

# 9

## Programmable Interval Timer 8253/54

### 9.1 Necessity and Introduction

The 8253/54 solves one of most common problem in any microcomputer system, the generation of accurate time delays under software control. Instead of setting up timing loops in system software, the programmer configures the 8253/54 to match his requirements, initializes one of the counters of the 8253/54 with the desired quantity, then upon command the 8253/54 will count out the delay and interrupt the CPU when it has completed its tasks. It is easy to see that the software overhead is minimum and that multiple delays can be easily be maintained by assignment of priority levels.

The 8253/54 includes three identical 16 bit counters that can operate independently. To operate a counter, a 16-bit count is loaded in its register and, on command, it begins to decrement the count until it reaches 0. At the end of the count, it generates a pulse that can be used to interrupt the CPU. The counter can count either in binary or BCD. In addition, a count can be read by the CPU while the counter is decrementing. In this chapter, we are going to study two timer ICs 8253 and 8254. The 8254 is a superset of 8253. The functioning of these two ICs are almost similar along with the pin configuration. Only the differences are :

8253	8254
1. Operating frequency 0 - 2.6 MHz.	1. Operating frequency 0 - 10 MHz.
2. Uses N-MOS technology.	2. Uses H-MOS technology.
3. Read-Back command not available.	3. Read-Back command available.
4. Reads and writes of the same counter can not be interleaved.	4. Reads and writes of the same counter can be interleaved.

In this chapter, where the things are common 8253/54 is mentioned and 8254 is mentioned specifically for giving only information about 8254.

### 9.2 Features

- 1) Three independent 16-bit down counters.
- 2) 8254 can handle inputs from DC to 10 MHz (5MHz 8254-5 8MHz 8254 10MHz 8254-2) where as 8253 can operate upto 2.6 MHz.

- 3) Three counters are identical presetable, and can be programmed for either binary or BCD count.
- 4) Counter can be programmed in six different modes.
- 5) Compatible with all Intel and most other microprocessors.
- 6) 8254 has powerful command called READ BACK command which allows the user to check the count value, programmed mode and current mode and current status of the counter.

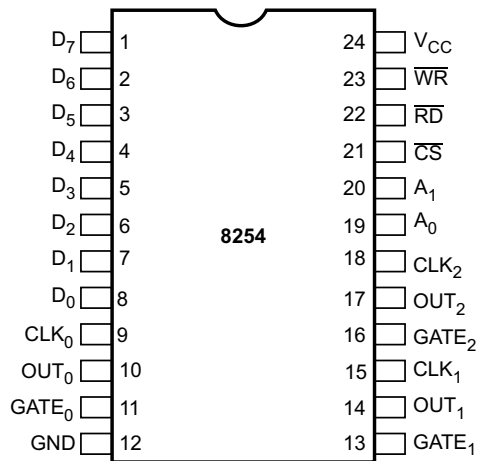


Fig. 9.1 Pin diagram of 8254

9.3 Block Diagram

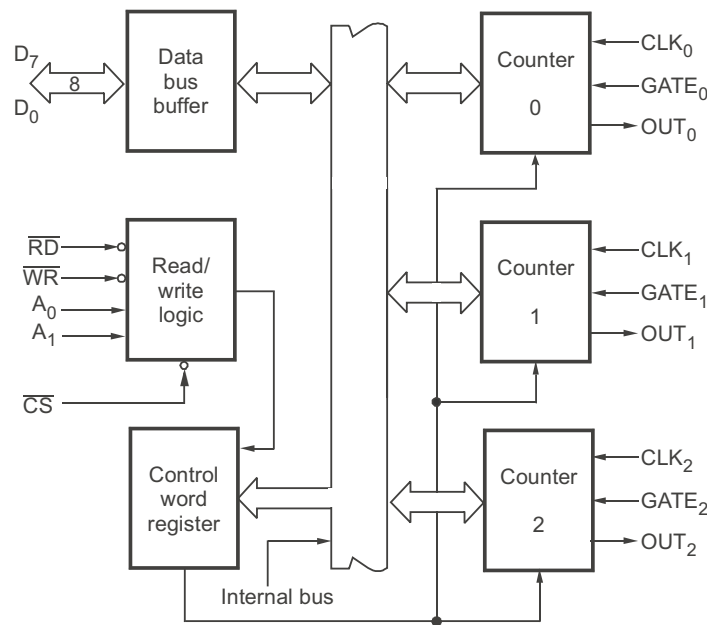


Fig. 9.2 Block diagram of 8254

Fig. 9.2 shows the block diagram of 8253/54. It includes three counters, a data bus buffer, Read/Write control logic, and a control register. Each counter has two input signals CLOCK and GATE and one output signal OUT.

#### Data Bus Buffer :

This tri-state, bi-directional, 8-bit buffer is used to interface the 8253/54 to the system data bus. The Data bus buffer has three basic functions.

