

**dce**  
2019

## Digital Systems

### FLIP-FLOPs

BK  
TP.HCM

**dce**  
2019

## Introduction

- So far we have seen Combinational Logic
  - The output(s) depends only on the current values of the input variables
- Here we will look at Sequential Logic circuits
  - The output(s) can depend on present and also past values of the input and the output variables
- Sequential circuits exist in one of a defined number of states at any one time
  - They move "sequentially" through a defined sequence of transitions from one state to the next
  - The output variables are used to describe the state of a sequential circuit either directly or by deriving state variables from them

BK  
TP.HCM

2

**dce**  
2019

## General Digital System

3

BK  
TP.HCM

**dce**  
2019

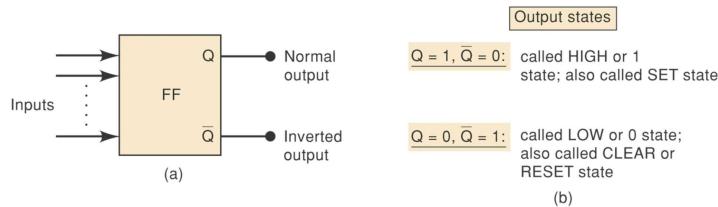
## Synchronous and Asynchronous Sequential Logic

- Synchronous
  - The timing of all state transitions is controlled by a common clock
  - Changes in all variables occur simultaneously
- Asynchronous
  - State transitions occur independently of any clock and normally dependent on the timing of transitions in the input variables
  - Changes in more than one output do not necessarily occur simultaneously
- Clock
  - A clock signal is a square wave of fixed frequency
  - Often, transitions will occur on one of the edges of clock pulses
    - i.e. the rising edge or the falling edge

BK  
TP.HCM

4

General flip-flop symbol and definition of its two possible output states



- We now introduce the concept of memory. The **flip-flop**, abbreviated FF, is a key memory element.
- The outputs of a flip flop are  $Q$  and  $Q'$
- $Q$  is understood to be the normal output,  $Q'$  is always the opposite.

5

## NAND Gate Latch

- The NAND gate latch or simply latch is a basic FF.
- The inputs are **set** and **clear (reset)**
- The inputs are active low, that is, the output will change when the input is pulsed low.
- When the latch is set

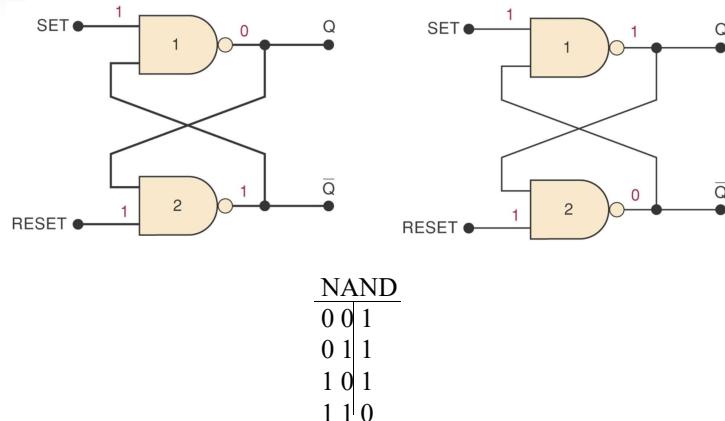
$$Q = 1 \text{ and } \bar{Q} = 0$$

- When the latch is clear or reset

$$Q = 0 \text{ and } \bar{Q} = 1$$

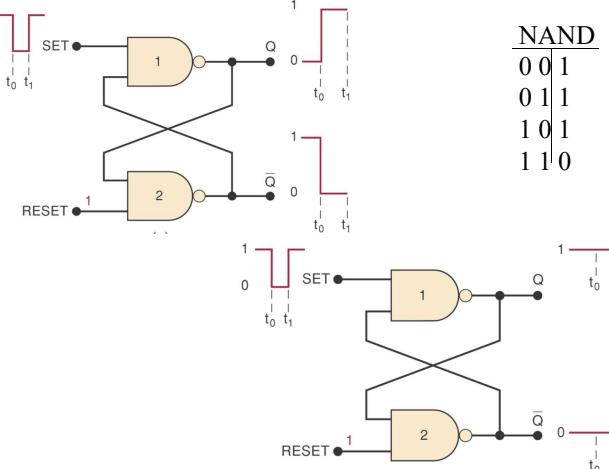
6

A NAND latch is an example of a bistable device



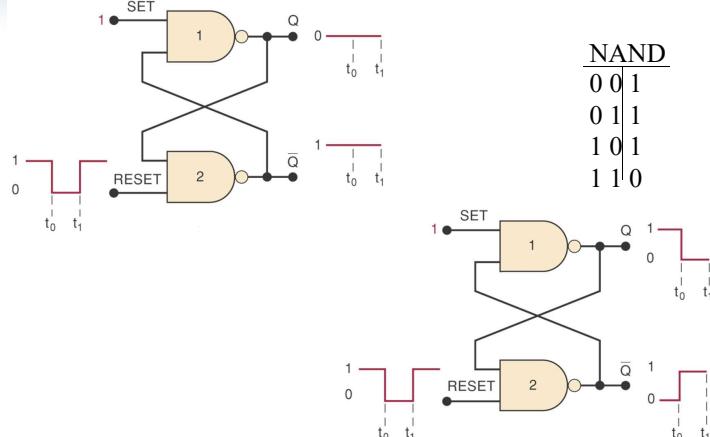
7

## Setting the NAND Flip-Flop



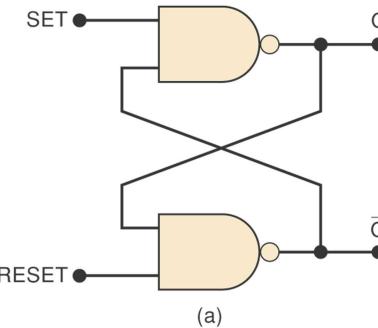
8

## Resetting the NAND Flip-Flop



9

## Function table of a NAND latch



Set	Reset	Output
1	1	No change
0	1	$Q = 1$
1	0	$Q = 0$
0	0	Invalid*

\*Produces  $Q = \bar{Q} = 1$ .

(b)

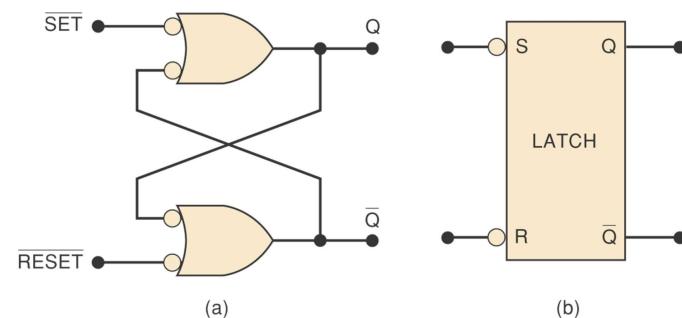
10

## NAND Gate Latch

- Summary of the NAND latch:
  - SET = RESET = 1. Normal resting state, outputs remain in state prior to input.
  - SET = 0, RESET = 1. Q will go high and remain high even if the SET input goes high.
  - SET = 1, RESET = 0. Q will go low and remain low even if the RESET input goes high.
  - SET = RESET = 0. Output is unpredictable because the latch is being set and reset at the same time.

