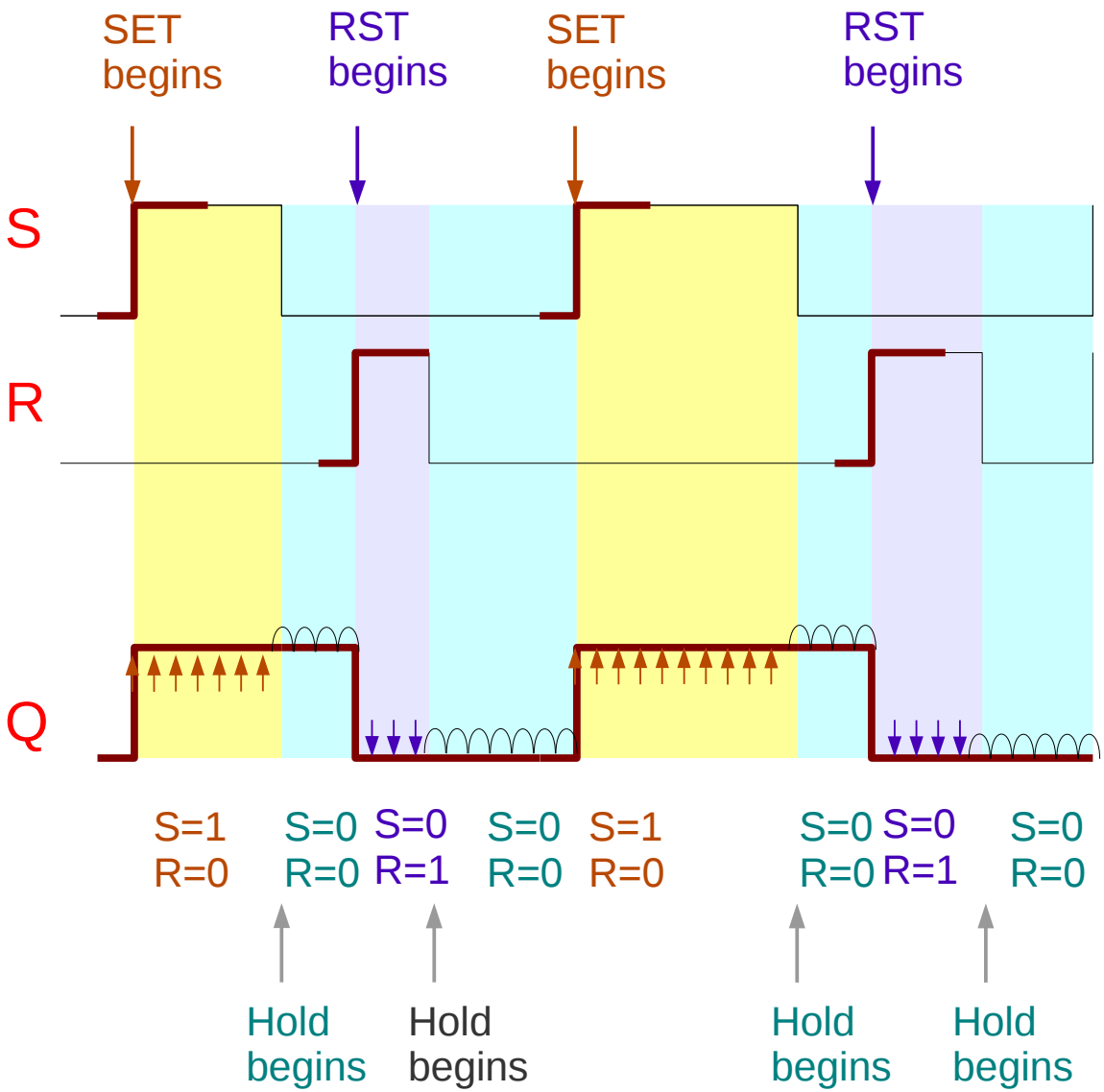
Latches and Flip-flops (1A)

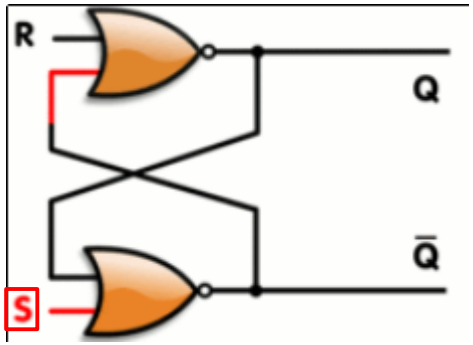
Young Won Lim
11/6/13

Please send corrections (or suggestions) to youngwlim@hotmail.com.

This document was produced by using OpenOffice and Octave.

