Module
3
Embedded Systems I/O

Lesson
14

Timers

Instructional Objectives

After going through this lesson the student would learn Standard Peripheral Devices most commonly used in single purpose processors. They are

- Timers and Counter Basics
- Various Modes of Timer Operation
- The internal Timer of 8051
- A programmable interval timer 8253
- Watchdog Timer and Watchdog circuit

Pre-Requisite

Digital Electronics, Microprocessors

14 Introduction

The Peripherals of an embedded processor can either be on the same chip as the processor or can be connected externally.

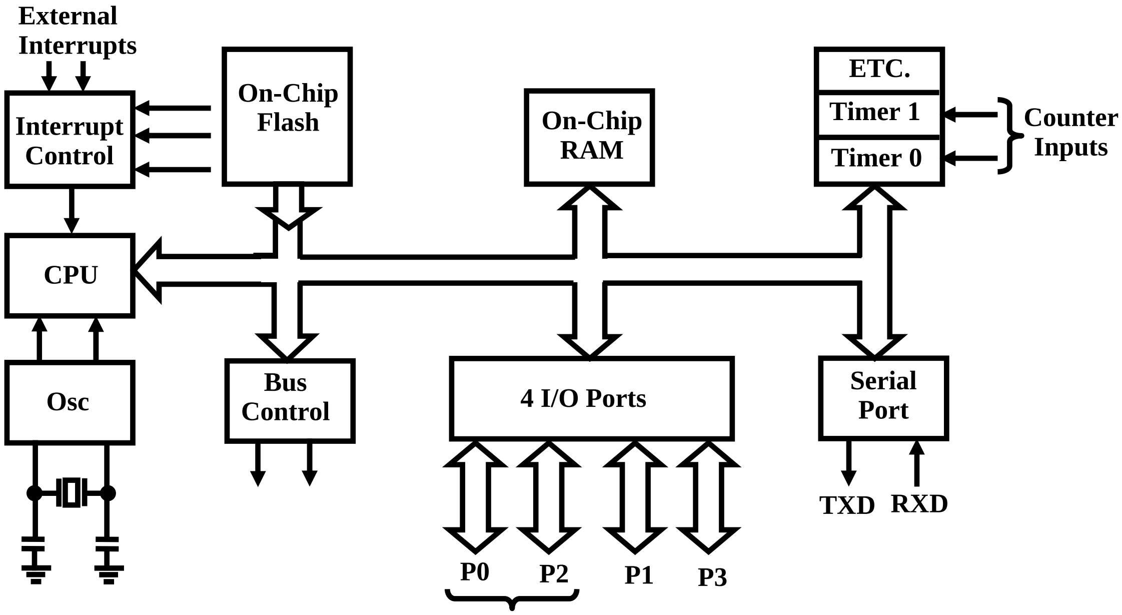


Fig. 14.1 Block Diagram of the basic 8051 Architecture

For example in a typical embedded processor as shown in Fig.14.1 timer, interrupt. Serial port and parallel ports reside on the single chip. These dedicated units are otherwise termed as single-purpose processor. These units are designed to achieve the following objectives. They can be a part of the microcontroller or can reside outside the chip and therefore should be properly interfaced with the processor.

The tasks generally carried out by such units are

- Timers, counters, watchdog timers
- serial transmission

- analog/digital conversions

Timer

Timer is a very common and useful peripheral. It is used to generate events at specific times or measures the duration of specific events which are external to the processor. It is a programmable device, i.e. the time period can be adjusted by writing specific bit patterns to some of the registers called timer-control registers.

Counter

A counter is a more general version of the timer. It is used to count events in the form of pulses which is fed to it.

Fig.14.2(a) shows the block diagram of a simple timer. This has a 16-bit up counter which increments with each input clock pulse. Thus the output value *Cnt* represents the number of pulses since the counter was last reset to zero. An additional output *top* indicates when the terminal count has been reached. It may go high for a predetermined time as set by the programmable control word inside the timer unit. The count can be loaded by the external program.

Fig.14.2(b) provides the structure of another timer where a multiplexer is used to choose between an internal clock or external clock. The mode bit when set or reset decided the selection. For internal clock(*Clk*) it behaves like the timer in Fig.14.2(a). For the external count in (*cnt_in*) it just counts the number of occurrences.

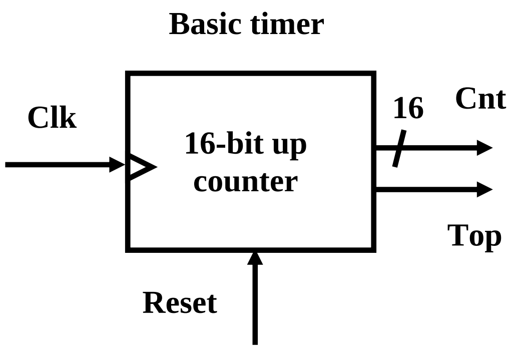


Fig. 14.2(a)

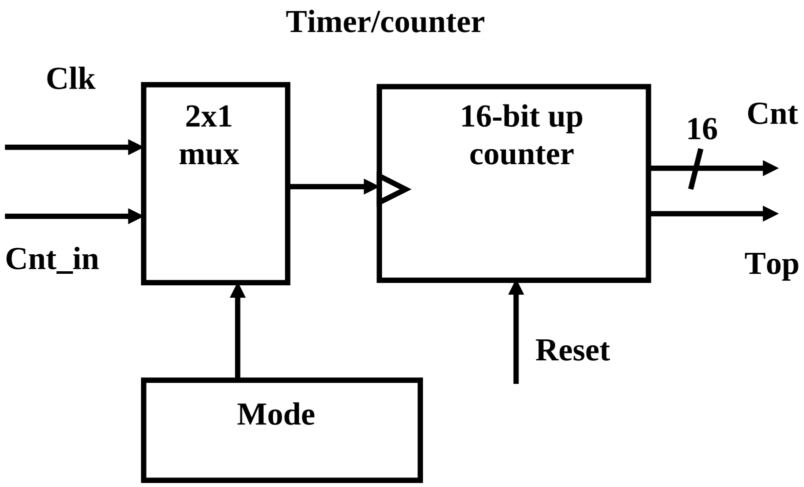


Fig. 14.2(b)

Fig.14.2(c) shows a timer with the terminal count. This can generate an event if a particular interval of time has been elapsed. The counter restarts after every terminal count.

