12-Latches and Flip Flops

Text: Unit 11

ECEGR/ISSC 201
Digital Operations and Computations
Winter 2011

Overview

- Sequential Circuits
- Set/Reset Latch
- Gated D Latch
- Flip-Flops

Sequential Circuits

- Sequential circuits:
  - Output depends upon current and past inputs
- Memory is required
- Two basic memory devices:
  - Latches
  - Flip-flops
- Feedback is needed to store output state

Sequential Circuits

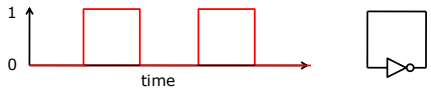
- Flip-Flop
  - Output changes based upon an input clock signal
  - Used in signal generation
- Latches
  - Output changes based upon data input

Feedback

- An output is fed to an "earlier" input
- Example of an inverter circuit

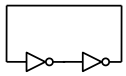
Feedback

- Assume that there is a delay before the inversion is complete
- If the signal starts as 0, then:



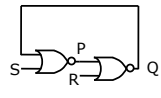
Feedback

- A stable state (persistent 1 or 0) is not reached
- If another inverter is added, then the output is stable
- Similar to debounced switch



Set-Reset Latch

- Consider two NOR gates
- In this configuration, they form a Set-Reset Latch, for reasons shown hereafter

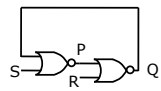


Set-Reset Latch

- *Present state*: the state of Q at the time the signals are applied (or changed), Q
- *Next state*: the state of Q after the latch has reacted to the input, Q^+

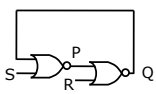
Set-Reset Latch

- If Q = 0, find Q^+ if S=R=0



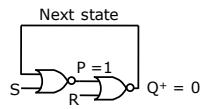
Set-Reset Latch

- If Q = 0, find Q^+ if S=R=0
  - P = (S + Q)′ = (0 + 0)′= 1
  - Q^+ = (R + P)′ = (0 + 1)′ = 0
- Is this stable?



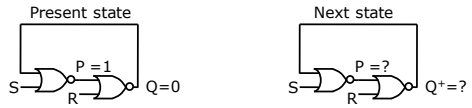
Set-Reset Latch

- If Q = 0, find Q^+ if S=R=0
  - P = (S + Q)′ = (0 + 0)′= 1
  - Q^+ = (R + P)′ = (0 + 1)′ = 0
- Is this stable?
  - Yes!
  - Q^+ = Q



Set-Reset Latch

- Now assume that S changes to 1
  - By our notation:
    - Next state on previous slide becomes the present state
    - Q^+ on previous slide becomes Q
- What does the output become?

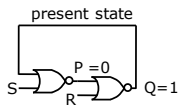


Set-Reset Latch

- Now assume that S changes to 1
  - Q = 0
  - => P = (S + 0)' = 0
  - => Q^+ = (R + 0)'= 1

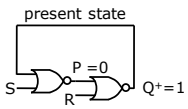
Set-Reset Latch

- Changing S from 0 to 1 changed the output from 0 to 1
- Is this state stable?



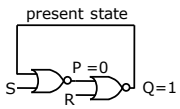
Set-Reset Latch

- Changing S from 0 to 1 changed the output from 0 to 1
- Is this state stable?
  - S=1, R = 0
  - P = (S + 1)' = 0
  - Q^+ = (R + 0)' = 1, Yes it is stable



Set-Reset Latch

- What if the S is returned to 0?



Set-Reset Latch

- What if the S is returned to 0?
  - S=0, R =0
  - P = (S +1)' = 0
  - Q^+ = (R +0)' =1, stable, no change in output

Set-Reset Latch

- Wait, I thought that when S = R =0, the output Q was 0. What happened?
- For S/R Latches, there are two possible stable outputs for each input, depending on the previous state of the latch

Set-Reset Latch

- Recap:
  - Started with S=0, R=0; output was 0 (stable)
  - Changed S to 1; output became 1 (stable)
  - Changed S back to 0, output stayed at 1 (stable)
- Changing S to 1 is known as setting the latch
- What happens now when R = 1?

Set-Reset Latch

- What happens now when R = 1 (S = 0)?
  - S=0, R =1
  - $Q^+ = (R + 0)' = 0$, output returns to zero
  - $P = (S + 0)' = 1$
- Is it stable?

Set-Reset Latch

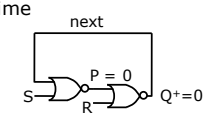
- What happens now when R = 1 (S = 0)?
  - S=0, R =1
  - $Q^+ = (R + 0)' = 0$, output returns to zero
  - $P = (S + 0)' = 1$
- Is it stable?
  - $P = (S + 0)' = 1$
  - $Q = (R + 1)' = 0$, yes!

Set-Reset Latch

- Recap:
  - Started with S=0, R=0; output was 0 (stable)
  - Changed S to 1; output became 1 (stable)
  - Changed S back to 0, output stayed at 1 (stable)
  - Changed R to 1, output reset to 0 (stable)
- Changing R to 1 is known as resetting the latch
- What happens when S = R = 1 (both setting and resetting the latch)?

