Lecture 9: 8253/4 Timer and Music

Prof. Xiangzhong FANG

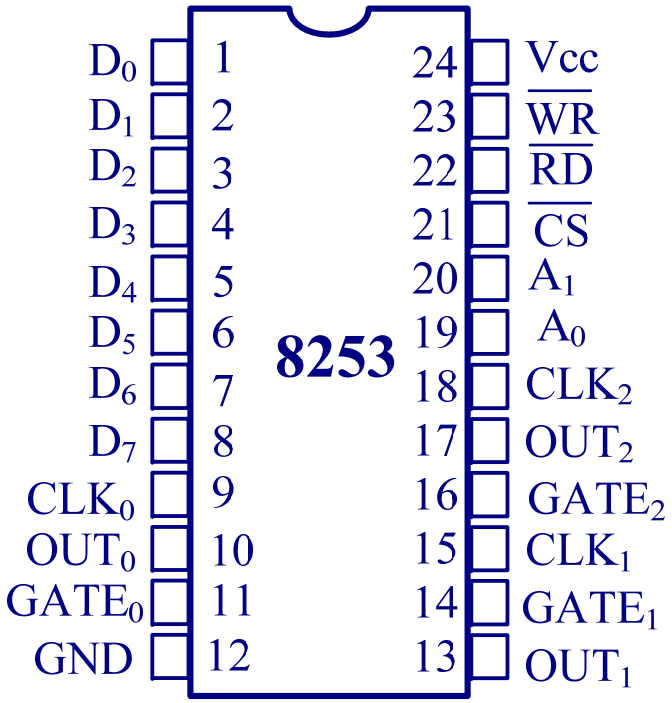
xzfang@sjtu.edu.cn

The 80x86 IBM PC and Compatible Computers

Chapter 13

8253/4 Time and Music

Package & Internal Structure



The 8253/54 Programmable interval timer is used to generate a lower frequency for various uses

e.g.,
Accurate time delays

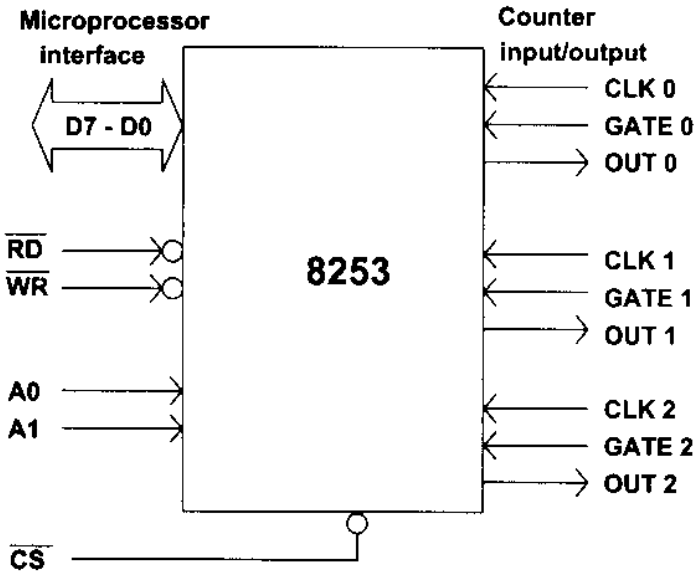
Software:
setting up a timing loop

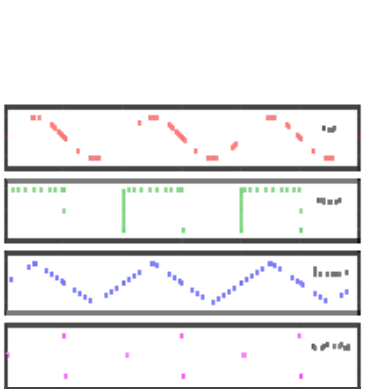
```
        MOV  CX, N
AGAIN: LOOP AGAIN
```

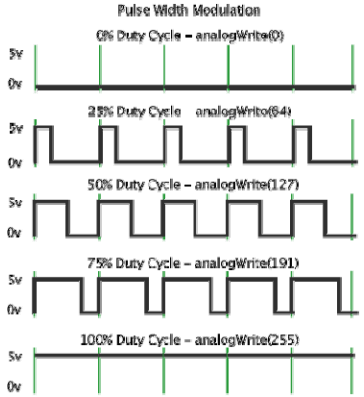
Hardware:
using 8253 to count out the delay and interrupt the CPU

Pros and cons?