Timer with a terminal count

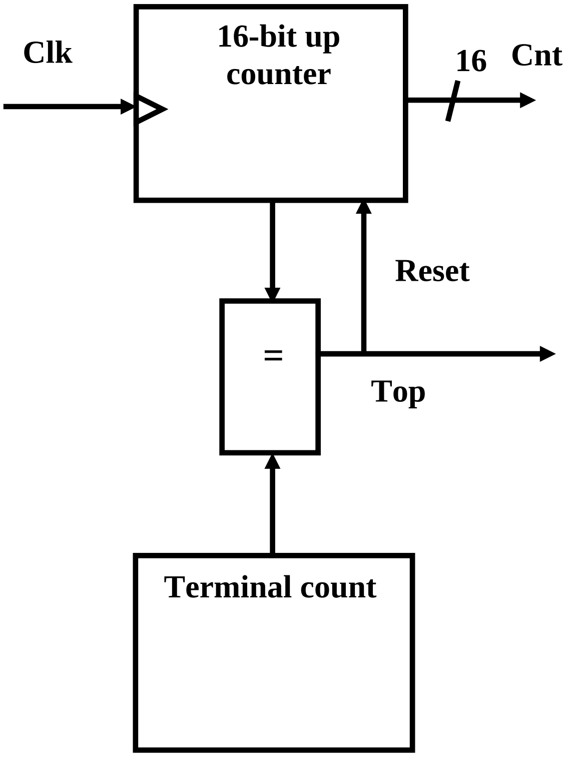


Fig. 14.2(c)

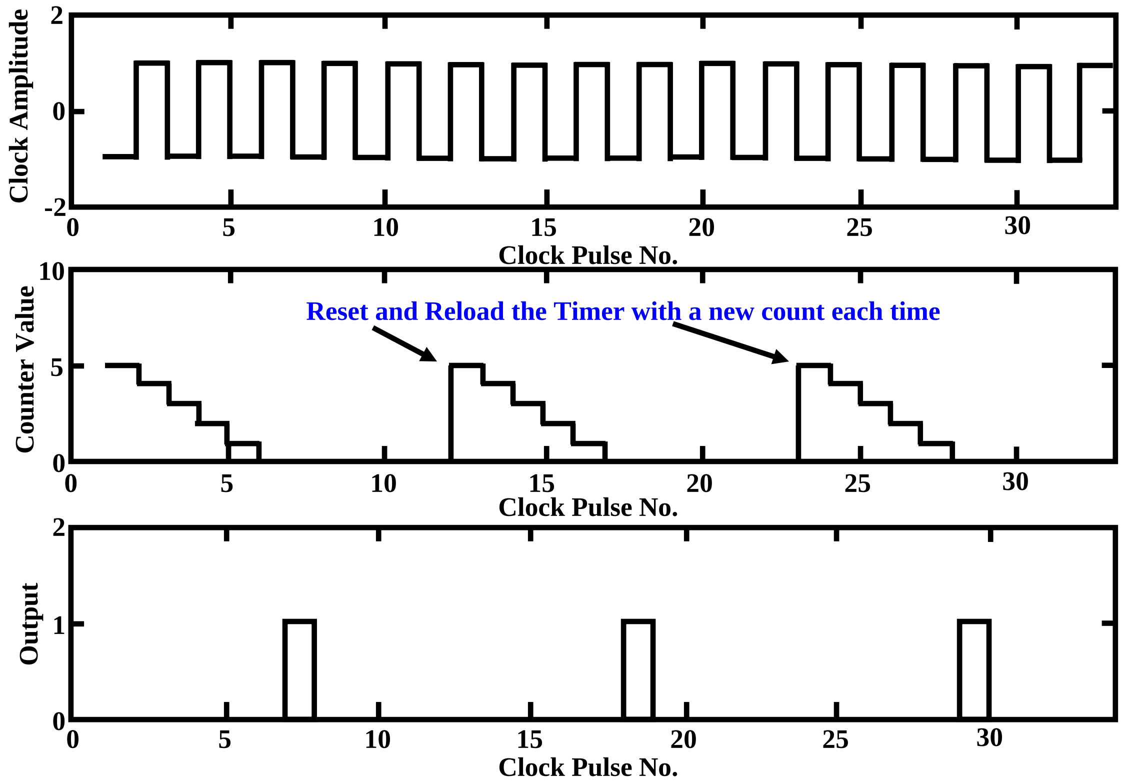


Fig. 14.3 The Timer Count and Output. The timer is in count-down mode. In every clock pulse the count is decremented by 1. When the count value reaches zero the output of the counter i.e. TOP goes high for a predetermined time. The counter has to be loaded with a new or previous value of the count by external program or it can be loaded automatically every time the count reaches zero.

Timer in 8051 Microcontroller

Fig.14.1 shows the architecture of 8051 which has got two timer units.

The 8051 comes equipped with two timers, both of which may be controlled, set, read, and configured individually. The 8051 timers have three general functions: 1) Keeping time and/or calculating the amount of time between events, 2) Counting the events themselves, or 3) Generating baud rates for the serial port.

As mentioned before, the 8051 has two timers which each function essentially the same way. One timer is TIMER0 and the other is TIMER1. The two timers share two Special Function Registers(SFR) (TMOD and TCON) which control the timers, and each timer also has two SFRs dedicated solely to itself (TH0/TL0 and TH1/TL1).

Timer0 and Timer1

The Timer and Counter functions are selected in the Special Function Register TMOD. These two Timer/Counter have four operating modes which are selected by bit-pairs (M1. M0) in TMOD. Modes 0, 1, and 2 are the same for both Timer/Counters.Mode3 is different.

MODE0

Either Timer in Mode0 is an 8-bit Counter with a divide-by-32 pre-scaler. In this mode, the Timer register is configured as a 13-Bit register. As the count rolls over from all 1s to all 0s, it sets the Timer interrupt flag TF1. The counted input is enabled to the Timer whenTR1 = 1and either GATE = 0 or $\overline{\text{INT1}}$ = 1. (Setting GATE = 1 allows the Timer to be controlled by external input INT1, to facilitate pulse width measurements.)

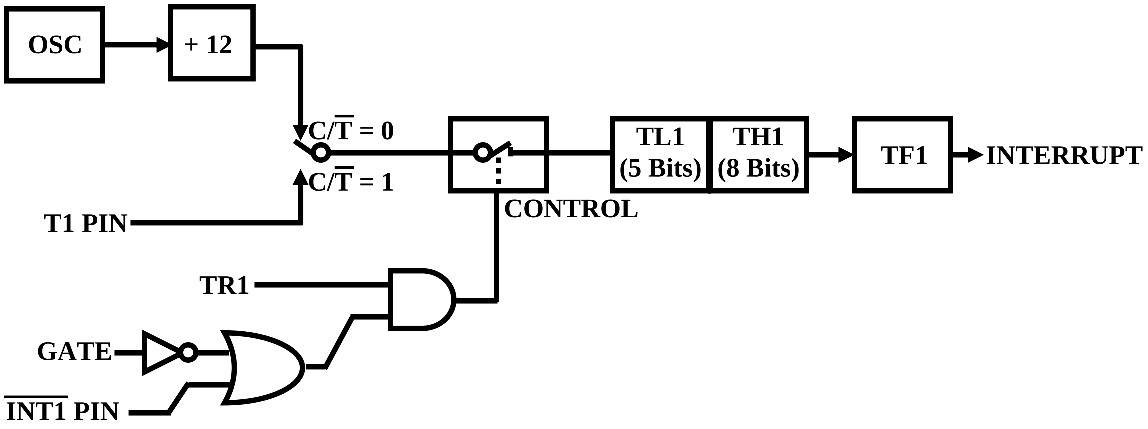

OSC
+ 12
C/$\overline{\text{T}}$ = 0
C/$\overline{\text{T}}$ = 1
T1 PIN
TL1 (5 Bits)
TH1 (8 Bits)
TF1
INTERRUPT
CONTROL
TR1
GATE
$\overline{\text{INT1}}$ PIN