11

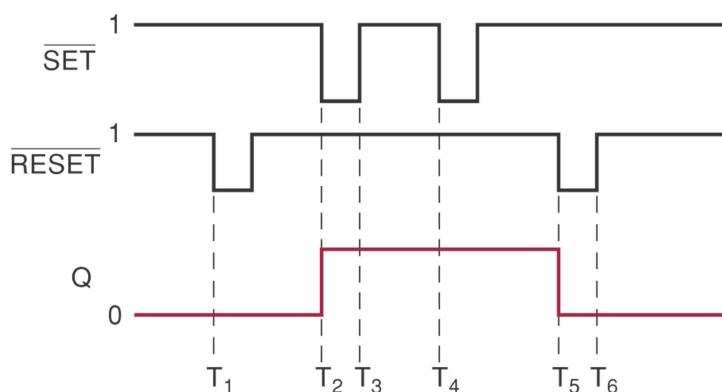
## Other Representations of a NAND latch



- Symbols indicate  $Q$  is set (high) when  $S$  is low.

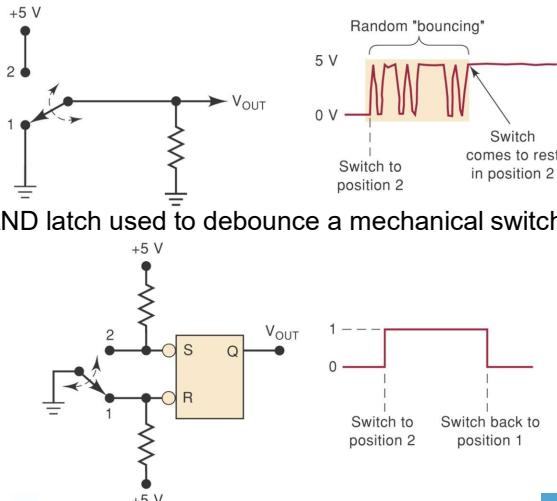
12

## Determine Q



13

## A example of NAND latch Application



14

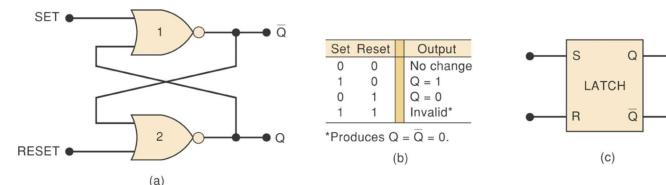
## NOR Gate Latch

- The NOR latch is similar to the NAND latch except that the **Q** and  **$\bar{Q}$**  outputs are **reversed**.
- The **SET** and **RESET** inputs are **active high**, that is, the output will change when the input is pulsed high.
- In order to ensure that a FF begins operation at a known level, a pulse may be applied to the **SET** or **RESET** inputs when a device is powered up.

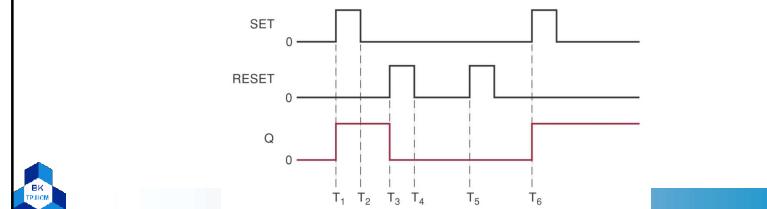
15

## NOR gate latch

- (a) NOR gate latch; (b) function table; (c) simplified block symbol.



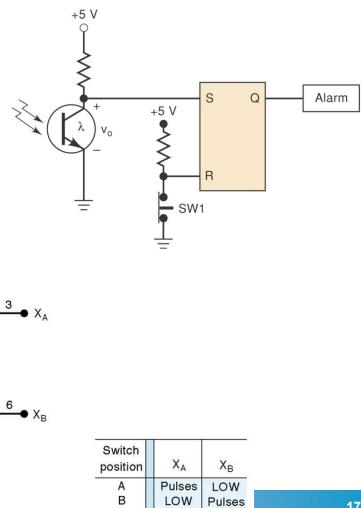
- Determine **Q** for a NOR latch given the inputs below



16

SR latch useful when temporary setting is used to activate switch

The NAND FF is used to provide a bounce free switch so that the 1 KHz pulse can propagate to the output without distortion.



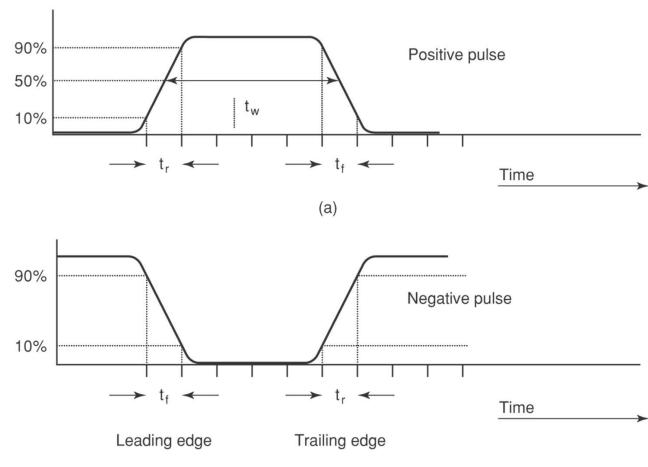
17

## Digital Pulses

- The transition from low to high on a positive pulse is called **rise time ( $t_r$ )**.
  - Rise time is measured between the 10% and 90% points on the leading edge of the voltage waveform.
- The transition from high to low on a positive pulse is called **fall time ( $t_f$ )**.
  - Fall time is measured between the 90% and 10% points on the trailing edge of the voltage waveform.

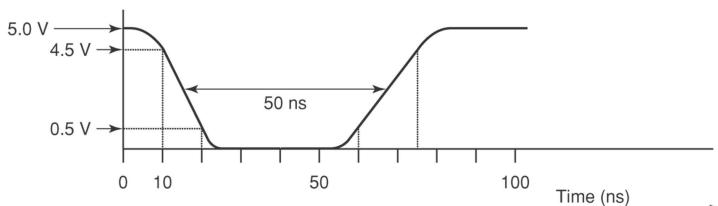
18

## Rise and Fall times



19

## Rise and Fall times



Signal that activates an active-low output with:  
 $t_w = 50\text{ns}$ ,  $t_r = 15\text{ns}$ , and  $t_f = 10\text{ ns}$ .

