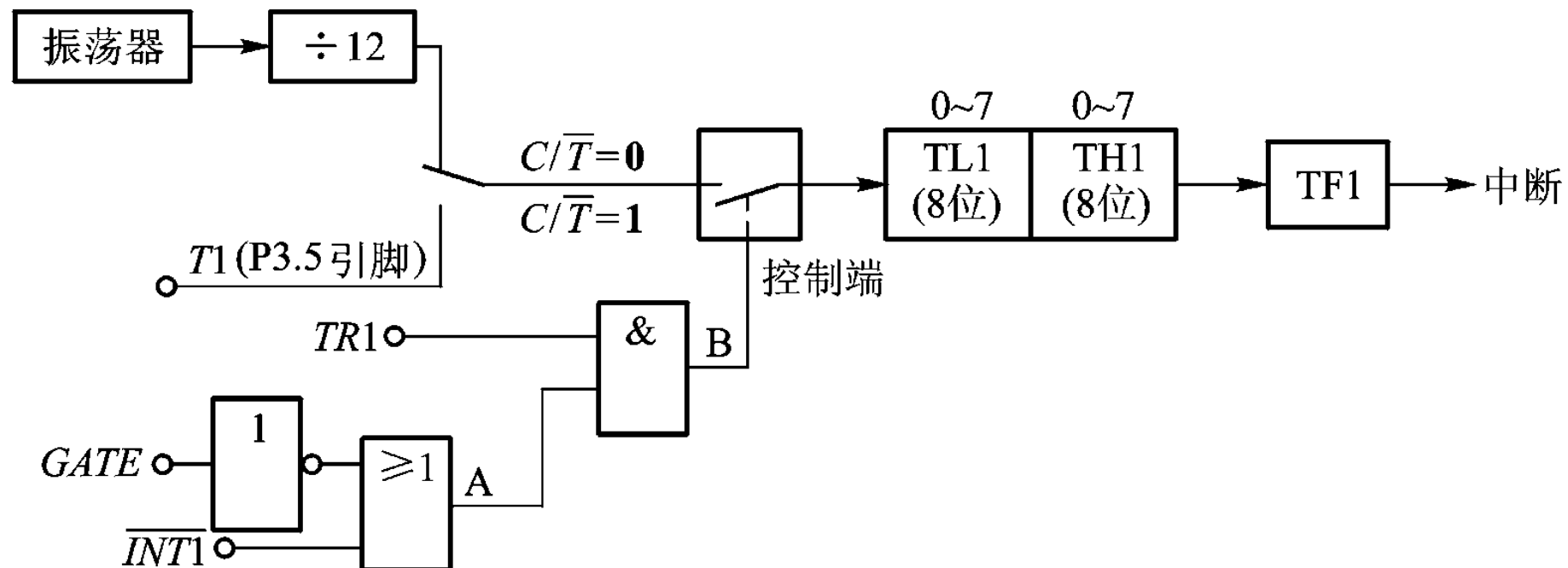


## (2) 门控信号GATE的应用—测量脉冲宽度

**例6-5: GATE1可使定时器/计数器T1的启动计数受INT1的控制, 可测量引脚INT1(P3.3)上正脉冲的宽度(机器周期数)。**



6-14



**ORG 0000H**

**RESET: AJMP MAIN ; 复位入口转主程序**

**ORG 0100H**

**MAIN: MOV SP,#60H**

**MOV TMOD,#90H ; T1为方式1定时控制字**

**MOV TL1,#00H**

**MOV TH1,#00H —**

**LOOP0: JB P3.3,LOOP0 — ÷ INT1高,则循环等待**

**SETB TR1 ; 如INT1为低,启动T1**

**LOOP1: JNB P3.3,LOOP1 ; INT1低,则循环**

**CLR      TR1                      ; 停止T1计数**

**MOV      A,TL1    ; T1计数值送A**

**将A中的T1计数  
值送显示缓冲区  
转换成显示的代码**