Fig. 14.4 Timer/Counter Mode 0: 13-BitCounter

(MSB)							(LSB)
GATE	C/T	M1	M0	GATE	C/T	M1	M0
Timer 1				Timer 0			

GATE Gating control when set. Timer/Counter "x" is enabled only while "$\overline{\text{INTx}}$" pin is high and "TRx" control pin is set. When cleared Timer "x" is enabled whenever "TRx" control bit is set.

C/T Timer or Counter Selector cleared for Timer operation (input from internal system clock). Set for Counter operation (input from "Tx" input pin).

M1	M0	Operating Mode
0	0	8-bit Timer/Counter "THx" with "TLx" as 5-bit prescaler.
0	1	16-bit Timer/Counter "THx" and "TLx" are cascaded; there is no prescaler.
1	0	8-bit auto-reload Timer/Counter "THx" holds a value which is to be reloaded into "TLx" each time it overflows.
1	1	(Timer 0) TL0 is an 8-bit Timer/Counter controlled by the standard Timer 0 control bits. THO is an 8-bit timer only controlled by Timer 1 controls bits.
1	1	(Timer 1) Timer/Counter 1 stopped.

Mode Control Register (TMOD)

(MSB)							(LSB)
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

Symbol	Position	Name and Significance	Symbol	Position	Name and Significance
TF1	TCON.7	Timer 1 overflow Flag. Set by hardware on Timer/Counter overflow. Cleared by hardware when processor vectors to interrupt routine.	IE1	TCON.3	Interrupt 1 Edge flag. Set by hardware when external interrupt edge detected. Cleared when interrupt processed.
TR1	TCON.6	Timer 1 Run control bit. Set/cleared by software to turn Timer/Counter on/off.	IT1	TCON.2	Interrupt 1 Type control bit. Set/cleared by software to specify falling edge/low level triggered external interrupts.
TF0	TCON.5	Timer 0 overflow Flag. Set by hardware on Timer/Counter overflow. Cleared by hardware when processor vectors to interrupt routine.	IE0	TCON.1	Interrupt 0 Edge flag. Set by hardware when external interrupt edge detected. Cleared when interrupt processed.
TR0	TCON.4	Timer 1 Run control bit. Set/cleared by software to turn Timer/Counter on/off.	IT0	TCON.0	Interrupt 0 Type control bit. Set/cleared by software to specify falling edge/low level triggered external interrupts.

Timer/Counter Control Register (TCON)

MODE 1: Mode 1 is the same as Mode 0, except that the Timer register is being run with all 16bits.

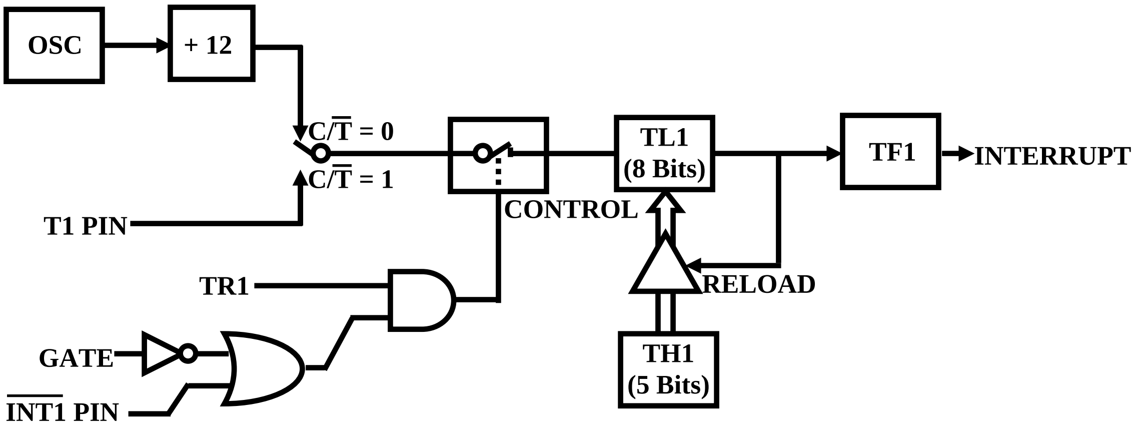


Fig. 14.5 MODE 2 configures the Timer register as an 8-bit counter with automatic reload

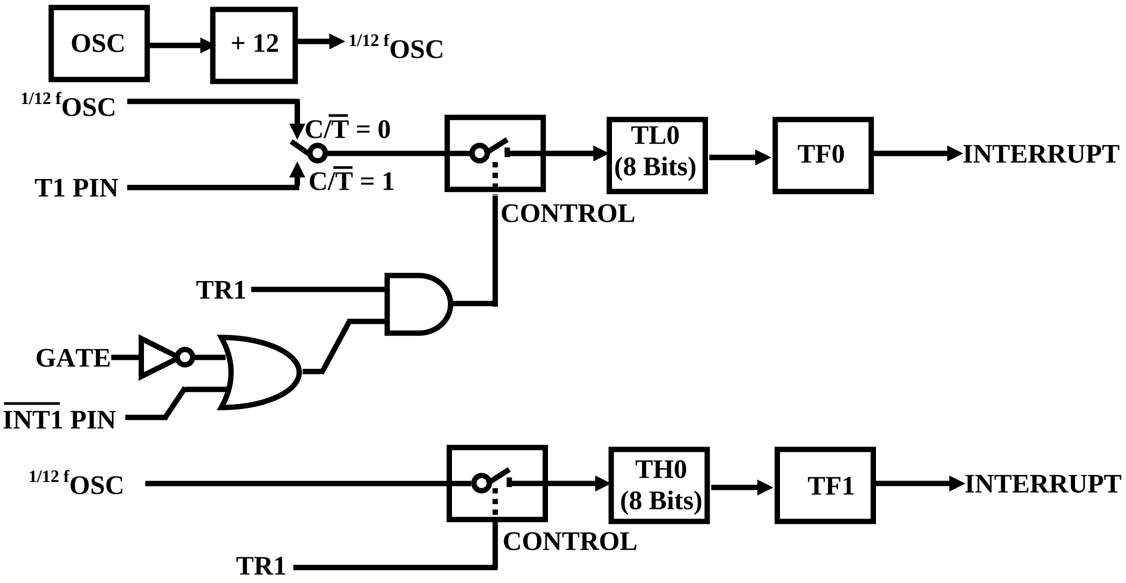
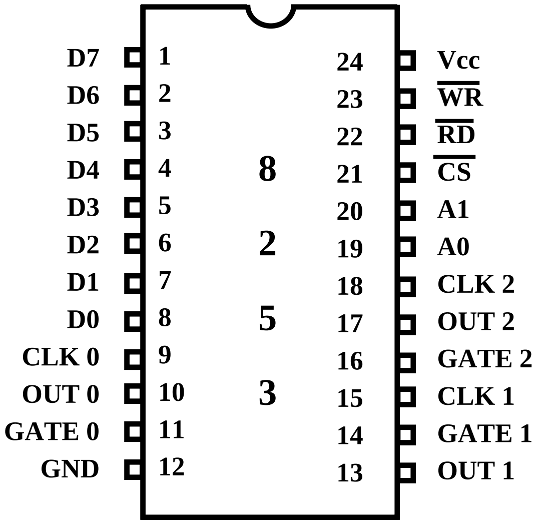