1. Programming the modes of 8253/54.
2. Loading the count registers.
3. Reading the count values.

**Read/Write Logic :** The Read/Write logic has five signals :  $\overline{RD}$ ,  $\overline{WR}$ ,  $\overline{CS}$  and the address lines  $A_0$  and  $A_1$ . In the peripheral I/O mode, the  $\overline{RD}$  and  $\overline{WR}$  signals are connected to  $\overline{IOR}$  and  $\overline{IOW}$ , respectively. In memory-mapped I/O, these are connected to  $\overline{MEMR}$  and  $\overline{MEMW}$ . Address lines  $A_0$  and  $A_1$  of the CPU are usually connected to lines  $A_0$  and  $A_1$  of the 8253/54, and  $\overline{CS}$  is tied to a decoded address. The control word register and counters are selected according to the signals on lines  $A_0$  and  $A_1$ .

$A_1$	$A_0$	Selection
0	0	Counter 0
0	1	Counter 1
1	0	Counter 2
1	1	Control word Register

**Control Word Register :** This register is accessed when lines  $A_0$  and  $A_1$  are at logic 1. It is used to write a command word which specifies the counter to be used (binary or BCD), its mode, and either a read or write operation.

**Counters :** These three functional blocks are identical in operation. Each counter consists of a single, 16 bit, pre-settable, down counter. The counter can operate in either binary or BCD and its input, gate and output are configured by the selection of modes stored in the control word register. The counters are fully independent. The programmer can read the contents of any of the three counters without disturbing the actual count in process.

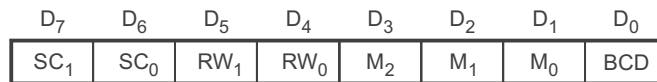
## 9.4 Operational Description

The complete functional definition of the 8253/54 is programmed by the system software. Once programmed, the 8253/54 is ready to perform whatever timing tasks it is assigned to accomplish.

**Programming the 8253/54 :** Each counter of the 8253/54 is individually programmed by writing a control word into the control word register ( $A_0 - A_1 = 11$ ). The Fig. 9.3 shows the control word format. Bits  $SC_1$  and  $SC_0$  select the counter, bits  $RW_1$  and  $RW_0$  select the read, write or latch command, bits  $M_2$ ,  $M_1$  and  $M_0$  select the mode of operation and bit BCD decides whether it is a BCD counter or binary counter.

**WRITE Operation :**

1. Write a control word into control register.
2. Load the low-order byte of a count in the counter register.
3. Load the high-order byte of count in the counter register.

**SC - Select counter**SC<sub>1</sub> SC<sub>0</sub>

0	0	Select counter 0
0	1	Select counter 1
1	0	Select counter 2
1	1	Illegal for 8253 Read -Back command for 8254 (See Read operations)

**M - Mode**M<sub>2</sub> M<sub>1</sub> M<sub>0</sub>

0	0	0	Mode 0
0	0	1	Mode 1
x	1	0	Mode 2
x	1	1	Mode 3
1	0	0	Mode 4
1	0	1	Mode 5

**RW - Read /Write**RW<sub>1</sub> RW<sub>0</sub>

0	0	Counter latch command (See Read operations)
0	1	Read / Write least significant byte only
1	0	Read / Write most significant byte only
1	1	Read / write least significant byte first, then most significant byte

**BCD :**

0	Binary counter 16 - bits
1	Binary coded decimal (BCD) Counter (4 Decades)

Note : Don't care bits (x) should be 0 to ensure compatibility with future Intel products

**Fig. 9.3 Control word format**

**READ Operation :** In some applications, especially in event counters, it is necessary to read the value of the count in process. This can be done by three possible methods:

1. **Simple Read :** It involves reading a count after inhibiting the counter by controlling the gate input or the clock input of the selected counter, and two I/O read operations are performed by the CPU. The first I/O operation reads the low-order byte, and the second I/O operation reads the high order byte.

2. **Counter Latch Command :** In the second method, an appropriate control word is written into the control register to latch a count in the output latch, and two I/O read operations are performed by the CPU. The first I/O operation reads the low-order byte, and the second I/O operation reads the high order byte.

3. **Read-Back Command (Available only for 8254) :** The third method uses the Read-Back command. This command allows the user to check the count value, programmed Mode, and current status of the OUT pin and Null count flag of the selected counter(s). Fig.9.4 shows the format of the control word register for Read-Back command.

$$A_0, A_1 = 11 \quad \overline{CS} = 0 \quad \overline{RD} = 1 \quad \overline{WR} = 0$$

D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>
1	1	$\overline{COUNT}$	$\overline{STATUS}$	CNT <sub>2</sub>	CNT <sub>1</sub>	CNT <sub>0</sub>	0

D<sub>5</sub> : 0 = Latch count of selected counter(s)

D<sub>4</sub> : 0 = Latch status of selected counter(s)

D<sub>3</sub> : 1 = Select counter 2

D<sub>2</sub> : 1 = Select counter 1

D<sub>1</sub> : 1 = Select counter 0

D<sub>0</sub> : Reserved for future expansion : must be 0.

