

ECE212/212s Lecture Notes

Lecture: Chapter 6

Dr. Clay Gloster, Jr.

Latches and flip flops

Latch - outputs change soon after inputs change

Flip flops - outputs change only with respect to the clock

Classes of Latches

- Unclocked (only data inputs)
- Level sensitive (enable input)

Classes of Flip Flops

- Positive-edge triggered
(Rising-edge triggered)
- Negative-edge triggered
- Master slave

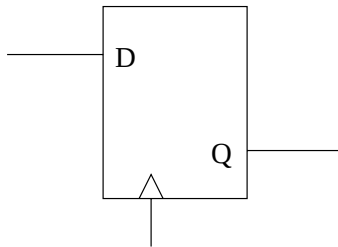


Figure 1: Positive Edge Triggered FF

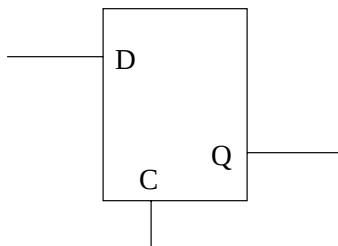


Figure 2: Level Sensitive Latch

A Simple R-S Latch

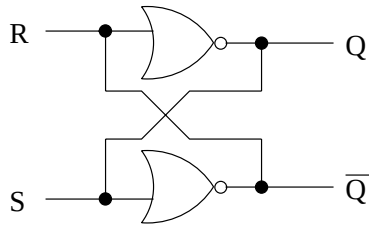
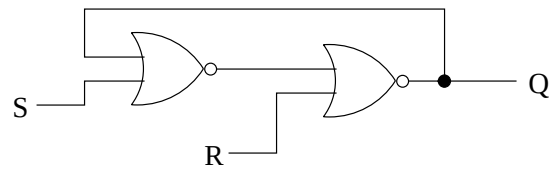


Figure 3: Cross-Coupled Nor (RS Latches)

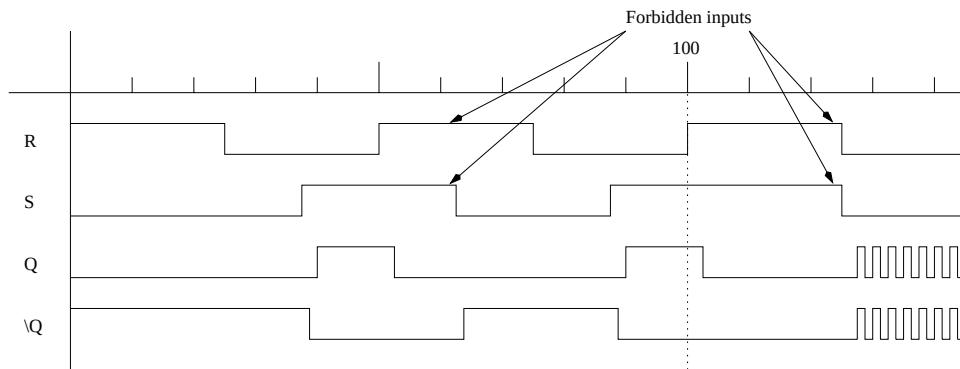
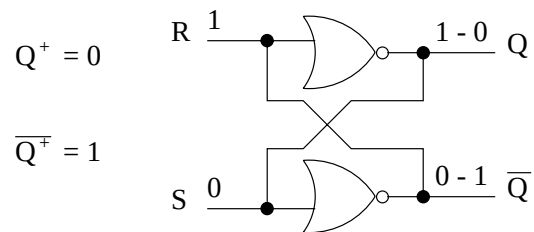
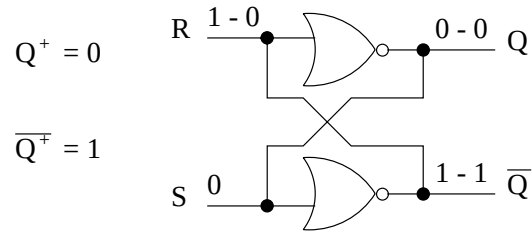