Fig. 14.6 MODE 3: Timer simply holds its count. Timer 0 in Mode 3 establishes TL0 and TH0 as two separate counters.

The Programmable Interval Timer 8253

For processors where the timer unit is not internal the programmable interval timer can be used. Fig.14.7 shows the signals for 8253 programmable interval timer.



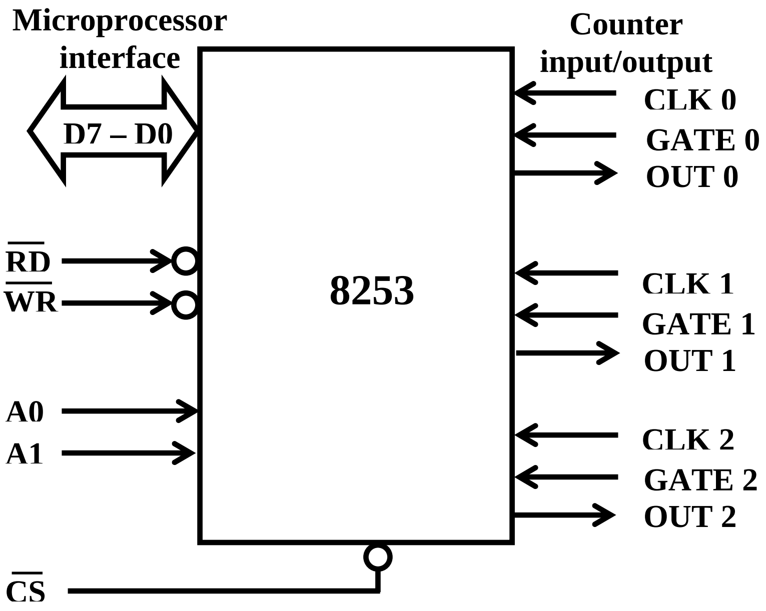


Fig. 14.7 The pin configuration of the timer

Fig.14.8 shows the internal block diagram. There are three separate counter units controlled by configuration register (Fig.14.9).

Each counter has two inputs, clock and gate and one output. The clock is signal that helps in counting by decrementing a preloaded value in the respective counter register. The gate serves as an enable input. If the gate is maintained low the counting is disabled. The timing diagram explains in detail about the various modes of operation of the timer.

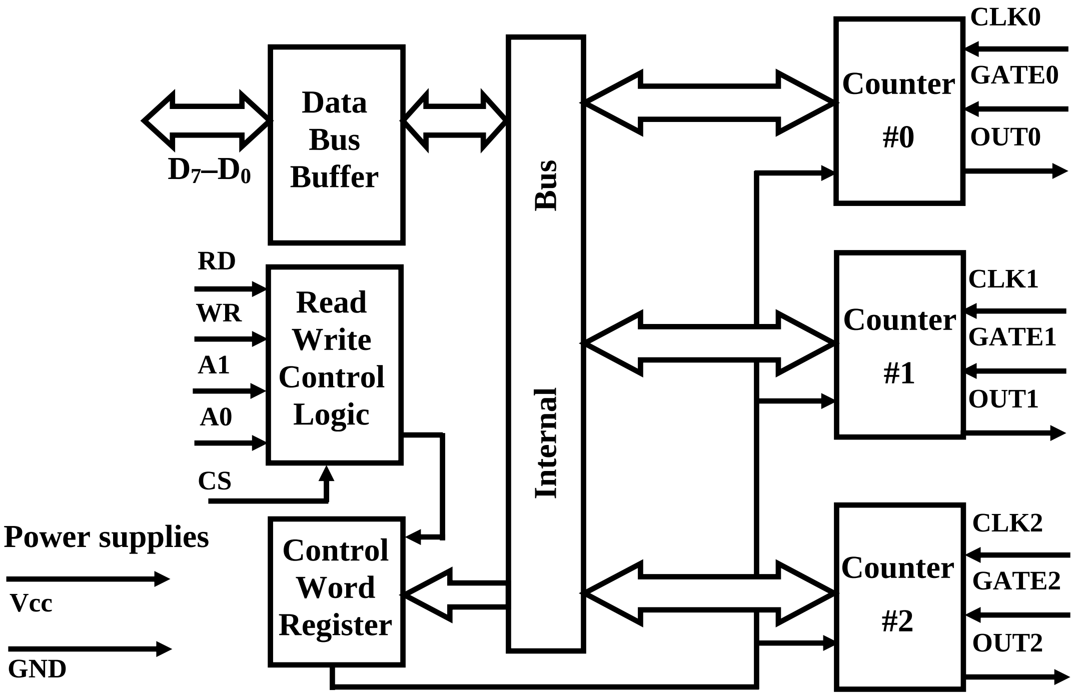
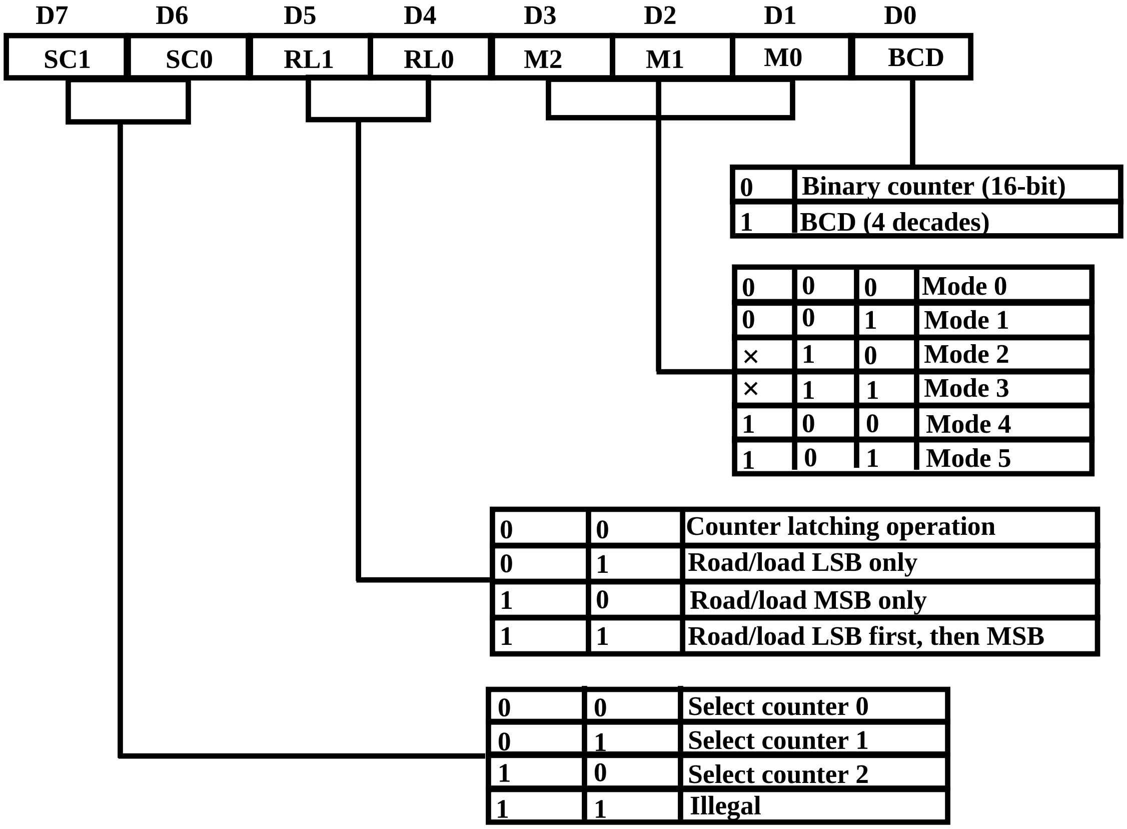