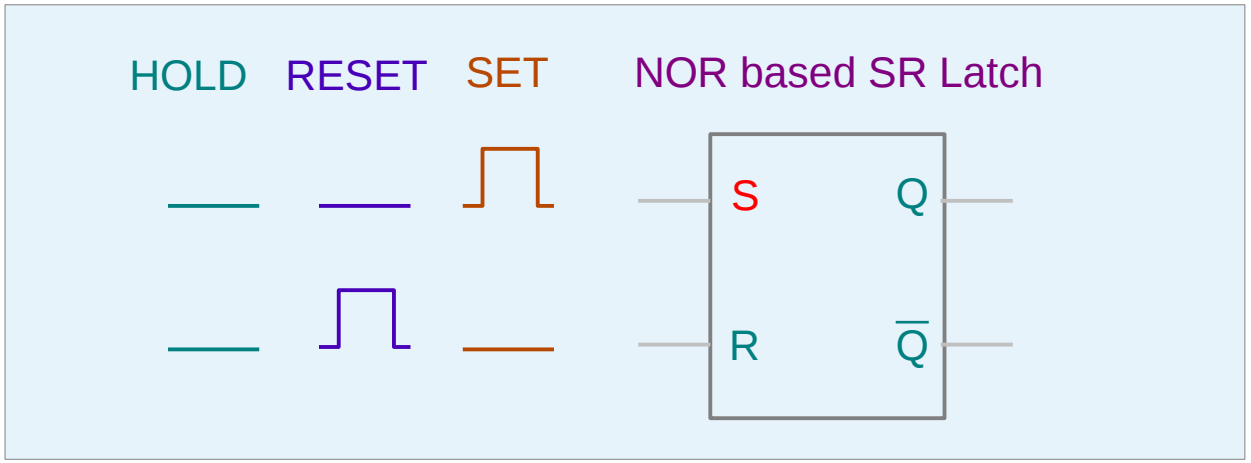
NOR-based SR Latch

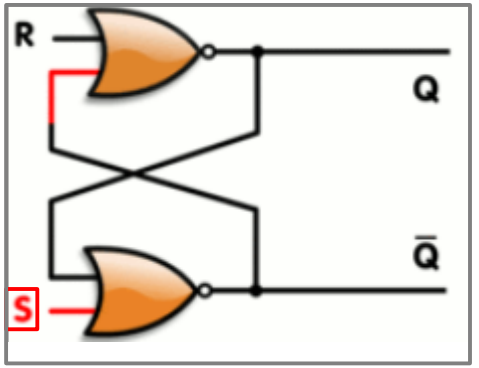


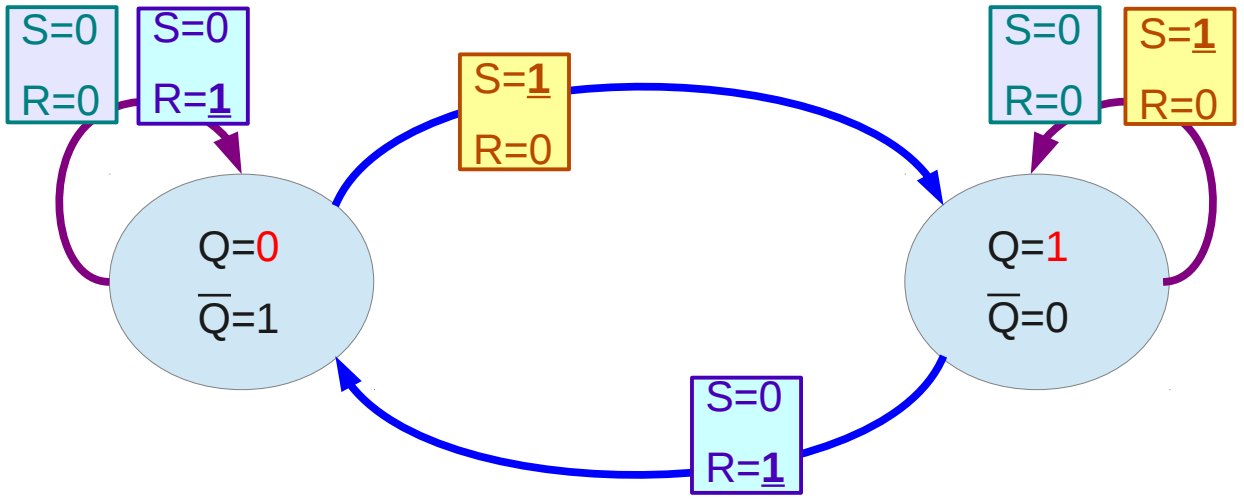


SET	S=1 R=0	Q=1 $\overline{Q}$=0
RESET	S=0 R=1	Q=0 $\overline{Q}$=1
HOLD	S=0 R=0	Q=old Q $\overline{Q}$=old $\overline{Q}$

NOR-based SR Latch States

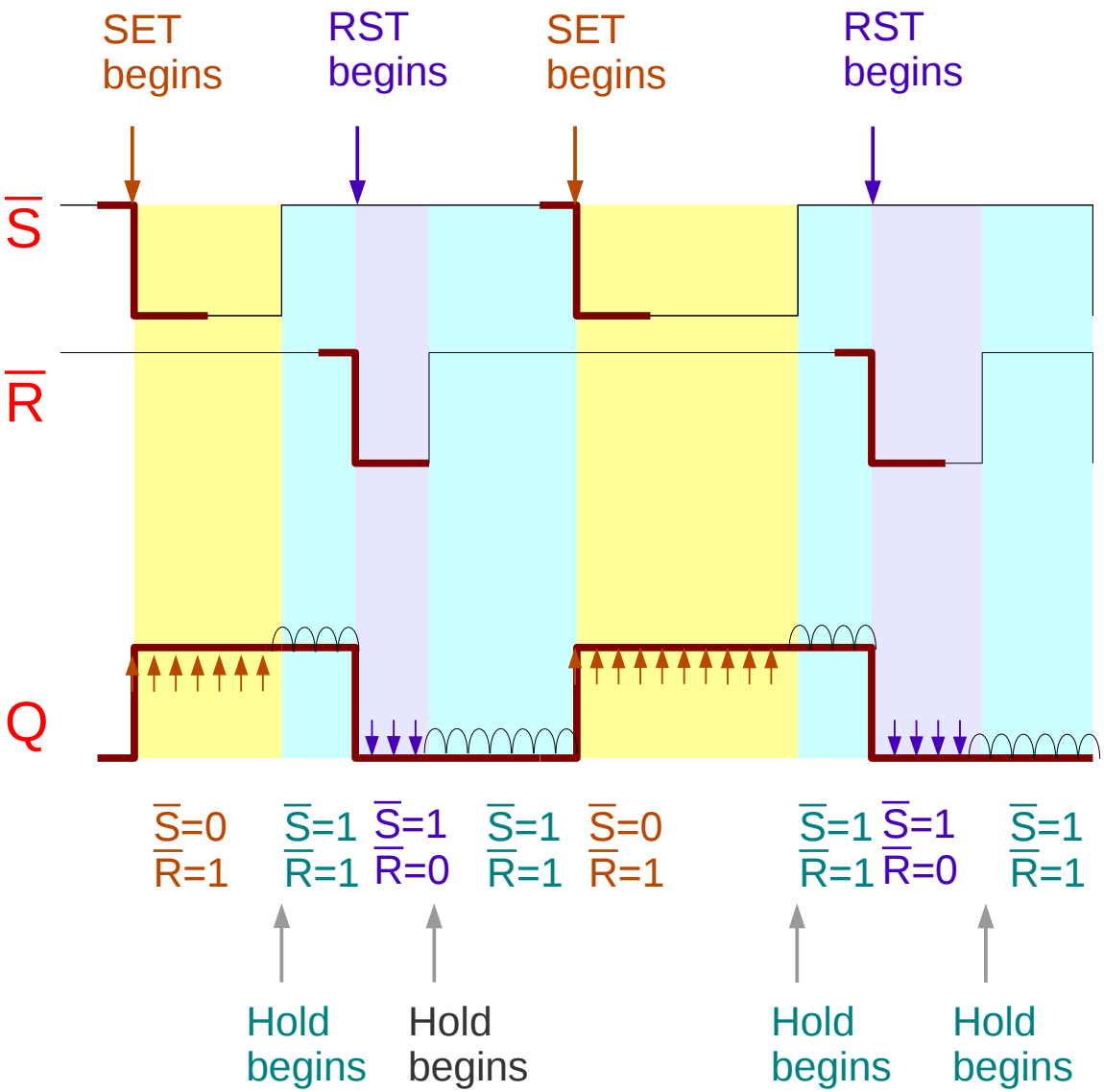


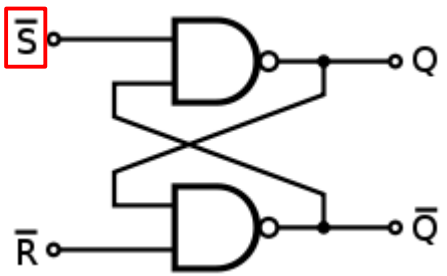




SET	S=1 R=0	Q=1 $\overline{Q}$=0
RESET	S=0 R=1	Q=0 $\overline{Q}$=1
HOLD	S=0 R=0	Q=old Q $\overline{Q}$=old $\overline{Q}$

