8051 Timers and Counters

8051 has **two counters/timers** which can be used as timer to **generate a time delay** and as a counter to **count events** happening outside the microcontroller.

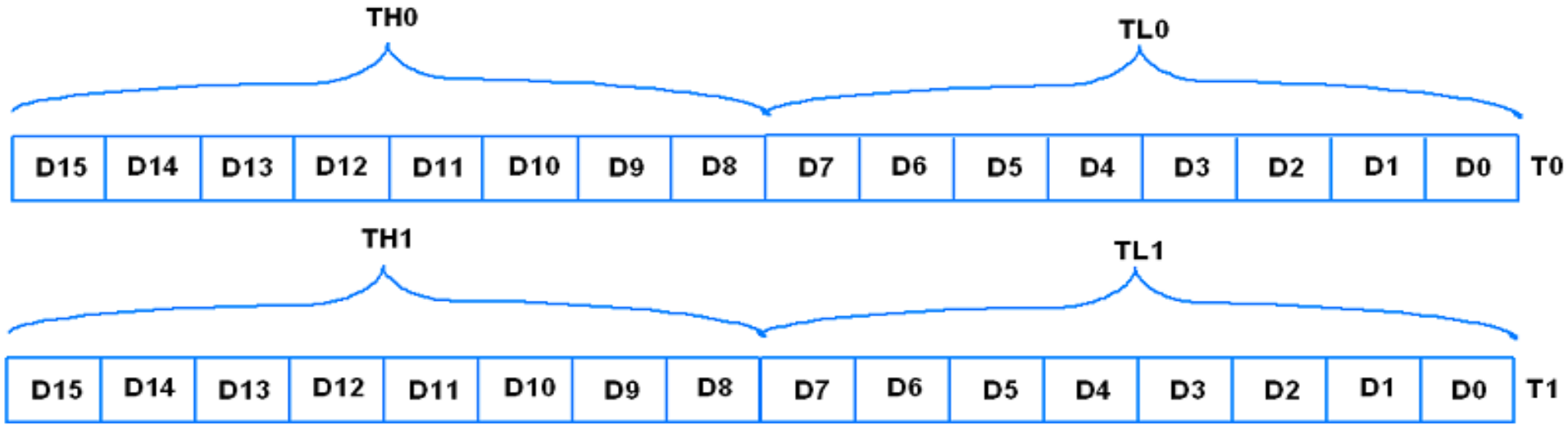
- ✓ Timer 0 and Timer 1.
- ✓ 16 bits wide – High byte (TH) and Low byte (TL).
- ✓ Each timer can also act as separate 8 bit register.
- ✓ 8051 counter always count up.
- ✓ Each counter has a 16-bit count register in the Special Function Register Area.
- ✓ It can count from 0000 to FFFF
- ✓ Overflow condition means the counter has reached its maximum value and again it is incremented. Overflow flag bit will be set in that condition.
- ✓ When their counts rolls over from the maximum count to 0000, the corresponding timer flag in TCON is set. TCON is timer control register. It is 8 bit register.

TIMERS / COUNTERS

- **Two 16 bit timer/counters**
- **Can be programmed independently** as – **timer or event counter**.
- **Four-SFR's connected** with **TIMER/COUNTER** operation
- **TMOD** – Timer Mode Register
- **TCON** – Timer Control Register
- **TH0, TL0** – Timer/Counter - 0
- **TH1, TL1** – Timer/Counter - 1
- **Two pins** of **8051 connected** with **Timer/counter**.
  - **T0** – Timer 0 external input – **P3.4**
  - **T1** – Timer 1 external input – **P3.5**
- $\overline{\text{INT0}}$ and $\overline{\text{INT1}}$ are also used for controlling the timer/counters.

Timer 0 : 16 Bits – TH0 and TL0

Timer 1 : 16 Bits – TH1 and TL1



TMOD (Timer mode register) and TCON (Timer control register) are two 8 bit registers called special function registers which are having special functions associated with it.