20

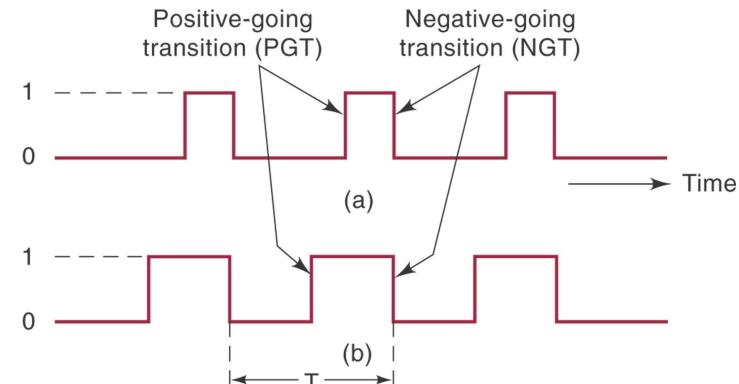
## Clock Signals and Clocked Flip-Flops

- **Asynchronous** system – outputs can change state at any time the input(s) change.
- **Synchronous** system – output can change state only at a specific time in the clock cycle.
  - The clock signal is a rectangular pulse train or square wave.
  - **Positive going transition (PGT)** – when clock pulse goes from 0 to 1.
  - **Negative going transition (NGT)** – when clock pulse goes from 1 to 0.
  - Transitions are also called edges.

21



## Ideal Clock Signals



22



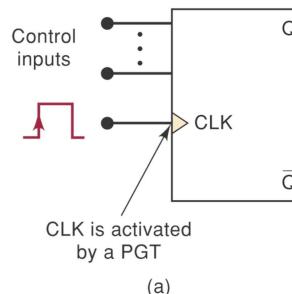
## Clock Signals and Clocked Flip-Flops

- Clocked FFs change state on one or the other clock transitions. Some common characteristics:
  - Clock inputs are labeled CLK, CK, or CP.
  - A **small triangle** at the CLK input indicates that the input is activated with a **PGT**.
  - A **bubble and a triangle** indicates that the CLK input is activated with a **NGT**.
  - Control inputs have an effect on the output only at the active clock transition (NGT or PGT). These are also called synchronous control inputs.
  - The control inputs get the FF outputs ready to change, but the change is not triggered until the CLK edge.

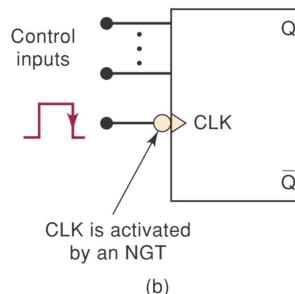
23



## Clocked Flip-Flops



CLK is activated  
by a PGT  
(a)

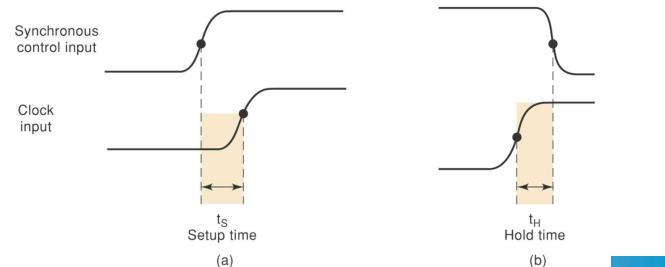


CLK is activated  
by an NGT  
(b)

24

## Clock Signals and Clocked Flip-Flops

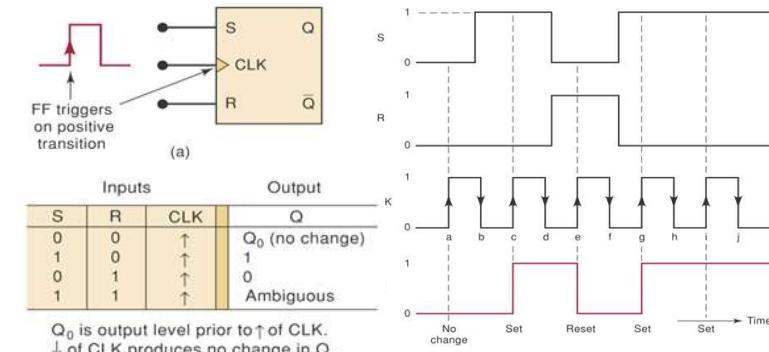
- Setup time ( $t_S$ ) is the minimum time interval **before** the active CLK transition that the **control input** must be kept at the proper level.
- Hold time ( $t_H$ ) is the time **after** the active CLK transition during which the **control input** must be kept at the proper level.



25

## Clocked S-R Flip-Flop

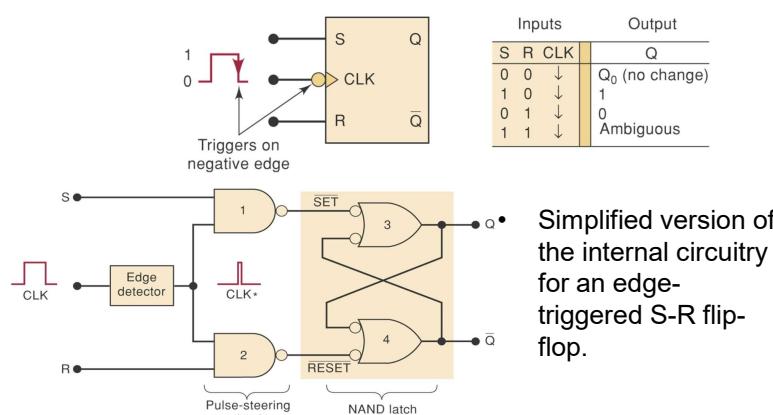
- The SET-RESET (or SET-CLEAR) FF will change states at the positive going or negative going clock edge.



26

## Clocked SR Flip-Flop

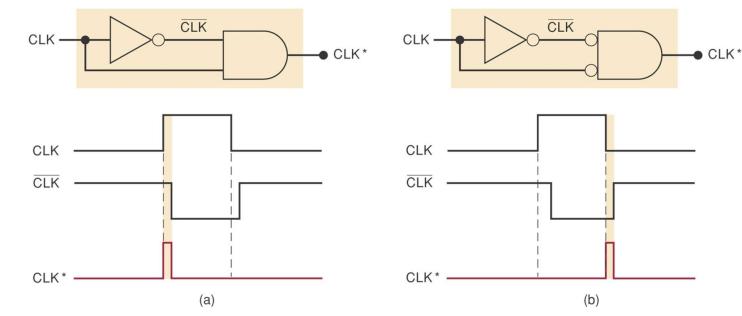
- Clocked S-R flip-flop that triggers only on negative-going transitions.



27

## Clocked SR Flip-Flop

- Implementation of edge-detector circuits used in edge-triggered flip-flops: (a) PGT; (b) NGT. The duration of the CLK\* pulses is typically 2–5 ns.



28

## Clocked J-K Flip-Flop

- Operates like the S-R FF. J is set, K is clear.
- When J and K are both high the output is **toggled** from whatever state it is in to the opposite state.
- May be positive going or negative going clock trigger.
- Has the ability to do everything the S-C FF does, plus operate in toggle mode.

