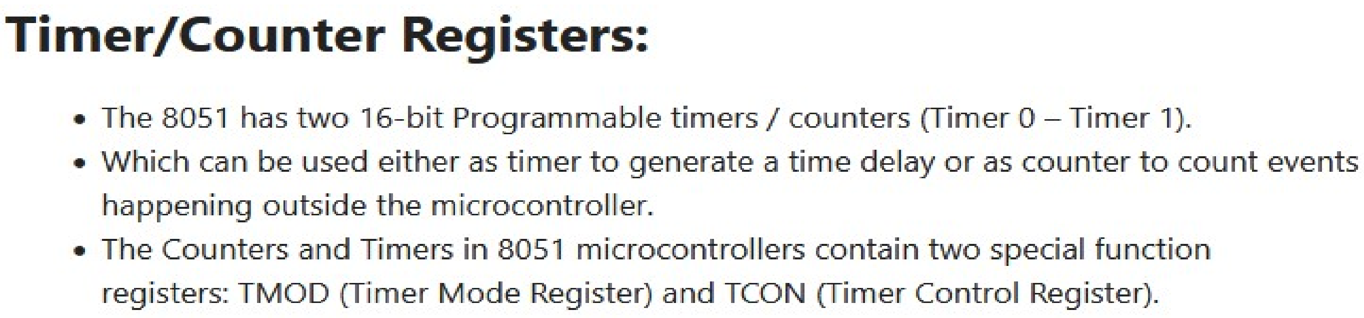
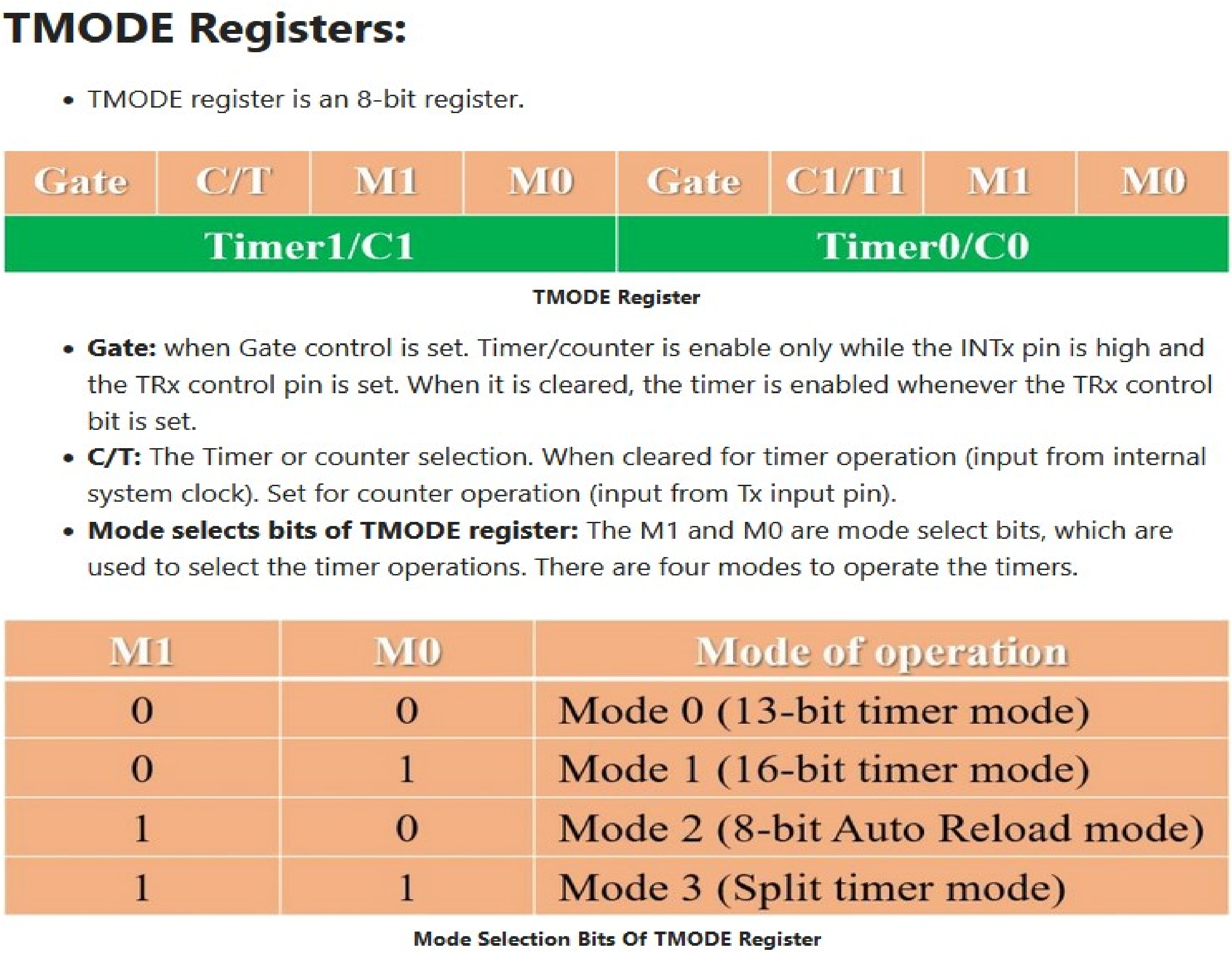
Module-2-Timers