**Fig. 9.4 Control word register for read-back command**

The Read-Back command may be used to latch multiple counter output latches by setting the  $\overline{COUNT}$  bit D<sub>5</sub> = 0 and selecting the desired counter (s). Each counter's latch count is held until it is read (or the counter is reprogrammed). That counter is automatically unlatched when read.

#### Other Features of Read - Back Command (Available only for 8254) :

The Read-Back command may also be used to latch status information of selected counter(s) by setting STATUS bit D<sub>4</sub> = 0. The contents of the counter must be latched before reading. The status of a counter is then accessed by a read from that counter. The Fig. 9.5 shows the counter status format.

D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>
OUTPUT	NULL COUNT	RW <sub>1</sub>	RW <sub>0</sub>	M <sub>2</sub>	M <sub>1</sub>	M <sub>0</sub>	BCD

D<sub>7</sub>            1 = OUT pin is 1.

                 0 = OUT pin is 0.

D<sub>6</sub>            1 = NULL count.

                 0 = Count available for reading.

D<sub>5</sub>-D<sub>0</sub>        = Counter programmed mode. (see Fig. 9.3)

**Fig. 9.5 Counter status format**

Bit D<sub>5</sub> - D<sub>0</sub> contains the counter's programmed mode exactly as written in the last mode control word. Bit D<sub>7</sub> contains the current status of the output pin. In 8254, it is not possible to read count from the counter, if the count is not loaded into the counting element (CE). The Bit D<sub>6</sub> indicates whether the counting element has count or not. If D<sub>6</sub> = 0, counting element has count otherwise null count.

**Interleaved Read and Write :** Another feature of the 8254 is that reads and writes of the same counter may be interleaved. For example, if the counter is programmed for the two byte counts, the following sequence is valid.

1. Read least significant byte.
2. Write new least significant byte.
3. Read most significant byte.
4. Write new most significant byte.