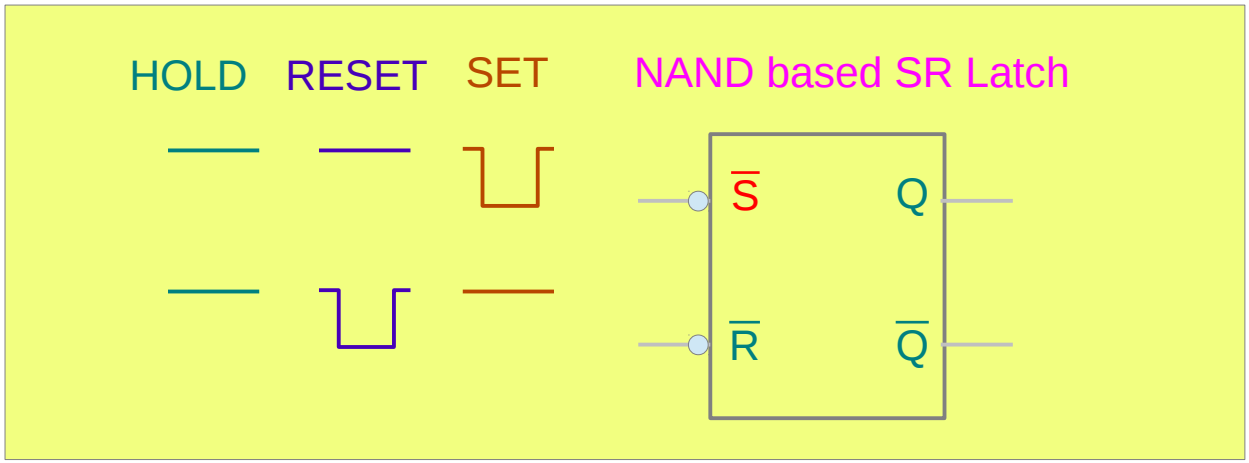
NAND-based SR Latch

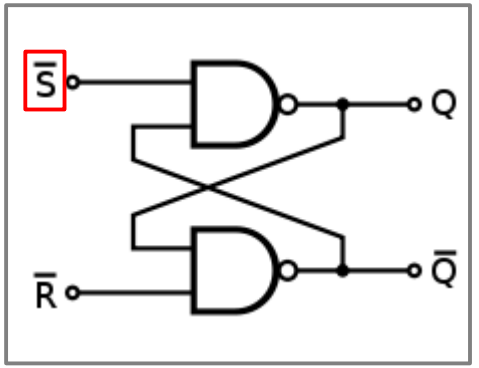


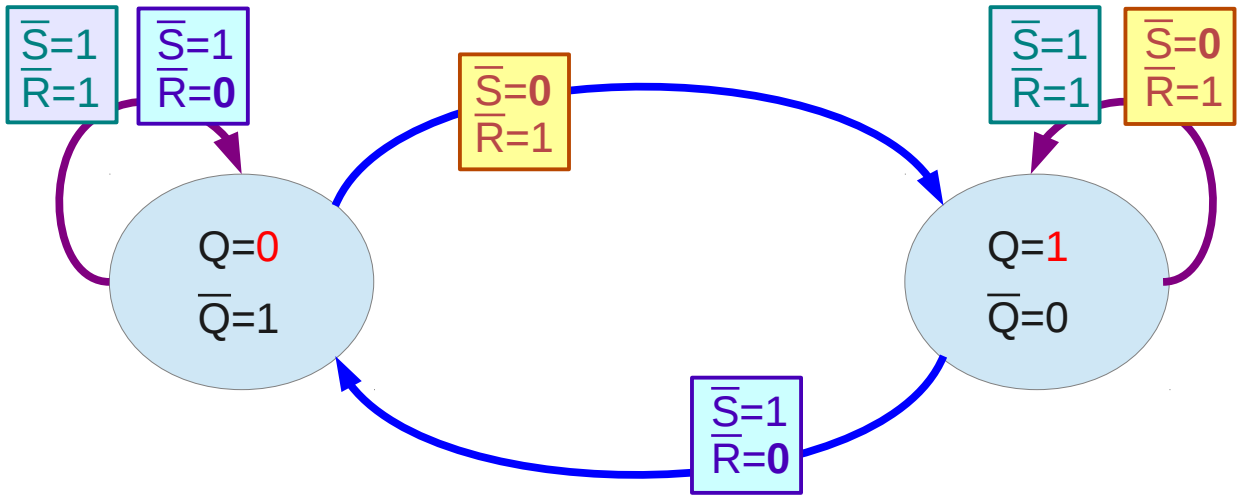


SET	$\overline{S}$=0 $\overline{R}$=1	Q=1 $\overline{Q}$=0
RESET	$\overline{S}$=1 $\overline{R}$=0	Q=0 $\overline{Q}$=1
HOLD	$\overline{S}$=1 $\overline{R}$=1	Q=old Q $\overline{Q}$=old $\overline{Q}$

NAND-based SR Latch States





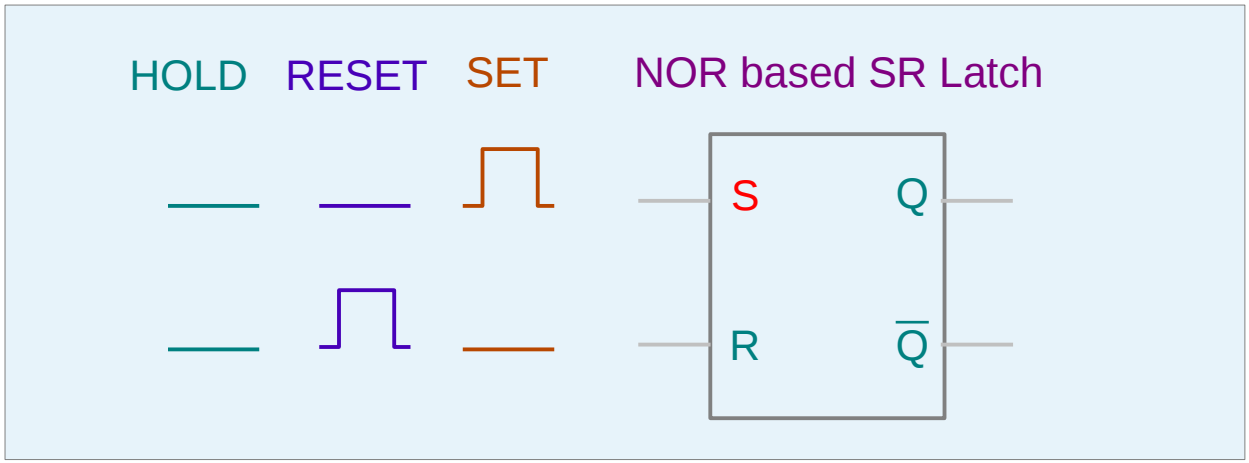


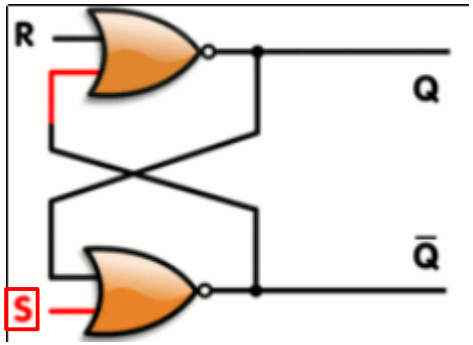
SET $\overline{S}=0$ $\overline{R}=1$ → Q=1, $\overline{Q}$=0