Timer/Counter Registers:

- The 8051 has two 16-bit Programmable timers / counters (Timer 0 – Timer 1).
- Which can be used either as timer to generate a time delay or as counter to count events happening outside the microcontroller.
- The Counters and Timers in 8051 microcontrollers contain two special function registers: TMOD (Timer Mode Register) and TCON (Timer Control Register).



TMODE Registers:

- TMODE register is an 8-bit register.

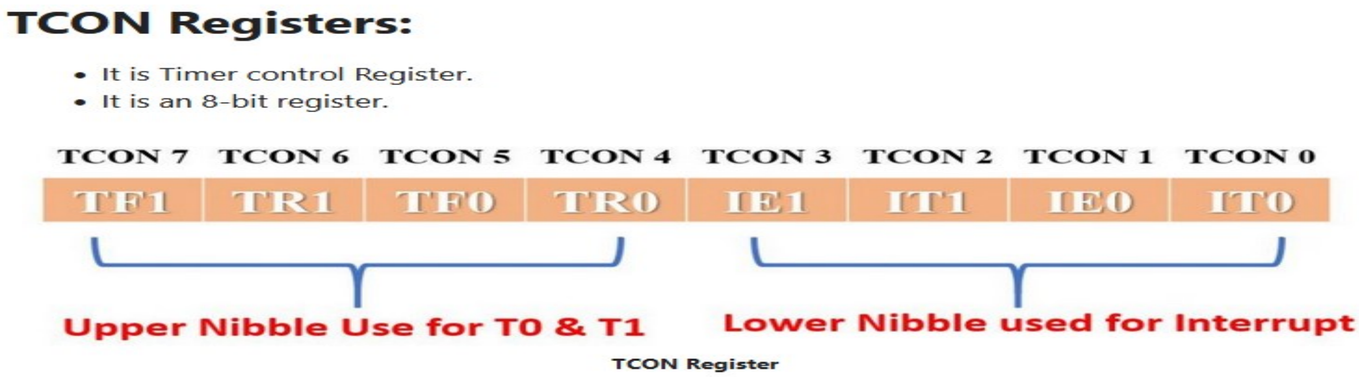
Gate	C/T	M1	M0	Gate	C1/T1	M1	M0
Timer1/C1				Timer0/C0			

TMODE Register

- **Gate:** when Gate control is set. Timer/counter is enable only while the INTx pin is high and the TRx control pin is set. When it is cleared, the timer is enabled whenever the TRx control bit is set.
- **C/T:** The Timer or counter selection. When cleared for timer operation (input from internal system clock). Set for counter operation (input from Tx input pin).
- **Mode selects bits of TMODE register:** The M1 and M0 are mode select bits, which are used to select the timer operations. There are four modes to operate the timers.