Figure 4: Cross-Coupled NOR timing diagram

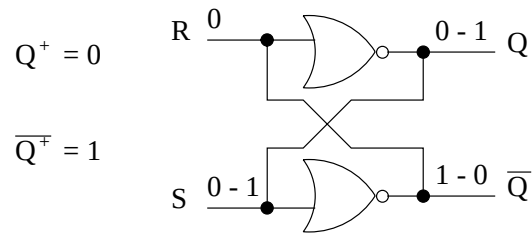
1) $Q = 1$, $\overline{Q} = 0$, $S = 0$, $R = 1$ Reset $R = 1$, $Q = 0$



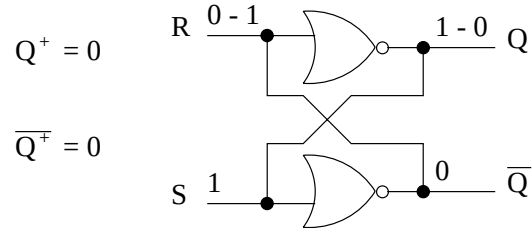
2) $Q = 0, \quad \overline{Q} = 1, \quad S = 0, \quad R = 0$ Hold $R, S = 0, \quad Q^+ = Q$



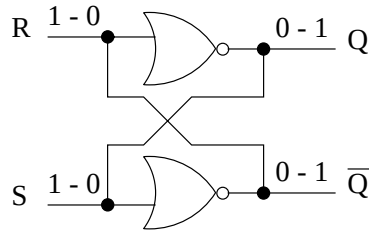
3) $Q = 0, \quad \overline{Q} = 1, \quad S = 1, \quad R = 0$ Set



4) $Q = 1, \quad \overline{Q} = 0, \quad S = 1, \quad R = 1$ Forbidden $Q = \overline{Q}$ Contradicts definitions



5) $Q = 0, \quad \overline{Q} = 0, \quad S = 0, \quad R = 0$



When R, S go to 0, $Q^+ = \overline{Q}^+ = 1$, then
 $Q^+ = \overline{Q}^+ = 0$ then
 $Q^+ = \overline{Q}^+ = 0$, etc.

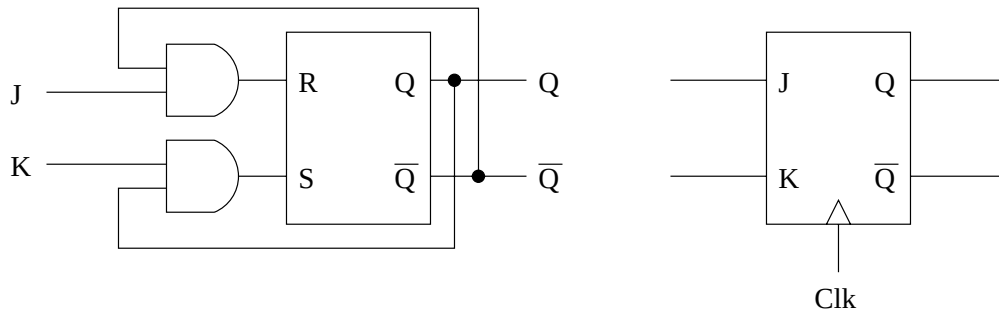
Truth table of cross-coupled NOR gates

S	R	$Q(+)$	$Q(t+1)$
0	0	$Q(+)$	$Q(t+1)$
1	0	$Q(+)$	1
0	1	$Q(+)$	0
1	1	$Q(+)$	Not allowed!

$$Q(t+1) = S(t) + \overline{R}(t)Q(t)$$

$$Q^+ = S + \overline{R}Q$$

J-K FF

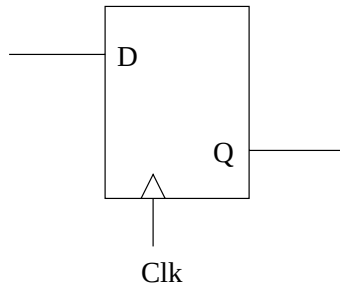


$J(t)$	$K(t)$	$Q(t)$	$Q(t+1)$
0	0	$Q(t)$	$Q(t)$
1	0	$Q(t)$	0
1	0	$Q(t)$	1
1	1	$Q(t)$	$\overline{Q(t)}$