9.5 Mode Definition

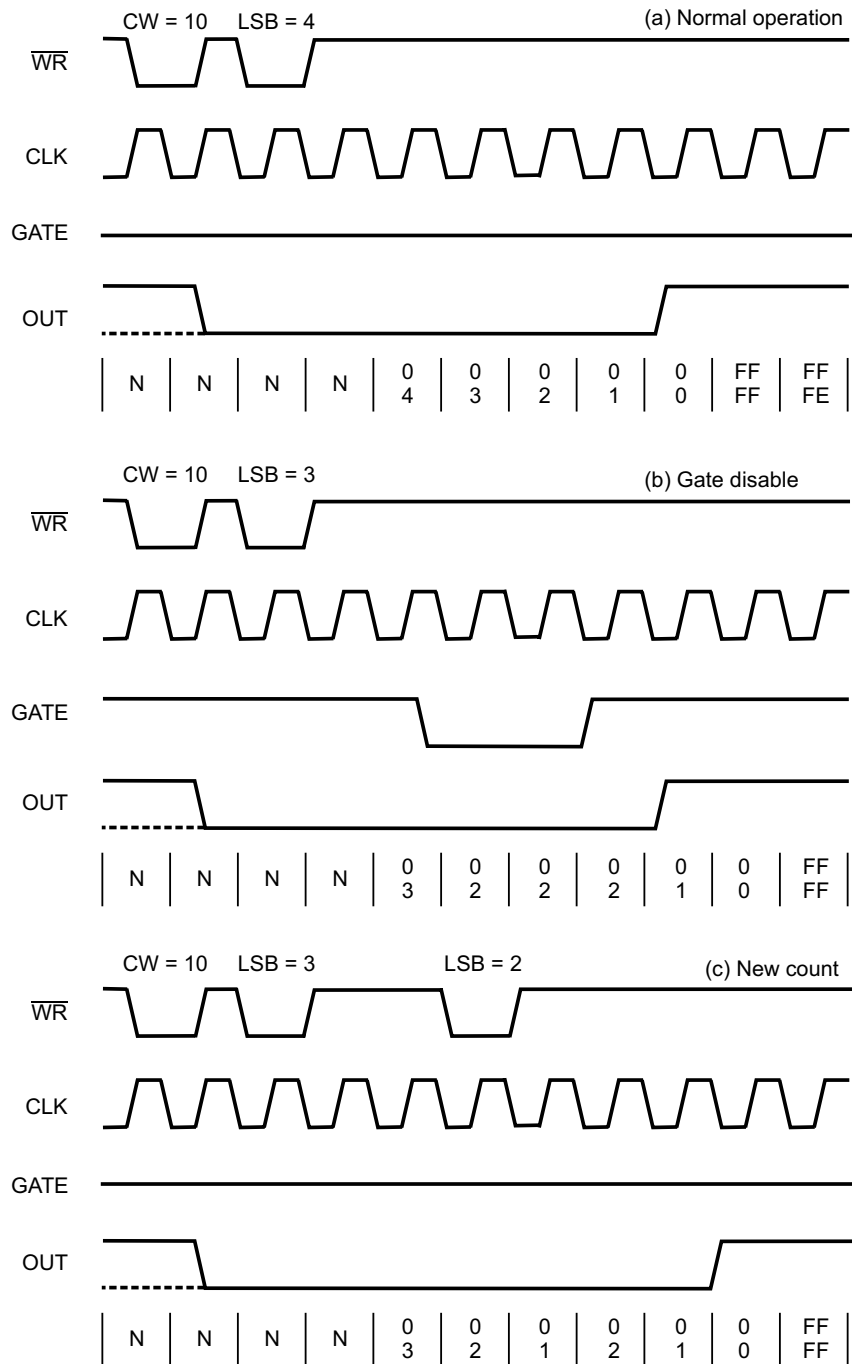


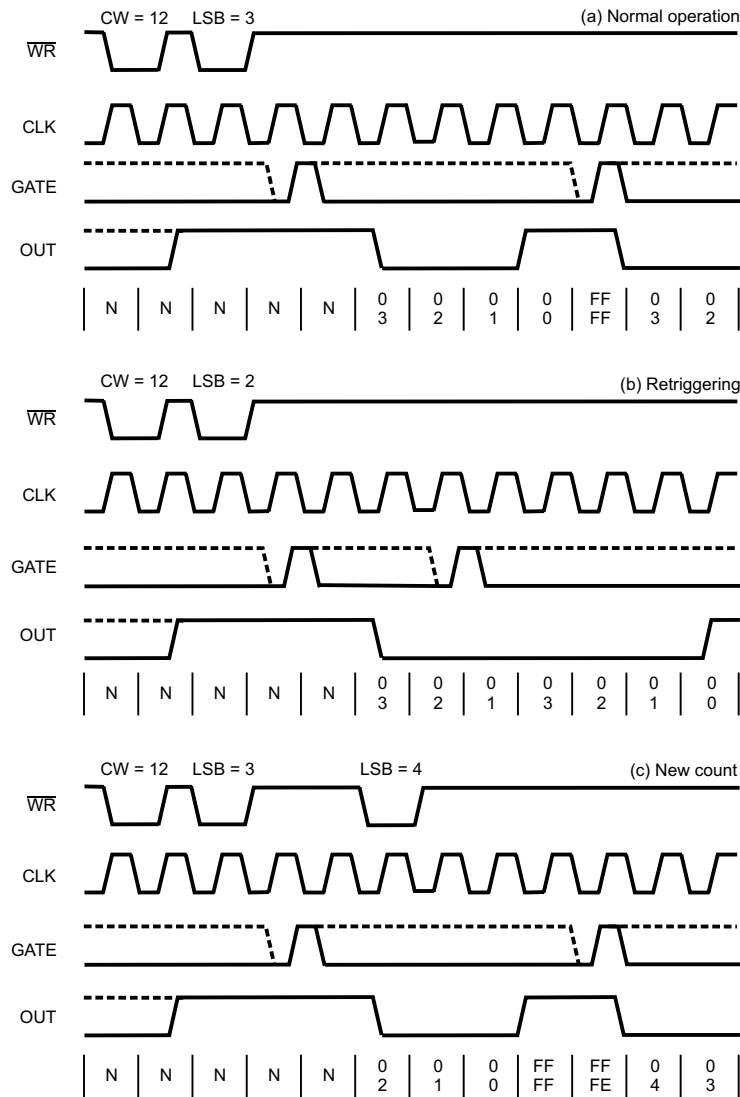
Fig. 9.6 Mode 0 interrupt on terminal count

**Mode 0 : Interrupt on terminal count**

**a) Normal Operation :** 1) The output will be initially low after the mode set operation. 2) After the count is loaded into the selected count Register the output will remain low and the counter will count. 3) When the terminal count is reached the output will go high and remain high until the selected count is reloaded.

**b) Gate Disable**

- 1) Gate = 1 enables counting.
  - 2) Gate = 0 disables counting.
- Note : Gate has no effect on OUT.



**Fig. 9.7 Mode 1 hardware retriggerable one-shot**

**c) New Count**

If a new count is written to the counter, it will be loaded on the next CLK pulse and counting will continue from the new count

**In case of two byte count :**

- 1) Writing the first byte disables counting.
- 2) Writing the second byte loads the new count on the next CLK pulse and counting will continue from the new count.