M1	M0	Mode of operation
0	0	Mode 0 (13-bit timer mode)
0	1	Mode 1 (16-bit timer mode)
1	0	Mode 2 (8-bit Auto Reload mode)
1	1	Mode 3 (Split timer mode)

Mode Selection Bits Of TMODE Register



TCON Registers:

- It is Timer control Register.
- It is an 8-bit register.

TCON 7	TCON 6	TCON 5	TCON 4	TCON 3	TCON 2	TCON 1	TCON 0
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

Upper Nibble Use for T0 & T1

Lower Nibble used for Interrupt

TCON Register

Delay Function to Generate 1 ms Delay

In order to generate a delay of 1ms, the calculations using above steps are as follows.

- ✓ **Crystal frequency= 11.0592 MHz**
- ✓ I/P clock = 11.0592MHz
- ✓ F_{in} = 1/12x11.0592Mhz = 921.6 Khz
- ✓ T_{in} = 1/F_{in} =1/921.6x10^3 = 1.085μ sec

N = 1 msec / 1.085μ sec ≈ 922

Timer value = Maximum count-N

= 65536-922 = 64616

In hexa-decimal notation, 64616 = 1111 1100 0110 0110

= F C 6 6 = **FC66h**.

The following function will generate a delay of 1 ms using 8051 Timer 0.

```
Void delay ()
{
        TMOD = 0x01;    // Timer 0 Mode 1
        TH0= 0xFC;     //initial value for 1ms
        TL0 = 0x66;
        TR0 = 1;     // timer start
        while (TF0 == 0); // check overflow condition
        TR0 = 0;    // Stop Timer
        TF0 = 0;   // Clear flag
}
```

Solved-problem:

Write C language program to generate a square wave of 2 KHz frequency on P1.1 pin by using timer 0 and mode 1. Assume XTAL frequency is 11.0592 MHz.

Steps to generate a square wave using 8051

1. Set the output of any port on the 8051 to logic high.
2. Wait for some time.
3. Set the output of the same port to logic low.
4. Again wait for the same amount of time as you did earlier.
5. Loop around the same.

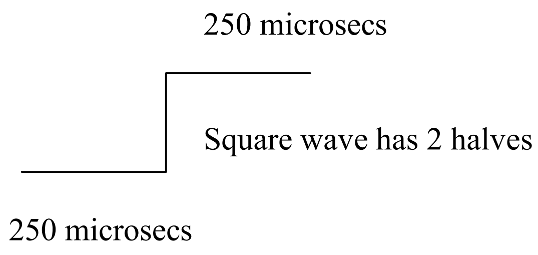
To obtain the desired frequency on the square wave, we have to manipulate with the delay. We know that the machine cycle frequency is 1/12 of the crystal oscillator frequency. So, with the crystal oscillator's frequency as **11.0592 Mhz** the machine cycle frequency is **921.6 Khz**. To sum up, that is equivalent to **1.085 µsecond**.

- ✓ **Crystal frequency= 11.0592 MHz**
- ✓ I/P clock = 11.0592MHz
- ✓ F_{in} = 1/12x11.0592Mhz = 921.6 Khz
- ✓ $T_{in} = 1/F_{in} = 1/921.6\text{x}10^3$ = 1.085µ sec

For 2 kHz square wave,

F_{out} = 2 KHz
$T_{out} = 1/2 \text{ X } 10^3$
T_{out} = 500µ sec

Consider half of it = T_{out} = 250µ sec
N = T_{out} / T_{in} = 250/1.085= 230



Timer value = Maximum count-N
= 65536-230 = **65306**

In hexa-decimal notation, 65306 = 1111 1111 0001 1010

= F F 1 A = **FF1A**.