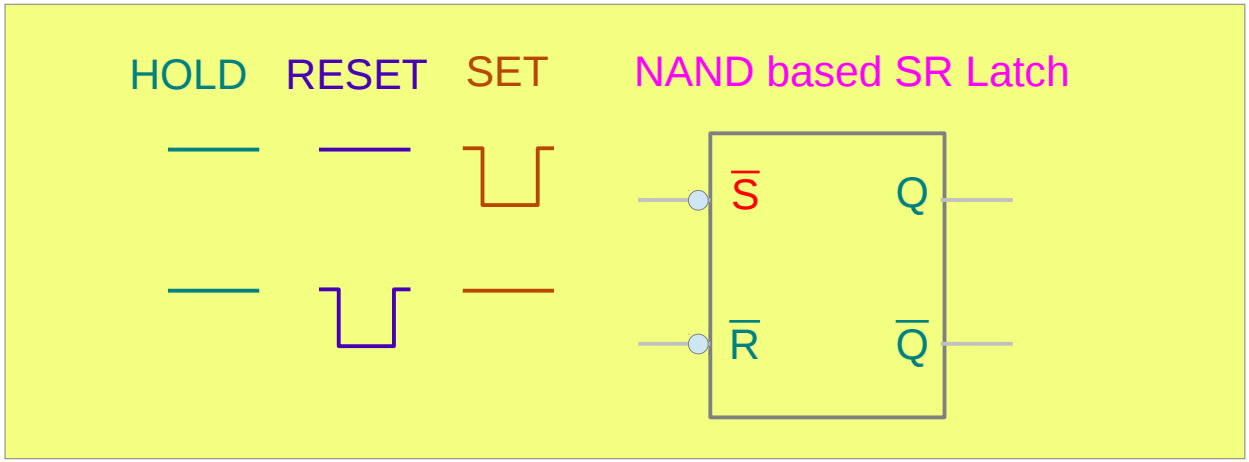
RESET $\overline{S}=1$ $\overline{R}=0$ → Q=0, $\overline{Q}$=1

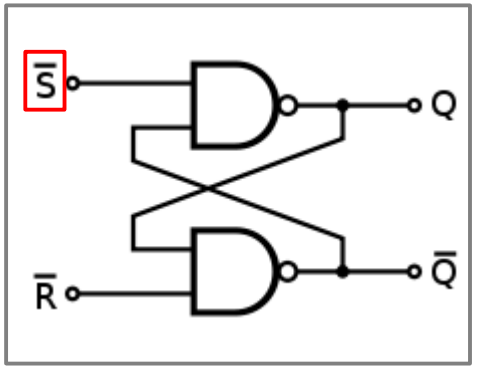
HOLD $\overline{S}=1$ $\overline{R}=1$ → Q=old Q, $\overline{Q}$=old $\overline{Q}$