Interface to the System

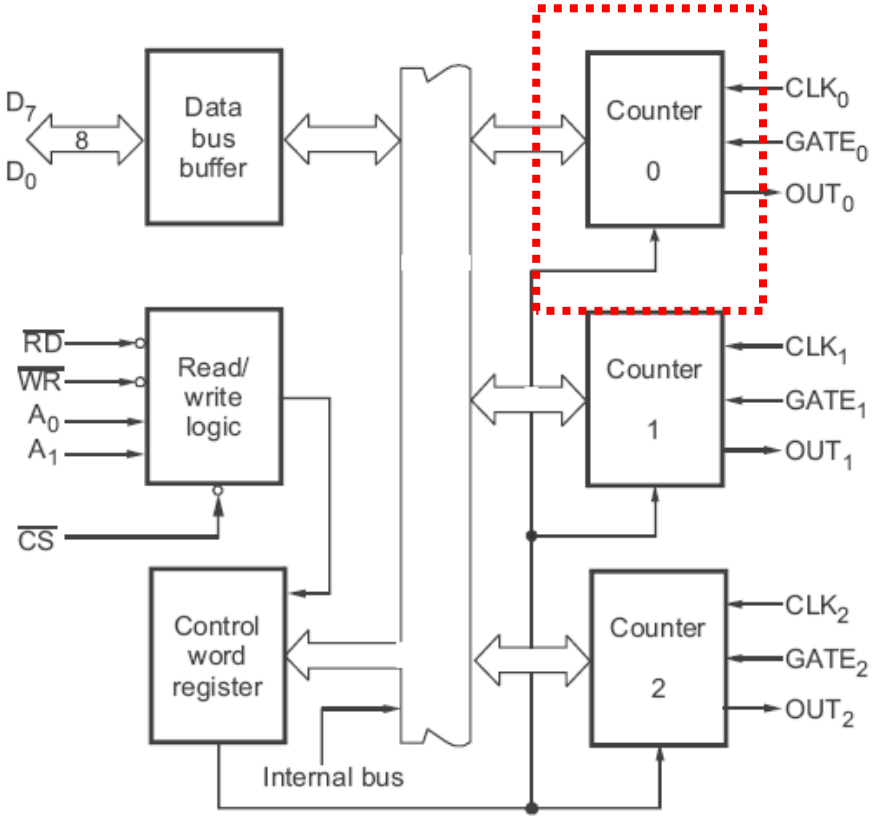


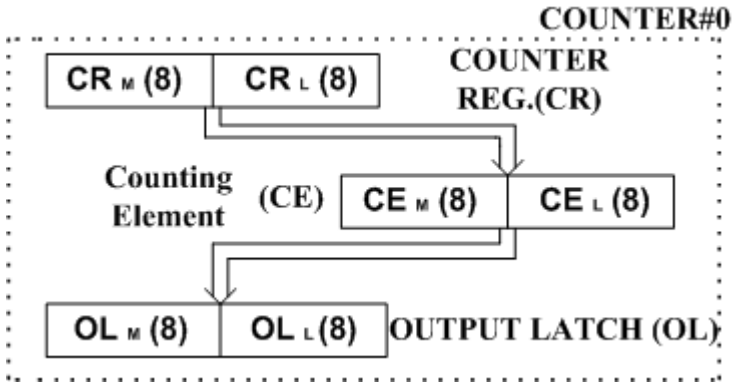




- There are three independent counters.
- The input frequency can be divided from 1 to 65536 (Binary), or 1 to 10000 (BCD)
- Shape of the output frequency:
  - Square-wave
  - One-shot
  - Square-wave with various duty cycles.
- Gate is used to enable (High) or disable (Low) the counter.
- Bidirectional bus D0-D7 is connected to D0-D7 of the system bus.

Internal Structure





To operate a 16-bit down counter

- a 16-bit count is loaded in the counter
- begins to decrement the count until it reaches 0
- generates a pulse that can be used to interrupt the CPU

Features

- Three independent 16-bit down counters
- 8254 can handle inputs from DC to 10 MHz (5MHz 8254-5 8MHz 8254 10MHz 8254-2) where as 8253 can operate up to 2.6 MHz
- Three counters are identical pre-settable, and can be programmed for either binary or BCD count
- Counter can be programmed in six different modes
- Compatible with all Intel and most other microprocessors
- 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

Internal Structure & Pins

- **Data bus buffer**
  - interface the 8253/4 to the system data bus
  - Bi-directional, tri-state, 8-bit

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

- **Read / Write control logic**
  - **~CS**
    - Tied to a decoded address
  - **~RD**, **~WR**
    - In isolated I/O: ~IOR, ~IOW
    - Memory-mapped I/O: ~MEMR, ~MEMW
  - **A_1, A_0**
    - Select the control word register and counters
    - usually connected to address lines A_1, A_0