Hence, TH0=0xFF

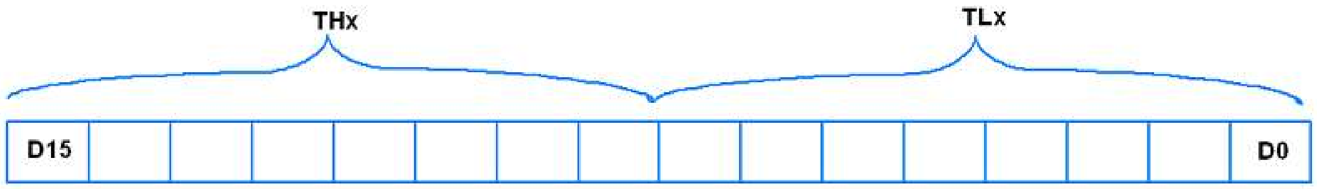
TL0=0x1A

C Program to generate Square Wave as per the given requirement

```
#include<reg51.h>
void delay(void);
sbit p=P1^5;
void main (void)
{
   while (1)
   {
       p=~p;
       delay();
   }
}
void delay()
{
    TMOD=0x01; //set timer 0 in mode 1 i.e. 16 bit number
    TL0=0x1AH; //load TL register with LSB of count
    TH0=0xFFH ; //Load TH register with MSB of count
    TR0 =1 //Start timer 0
    while(TF0==0) //wait until timer rolls over
    TR0=0; //Stop timer 0
    TF0=0; //Clear timer flag 0
}
```

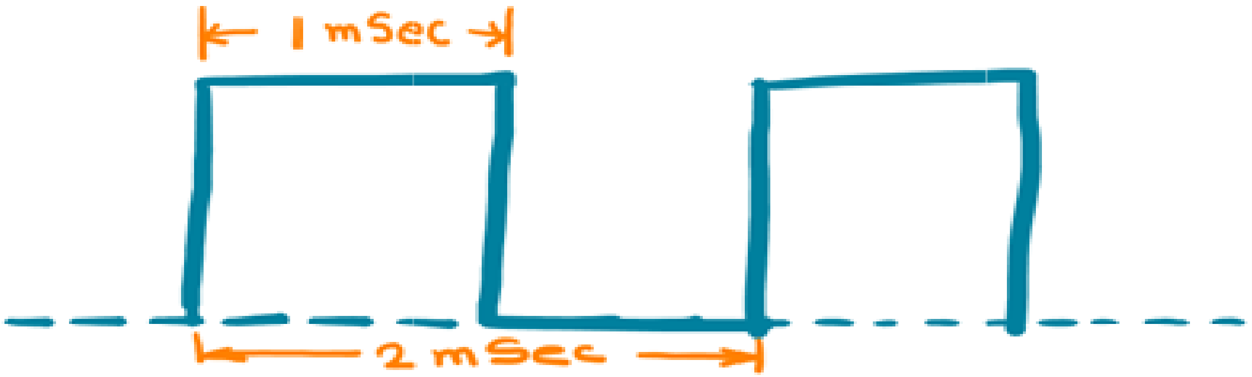
Mode1 (16-bit timer mode)

Mode 1 is a 16-bit timer mode used to generate a delay, it uses 8-bit of THx and 8-bit of TLx to form a total 16-bit register.



Example

Let's generate a square wave of 2msec time period using an AT89C51 microcontroller with timer0 in mode1 on the P1.0 pin of port1. Assume Xtal oscillator frequency of 11.0592 MHz.



As Xtal is 11.0592 MHz we have a machine cycle of 1.085usec.

Hence, the required count to generate a delay of 1msec. is,

Count =(1×10^(-3)) / (1.085×10^(-6)) ≈ 922

And mode1 has a max count is 2^16 (0 - 65535) and it increments from 0 to 65535 so we need to load value which is 922 less from its max. count i.e. 65535.

Hence value needed to be loaded is,

Value = (65535-Count) =64613= (FC65) hex

So we need to load FC65 Hex value higher byte in TH0 and lower byte in TL0 as, TH0 = 0xFC & TL0 = 0x65.