SR Latch Symbols



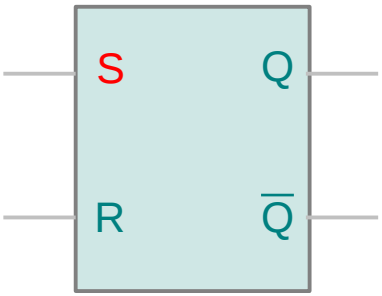




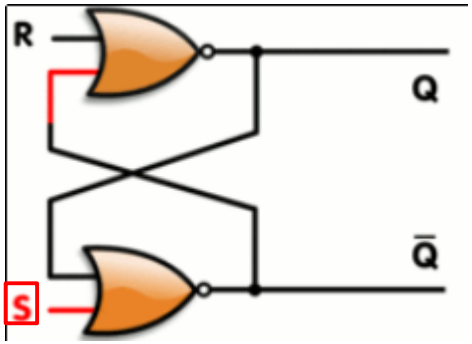


Active High and Low Inputs

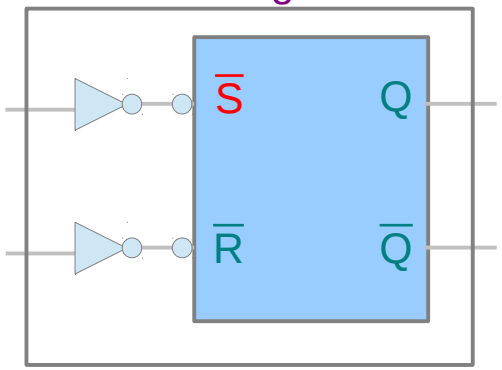
Active High



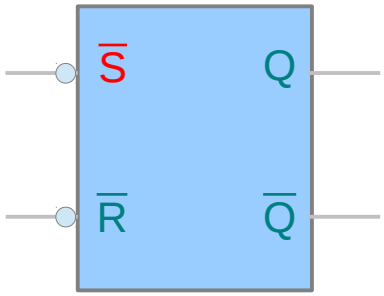
Active High



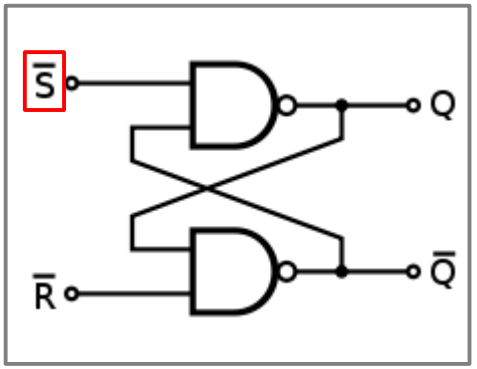
Active High



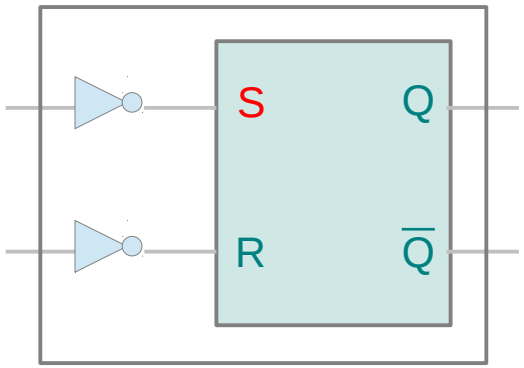
Active Low