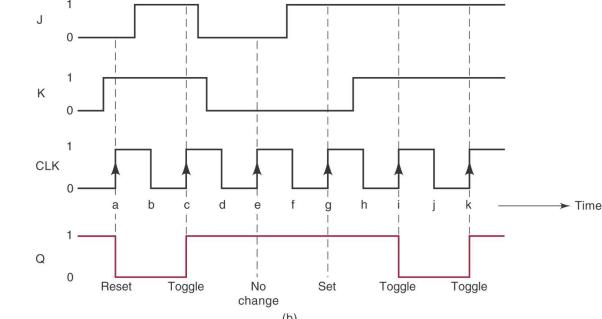
29



## Clocked JK Flip-Flop



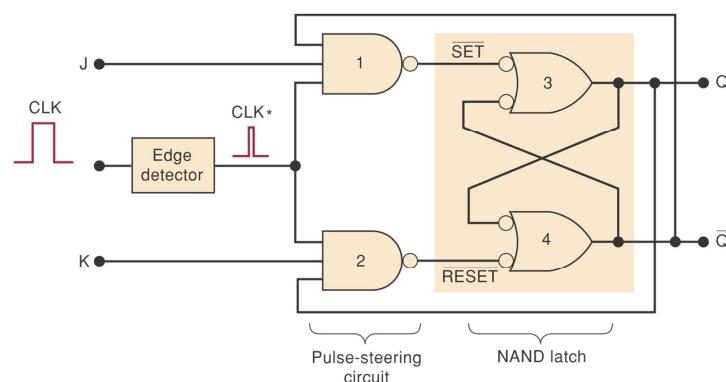
(a)



(b)

30

## Edge-triggered J-K flip-flop



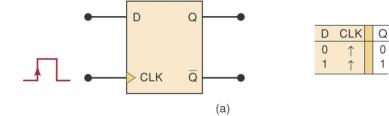
CLK\* must be high for FF to change states. This condition only occurs at the edge of a CLK transition.

31

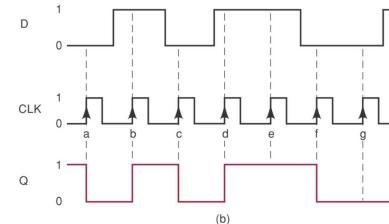


## Clocked D Flip-Flop

- One data input.
- The output changes to the value of the input at either the positive going or negative going clock trigger.



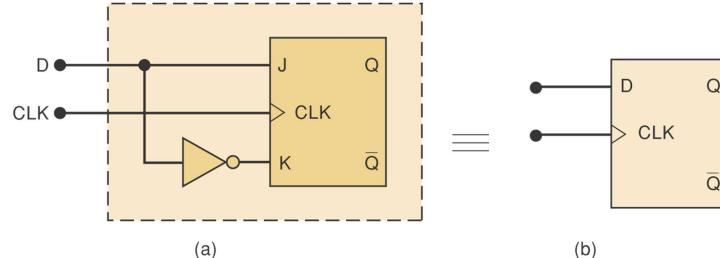
(a)



(b)

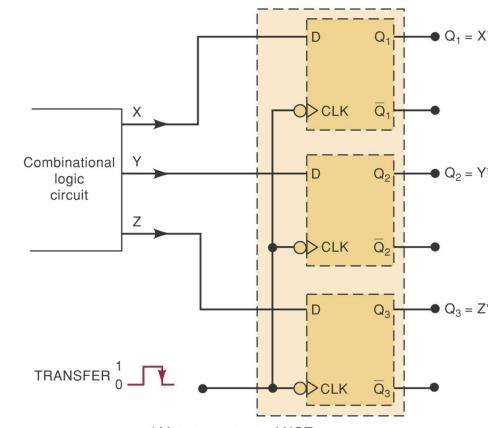
32

## Edge-triggered D flip-flop implementation from a J-K flip-flop



33

## Parallel transfer of binary data using D flip-flops



34

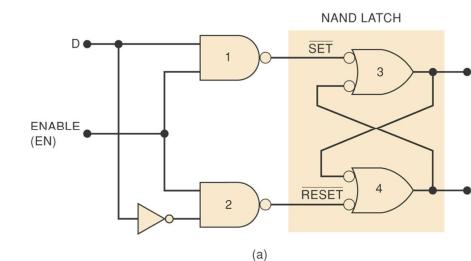
## D Latch (Transparent Latch)

- One data input.
- The **clock has been replaced by an enable line**.
- The device is **NOT** edge triggered.
- The output follows the input only when EN is high.

35

## D Latch

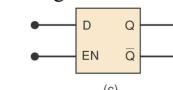
- D latch: (a) structure; (b) function table; (c) logic symbol.



Inputs		Output
EN	D	Q
0	X	$Q_0$ (no change)
1	0	0
1	1	1

"X" indicates "don't care."  
 $Q_0$  is state Q just prior to EN going LOW.

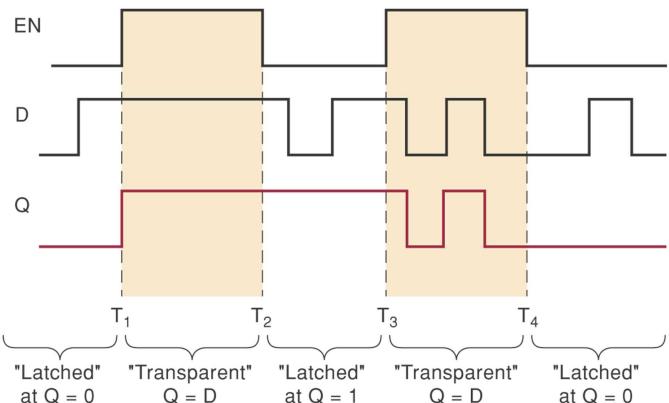
EN must be high for D-Latch to change states.



36

## D Latch

- Waveforms showing the two modes of operation of the transparent D latch.



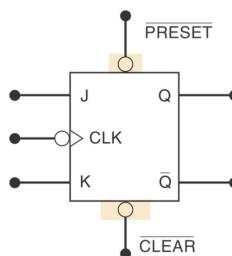
37

## Asynchronous Inputs

- Inputs that depend on the clock are synchronous.
- Most clocked FFs have **asynchronous inputs** that **do not depend on the clock**.
- The labels **PRE** and **CLR** are used for asynchronous inputs.
- Active low asynchronous inputs will have a bar over the labels and inversion bubbles.
- If the asynchronous inputs are not used they will be tied to their inactive state.