Programming steps for delay function

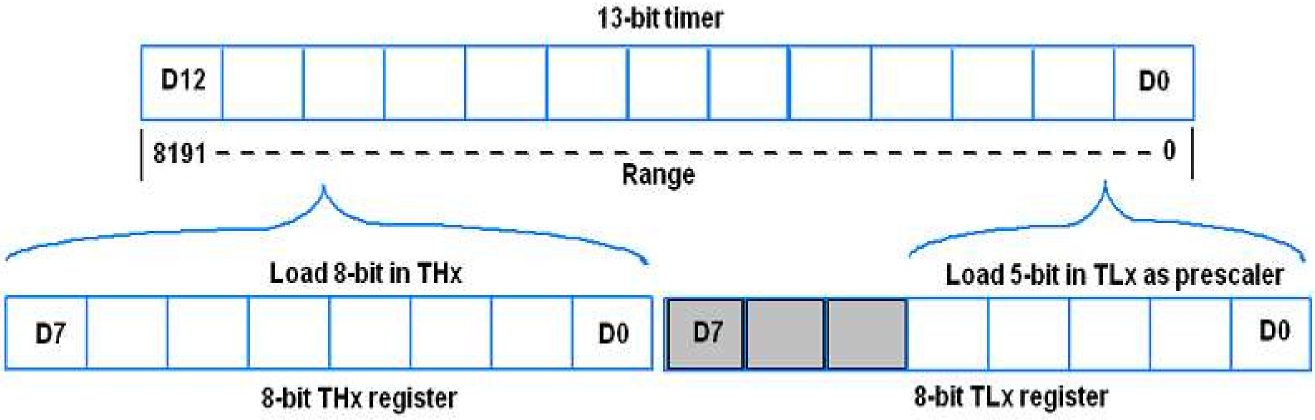
1. Load Tmod register value i.e. TMOD = 0x01 for Timer0 mode1 (16-bit timer mode).
2. Load calculated THx value i.e. here TH0 = 0xFC.
3. Load calculated TLx value i.e. here TL0 = 0x65.
4. Start the timer by setting a TRx bit. i.e. here TR0 = 1.
5. Poll TFx flag till it does not get set.
6. Stop the timer by clearing TRx bit. i.e. here TR0 = 0.
7. Clear timer flag TFx bit i.e. here TF0 = 0.
8. Repeat from step 1 to 7 for the delay again.

C Program to generate Square Wave as per the given requirement

```
#include <reg51.h>          /* Include x51 header file */
sbit test = P1^0;           /* set test pin0 of port1 */
void timer_delay()          /* Timer0 delay function */
{
        TH0 = 0xFC;         /* Load higher 8-bit in TH0 */
        TL0 = 0x74;         /* Load lower 8-bit in TL0 */
        TR0 = 1;            /* Start timer0 */
        while(TF0 == 0);    /* Wait until timer0 flag set */
        TR0 = 0;           /* Stop timer0 */
        TF0 = 0;           /* Clear timer0 flag */
}
void main()
{
        TMOD = 0x01;        /* Timer0 mode1 (16-bit timer mode) */
        while(1)
        {
                test = ~test;      /* Toggle test pin */
                timer_delay();  /* Call timer0 delay */
        }
}
```

Mode 0 (13-bit timer mode)