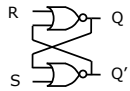
Set-Reset Latch

- What happens when S = R = 1 (both setting and resetting the latch)?
  - $P = (1 + Q)' = 0$ (regardless of Q)
  - $Q^+ = (1 + P)' = 0$ (regardless of R)
- Oscillations are possible if S and R change back to 0 (depending propagation delays)
- Generally, we do not allow S and R to be equal to 1 at the same time

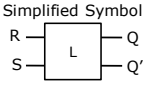


Set-Reset Latch

- With the limitations on the input states, stable states are always complements
  - P = Q'

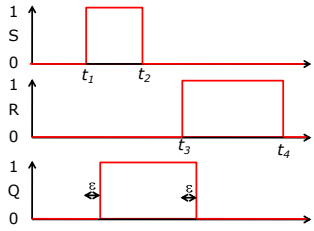


Simplified Symbol



Set-Reset Latch

- Timing Diagram



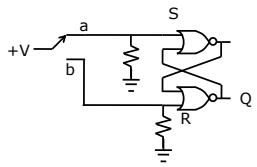
Set-Reset Latch

- $Q^+ = S + R'Q$ (leave off the time index)
  - Given that SR = 0
  - Referred to as the characteristic or next-state equation

S(t)	R(t)	Q(t)	Q(t +e)
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	X
1	1	1	X

Switch Debouncing

- S/R can be used to make a debounced switch
- Pull down resistors are used for 0 input
- Bouncing to the input to S does not propagate through to Q, similar result for R bounces



Gated D Latch

- Data input (D) and Gate input (G)
- When G = 1, the input to D is passed through to Q
- When G = 0, Q does not change
- Known as a transparent latch

Gated D Latch

- Based on the description, let's populate the truth table
- How many columns to our truth table?

Gated D Latch

- Based on the description, let's populate the truth table
- How many columns to our truth table?
  - G
  - D
  - Q
  - Q^+

Gated D Latch

- When G = 1, the input to D is passed through to Q
- When G = 0, Q does not change
- Complete the truth table

G	D	Q	Q^+
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

Gated D Latch

- When G = 1, the input to D is passed through to Q
- When G = 0, Q does not change
- Complete the truth table

G	D	Q	Q^+
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

Gated D Latch

- The resulting K-map is shown
- What is the characteristic equation?

G	D	Q	Q^+
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

Q \ GD	00	01	11	10
0	0	0	1	0
1	1	1	1	0

Gated D Latch

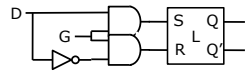
- $Q^+ = G'Q + GD$

G	D	Q	Q^+
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

Q \ GD	00	01	11	10
0	0	0	1	0
1	1	1	1	0

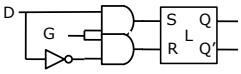
Gated D Latch

- $Q^+ = G'Q + GD$
- Realization with S/R Latch
  - $Q^+ = S + R'Q$
  - $S = GD$
  - $R = GD'$
- $Q^+ = GD + (GD')'Q = GD + (G' + D)Q = GD + G'Q + DQ$
  $= G'Q + GD$ (consensus theorem)

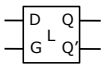


Gated D Latch

- Realization with S/R Latch
- $Q^+ = G'Q + GD$

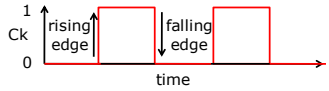


- Symbol



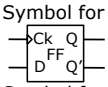
Edge Triggered D Flip-Flop

- Flip-flops change their output based upon a clock signal, Clock or Clk or Ck
- Data input D
- Output Q, Q'
- Rising Edge or Falling Edge are possible
- The edge (rising or falling) that triggers the operation is known as the *active edge*

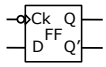


Edge Triggered D Flip-Flop

- Symbol for Rising Edge



- Symbol for Falling Edge



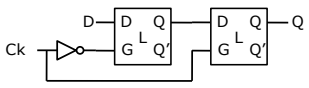
Edge Triggered D Flip-Flop

- Truth table for edge-triggered D Flip-Flop
- The output Q^+ is equal to the input just before the active edge
- $Q^+ = D$

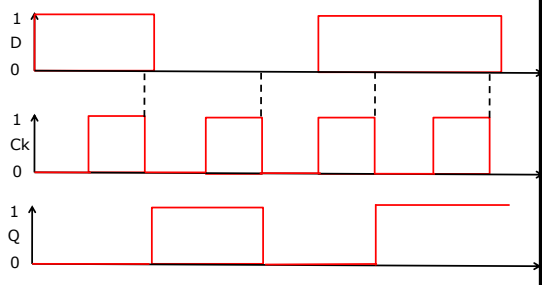
D	Q	Q^+
0	0	0
0	1	0
1	0	1
1	1	1

Edge Triggered D Flip-Flop

- Realization of Rising Edge Triggered D Flip-Flop



Falling Edge D Flip-Flop

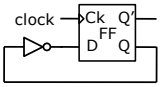


Rising Edge D Flip-Flop

- Operation is time sensitive
- D must be held at its value for a slight period before and after active edge
  - t_{su}: setup time (prior to active edge)
  - t_h: hold time (after active edge)
  - t_p: propagation delay (time from after an active edge to change in Q)