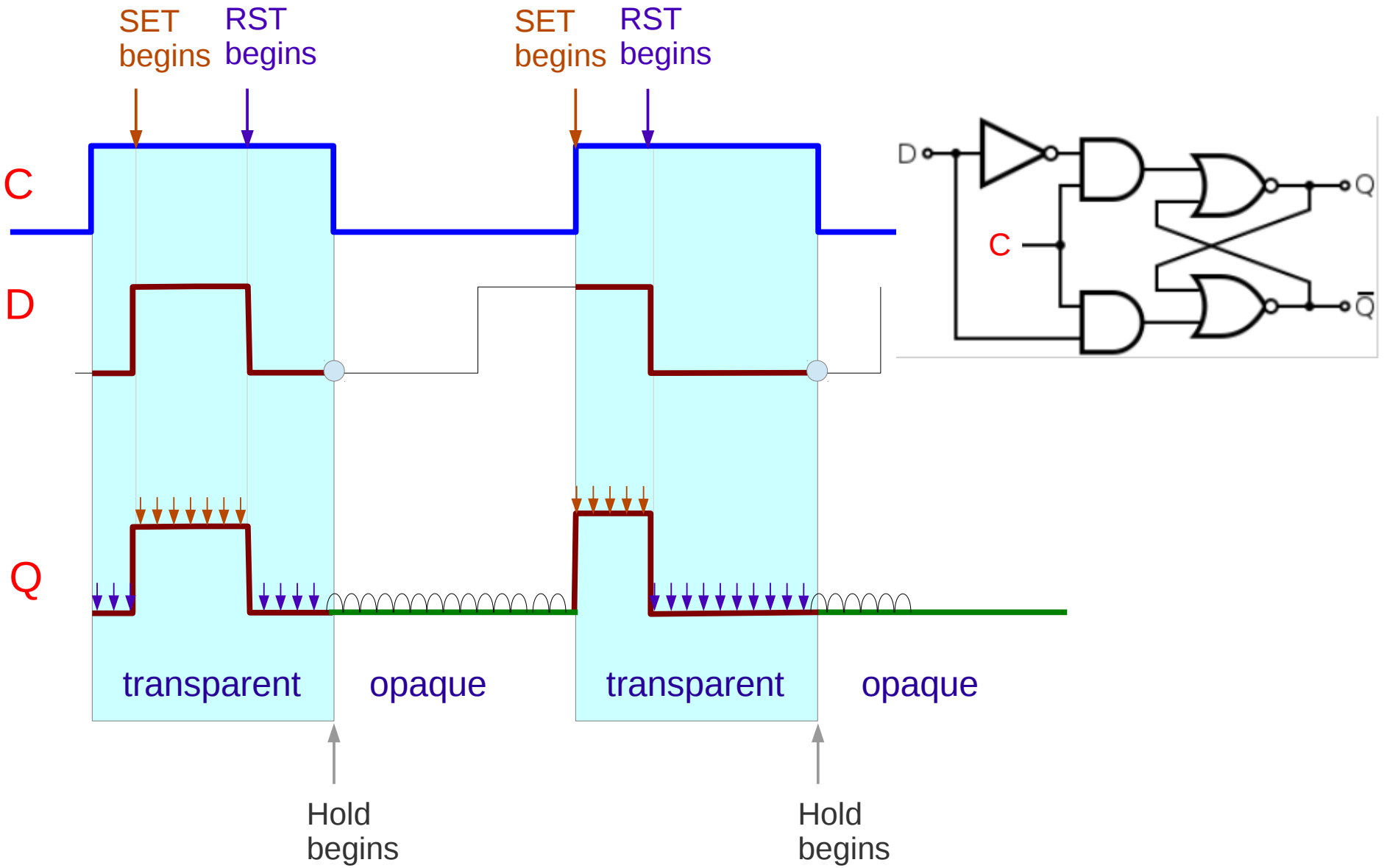
Active Low



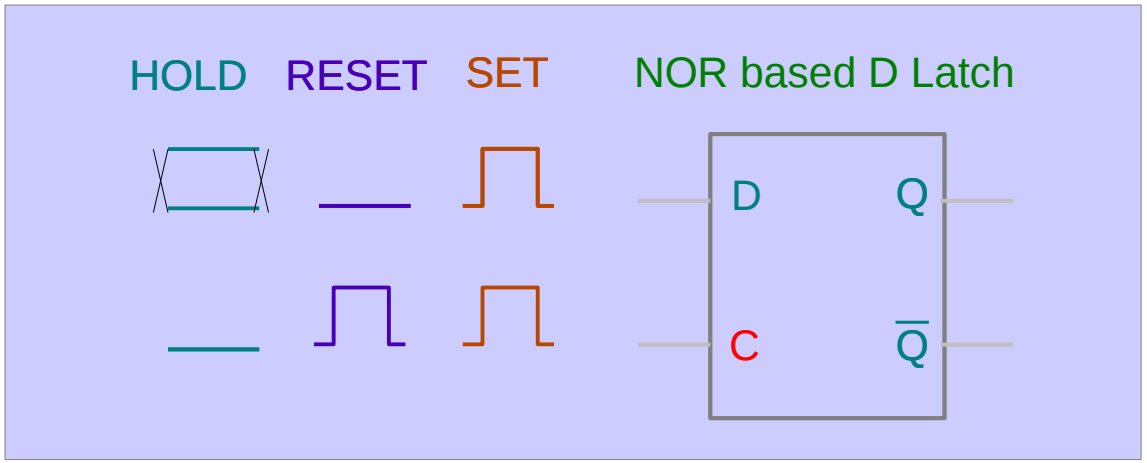
Active Low

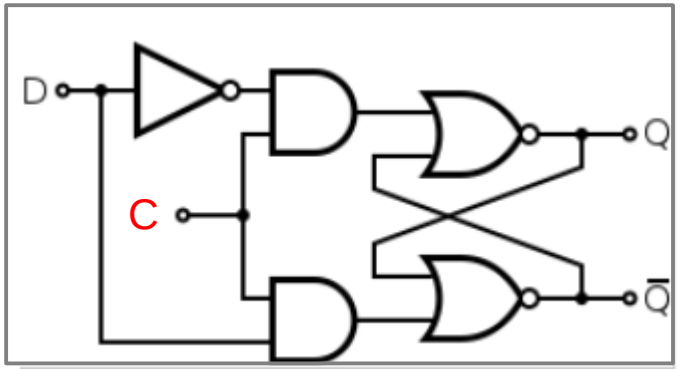


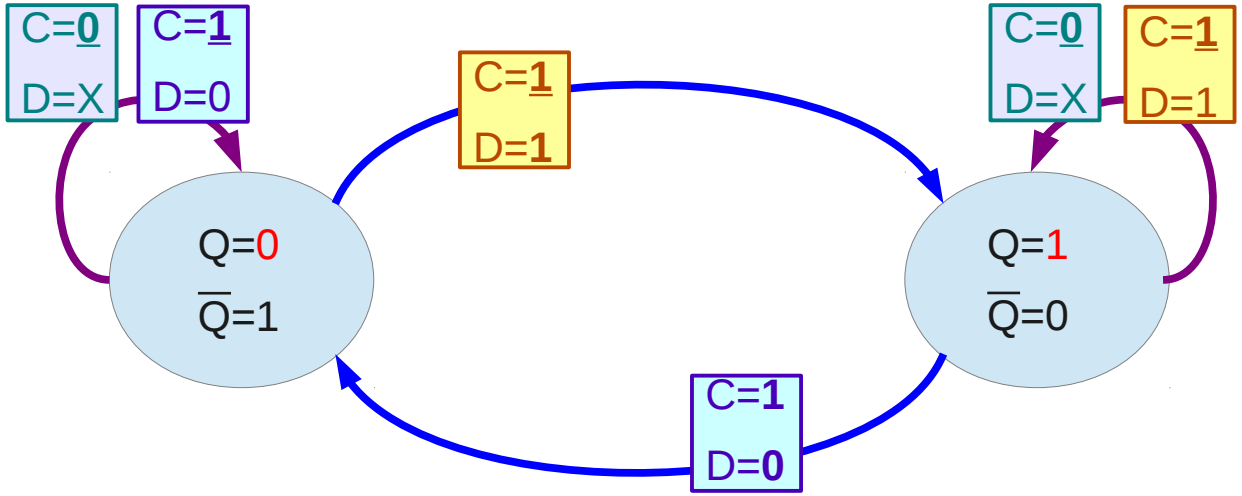
NOR-based D Latch



NOR-based D Latch

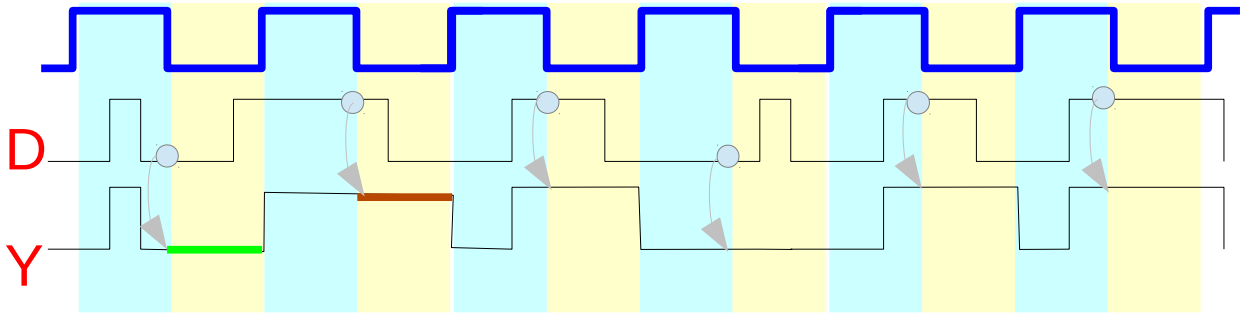


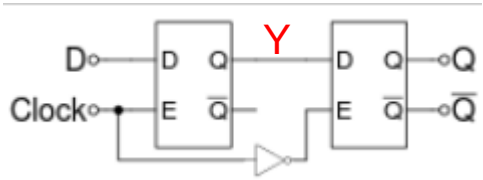




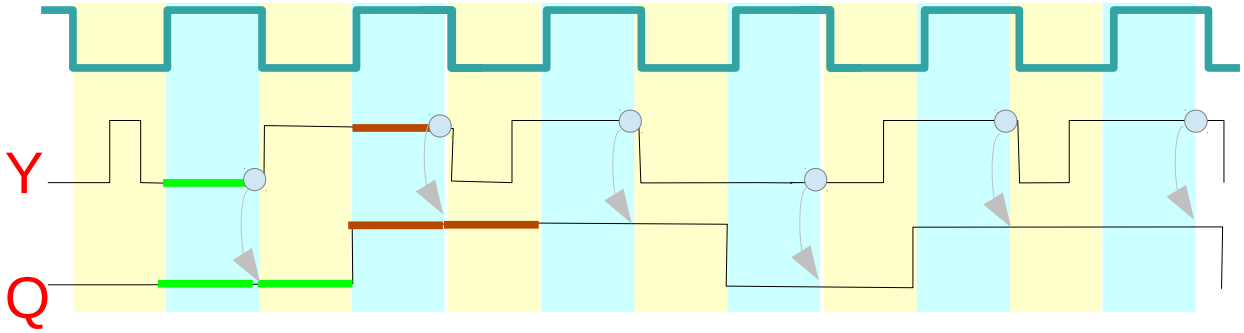