/CS	/RD	/WR	A1A0	FUNCTION
0	1	0	00	Write counter0(to CR0)
0	1	0	01	Write counter1(to CR1)
0	1	0	10	Write counter2(to CR2)
0	1	0	11	Write control port
0	0	1	00	Read counter0(from OL0)
0	0	1	01	Read counter1(from OL1)
0	0	1	10	Read counter2(from OL2)
0	0	1	11	Read control port (*8254*)
1	X	X	XX	Not available

Internal Structure & Pins

- **Control Word Register:**
  - Selected when A_1=1, A_0=1
  - Used to specify which counter to be used, its mode, and a read or write operation
- **Counters:**
  - Each consists of a single, 16-bit, pre-settable, down counter
  - Can operate in either binary or BCD
  - Input, gate and output are configured by the selection of modes
  - Reading from a counter does not disturb the actual count in process

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
SC_1	SC_0	RW_1	RW_0	M_2	M_1	M_0	BCD

SC - Select counter

SC_1	SC_0	
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)

RW - Read /Write

RW_1	RW_0	
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

M - Mode

M_2	M_1	M_0	
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

BCD :

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

Write/Read Operations

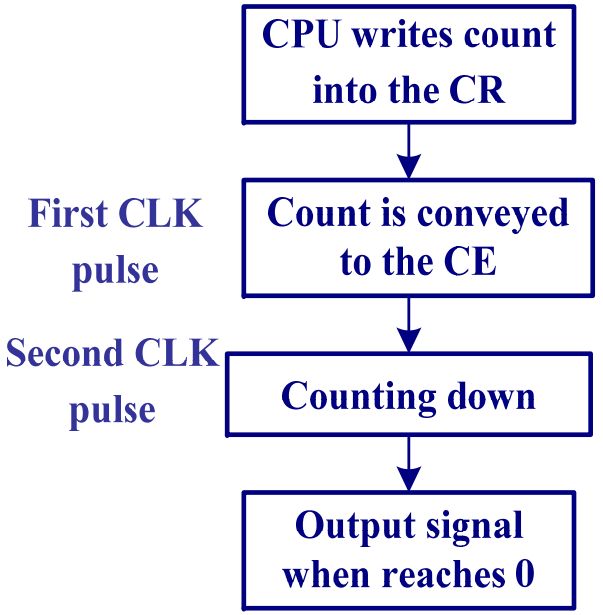
- WRITE:
  - Write a control word into control register
  - Load the low-order byte of a count in the counter register
  - Load the high-order byte of a count in the counter register
- READ:
  - **Simple Read**: two I/O read operations, first one for low-order byte and last one for the high order byte
  - **Counter Latch Command**: one I/O write operation used to write a control word to the control register to latch a count in the output latch, then two I/O read operations are used to read the latched count as in Simple Read.
  - **Read-Back Command**: for 8254 only

Features of 8253

- 8253 takes one CLK pulse to convey the count from CR to CE
- CE will start to count only when GATE = 1
  - *When check the GATE?*
  - On every CLK pulse's rising edge
  - *When act?*
  - On every CLK pulse's falling edge

Mode 0 : Interrupt on Terminal Count (1)