**LOOP3: LCALL DIR                      ; 调用显示子程序**

**DIR  
数值**

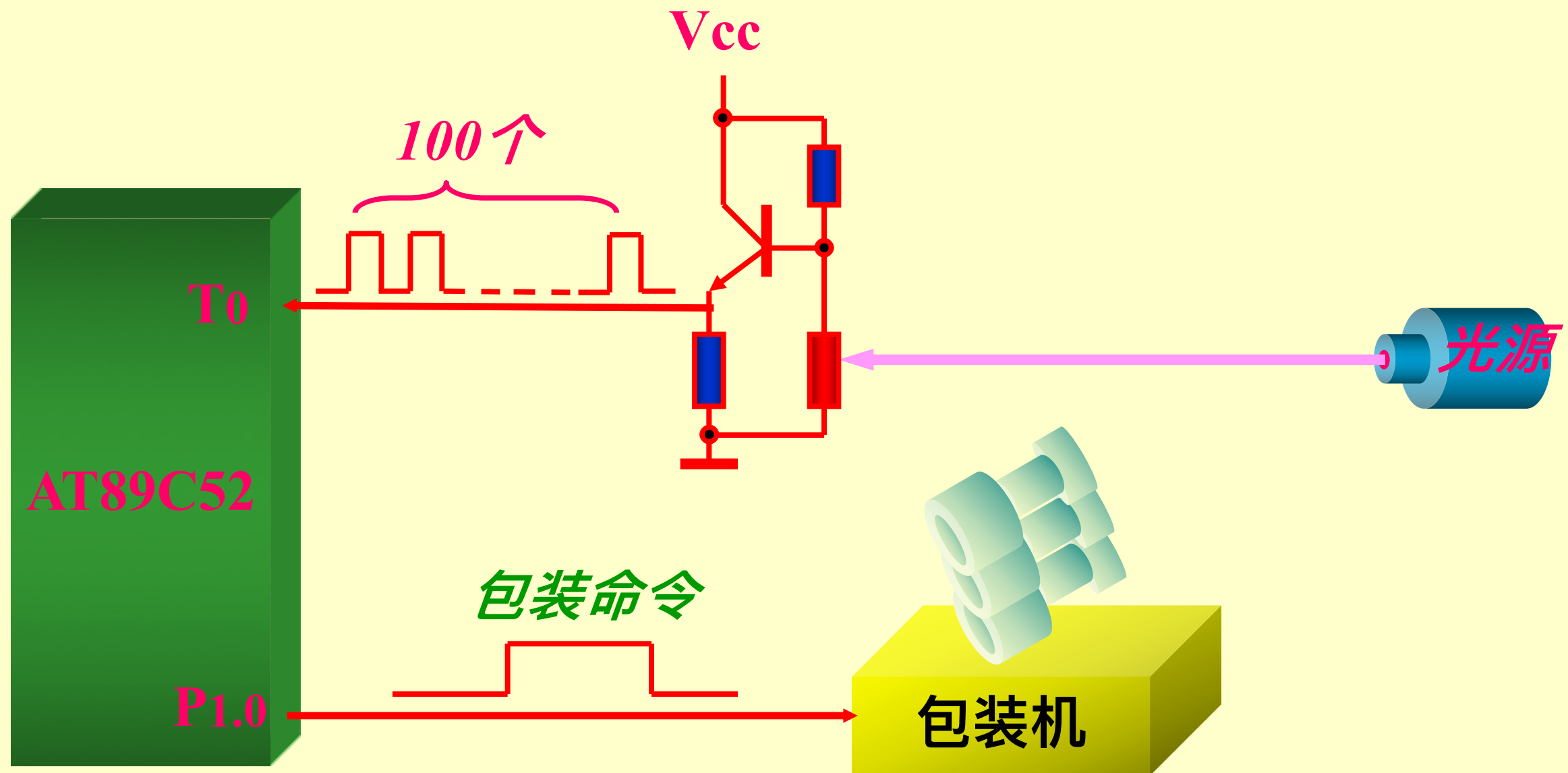
**; 显示T1计**

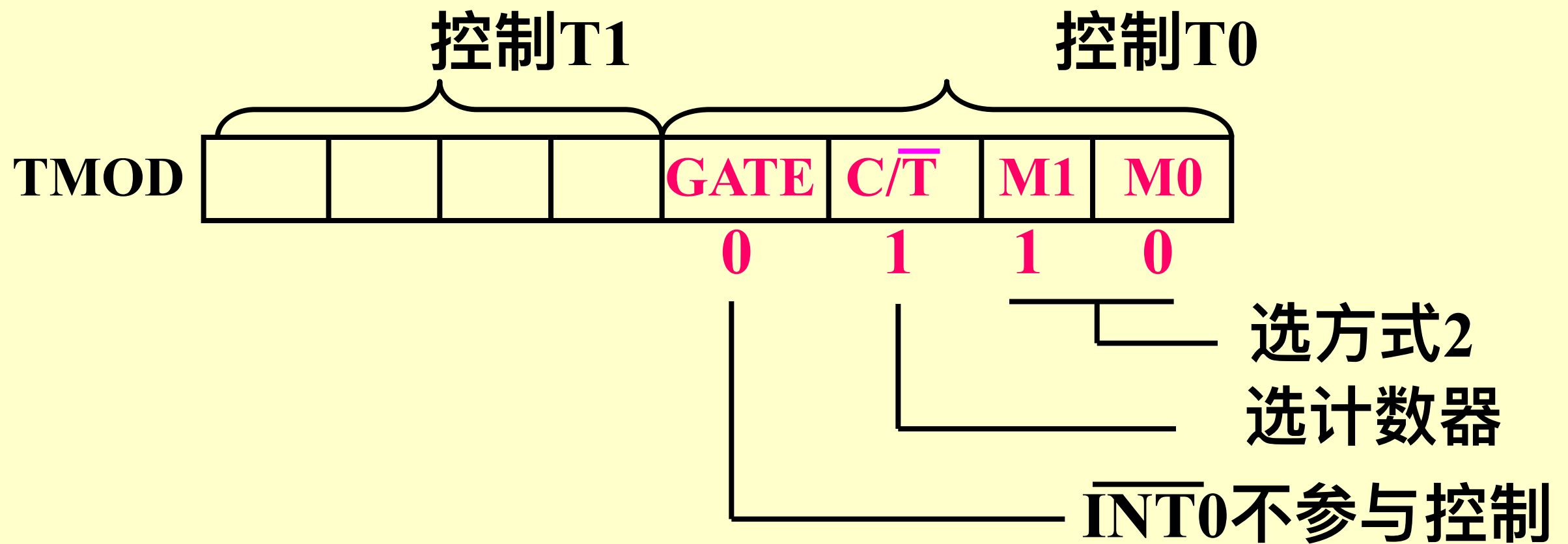
**AJMP LOOP3            ;**

**执行程序，使INT1引脚上出现的正脉冲宽度以机器周期数的形式显示在显示器上。**

### (3) 计数器应用

**例6-7：** 用T0监视一生产流水线，每生产100个工件，发出一包装命令，包装成一箱，并记录其箱数。





- (1) 方式字TMOD = 06H
- (2) 计数初值X = M - 64H = 9CH
- (3) 用P1.0启动外设发包装命令
- (4) 用R5R4作箱数计数器

## 程序:

<b>MOV</b>	<b>P1.0, #0</b>	<b>;P1.0为低</b>
<b>MOV</b>	<b>R5, #0</b>	<b>;</b>
<b>MOV</b>	<b>R4, #0</b>	<b>;箱数计数器清“0”</b>
<b>MOV</b>	<b>TMOD, #6</b>	<b>;置T0工作方式</b>
<b>MOV</b>	<b>TH0, #9CH</b>	<b>;</b>
<b>MOV</b>	<b>TL0, #9CH</b>	<b>;计数初值送计数器</b>
<b>SETB</b>	<b>EA</b>	<b>;CPU开中断</b>
<b>SETB</b>	<b>ET0</b>	<b>;T0开中断</b>
<b>SETB</b>	<b>TR0</b>	<b>;启动T0</b>
<b>SJMP</b>	<b>\$</b>	<b>;模拟主程序</b>
<b>ORG</b>	<b>000BH</b>	<b>;T0中断入口</b>
<b>AJMP</b>	<b>COUNT</b>	<b>;转向中断服务</b>

## 中断服务:

```
COUNT:  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      ;启动外设包装
        MOV    R3, #100  ;
DLY:    NOP              ;给外设足够时间
        DJNZ   R3, DLY   ;延时
        CLR    P1.0      ;停止包装
        RETI             ;中断返回
```