**MODE 1 : Hardware Retriggerable One-shot****a) Normal operation**

- 1) The output will be initially high
- 2) The output will go low on the CLK pulse following the rising edge at the gate input.
- 3) The output will go high on the terminal count and remain high until the next rising edge at the gate input.

**b) Retriggering**

The one shot is retriggerable, hence the output will remain low for the full count after any rising edge of the gate input.

**c) New count**

If the counter is loaded during one shot pulse, the current one shot is not affected unless the counter is retriggered. If retriggered, the counter is loaded with the new count and the one-shot pulse continues until the new count expires.

**MODE 2 : Rate generator**

This mode functions like a divide by-N counter.

**a) Normal Operation**

- 1) The output will be initially high.
- 2) The output will go low for one clock pulse before the terminal count.
- 3) The output then goes high, the counter reloads the initial count and the process is repeated.
- 4) The period from one output pulse to the next equals the number of input counts in the count register.

**b) Gate Disable**

- 1) If Gate = 1 it enables a counting otherwise it disables counting (Gate = 0 ).
- 2) If Gate goes low during an low output pulse, output is set immediately high. A trigger reloads the count and the normal sequence is repeated.



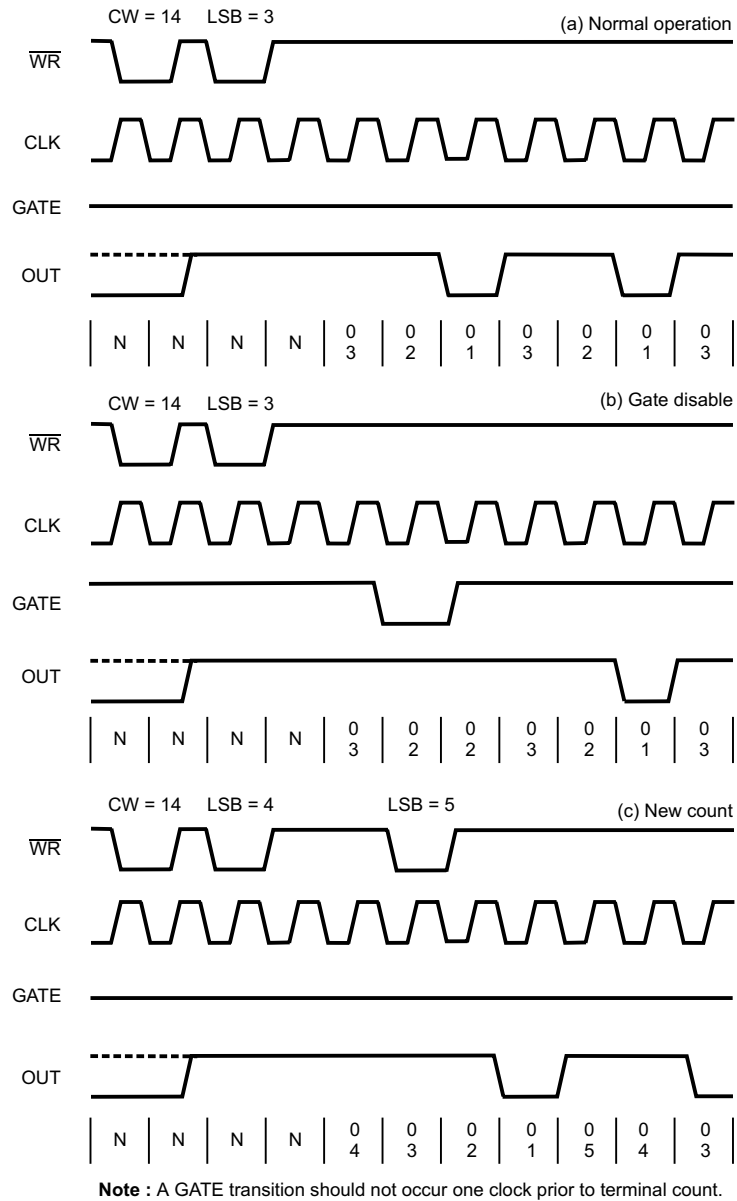


Fig. 9.8 Mode 2 rate generator

**c) New count** The current counting sequence does not affect when the new count is written. If a trigger is received after writing a new count but before the end of the current period, the new count will be loaded with the new count on the next CLK pulse and counting will continue from the new count. Otherwise, the new count will be loaded at the end of the current counting cycle.