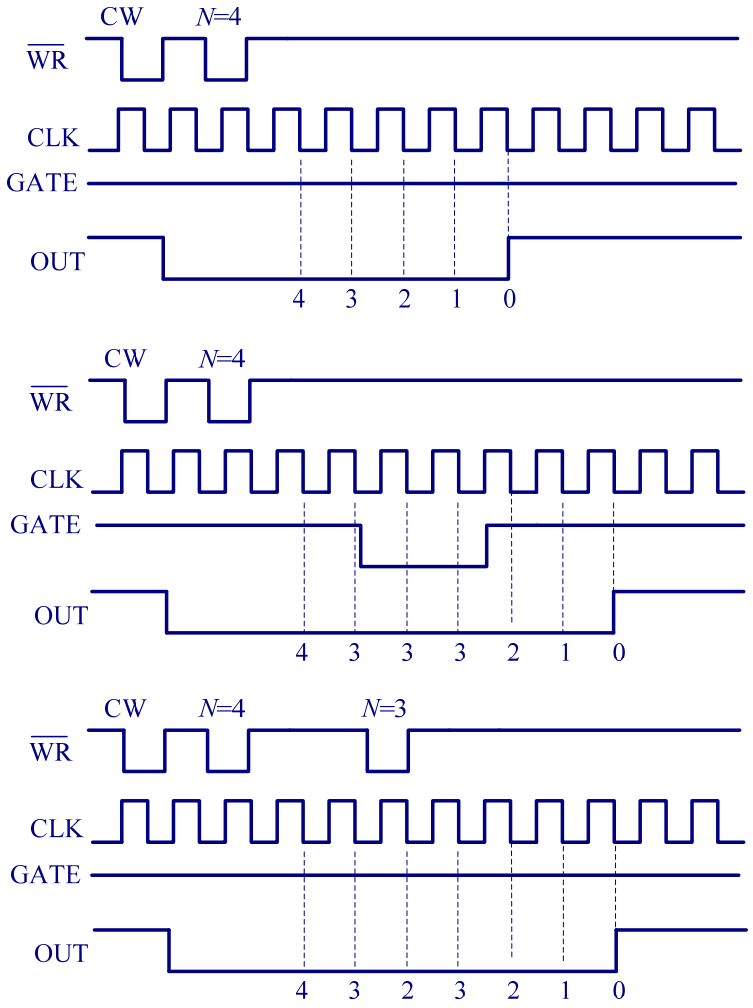
- Normal Operation:
  - The output will be initially **low** after the mode set operation;
  - After the count is loaded into the selected CR the output will remain **low**
  - When the terminal count is reached, the output will go **high** and remain high until the selected counter is reloaded



Mode 0 : Interrupt on Terminal Count (2)

- Gate disable:
  - Gate = 1 enables counting
  - Gate = 0 disables counting
- 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:
    - Writing the first byte disables counting
    - Writing the second byte loads the new count on the next CLK pulse and counting will continue from the new count

Mode 0: Interrupt on Terminal Count (3)



When loading a new count *N*, the actual number of CLK pulses in OUT is *N*+1

Does not automatically repeat

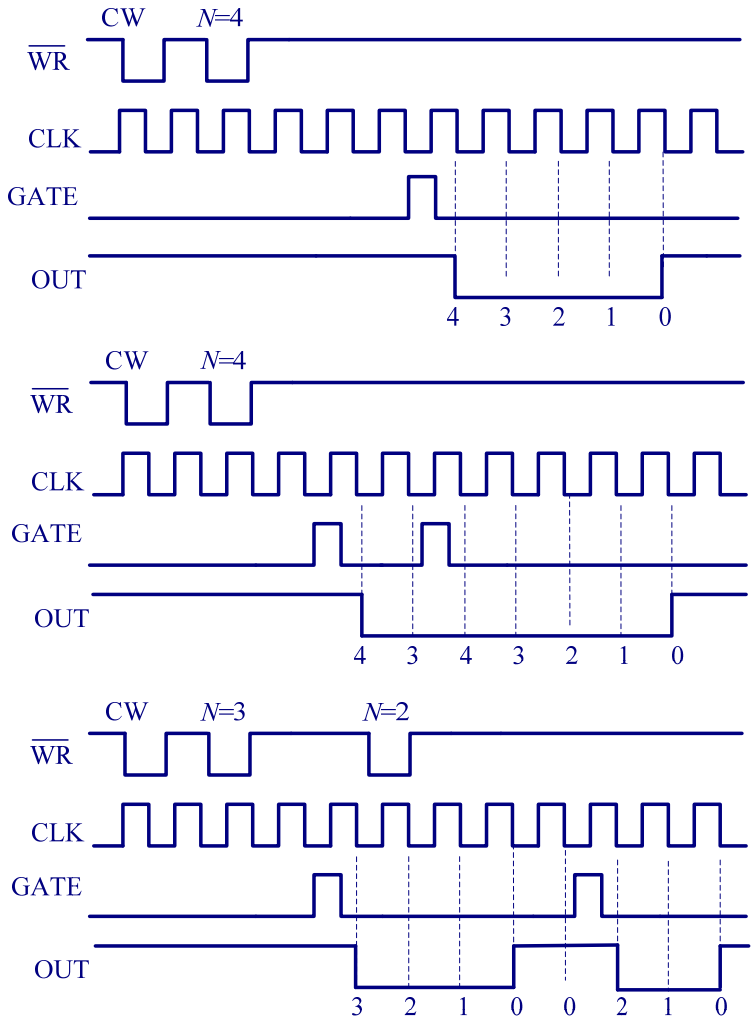
Mode 1: Hardware Retriggerable One-shot (1)

- Normal Operation:
  - The output will be initially **high** after the mode set operation;
  - The output will go **low** on the CLK pulse following the rising edge at the gate input;
  - The output will go **high** on the terminal count and remain high until the next rising edge at the gate input.