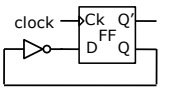
Rising Edge D Flip-Flop

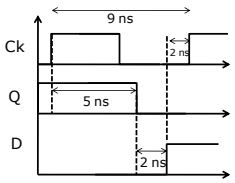
- Example
- Inverter characteristics
  - t_p: 2ns
- Flip-Flop characteristics
  - t_{su}: 3ns
  - t_h: 0 ns
  - t_p: 5ns
- Assume the clock period is 9ns



Rising Edge D Flip-Flop

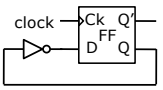
- Inverter characteristics
  - t_p: 2ns
- Flip-Flop characteristics
  - t_{su}: 3ns
  - t_h: 0 ns
  - t_p: 5ns
- Assume the clock period is 9ns

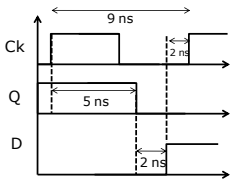




Rising Edge D Flip-Flop

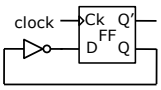
- Set-up time of 3ns is not satisfied
- Solution: increase clock period (decrease clock frequency)





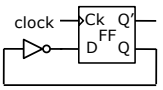
Rising Edge D Flip-Flop

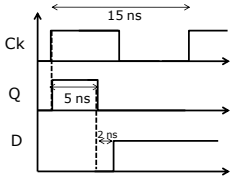
- Let the clock period be 15 ns
- Does this work?



Rising Edge D Flip-Flop

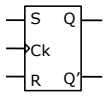
- Let the clock period be 15 ns
- Does this work?
  - Yes!
  - Set-up time of 3ns is satisfied
- Minimum clock period is 10 ns





S-R Flip-Flop

- Similar in concept to SR Latch
  - S=1 sets Q to 1
  - R = 1 resets Q to 0
- Q can only change after clock active edge

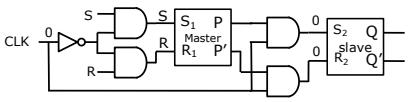


S-R Flip-Flop

- Truth table and characteristic equation is the same as an S-R latch, but Q^+ is interpreted differently
- Q^+ is the value of Q after the next active edge

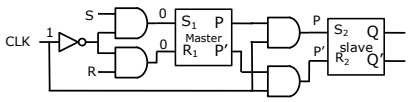
S-R Flip-Flop

- S-R Flip-Flop construction (master-slave)
- When CLK = 0
  - S/R set the master latch
  - Slave latch holds the previous value of Q



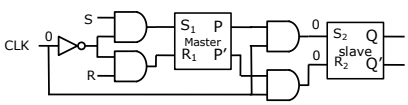
S-R Flip-Flop

- When CLK changes from 0 to 1
  - Output of the master latch is transferred to slave latch
- While CLK = 1, the output does not change



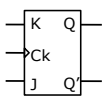
S-R Flip-Flop

- When CLK changes from 1 to 0
  - Q value is latched in the slave
  - New inputs to the master can be processed
- See page 304 for a timing diagram
  - Note how this is different from an edge-triggered flip-flop



J/K Flip Flop

- Similar to S-R Flip-Flop
  - J => S
  - K => R
- J and K may both be 1
  - Q changes from 0 to 1 or 1 to 0 on next active edge

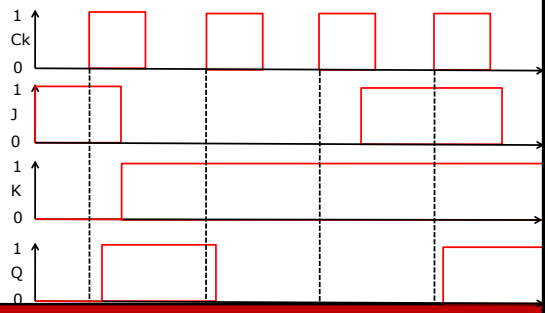


J/K Flip Flop

- $Q^+ = JQ' + K'Q$

J	K	Q	Q^+
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

J/K Flip Flop (rising edge)



T (toggle) Flip-Flop

- Used often in counters
- T input
- Clock input
- T =1, Q changes state after next active edge
- T =0, no state change occurs

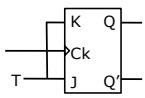
T Flip-Flop

- $Q^+ = T'Q + TQ' = T \oplus Q$

T	Q	Q^+
0	0	0
0	1	1
1	0	1
1	1	0

T Flip-Flop

- Implementation
  - Connect the J and K inputs of a J/K flip flop together and connect that to T input



Flip-Flops with Additional Inputs

- See text Section 11.8 for more information