$$J(t) \Rightarrow \text{Set} \quad K(t) \Rightarrow \text{Like Reset}$$

$$Q^+ = Q\overline{K} + \overline{Q}J$$

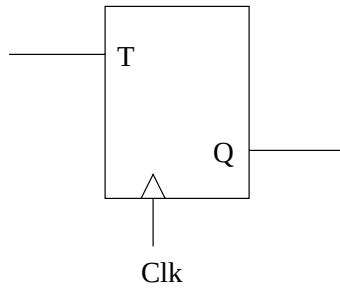
D FF



D(t)	Q(t)	Q(t+1)
0	Q(t)	0
1	Q(t)	1

$$Q^+ = D$$

T FF



T(t)	Q(t)	Q(t+1)
0	Q(t)	Q(t)
1	Q(t)	$\overline{Q(t)}$

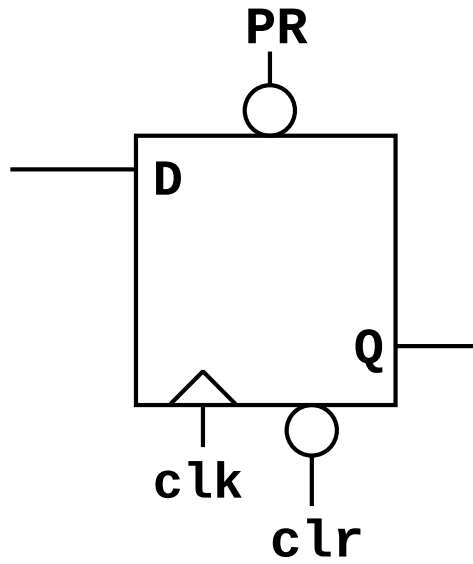
$$Q^+ = T \overline{Q} + (\overline{T}) Q$$

Excitation tables for R-S, J-K, T, and D flip flops

Q	Q^+	R	S	J	K	T	D
0	0	X	0	0	X	0	0
0	1	0	1	1	X	1	1
1	0	1	0	X	1	1	0
1	1	0	X	X	0	0	1

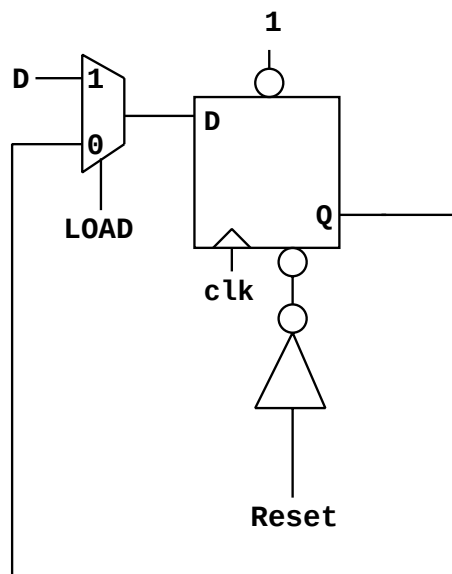
Memorize!

A D Flip Flop



Initially at X. How do we initialize?

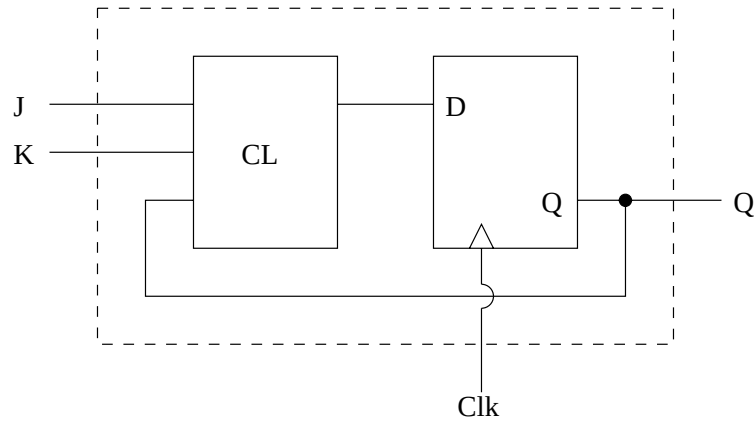
- Positive edge triggered (CLK)
- Preset (set) active low
- CLR (Reset) active low



Exercise: Assume that our library only has D flip-flops! How would we simulate our project design which has JK flip-flops?