Fig. 14.8 The internal block diagram of 8253 Table The address map

$\overline{CS}$	A1	A0	Port
0	0	0	Counter 0
0	0	1	Counter 1
0	1	0	Counter 2
0	1	1	Control register



D7	D6	D5	D4	D3	D2	D1	D0
SC1	SC0	RL1	RL0	M2	M1	M0	BCD

0	Binary counter (16-bit)
1	BCD (4 decades)

0	0	0	Mode 0
0	0	1	Mode 1
×	1	0	Mode 2
×	1	1	Mode 3
1	0	0	Mode 4
1	0	1	Mode 5

0	0	Counter latching operation
0	1	Road/load LSB only
1	0	Road/load MSB only
1	1	Road/load LSB first, then MSB

0	0	Select counter 0
0	1	Select counter 1
1	0	Select counter 2
1	1	Illegal

Fig. 14.9 Control Register

8253 Operating Modes

Mode 0 Interrupt on terminal count
Mode 1 Programmable one shot
Mode 2 Rate Generator
Mode 3 Square wave rate Generator
Mode 4 Software triggered strobe
Mode 5 Hardware triggered strobe

Mode 0: The output goes high after the terminal count is reached. The counter stops if the Gate is low. (Fig.14.10(a) & (b)). The timer count register is loaded with a count (say 6) when the WR line is made low by the processor. The counter unit starts counting down with each clock pulse. The output goes high when the register value reaches zero. In the mean time if the GATE is made low (Fig.14.10(b)) the count is suspended at the value(3) till the GATE is enabled again.

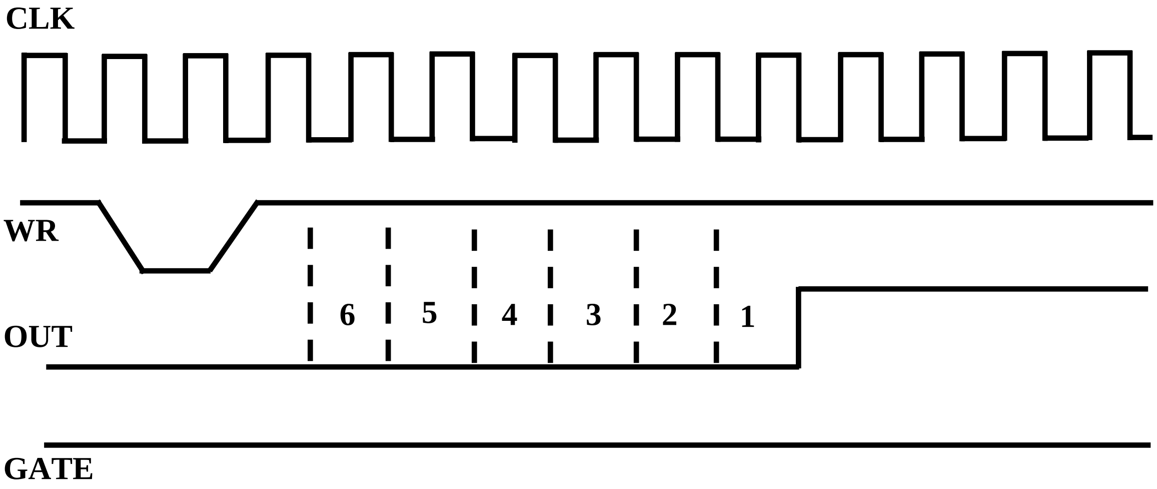


Fig. 14.10(a) Mode 0 count when Gate is high (enabled)

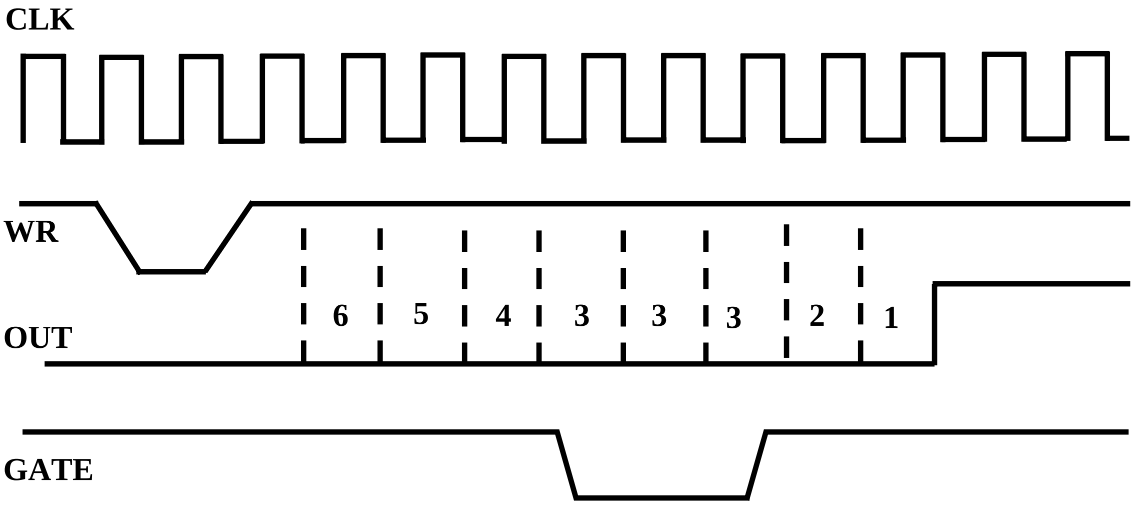


Fig. 14.10(b) Mode 0 count when Gate is low temporarily (disabled)