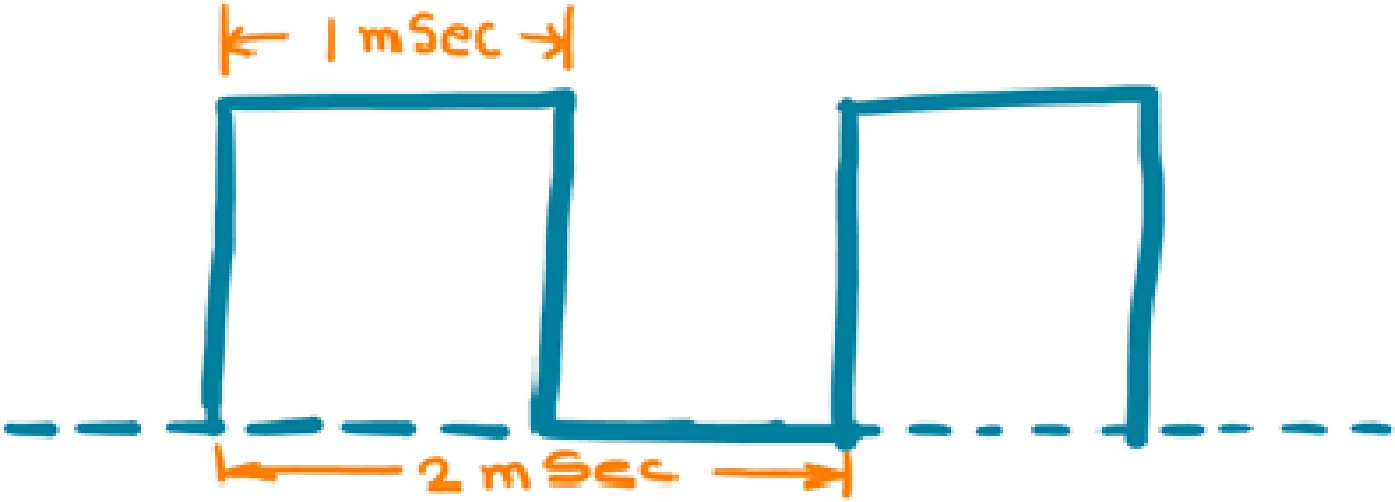
Mode 0 is a 13-bit timer mode for which 8-bit of THx and 5-bit of TLx (as Prescaler) are used. It is mostly used for interfacing with old MCS-48 family microcontrollers.



As shown in the above figure, 8-bit of THx and lower 5-bit of TLx used to form a total 13-bit timer. Higher 3-bits of TLx should be written as zero while using timer mode0, or it will affect the result.

Example

Let's generate a square wave of 2msec period using an AT89C51 microcontroller with timer0 in mode0 on the P1.0 pin of port1. Assume xtal oscillator frequency of 11.0592 MHz.



As the Xtal oscillator frequency is 11.0592 MHz, we have a machine cycle of 1.085uSec. Hence, the required count to generate a delay of 1mSec. is,

$$\text{Count} = (1\times10^{-3})/(1.085\times10^{-6}) \approx 922$$

The maximum count of Mode0 is 2^13 (0 - 8191) and the Timer0 count will increment from 0 – 8191. So we need to load value which is 922 less from its maximum count i.e. 8191. Hence value needed to be loaded is,

$$\text{Value}=(8191-\text{Count}) = 7269 = 0\text{x}1\text{C}65$$

So we need to load 0x1C65 value in Timer0.

1C65 = 000**1 1100 0110 0101**,

now load lower 5-bit in TL0 and next 8-bit in TH0

We get, TL0 = 000**1 1100** = 0x1C and TH0 = 0110 0101 = 0x65

Programming steps for delay function

1. Load Tmod register value i.e. TMOD = 0x00 for Timer0/1 mode0 (13-bit timer mode).
2. Load calculated THx value i.e. here TH0 = 0x65.
3. Load calculated TLx value i.e. here TL0 = 0x1C.
4. Start the timer by setting a TRx bit. i.e. here TR0 = 1.
5. Poll TFx flag till it does not get set.
6. Stop the timer by clearing TRx bit. i.e. here TR0 = 0.
7. Clear timer flag TFx bit i.e. here TF0 = 0.
8. Repeat from step 1 to 7 for the delay again.

C Program to generate Square Wave as per the given requirement

```
#include <reg51.h>                          /* Include x51 header file */
sbit test = P1^0;                  /* set test pin 0 of port1 */
void timer_delay()                          /* Timer0 delay function */
{
    TH0 = 0x65;             /* Load 8-bit in TH0 (here Timer0 used) */
    TL0 = 0x1C;                    /* Load 5-bit in TL0 */
    TR0 = 1;                       /* Start timer0 */
    while(TF0 == 0);               /* Wait until timer0 flag set */
    TR0 = 0;                       /* Stop timer0 */
    TF0 = 0;                       /* Clear timer0 flag */
}
void main()
{
    TMOD = 0x00;         /* Timer0/1 mode0 (13-bit timer mode) */
    while(1)
    {
        test = ~test;                       /* Toggle test pin */
        timer_delay();                      /* Call timer0 delay */
    }
}
```