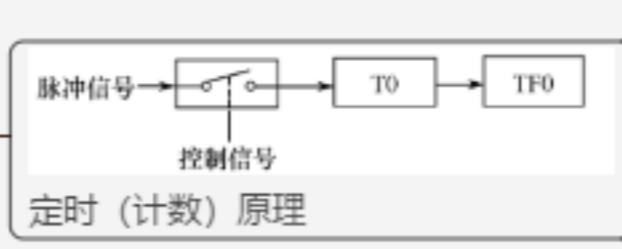


第6章 单片机的定时/计数器

定时/计数器的工作原理

- 需要进行定时控制的情况 如定时输出、定时检测、定时扫描等；
- 需要对外部事件进行计数的情况，如统计按键次数、脉冲个数等。



定时(计数)原理

在启动计数之前将TFx清0，并将一个称为计数初值a的整数先置入加1计数器T0中，则当观察到TFx为1时表明已经加入了(2^n - a)个脉冲，如此便能计算出脉冲的到达数量了。

若上述脉冲信号是来自单片机内部的时钟信号，则由于单片机的振荡周期非常精准，故而溢出时统计的脉冲数便可换算成定时时间，因此可作为定时器使用。

$$t = (\text{计数器满计数值} - \text{计数初值}) \times \text{机器周期}$$
$$= (2^n - a) \cdot \frac{12}{f_{osc}}$$

在时钟频率fosc和计数器容量n一定的情况下，定时时间t与计数初值a有关，计数初值越大，定时时间越短。

N = 计数器满计数值 - 计数初值 = 2^n - a
计数统计值N与计数初值a有关，计数初值越大，计满次数就越短。

基本原理

结构组成

- T0和T1: T0由TH0和TL0组成，T1由TH1和TL1组成。
- TMOD和TCON: TMOD是定时方式控制寄存器，由它确定定时/计数器的工作方式和功能；TCON是定时控制寄存器，用于管理T0和T1的启停、溢出和中断。

定时/计数器的控制

- TMOD寄存器: TMOD的低4位为T0的方式字，高4位为T1的方式字；C/T为定时/计数功能选择位，=0时为定时方式，=1时为计数方式；GATE为门控位: 当GATE=1时，只有TR0和/INT0都为高电平时，才能启动启动T0；当TR0=1和/INT0为低电平时，才能停止T0 (用来测量在/INT0引脚出现的正脉冲的宽度)；当GATE=0时，则只要TR0=1就能使T0启动，TR0=0就能使T0停止，而与/INT0的状态无关 (允许TR0启动计数器)；M1和M0: 工作方式选择位，可以设置4种工作方式。
- TCON寄存器: TR1和TR0: T1和T0的运行控制位，其他位都是中断有关的位；系统复位时，TCON初值为0，即默认的设置: TR0和TR1均为关闭状态。

定时/计数器的工作方式

- 方式1 (M1M0=01时): 由高8位THx和低8位TLx组成一个16位的加1计数器，满计数值M=2^16；C/T=0时为定时器方式，当时钟频率为12MHz，方式1的定时范围为1~65536μs；C/T=1时为计数器方式，计数范围为1~65536个脉冲；实例6.1 定时方式1应用: 定时器初始化: a = 2^16 - t * f_osc / 12; TH0 = (2^16 - t * f_osc / 12) / 256; TL0 = (2^16 - t * f_osc / 12) % 256; 直接赋值; TMOD、TH1、TH0、TR1; 查询法编程: 利用语句do{ } while(!TF1); 查询等待TF1置1; 循环体中不断装载计数初值，复位TF1; 中断法编程: 中断初始化: EA、ET1; 系统自动检查TF1，中断响应函数中装载计数初值，自动复位TF1。

- 方式2 (M1M0=10时): TLx作为加1计数器，满计数值为2^8，THx用以存放8位初值；若TL1计数溢出，TH1会自动将其初值重新装入TL1中。重新装入的过程不改变TH1中的值；方式2可产生非常精确的定时时间，尤其适合于作为串行口波特率发生器。