Mode 1 : Hardware Retriggerable One-shot (2)

- Retriggering:
  - retriggerable, hence the output will remain low for the full count after any rising edge of the gate input
- 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 1 : Hardware Retriggerable One-shot (3)



When loading a new count *N*, the current counting will not be affected

Does not automatically repeat

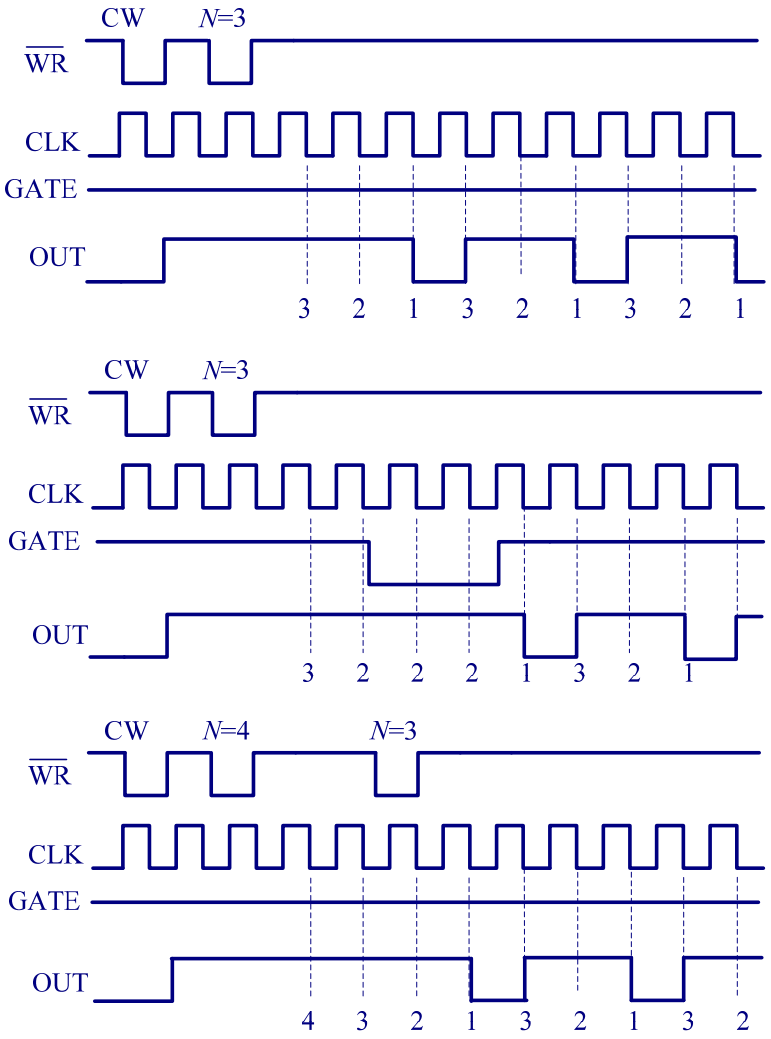
Mode 2: Rate Generator (1)

- Normal Operation:
  - The output will be initially **high**;
  - The output will go **low** for one clock pulse before the terminal count;
  - The output then goes **high**, the counter reloads the initial count and the process is repeated
  - *The period from one output pulse to the next **equals** the number of input counts in the count register*

Mode 2: Rate Generator (2)

- Gate disable:
  - If Gate=1 it enables a counting otherwise it disables counting (Gate=0)
  - If Gate goes low during an low output pulse, output is set immediately high
- New count:
  - The current counting sequence is not affected 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 2: Rate Generator (3)

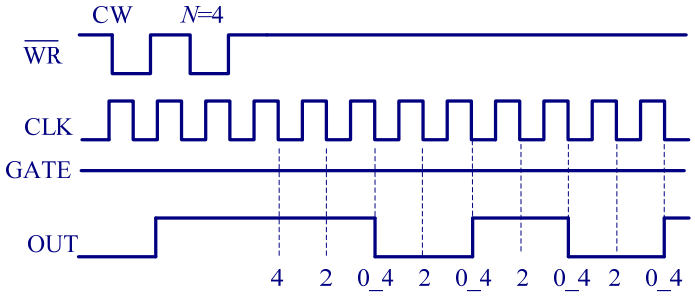


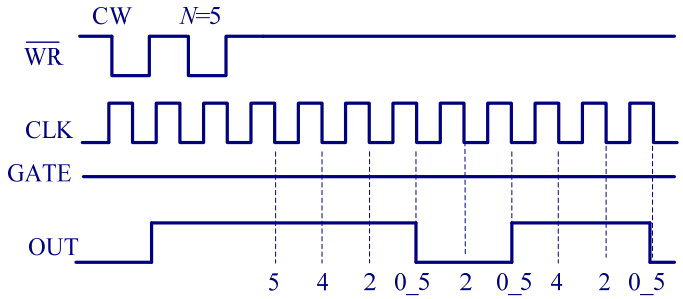
When loading a new count *N*, the current counting will not be affected