**Note :** In mode 2, a count of 1 is illegal.

## MODE 3 : Square Wave Rate Generator

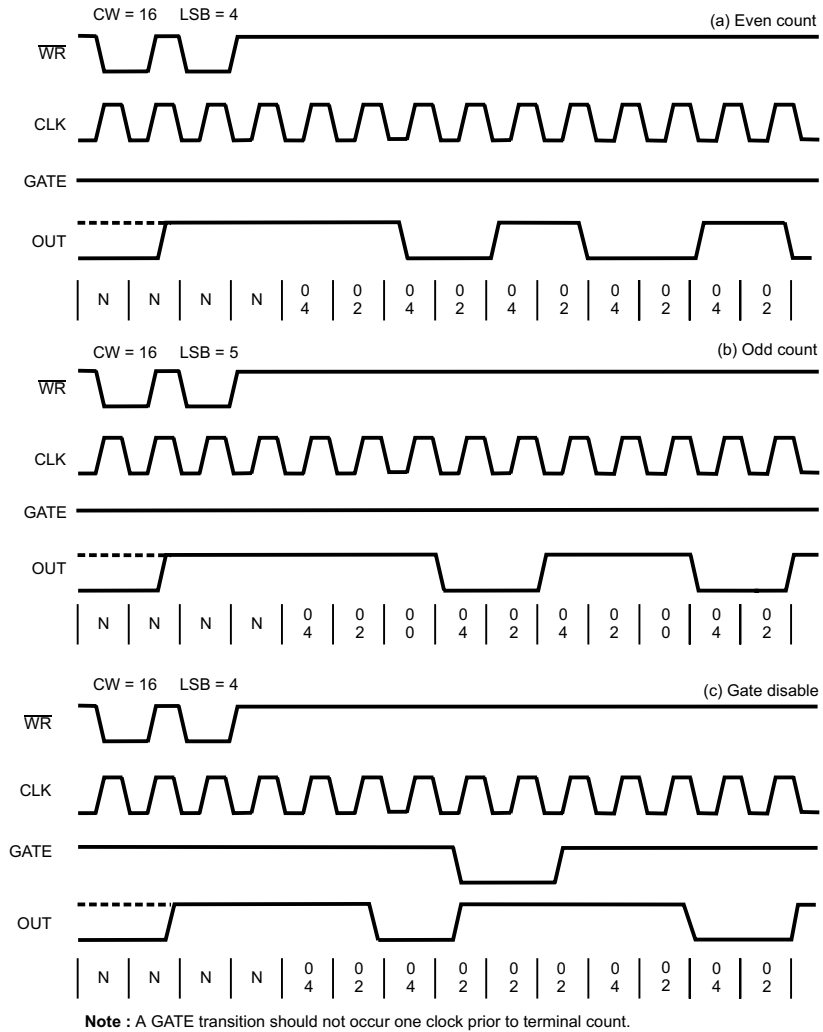


Fig. 9.9 Mode 3 square wave mode

## a) Normal operation

- 1) Initially output is high.
- 2) For even count, counter is decremented by 2 on the falling edge of each clock pulse. When the counter reaches terminal count, the state of the output is changed and the counter is reloaded with the full count and the whole process is repeated.
- 3) If the count is odd and the output is high the first clock pulse (after the count is loaded) decrements the count by 1. Subsequent clock pulses decrement the clock by 2. After timeout, the output goes low and the full count is reloaded. The first clock pulse (following the reload) decrements the count by 3 and subsequent clock pulse decrement the count by two. Then the whole process is

repeated. In this way, if the count is odd, the output will be high for  $(n+1)/2$  counts and low for  $(n-1)/2$  counts.

**b) Gate Disable**

If Gate is 1 counting is enabled otherwise it is disabled. If Gate goes low while output is low, output is set high immediately. After this, When Gate goes high, the counter is loaded with the initial count on the next clock pulse and the sequence is repeated.

**c) New Count**

The current counting sequence does not affect when the new count is written. If a trigger is received after writing a new count but before the end of the current half-cycle of the square wave, the counter will be loaded with the new count on the next CLK pulse and counting will continue from the new count. otherwise, the new count will be loaded at end of the current half-cycle.

**MODE 4 : Software Triggered Strobe.****a) Normal operation**

- 1) The output will be initially high
- 2) The output will go low for one CLK pulse after the terminal count (TC).

**b) Gate Disable**

If Gate is one the counting is enabled otherwise it is disabled. The Gate has no effect on the output.

**c) New count**

If a new count is written during counting, it will be loaded on the next CLK pulse and counting will continue from the new count. If the count is two byte then

- 1) Writing the first byte has no effect on counting.
- 2) Writing the second byte allows the new count to be loaded on the next CLK pulse.

**MODE 5 : Hardware triggered strobe (Retriggerable).****a) Normal operation**

- 1) The output will be initially high.
- 2) The counting is triggered by the rising edge of the Gate.
- 3) The output will go low for one CLK pulse after the terminal count (TC).

**b) Retriggering**

If the triggering occurs on the Gate input during the counting, the initial count is loaded on the next CLK pulse and the counting will be continued until the terminal count is reached.