38

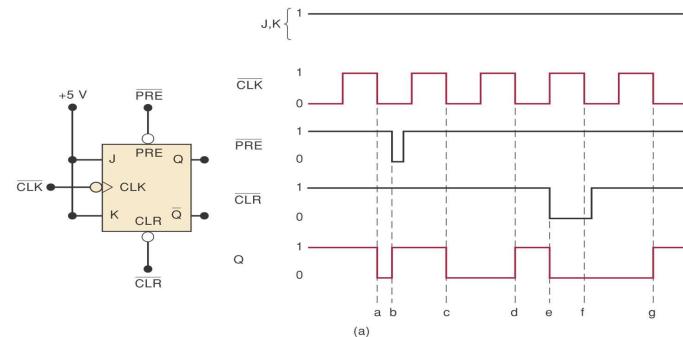
## Clocked J-K flip-flop with asynchronous inputs



J	K	Clk	PRE	CLR	Q
0	0	↓	1	1	Q (no change)
0	1	↓	1	1	0 (Synch reset)
1	0	↓	1	1	1 (Synch set)
1	1	↓	1	1	$\bar{Q}$ (Synch toggle)
x	x	x	1	1	Q (no change)
x	x	x	1	0	0 (asynch clear)
x	x	x	0	1	1 (asynch preset)
x	x	x	0	0	(Invalid)

39

## Clocked J-K flip-flop with asynchronous inputs



Point	Operation
a	Synchronous toggle on NGT of CLK
b	Asynchronous set on PRE = 0
c	Synchronous toggle
d	Synchronous toggle
e	Asynchronous clear on CLR = 0
f	CLR overrides the NGT of CLK
g	Synchronous toggle

(a)

40

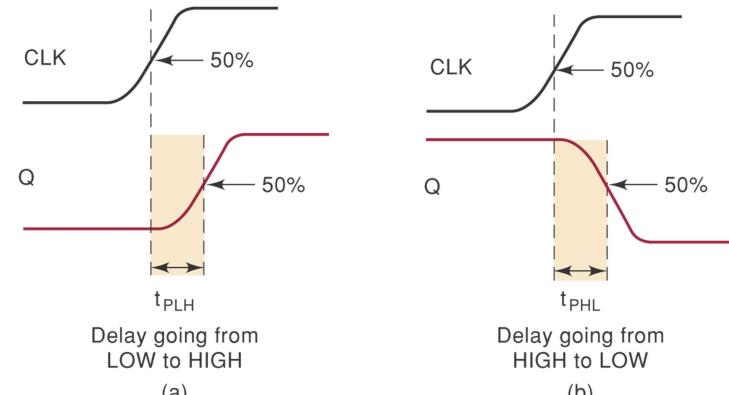
## Flip-Flop Timing Considerations

- Important timing parameters:
  - Setup and hold times
  - Propagation delay: the time for a signal at the input to be shown at the output.
  - Maximum clocking frequency: highest clock frequency that will give a reliable output.
  - Clock pulse high and low times: minimum time that the clock must be high before going low, and low before going high.
  - Asynchronous active pulse width: the minimum time PRESET or CLEAR must be held for the FF to set or clear reliably.
  - Clock transition times: maximum time for the clock transitions, generally less than 50 ns for TTL, or 200 ns for CMOS devices.

41



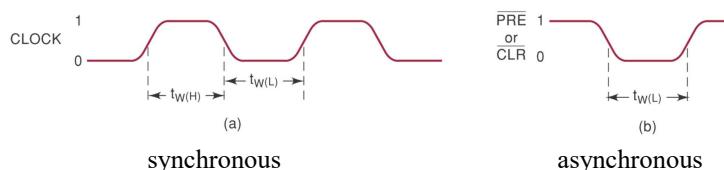
## Flip-Flop Propagation Delays



42



## Clock LOW and HIGH time



$t_{w(L)}$  is the minimum time that the CLK must remain low before it goes high.

$t_{w(H)}$  is the minimum time that the CLK must remain high before it goes low.

Similarly for asynchronous signals - but may have a different value than the CLK signal.

43

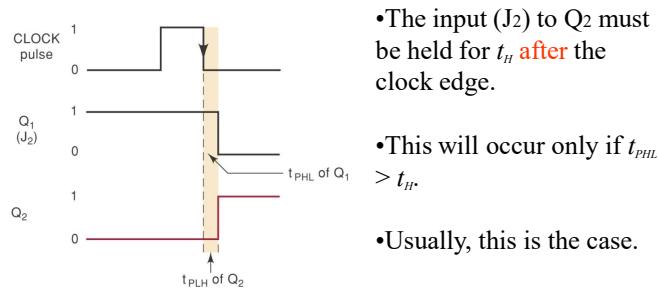
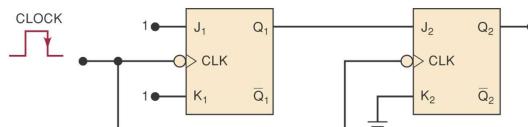


## Potential Timing Problems in FF Circuits

- When the output of one FF is connected to the input of another FF and both devices are triggered by the same clock, there is a potential timing problem.
- Propagation delay may cause unpredictable outputs.
- The low hold time parameter of most FFs mean this won't normally be a problem.

44

## Propagation Delay in Synchronous Circuits



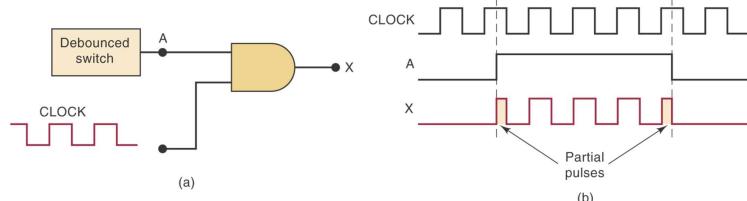
45

## Flip-Flop Synchronization

- Most systems are primarily synchronous in operation, in that changes depend on the clock.
- Asynchronous and synchronous operations are often combined.
- The random nature of asynchronous inputs can result in unpredictable results.

46

## Asynchronous Signals may have Undesirable Side Effects

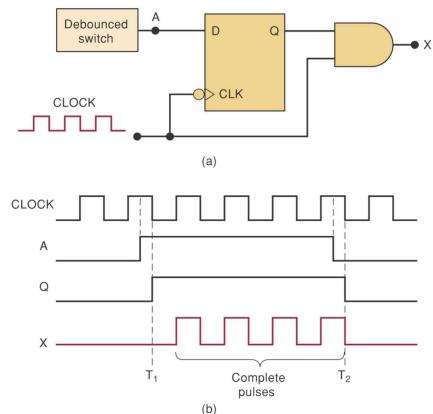


- Asynchronous signal A can produce partial pulses at X

47

## Edge-triggered flip-flop can Synchronize Circuit

- The signal A has no effect until negative edge of clock.