Problem Restated: Show how to implement a JK flip-flop starting with a D flip flop.

Solution:

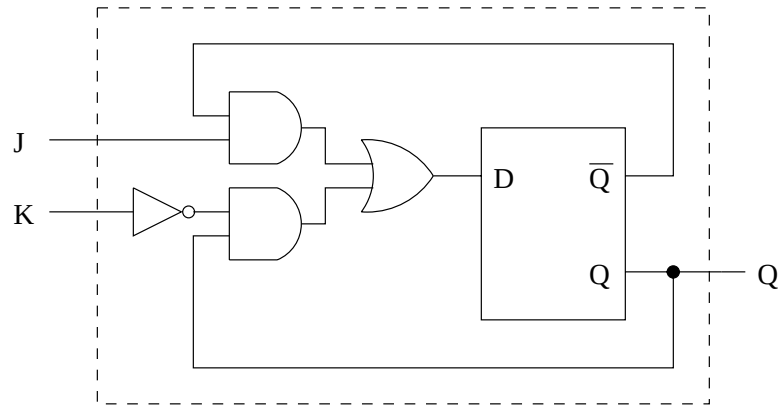


(set)

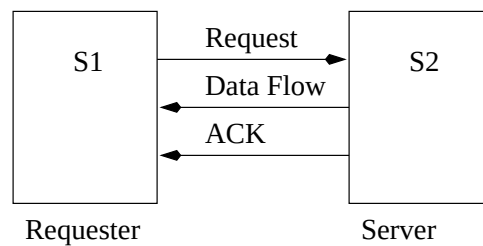
J	K	Q(t)	Q(t+1)	D
0	0	0	0	0
0	0	1	1	1
0	1	0	0	0
0	1	1	0	0
1	0	0	1	1
1	0	1	1	1
1	1	0	1	1
1	1	1	0	0

JK Q	00	01	11	10
			1	1
	1			1

$$D = J\bar{Q} + \bar{K}Q$$



Asynchronous Communication



S1 Signals Request

S2 Sends Data

S2 Sends ACKNOWLEDGE (Data is ready)

Sometimes used in synchronous systems. If it is known that data will be valid in k cycles ACK is not needed.

Request line is sometimes called the enable.

Sequential Circuit Design Example: A particular sequential circuit has one input and one output. The output is equal to one when the past three previous inputs were 1,0,1. Design the sequential circuit using J-K Flip-Flops.

Finite State Machine Design Steps (Page 388)

1. Understand the Problem (try some input sequences)
2. Draw State Diagram.
3. Perform State Minimization (Formal method later)
4. Perform State Assignment
5. Draw State Transition Table (Symbolic)
6. Draw State Transition Table (Encoded)
7. Choose FF type (J-K's tend to reduce gate count , D's are easier)
8. Use excitation tables to fill in K-Maps for FF inputs
9. Perform logic simplification
10. Implement the FSM

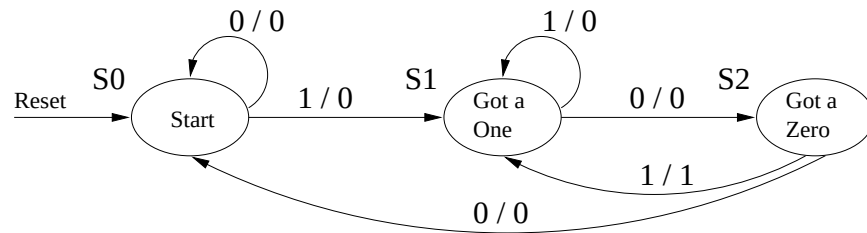


Figure 5: State Diagram

State Transition Table

PS	NS	
	A = 0	A = 1
S_0	$S_0,0$	$S_1,0$
S_1	$S_2,0$	$S_1,0$
S_2	$S_0,0$	$S_1,1$