Mode 1 Programmable mono-shot

The output goes low with the Gate pulse for a predetermined period depending on the counter. The counter is disabled if the GATE pulse goes momentarily low.

The counter register is loaded with a count value as in the previous case (say 5) (Fig.14.11(a)). The output responds to the GATE input and goes low for period that equals the count down period of the register (5 clock pulses in this period). By changing the value of this count the duration of the output pulse can be changed. If the GATE becomes low before the count down is completed then the counter will be suspended at that state as long as GATE is low (Fig.14.11(b)). Thus it works as a mono-shot.

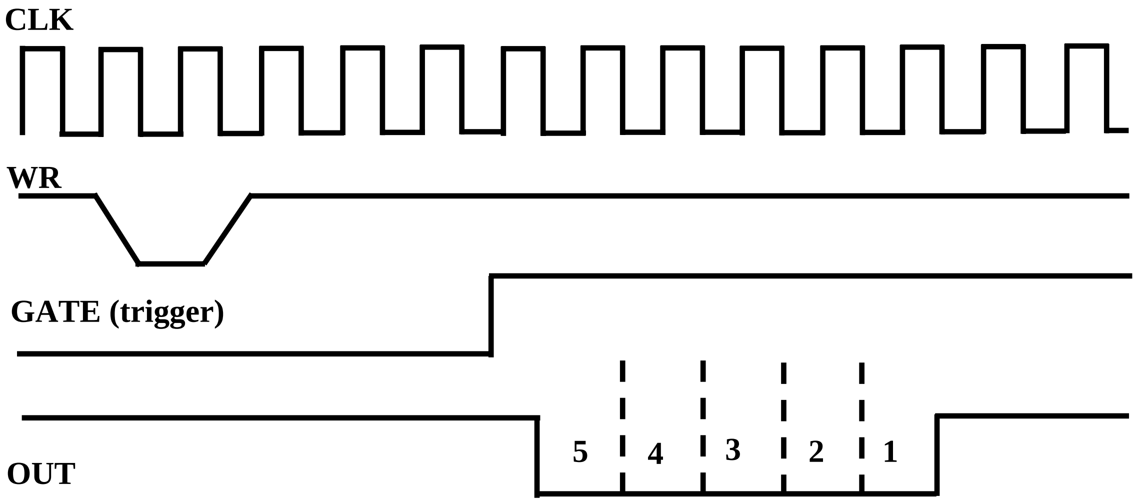


Fig. 14.11(a) Mode 1 The Gate goes high. The output goes low for the period depending on the count

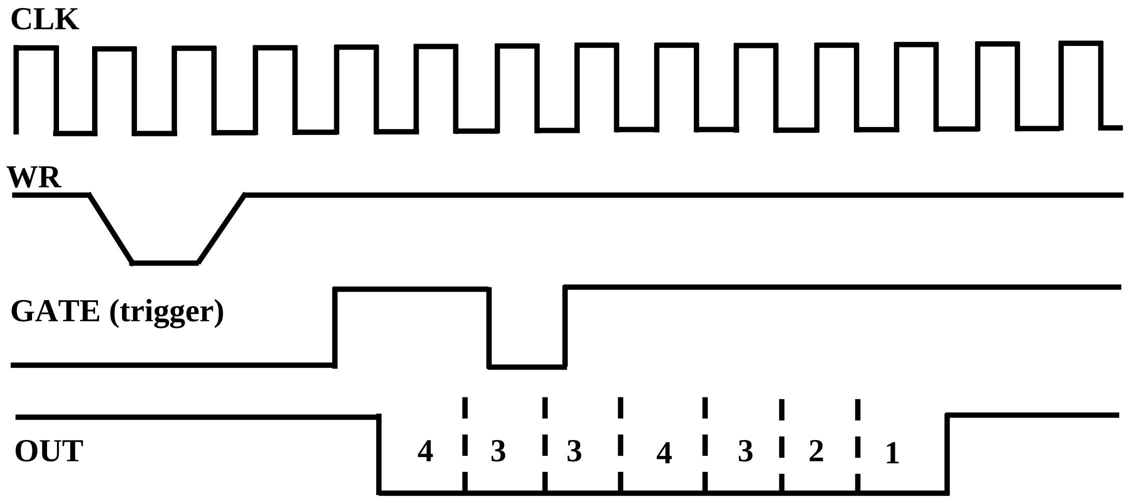


Fig. 14.11(b) Mode 1 The Gate pulse is disabled momentarily causing the counter to stop.

Mode 2 Programmable Rate Generator

Fig.14.12(a) and (b) shows the waveforms corresponding the Timer operation in this mode. In this mode it operates as a rate generator. The output goes high for a period that equals the time of count down of the count register (3 in this case). The output goes low exactly for one clock period before it becomes high again. This is a periodic operation.