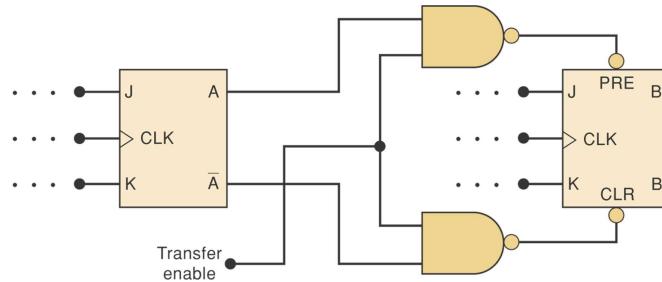


48

## Data Storage and Transfer

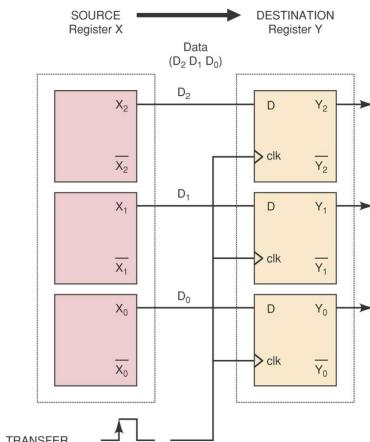
- Asynchronous transfers are controlled by **PRE** and **CLR** inputs.
- Transferring the bits of a register simultaneously is a parallel transfer.
- Transferring the bits of a register a bit at a time is a serial transfer.

## Asynchronous Data Transfer Operation



- Uses PRE and CLR inputs to load data into FF
  - PRE and CLR won't be both low at the same time
- $A = 1, EN = 1, PRE = 0$ , sets  $B = 1$
- $A = 0, EN = 1, CLR = 0$ , sets  $B = 0$

## Synchronous transfer of contents of register X into register Y



## Serial Data Transfer: Shift Registers

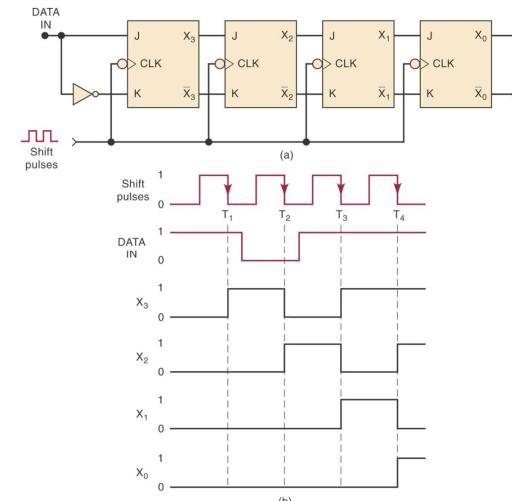
- When FFs are arranged as a shift register, **bits will shift with each clock pulse**.
- FFs used as shift registers must have very low hold time parameters to perform predictably. Modern FFs have  $t_H$  values well within what is required.
- The direction of data shifts will depend on the circuit requirements and the design.

## Serial Data Transfer: Shift Registers

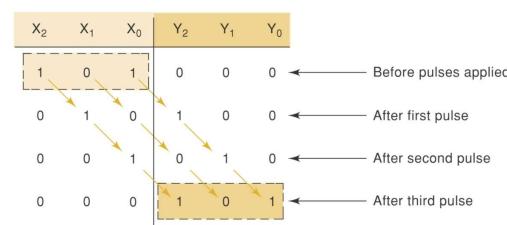
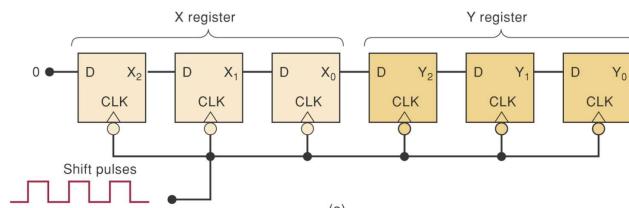
- **Parallel transfers** – register contents are transferred simultaneously with a single clock cycle.
- **Serial transfers** – register contents are transferred one bit at a time, with a clock pulse for each bit.
- Serial transfers are slower, but the circuitry is simpler. Parallel transfers are faster, but circuitry is more complex.
- Serial and parallel are often combined to exploit the benefits of each.



## Four-bit Shift Register



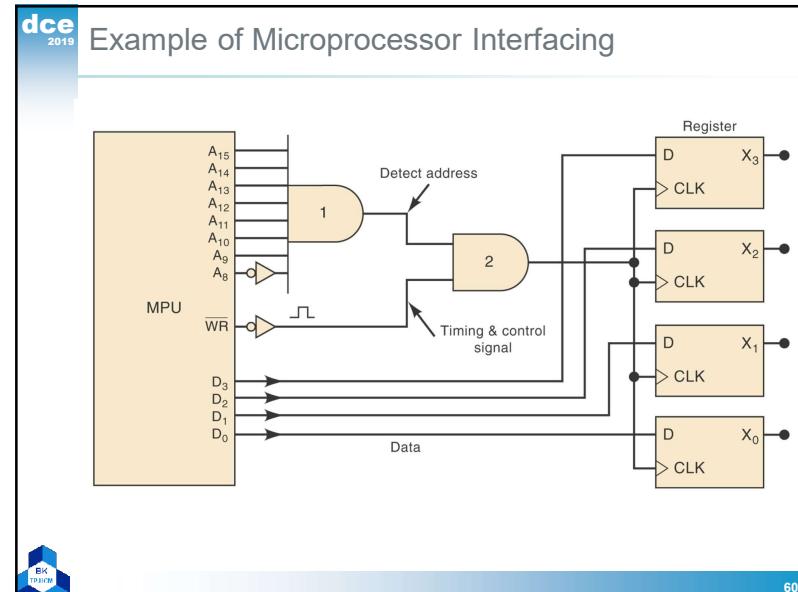
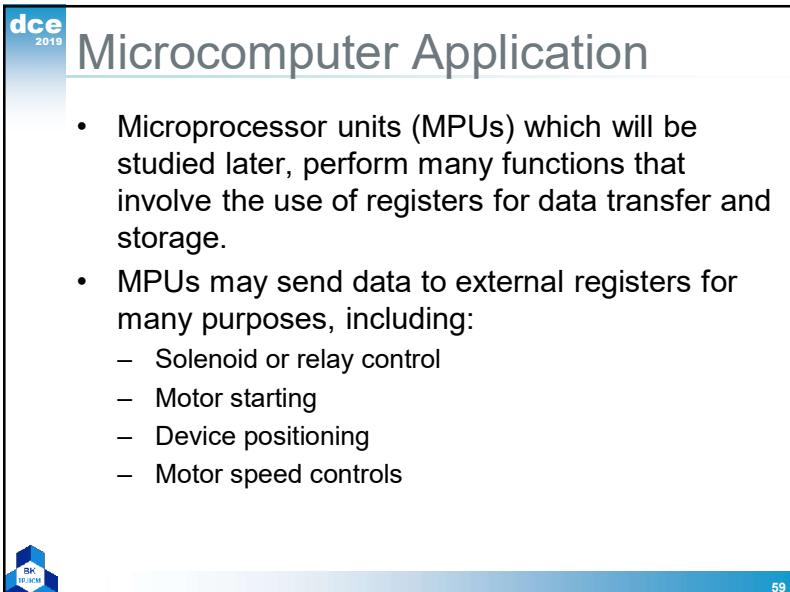
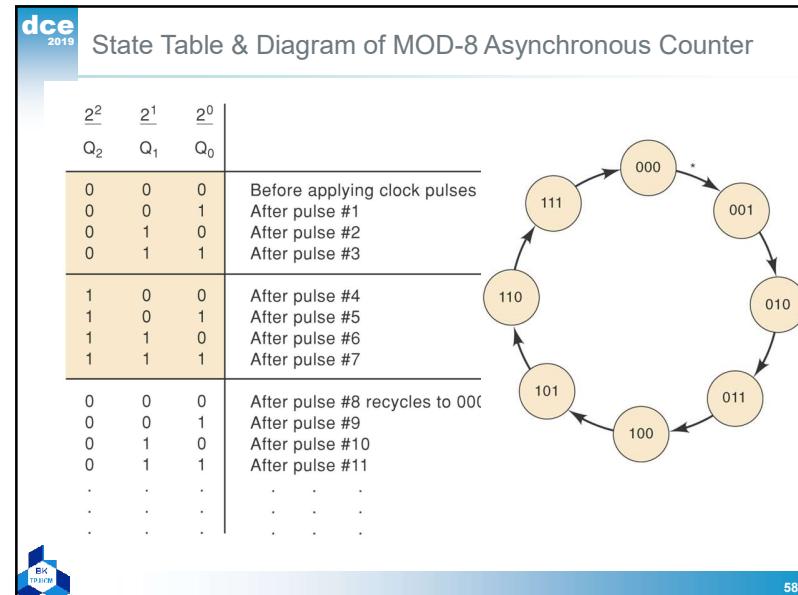
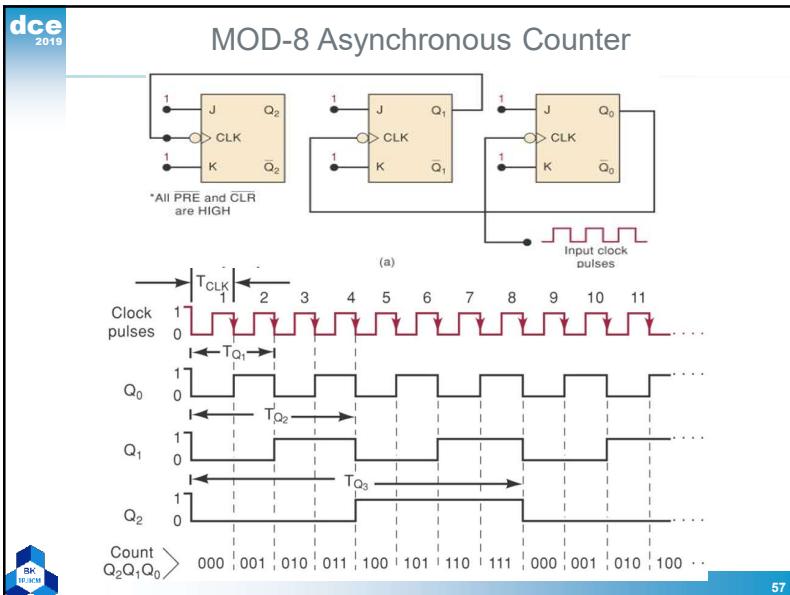
## Serial transfer from X register into Y register