TMOD register is going to decide that in which mode the timer is going to operate. TCON register indicates the status of the timers.

Timer Mode Register (TMOD)

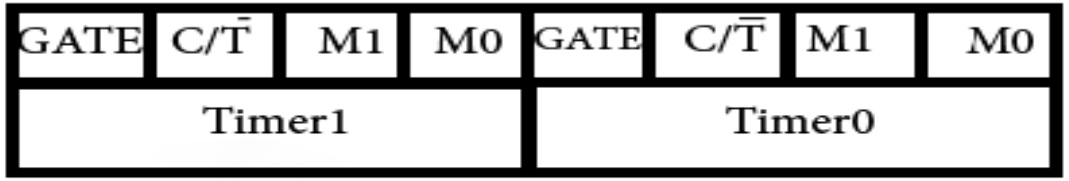


Fig. TMOD Register

Gate bit : 1, Hardware Start/Stop and 0, Software Start/Stop. Hardware (Interrupt and Reset).

C/T: Used to decide whether a timer is used as a delay generator or an event counter.

1: Counter 0: Timer – Delay is generated

M1 and M0: Decides the modes of operation in timer.

Timer Mode Control Register - TMOD

	← Timer 1 →				← Timer 0 →			
Bit no.	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Symbol	**Gate**	**C/$\overline{T}$**	**M1**	**M0**	**Gate**	**C/$\overline{T}$**	**M1**	**M0**

M1 and M0 specify the mode as follows:

M1	M0	Mode	*Description in brief*
0	0	0	13-bit counter
0	1	1	16-bit counter
1	0	2	8-bit counter with autoreload
1	1	3	Split Timer 0 into two 8-bit counters or to stop Timer 1

Mode 0 : 13 bit timer mode :

0000 to 1FFF : TH and TL

Start: SETB TR0 : Timer 0

SETB TR1 : Timer 1

After roll over the flag bit TF0 and TF1 will be set (1)

Stop: CLR TR0

CLR TR1

Mode 1 : 16 bit timer mode :

0000 to FFFF : TH and TL

Mode 2: 8 bit auto reload mode :

TH and TL : 00 to FF : The initialized value of TH will be copied into the TL

After roll over the flag bit TF0 and TF1 will be set (1)

The programmers need not to reload the count here. The contents of TH will automatically reload into TL. The count is first loaded in TH and remains there. It will not be modified.

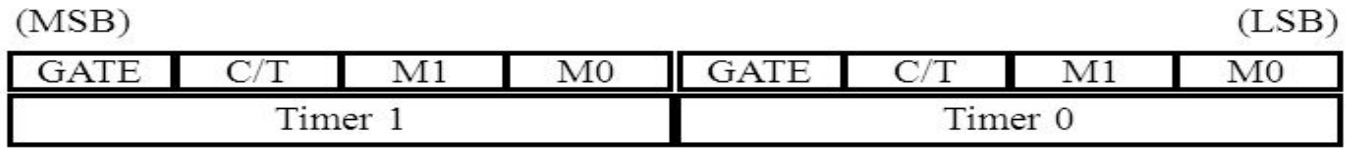
Mode 3 : Split Timer Mode

In mode 0, 1 and 2, the operations of the timer 0 and timer 1 are independent of each other. If suppose timer 0 is operating in mode 0, the timer 1 can operate in mode 2. their functions are completely different. They are independent to each other. But in mode 3, they are not independent to each other. They are collectively work.

Timer 1 : Mode 3: TR1 and TF1 : Timer 0

Find the value for TMOD if we want to program timer 0 in mode 2, use 8051 XTAL for the clock source, and use instructions to start and stop the timer.

Solution:



TMOD= 0000 0010

Timer 1 is not used.

Timer 0, **mode 2**,

C/T = 0 to use XTAL clock source (timer)

gate = 0 to use internal (**software**) start and stop method.

Timer Control Register (TCON)