Master-Slave D FlipFlop

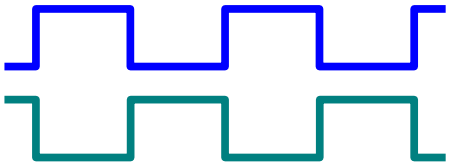
Master D Latch

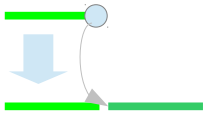




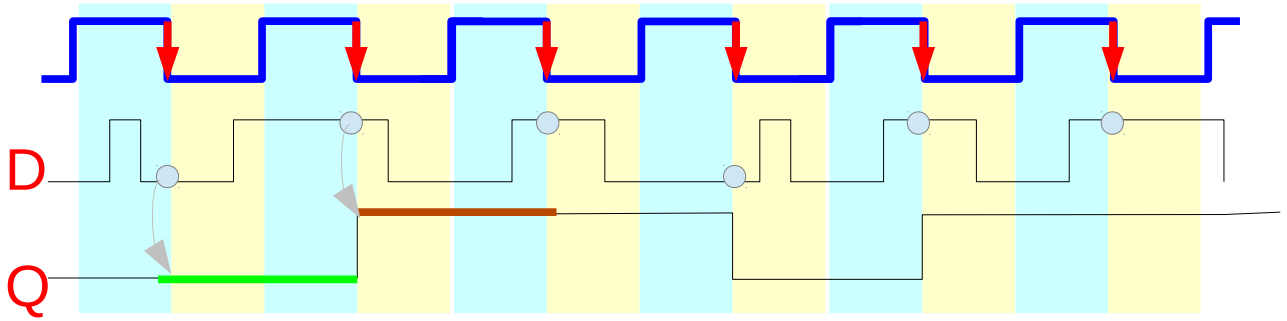
Slave D Latch







Master-Slave D F/F

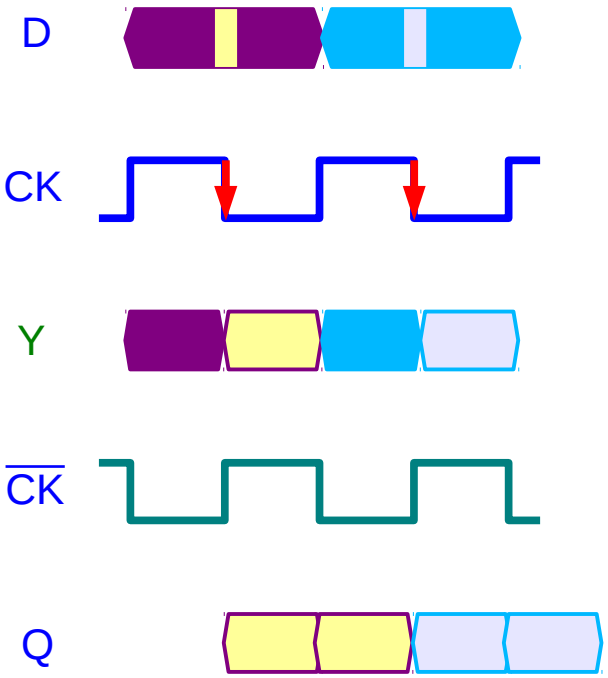


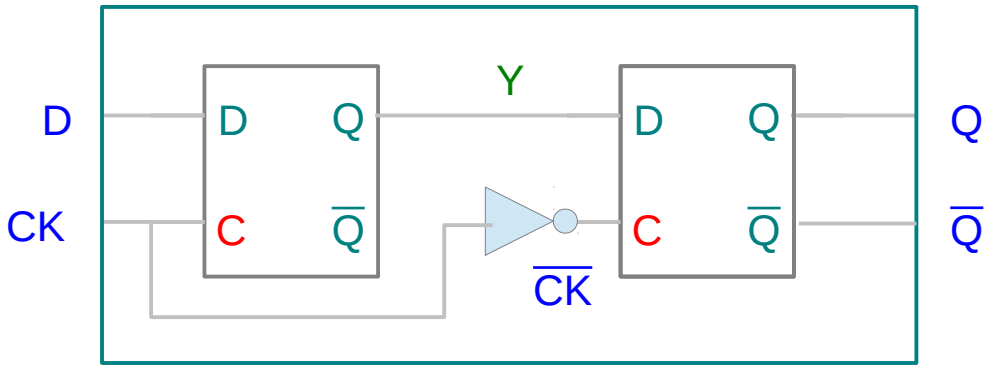
the hold output of the master is transparently reaches the output of the slave