## Frequency Division and Counting

- FFs are often used to divide a frequency as illustrated in next slide. Here the output frequency is 1/8<sup>th</sup> the input (clock) frequency.
- The same circuit is also acting as a binary counter. The outputs will count from 000<sub>2</sub> to 111<sub>2</sub>.
- The number of states possible in a counter is the modulus or MOD number. Next slide is a MOD-8 (2<sup>3</sup>) counter. If another FF is added it would become a MOD-16 (2<sup>4</sup>) counter.





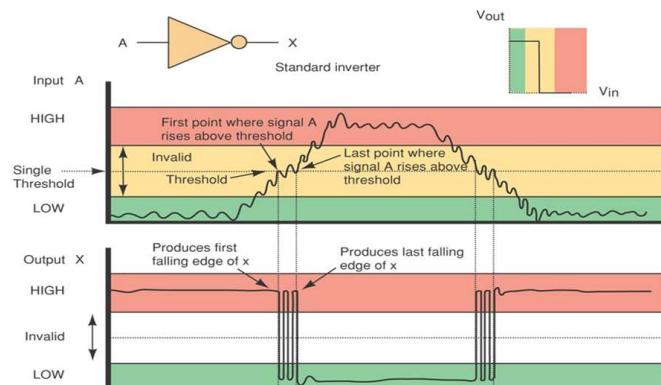
## Schmitt-Trigger Devices

- Not a FF but shows a memory characteristic
- Accepts slow changing signals and produces a signal that transitions quickly.
- A Schmitt trigger device will not respond to an input until it exceeds the positive or negative going threshold.
- There is a separation between the two threshold levels. This means that the device will “remember” the last threshold exceeded until the input goes to the opposite threshold.



61

## Schmitt-Trigger Response (two thresholds)

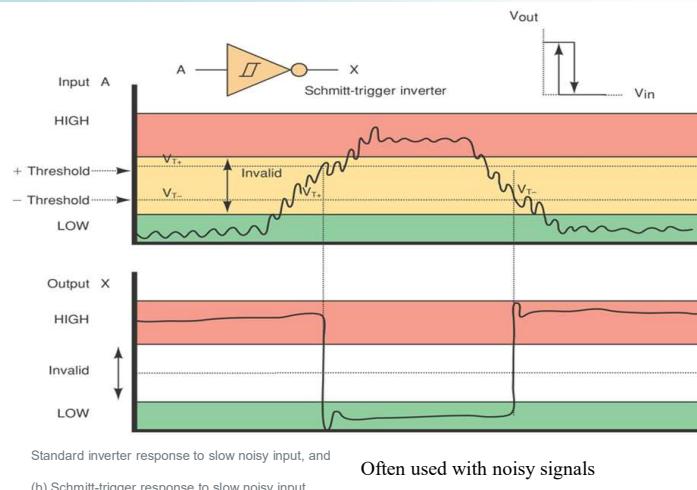


Standard inverter response to slow noisy input, and  
(b) Schmitt-trigger response to slow noisy input.

Often used with noisy signals

62

## Schmitt-Trigger Response (two thresholds)



Standard inverter response to slow noisy input, and  
(b) Schmitt-trigger response to slow noisy input.