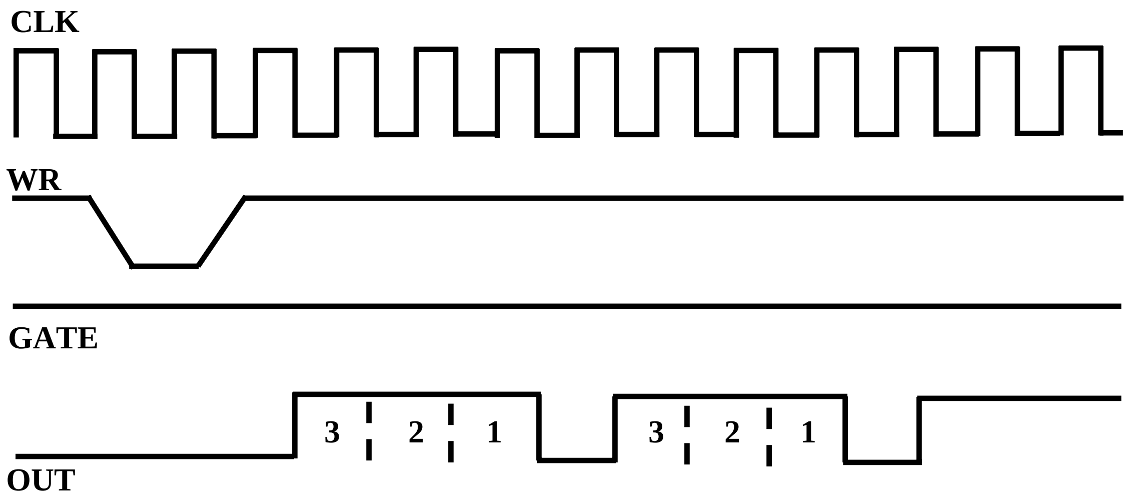


Fig. 14.12(a) Mode 2 Operation when the GATE is kept high

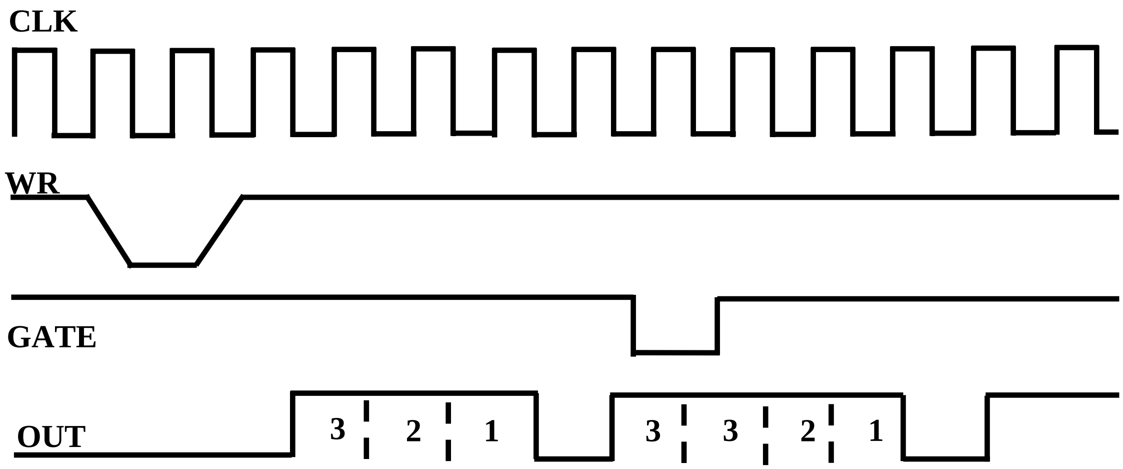


Fig. 14.12(b) Mode 2 operation when the GATE is disabled momentarily.

Mode 3 Programmable Square Wave Rate Generator

It is similar to Mode 2 but the output high and low period is symmetrical. The output goes high after the count is loaded and it remains high for period which equals the count down period of the counter register. The output subsequently goes low for an equal period and hence generates a symmetrical square wave unlike Mode 2. The GATE has no role here. (Fig.14.13).

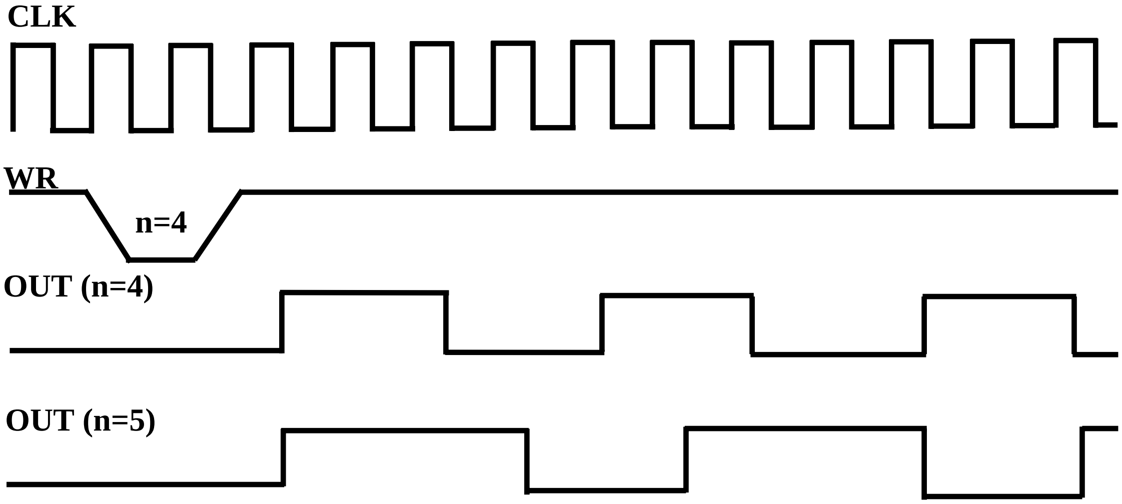


Fig. 14.13 Mode3 Operation: Square Wave generator

Mode 4 Software Triggered Strobe

In this mode after the count is loaded by the processor the count down starts. The output goes low for one clock period after the count down is complete. The count down can be suspended by making the GATE low (Fig.14.14(a) (b)). This is also called a software triggered strobe as the count down is initiated by a program.