this value is held for another half period

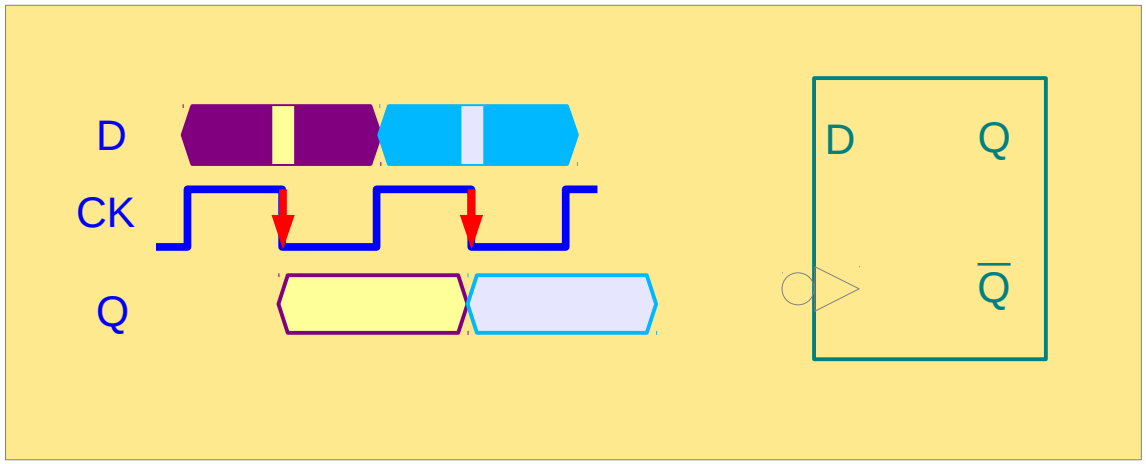
Master-Slave D FlipFlop – Falling Edge

Master D Latch



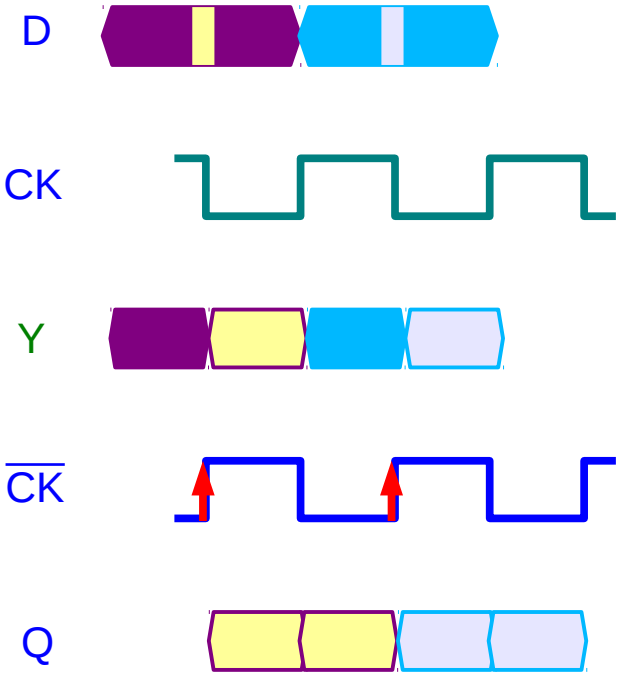


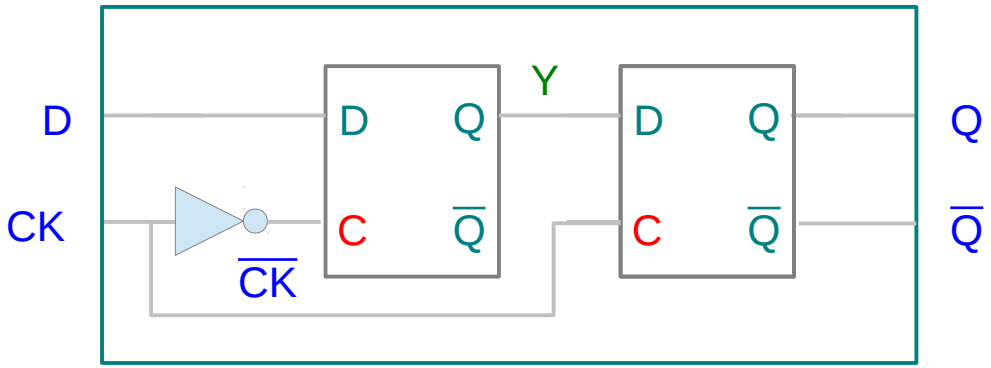
Slave D Latch



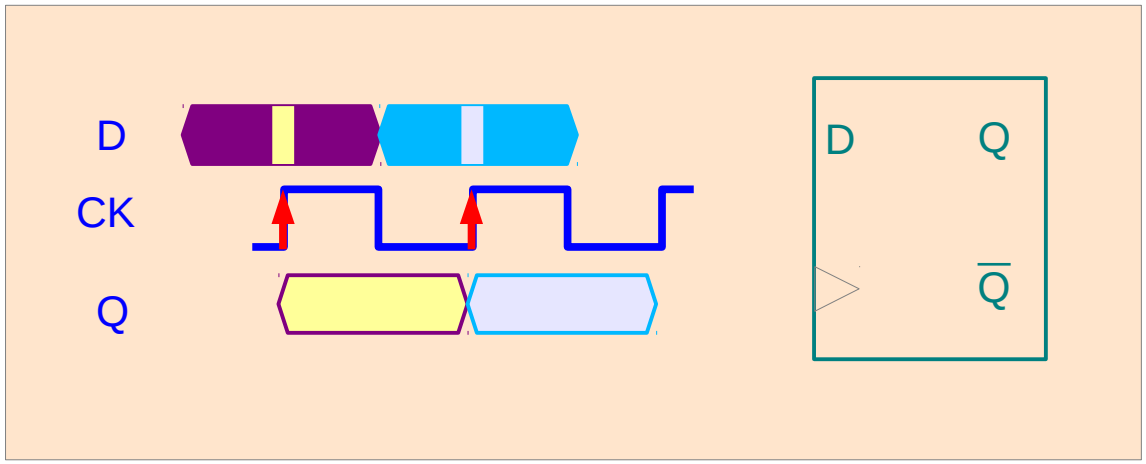
Master-Slave D FlipFlop – Rising Edge

Master D Latch





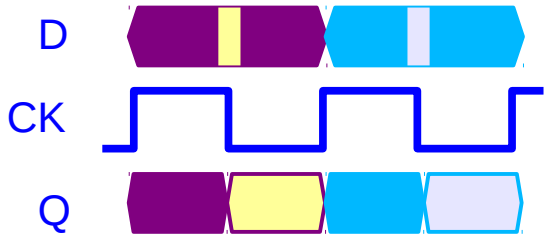
Slave D Latch

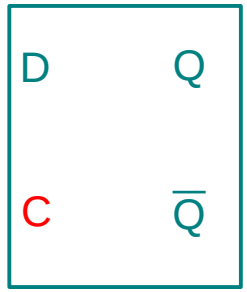


D Latch & D FlipFlop

Level Sensitive D Latch

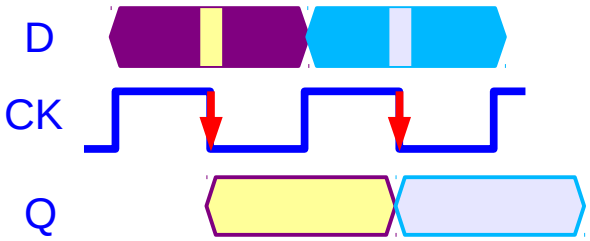
CK=1 transparent
CK=0 opaque

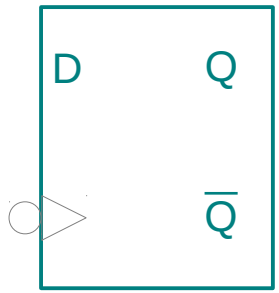




Edge Sensitive D FlipFlop

CK=1→0 transparent
else opaque





Advantages of Latches over FFs

Flipflop designs are very **easy to verify timing**

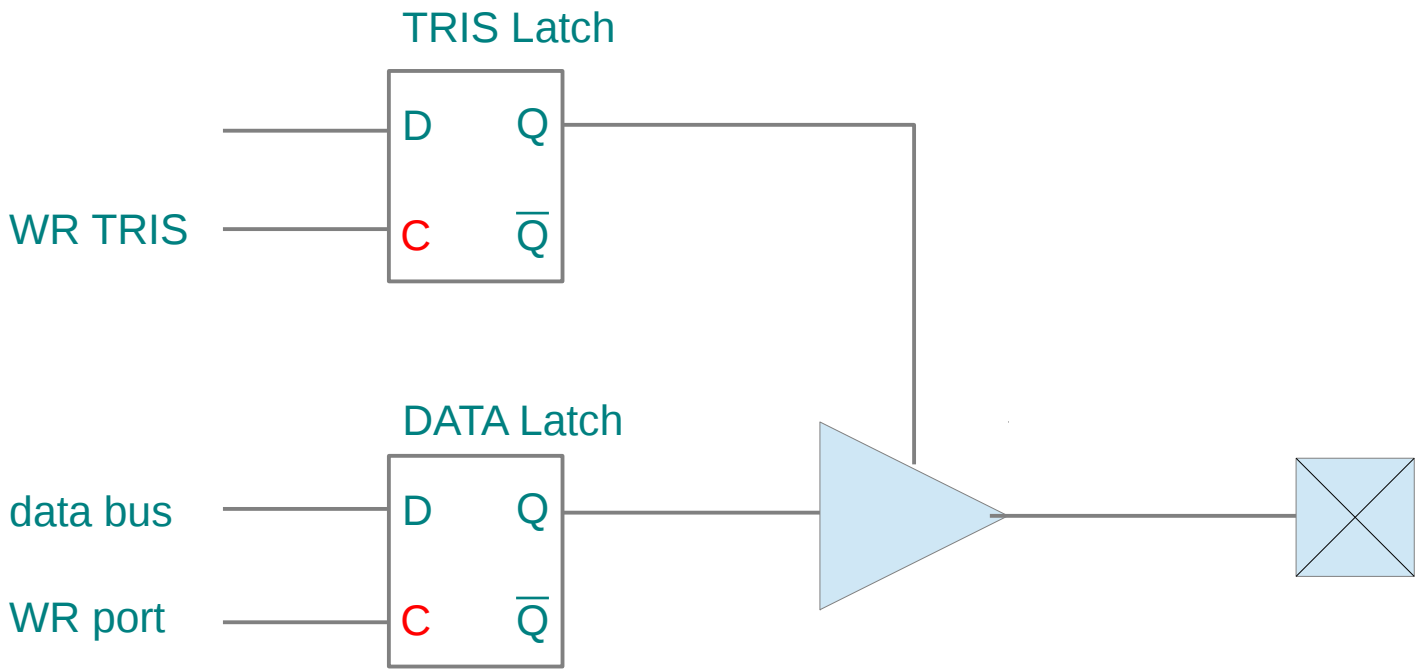
- Each path between flip-flops must be less than the clock period
- Tools check for skew, setup, and hold time violations
- Short paths are padded
  (buffers are added to slow down the signals)
- Skew in flip-flop based systems affects the critical path

Most designs in industry are based on flip-flops

Latch designs are **more flexible** than a flip-flop design

- Need to CAD tools to make sure it works
- Can borrow time to allow a path to be longer than clock period
- Can tolerate clock skew
  -- skew does not directly add to cycle time
- Less silicon area

Latches at the output ports



References

[1] http://en.wikipedia.org/
[2] M. M. Mano, C. R. Kime, “Logic and Computer Design Fundamentals”, 4th ed.
[3] D.M. Harris, S. L. Harris, “Digital Design and Computer Architecture”
[4] http://www.ee.ic.ac.uk/pcheung lecture note “Topic 7 – clocking strategies”