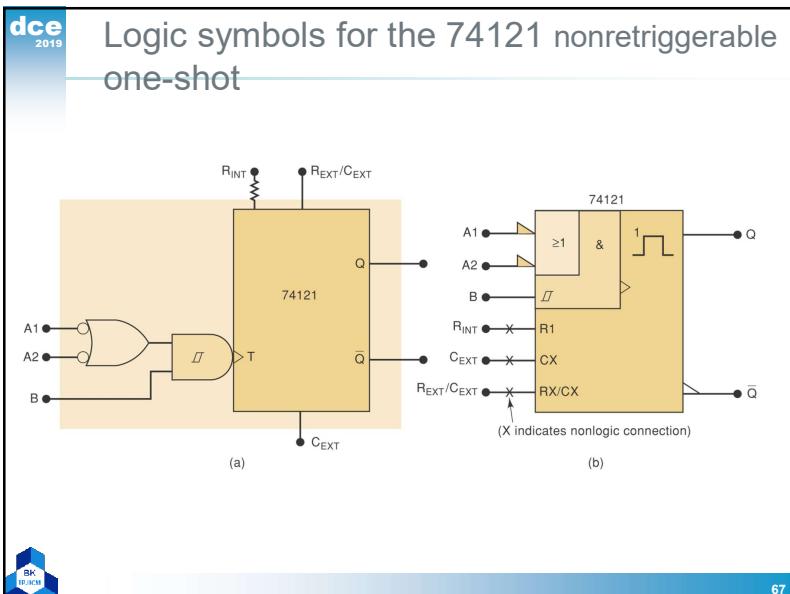
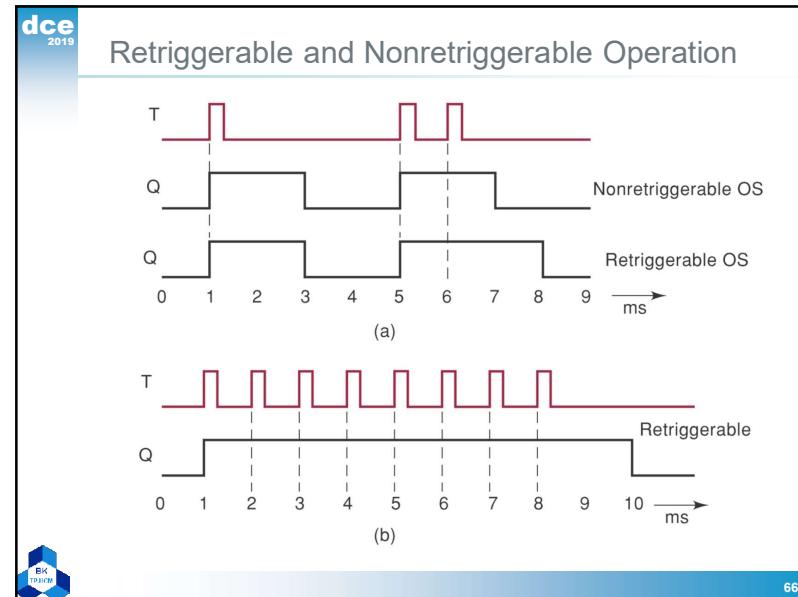
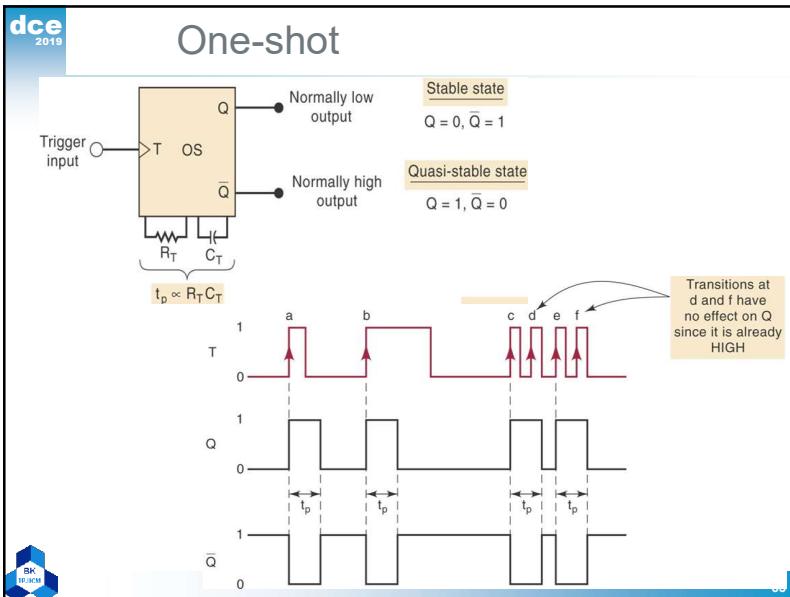
63

## One-shot (Monostable Multivibrator)

- Changes from stable state to quasi-stable state for a period of time determined by external components (usually resistors and capacitors).
- Nonretriggerable devices will trigger and return to stable state.
- Retriggerable devices can be triggered while in the quasi-stable state to begin another pulse.
- One shots are called monostable multivibrators because they have only one stable state.
- They are prone to triggering by noise so, tend to be used in simple timing applications.



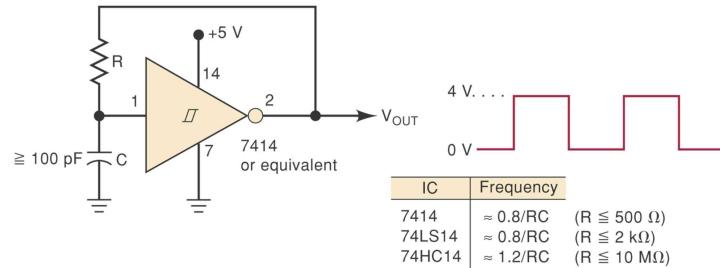
64



- dce 2019**
- ## Clock Generator Circuits
- FFs have two stable states, so are considered bistable multivibrators.
  - One shots have one stable state and are considered monostable multivibrators.
  - Astable or free-running multivibrators switch back and forth between two unstable states. This makes it useful for generating clock signals for synchronous circuits.
  - Crystal control may be used if a very stable clock is needed. Crystal control is used in microprocessor based systems and microcomputers where accurate timing intervals are essential.

## Clock Generator Circuit: Schmitt-trigger Oscillator

Schmitt-trigger oscillator using a 7414 INVERTER. A 7413 Schmitt-trigger NAND may also be used.

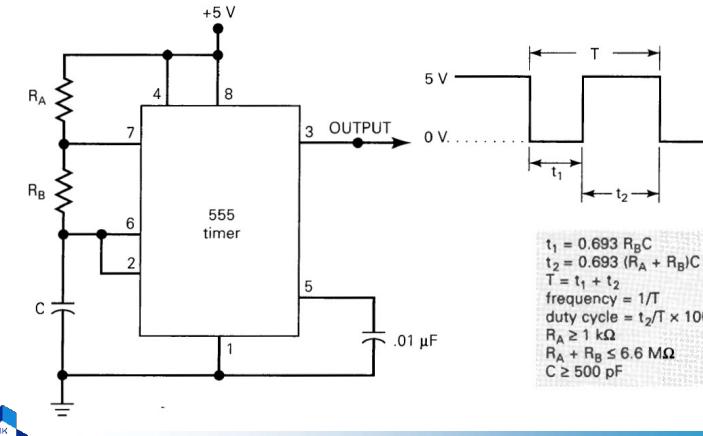


Circuit will not oscillate if  $R$  is not kept within these limits.

69

## Clock Generator Circuit: 555 Timer

555 timer IC used astable multivibrator.



70