**c) New count**

If a new count is written during counting, the current counting sequence will not be affected. If the trigger occurs after the new count is written but before the terminal

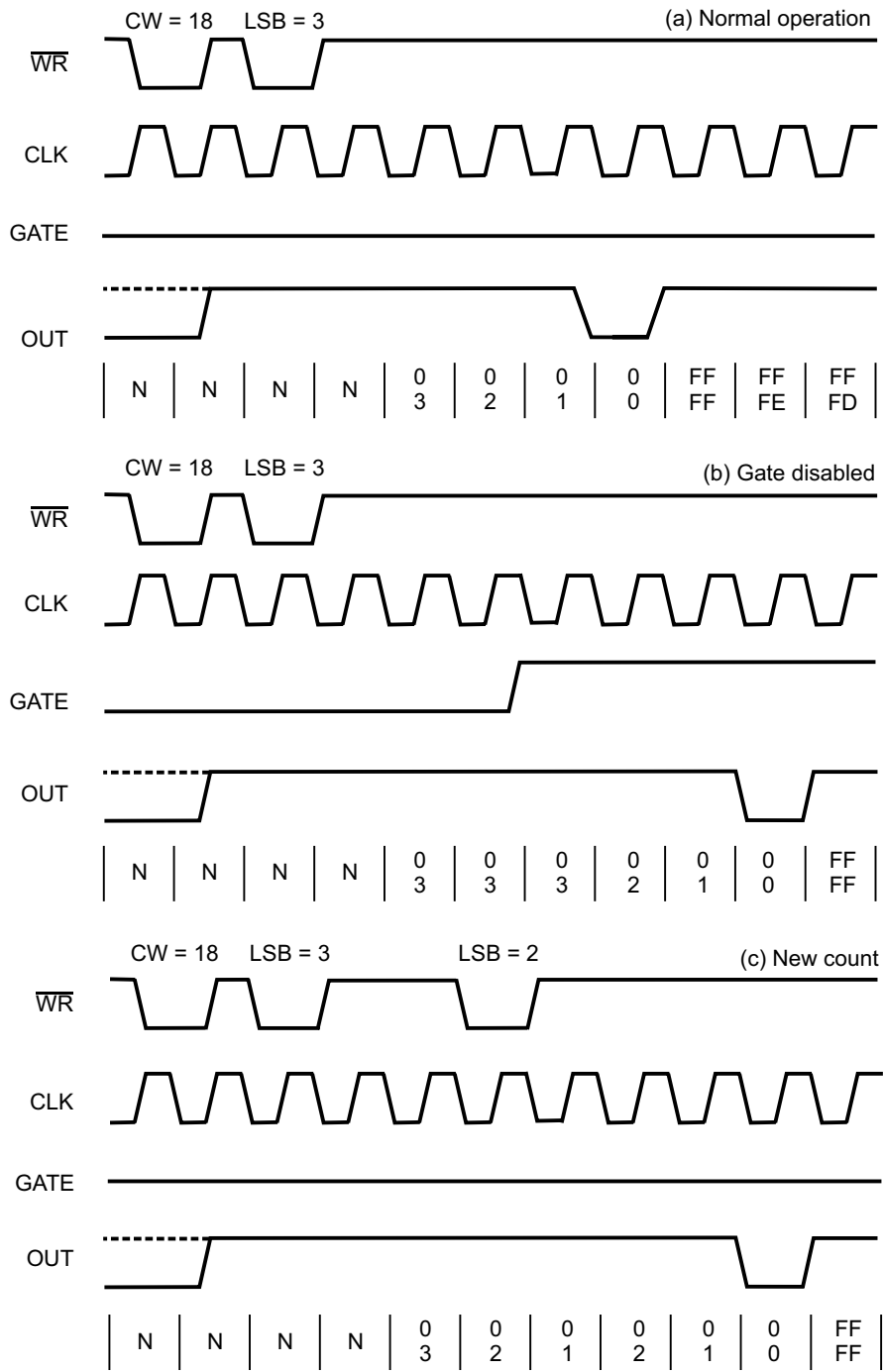


Fig. 9.10 Mode 4 software triggered strobe

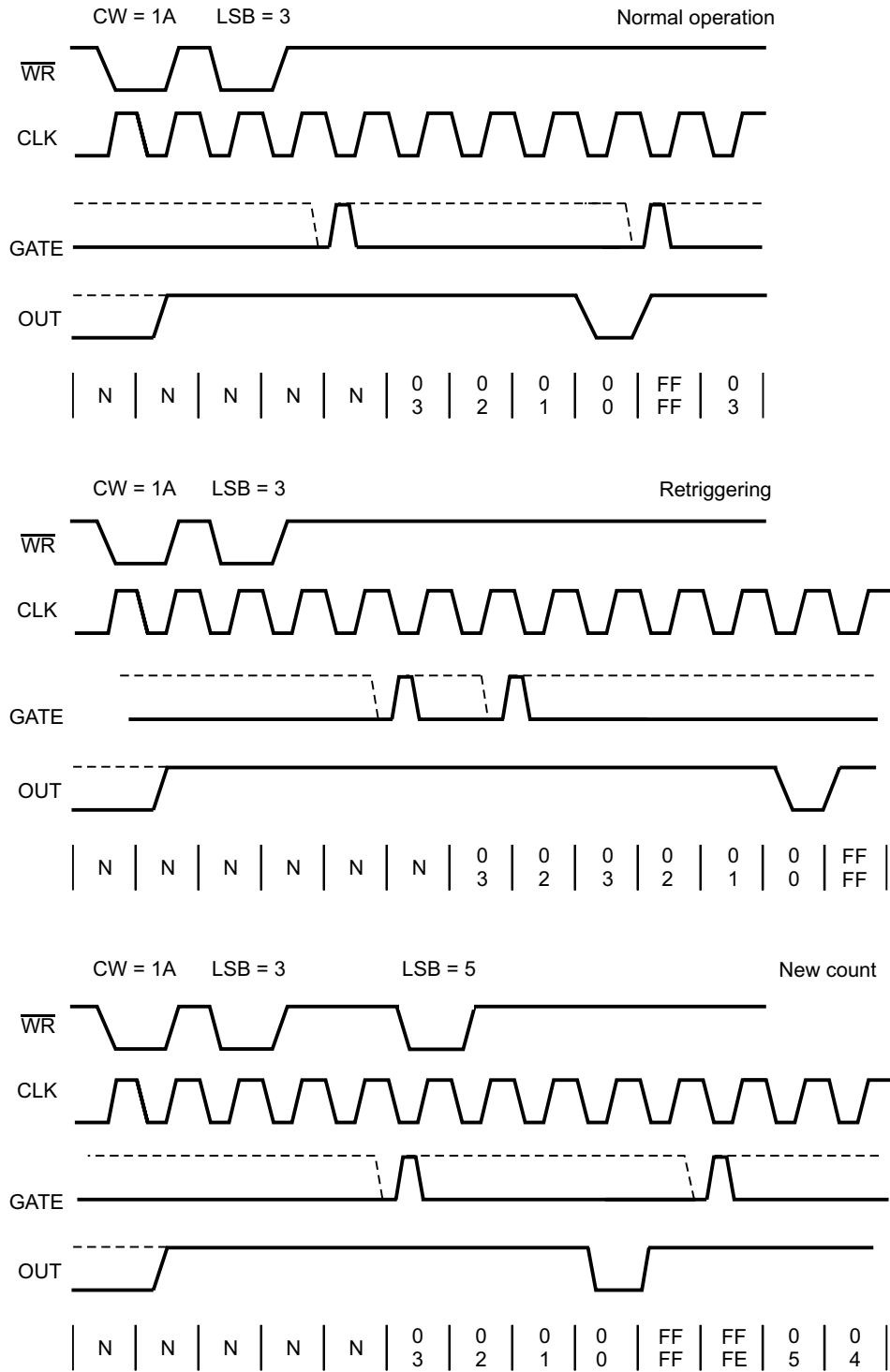


Fig. 9.11 Mode 5 hardware triggered strobe

count, the counter will be loaded with the new count on the next CLK pulse and counting will continue from there.

Table 9.1 shows the gate pin operations.

Signal Status Modes	Low or Going Low	Rising	High
0	Disables counting	—	Enables counting
1	—	i) Initiates counting ii) Resets output after next clock	—
2	i) Disables counting ii) Sets output immediately high	Initiates counting	Enables counting
3	i) Disables counting ii) Sets output immediately high	Initiates counting	Enables counting
4	Disables counting	—	Enables counting
5	—	Initiates counting	—

Table 9.1 Gate pin operations

## 9.6 Programming Examples

**Example 1:** Write a program to initialize counter 2 in mode 0 with a count of C030H. Assume address for control register = 0BH, counter 0 = 08H, counter 1 = 09H and counter 2 = 0AH.

**Sol. :** Control word

D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>	
SC <sub>1</sub>	SC <sub>2</sub>	RW <sub>1</sub>	RW <sub>0</sub>	M <sub>2</sub>	M <sub>1</sub>	M <sub>0</sub>	BCD	
1	0	1	1	0	0	0	0	= B0H

### Source Program

```

MOV AL,B0H
OUT 0BH,AL           ; Loads control word (B0H) in the control
                     ; register.

MOV AL,30H
OUT 0AH,AL           ; Loads lower byte of (30H)the count.

MOV AL,C0H
OUT 0AH,AL           ; Loads higher byte (C0H) of the count.

```