Automatically repeat on terminal count

Mode 3: Square Wave Rate Generator (1)

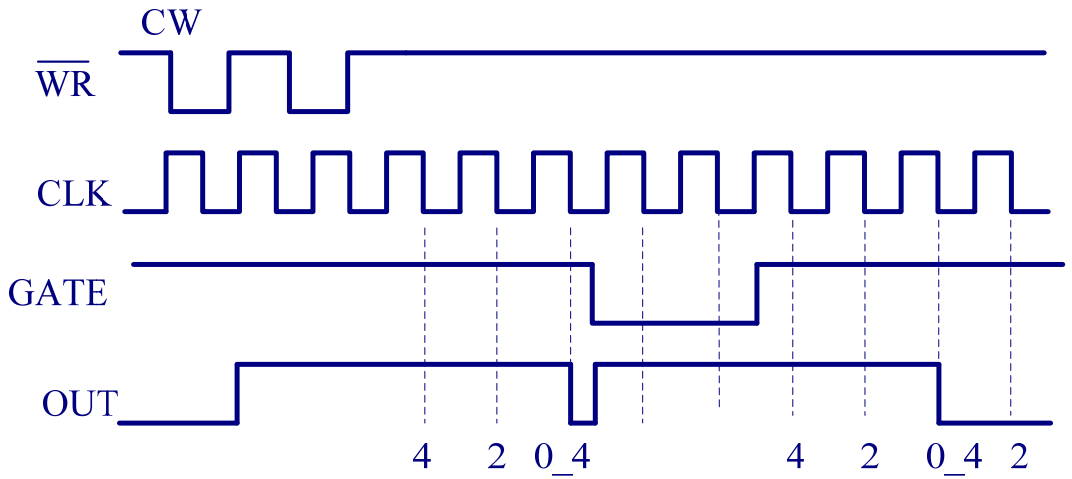
- Normal Operation:
  - The output will be initially high;
  - For even count, counter is decremented by 2 on the falling edge of each clock pulse; when reaches terminal count, the state of the output is changed and the counter is reloaded with the full count and the whole process is repeated
  - For odd count, the first clock pulse 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.***





Mode 3: Square Wave Rate Generator (2)

- 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.



Mode 3: Square Wave Rate Generator (3)

- 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.

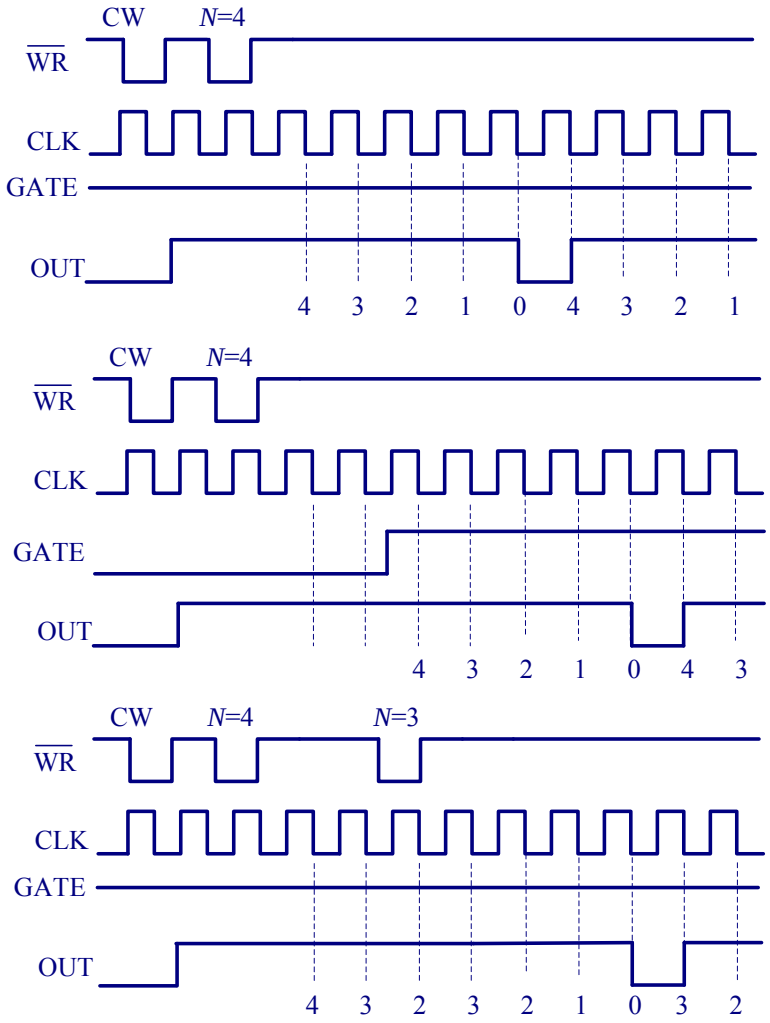
When loading a new count *N*, the current half will not be affected

