Week 12 + 13

CS2100 - L21a - Sequential Logic (Building Blocks)

21a.1 - Memory elements

21a.2 - Latches

- S-R latch
- D latch

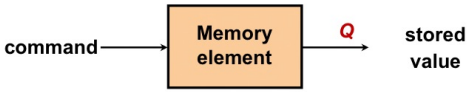
21a.3 - Flip-flops

- S-R flip-flop
- D flip-flop
- J-K flip-flop
- T flip-flop

21a.1 – Memory elements

Memory Elements: Overview

- A device which can remember value indefinitely, or change value on command from its inputs.



- **Characteristic table**:

Diff from truth table

Command (at time *t*)	*Q(t)* (now)	*Q(t+1)* (current)
Set	X	1
Reset	X	0
Memorise / No Change	0	0
	1	1

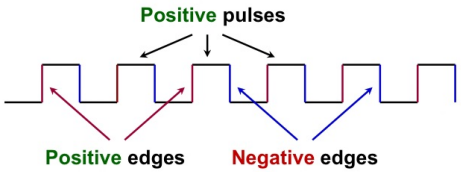
Q(t) or ***Q***: current state

Q(t+1) or ***Q*⁺**: next state

Memory Elements: Triggering/Activation

- Two types of triggering/activation

1. **Pulse-triggered**
   - **Latches**
   - ON = 1, OFF = 0



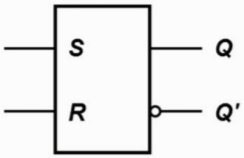
2. **Edge-triggered**
   - **Flip-flops**
   - Positive edge-triggered (ON = from 0 to 1; OFF = other time)
   - Negative edge-triggered (ON = from 1 to 0; OFF = other time)

21a.2 - Latches

S-R latch

S-R Latch: Overview

- **Two inputs**: [***S*** and ***R***]
- **Two complementary outputs**: [***Q*** and ***Q'***]
  - When ***Q* = HIGH** ➔ latch is in **SET** state
  - When ***Q* = LOW** ➔ latch is in **RESET** state



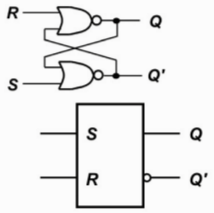
- **Two variants**:
  - **Active high input S-R Latch** (1 = active)
  - **Active low input S-R Latch** (0 = active)

Active High *S-R* Latch: Characteristic

Set Reset

- **Characteristic table**:

S	R	Q	Q'	Action
0	0	NC	NC	No change. Latch remained in present state.
1	0	1	0	Latch SET.
0	1	0	1	Latch RESET.
1	1	0	0	Invalid condition.



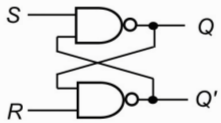
S	R	Q(t+1) or Q^+	
0	0	Q(t)	No change
0	1	0	Reset
1	0	1	Set
1	1	indeterminate	

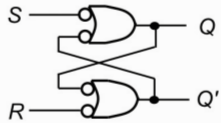
$Q(t+1) = ?\quad S + R' \cdot Q$

$S \cdot R = 0$

Active-Low *S-R* Latch

- **Active-low input** *S-R* latches are similarly constructed with cross-coupled NAND gates:



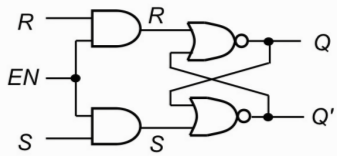


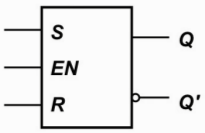
(Sometimes, the inputs are labelled as *S'* and *R'*.)

- Note the alternative drawing on the right, where NAND gate is drawn as Negative-OR gate
- In this case,
  - when ***R*=0** and ***S*=1**, the latch is **reset** (i.e. Q becomes 0)
  - when ***R*=1** and ***S*=0**, the latch is **set** (i.e. Q becomes 1)
  - when ***S*=*R*=1**, it is a **no-change** command.
  - when ***S*=*R*=0**, it is an **invalid** command.

Gated *S-R* Latch (with enable)

- *S-R* latch + *enable input* (*EN*) and 2 AND gates → a **gated *S-R* latch**



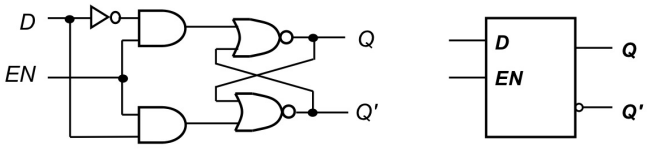


- Outputs change (if necessary) only when ***EN* is high**

D latch

Gated *D* Latch: Overview

- Make input *R* equal to *S'* → **gated *D* latch**



- *D* latch eliminates the **undesirable condition of invalid state in the *S-R* latch**

Gated *D* Latch: Characteristic

- When *EN* is high,
  - *D* = HIGH → latch is SET
  - *D* = LOW → latch is RESET
- Hence when EN is high, *Q* "follows" the *D* (data) input.
- **Characteristic table**:

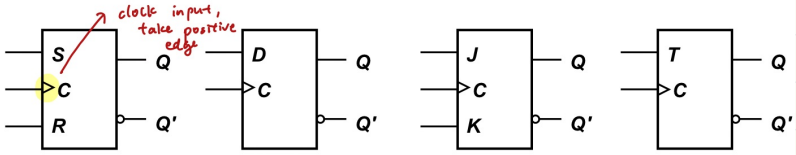
EN	*D*	*Q(t+1)*	
1	0	0	Reset
1	1	1	Set
0	X	*Q(t)*	No change

When *EN*=1, ***Q(t+1) = ?*** D

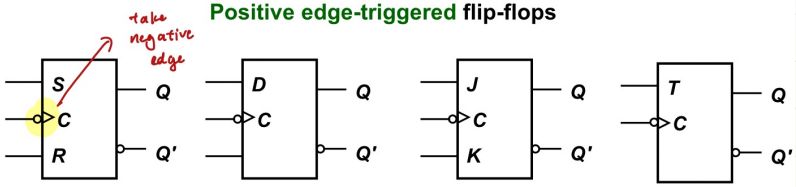
21a.3 - Flip-flops

Flip-Flops: Variants

- ***S-R* flip-flop**, ***D* flip-flop**, ***J-K* flip-flop** and ***T* flip-flop**
- Note the ">" symbol at the clock input.



Positive edge-triggered flip-flops

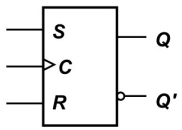


Negative edge-triggered flip-flops

S-R flip-flop

S-R Flip-Flop

- **On the triggering edge of the clock pulse**,
  - *R* = HIGH and *S* = LOW ➔ *Q* becomes LOW (RESET state)
  - *S* = HIGH and *R* = LOW ➔ *Q* becomes HIGH (SET state)
  - Both *R* and *S* are LOW ➔No change in output *Q*
  - Both *R* and *S* are HIGH ➔Invalid!
- **Characteristic table** of positive edge-triggered *S-R* flip-flop:



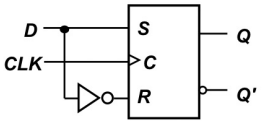
S	*R*	*CLK*	*Q(t+1)*	Comments
0	0	X	*Q(t)*	No change
0	1	↑	0	Reset
1	0	↑	1	Set
1	1	↑	?	Invalid

X = irrelevant ("don't care")
↑ = clock transition LOW to HIGH

D flip-flop

D Flip-Flop: Summary

- Single input *D* (data), on the triggering edge of the clock pulse:
  - *D* = HIGH ➔ *Q* becomes HIGH (SET state)
  - *D* = LOW ➔ *Q* becomes LOW (RESET state)
- Hence, *Q* "follows" *D* at the clock edge.
- Convert *S-R* flip-flop into a *D* flip-flop: **add an inverter**



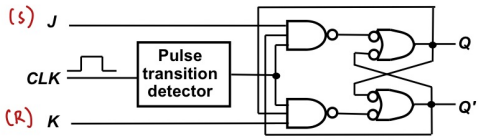
D	CLK	Q(t+1)	Comments
1	↑	1	Set
0	↑	0	Reset

↑ = clock transition LOW to HIGH

J-K flip-flop

J-K Flip-Flop

- ***Q*** and ***Q'*** are fed back to the pulse-steering NAND gates



- **Characteristic table**:

J	K	CLK	Q(t+1)	Comments
0	0	↑	Q(t)	No change
0	1	↑	0	Reset
1	0	↑	1	Set
1	1	↑	Q(t)'	Toggle

No illegal state!

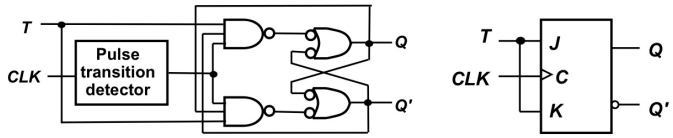
Q(t+1) = ?

T flip-flop

T Flip-Flop

toggle

- Formed by tying both inputs of J-K Flip-Flop together:



- **Characteristic table:**

T	CLK	Q(t+1)	Comments
0	↑	Q(t)	No change
1	↑	Q(t)'	Toggle

Q	T	Q(t+1)
0	0	0
0	1	1
1	0	1
1	1	0

Q(t+1) = ?

Flip-Flop Characteristic Tables

- Summary of the 4 flip-flops covered:

J	K	Q(t+1)	Comments
0	0	Q(t)	No change
0	1	0	Reset
1	0	1	Set
1	1	Q(t)'	Toggle

S	R	Q(t+1)	Comments
0	0	Q(t)	No change
0	1	0	Reset
1	0	1	Set
1	1	?	Unpredictable

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

T	Q(t+1)	
0	Q(t)	No change
1	Q(t)'	Toggle