- 实例6.2 定时方式2应用: 计算初值: a = 2^8 - t * f_osc / 12; 采用中断法检测溢出，中断函数中无需装载计数初值。

- 实例6.3 计数方式2应用: 电路修改: 按键改由T0 (P3.4) 引脚接入; 编程要点: 将T0设置为计数器方式2，计数初值为0xff，1个外部脉冲就可产生中断请求。

- 方式0 (M1M0=00时): 采用低5位TLx和高8位THx组成一个13位的加1计数器，满计数值为2^13，初值不能自动重载；方式0的TLx中高3位是无效的，可为任意值，计算初值时必须特别留意。

- 方式3 (M1M0=11时): 可以组合出3种定时/计数器关系: ① TH0+TF1+TR1组成的带中断功能的8位定时器；② TL0+TF0+TR0组成的带中断功能的8位定时/计数器；③ T1组成的无中断功能的定时/计数器。

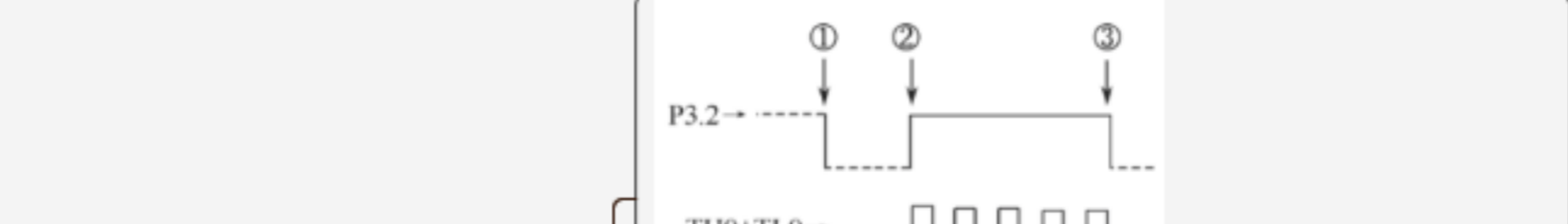
特点: 方式3下T0可有两个具有中断功能的8位定时器;在定时器T0用作工作方式3时，T1仍可设置为工作方式0~2

定时/计数器的应用实例

- 实例6.5 产生同步脉冲信号: 编程要点: ① 采用T0计数方式2对P3.4的外部脉冲进行计数，选择初值0xff (=256-1) 使其一次脉冲即产生T0溢出，如此监测P3.4引脚负脉冲的出现；② 当TF0=1时改用T0定时方式2进行定时操作，选择初值0x06 (=256-500*6/12) 使其产生500μs定时，同时使P3.0输出低电平；③ 当TF0再次为1时，使P3.0输出高电平，再改用T0计数方式2，如此往复进行。

- 实例6.6 产生占空比信号: 编程要点: 采用定时与软件计数联合的办法，将250次的10ms定时累积起来，便可达到2.5s的定时效果；先对代表中断次数的全局变量进行累加，然后进行超限判断，并根据判断结果调整输出电平或变量清0。

- 实例6.7 定时中断控制的流水灯: 编程要点: 为避免中断函数内执行的任务过多，耗时太长，总体定时准确性将下降的问题，采用了在中断函数中仅做中断次数统计和计数初值重入，而将流水灯数据输出改在主函数中进行的新方案。



实例6.8 脉冲宽度测试: 编程要点: 利用查询方式找到①点的出现时刻→利用/INT0信号的上升沿在②点启动T0定时方式1→利用/INT0信号的下降沿在③点中止T0定时→取出反映了脉冲宽度的T0计数值。

- 在C51中进行单片机片内存储器操作的方法是，定义指针变量并赋值地址→按指针变量对片内存储器数据进行读/写操作；
- 十六进制数转BCD码的方法是: 从最低位开始进行模10计算→删去最末位(相当于整除10)→继续模10计算，直至整除10的结果为0。