Automatically repeat on terminal count

Mode 4: Software Triggered Strobe (1)

- Normal Operation:
  - The output will be initially **high**;
  - The output will go **low** for one CLK pulse after the terminal count
- Gate disable:
  - If Gate is one, the counting is enabled; otherwise, it is disabled
- 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:
    - Writing the first byte has no effect on counting
    - Writing the second byte allows the new count to be loaded on the next CLK pulse

Mode 4: Software Triggered Strobe (2)



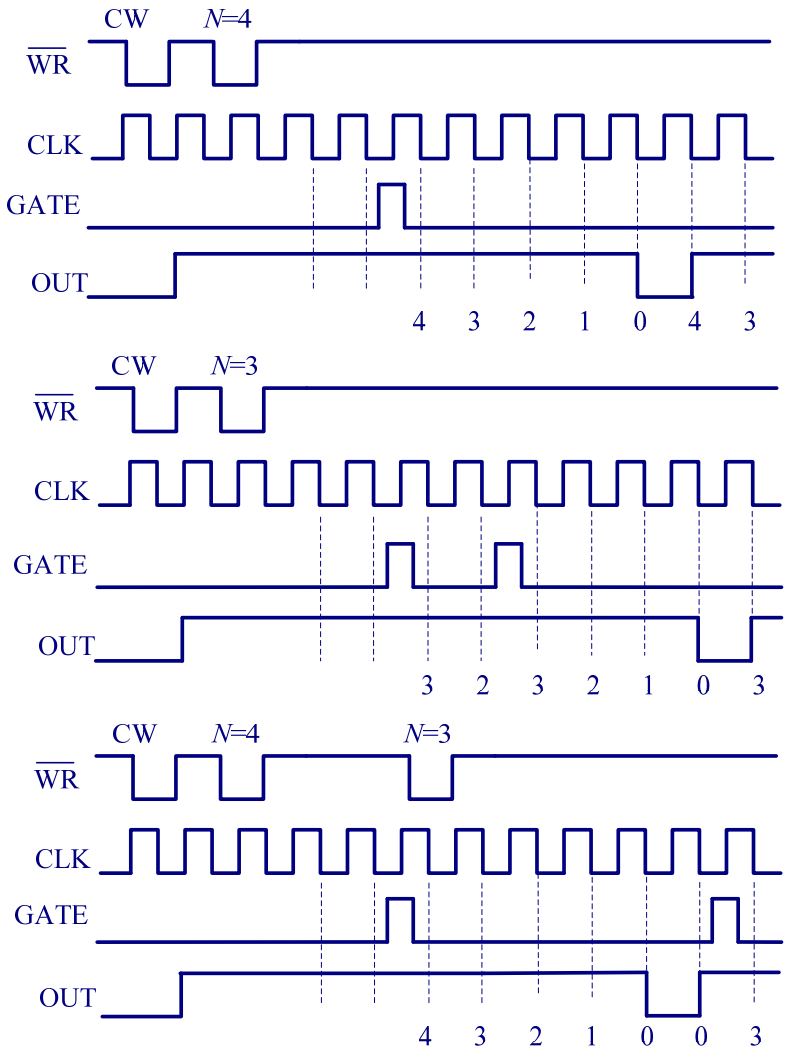
When loading a new count *N*, the actual number of CLK pulses in OUT is *N*+1

Automatically repeat

Mode 5: Hardware Triggered Strobe (Retriggerable) (1)

- Normal Operation:
  - The output will be initially **high**;
  - The counting is triggered by the rising edge of the Gate
  - The output will go **low** for one CLK pulse after the terminal count
- Retriggering:
  - If the triggering occurs 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
- New count:
  - the current counting sequence will not be affected. If the trigger occurs after the new count but before the terminal count, the counter will be loaded with the new count on the next CLK pulse and counting will continue from there