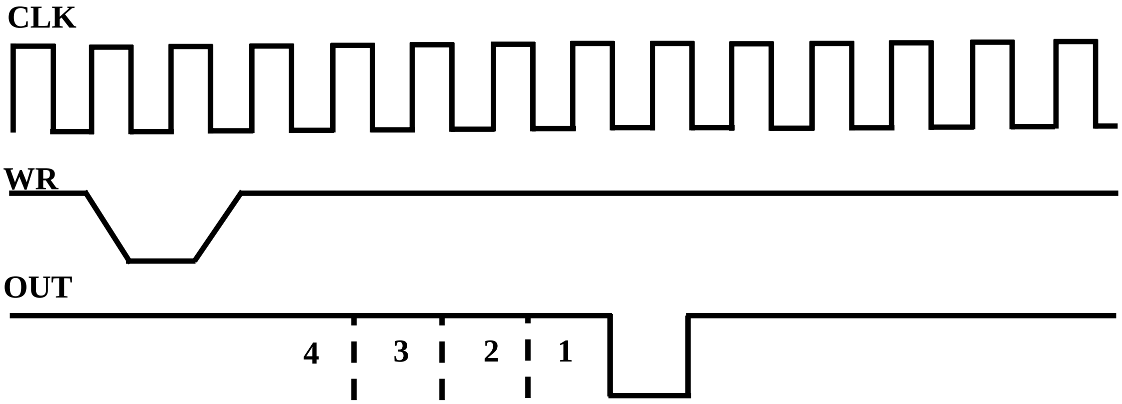


Fig. 14.14(a) Mode 4 Software Triggered Strobe when GATE is high

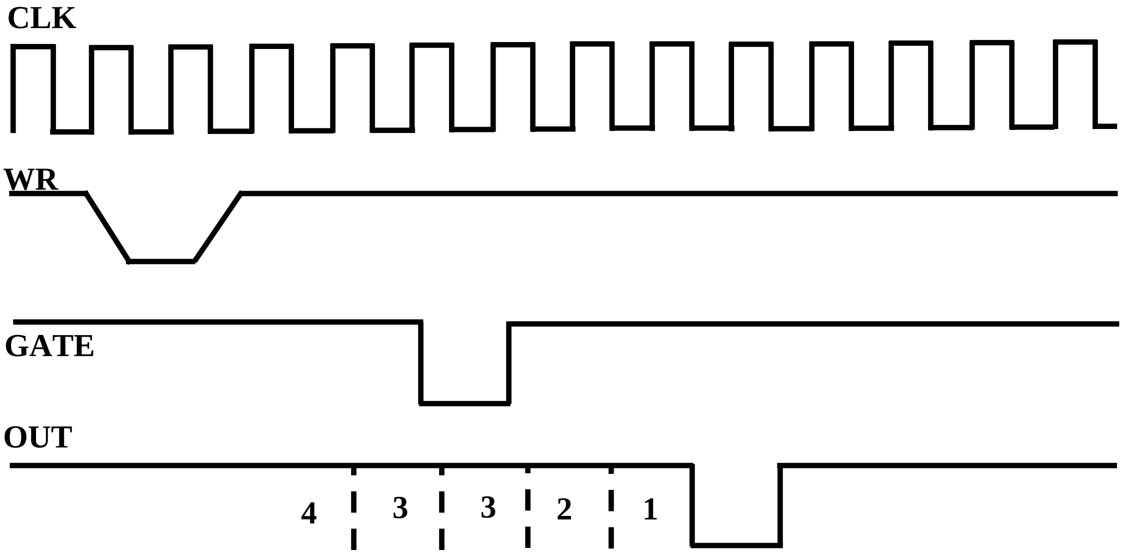


Fig. 14.14(b) Mode 4 Software Triggered Strobe when GATE is momentarily low

Mode 5 Hardware Triggered Strobe

The count is loaded by the processor but the count down is initiated by the GATE pulse. The transition from low to high of the GATE pulse enables count down. The output goes low for one clock period after the count down is complete (Fig.14.15).

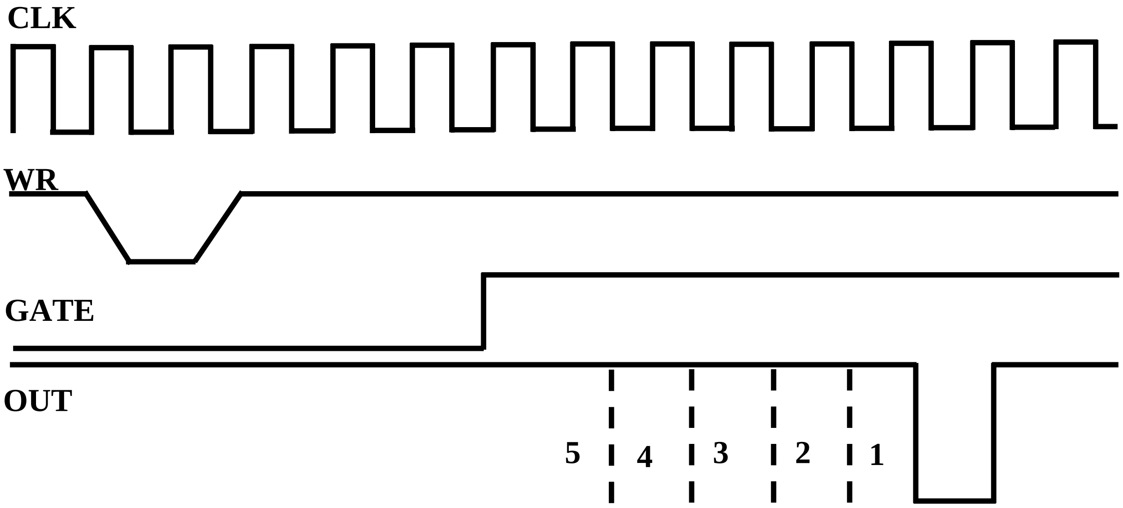


Fig. 14.15 Mode 5 Hardware Triggered Strobe

Watchdog timer

A Watchdog Timer is a circuit that automatically invokes a reset unless the system being watched sends regular hold-off signals to the Watchdog.

Watchdog Circuit

To make sure that a particular program is executing properly the Watchdog circuit is used. For instance the program may reset a particular flip-flop periodically. And the flip-flop is set by an external circuit. Suppose the flip-flop is not reset for long time it can be known by using external hardware. This will indicate that the program is not executed properly and hence an exception or interrupt can be generated.

Watch Dog Timer(WDT) provides a unique clock, which is independent of any external clock. When the WDT is enabled, a counter starts at 00 and increments by 1 until it reaches FF. When it goes from FF to 00 (which is FF + 1) then the processor will be reset or an exception will be generated. The only way to stop the WDT from resetting the processor or generating an exception or interrupt is to periodically reset the WDT back to 00 throughout the program. If the program gets stuck for some reason, then the WDT will not be set. The WDT will then reset or interrupt the processor. An interrupt service routine will be invoked to take into account the erroneous operation of the program. (getting stuck or going into infinite loop).

Conclusion

In this chapter you have learnt about the programmable timer/counter. For most of the embedded processors the timer is internal and exists along with the processor on the same chip. The 8051 microcontroller has 3 different internal timers which can be programmed in various modes by the configuration and mode control register. An external timer chip namely 8253 has also been discussed. It has 8 data lines 2 data lines, 1 chip select line and one read and one write control line. The 16 bit counts of the corresponding registers can be loaded with two consecutive write operations. Counters and Timers are used for triggering, trapping and managing various real time events. The least count of the timer depend on the clock. The stability of the clock decides the accuracy of the timings. Timers can be used to generate specific baud rate clocks for asynchronous serial communications. It can be used to measure speed, frequency and analog voltages after Voltage to Frequency conversion. One important application of timer is to generate Pulse-Width-Modulated (PWM) waveforms. In 8253 the GATE and pulse together can be used together to generate pulse with different widths. These modulated pulses are used in electronic power control to reduce harmonics and hence distortions.

You also learnt about the Watch dog circuit and Watch dog timers. These are used to monitor the activity of a program and the processor.

Questions

Q1. Design a circuit using 8253 to measure the speed of any motor by counting the number of pulses in definite period.

Q2. Write a pseudo code (any assembly code) to generate sinusoidal pulse width modulated waveform from the 8253 timer.

Q3. Design a scheme to read temperature from a thermister circuit using a V/F converter and Timer.

Q4. What are the differences in Mode 4 and Mode 5 operation of 8253 Timer?

Q5. Explain the circuit given in Fig.14.5.