# 扩大计数个数和定时时间

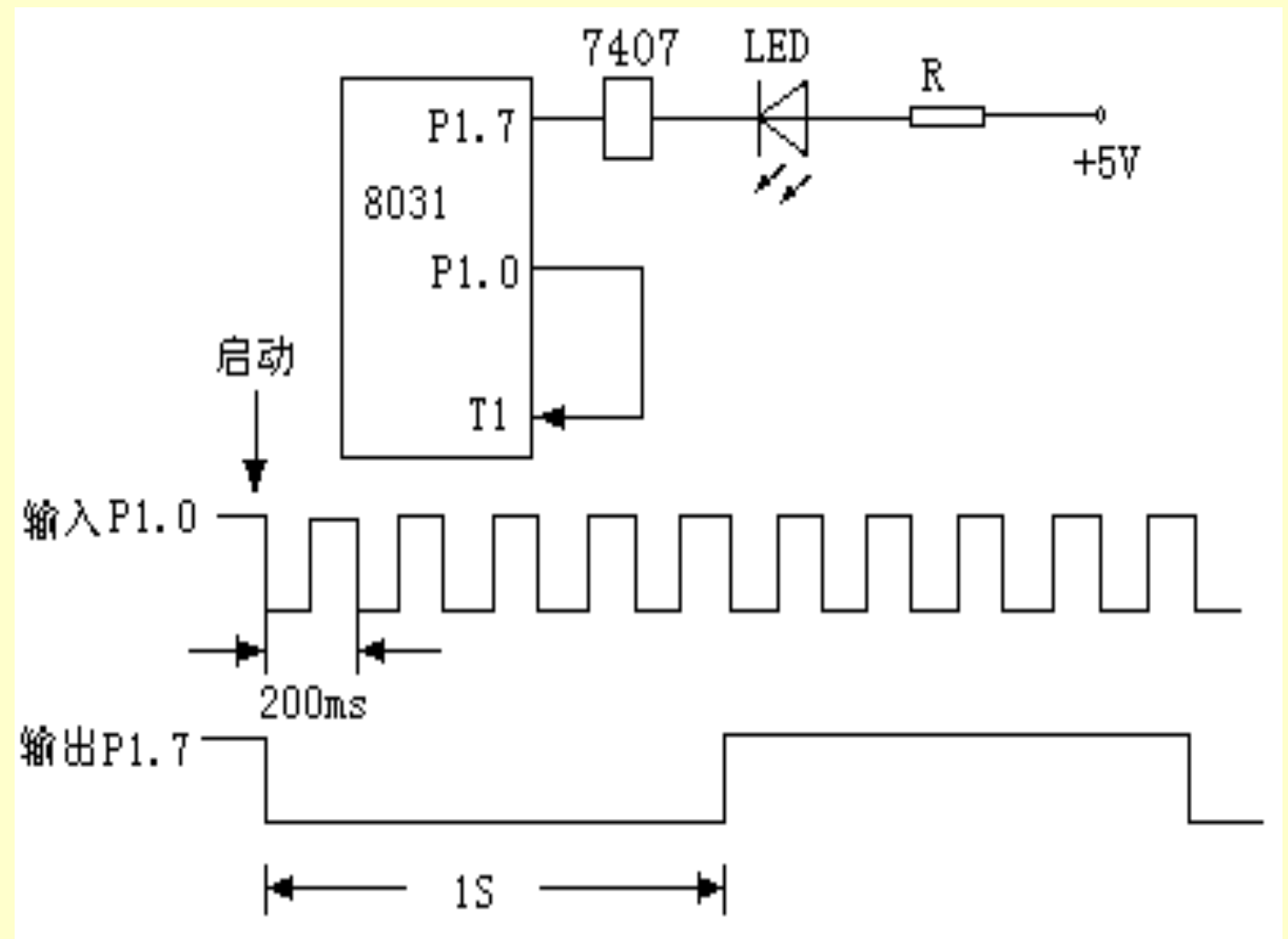
[例] 利用MCS-51单片机的T/C，产生电子时钟的1S基时，并且由P1.7输出2S的方波。 $F_{osc} = 12\text{MHz}$ 。

**焦点：**

16位计数器最大定时65.536ms，要产生比这个时间长的定时，怎么办？

1. 两个定时器级联

2. 硬件定时加软件计数相结合。





## 两个T/C级联的方法：

一个T/C定时，回0溢出时，使P1.0输出一个负脉冲送到另一个T/C的外部脉冲输入端用以计数。

当 $f_{osc}=12\text{MHz}$ 时，最大时间可以达到：

$$T = (65536 \times 65536) \times 1\mu\text{s}$$

$$= 4294967296\mu\text{s}$$

$$= 4294967.296\text{ms}$$

$$= 4294.967296\text{S}。$$

若再与软件计数相结合,会产生更长的时间。

**本讲结束！**