Mode 5: Hardware Triggered Strobe (Retriggerable) (2)



When loading a new count *N*, the current counting will not be affected

Automatically repeat on terminal count

Programming Example

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_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
SC_1	**SC_2**	**RW_1**	**RW_0**	**M_2**	**M_1**	**M_0**	**BCD**
1	0	1	1	0	0	0	0

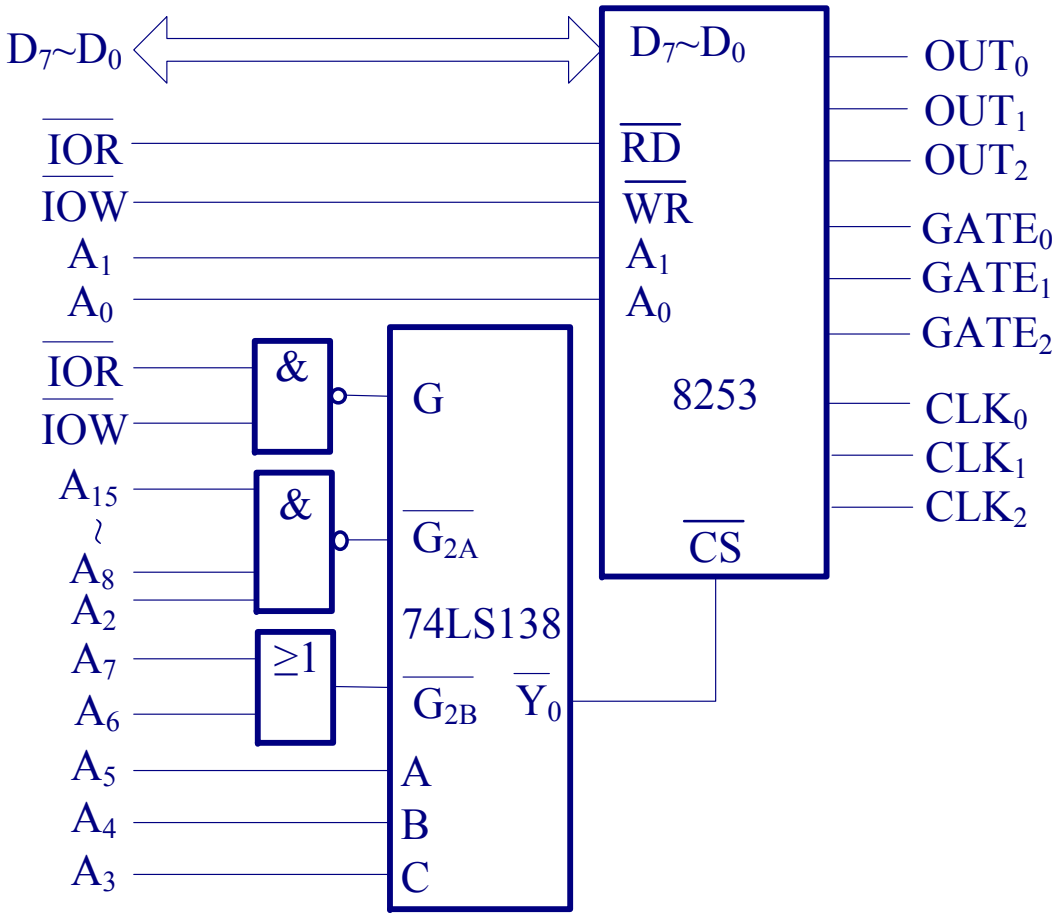
= B0H

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

Example & Quiz

- The frequency of CLK is 2MHz, write initiation program to let counter 0 generate an interruption request after 100μs, let counter 1 generate 50% duty cycle square wave with a period of 10μs, and let counter 2 generate a negative pulse every 1ms.



```
MOV DX, 0FF07H
MOV AL, 00010000B          ;counter 0, write LSB only, mode 0, binary
OUT DX, AL
MOV AL, 01010110B          ;counter 1, write LSB only, mode 3, binary
OUT DX, AL

MOV DX, 0FF04H
MOV AL, 200                ; initial count for counter 0
OUT DX, AL
MOV DX, 0FF05H
MOV AL, 20                 ;initial count for counter 1
OUT DX, AL
MOV DX, 0FF06H

MOV DX, 0FF07H
MOV AL, 10110100B          ;counter 2, write LSB and MSB, mode 2
OUT DX, AL

MOV DX, 0FF06H
MOV AX, 2000               ; initial count for counter 2
OUT DX, AL
MOV AL, AH
OUT DX, AL
```