ECN 104

Digital Logic Design

Spring 2023

Tanmoy Pramanik

http://ece.iitr.ac.in/tanmoy_pramanik

Acknowledgment: Content mostly taken from many textbooks

Multivibrator circuits

- Three operating modes:
  - Astable: also called “free-running,” output is neither stable at logic high nor logic low – keeps switching on its own between the two levels at some frequency decided by the delay elements in the circuit
  - Monostable: also called “single-shot”, the output is stable at one of the levels and can be momentarily switched to the other state by a trigger signal
  - Bistable: Both states are stable unless switched by external inputs; These are similar in functionality as flip-flops.

Astable 555 IC Timer

- Commonly used for generating pulses with a programmable duty cycle

The charge time (output high) is given by:

$$t_1 = 0.693\ (R_A + R_B)\ C$$

And the discharge time (output low) by:

$$t_2 = 0.693\ (R_B)\ C$$

Thus the total period is:

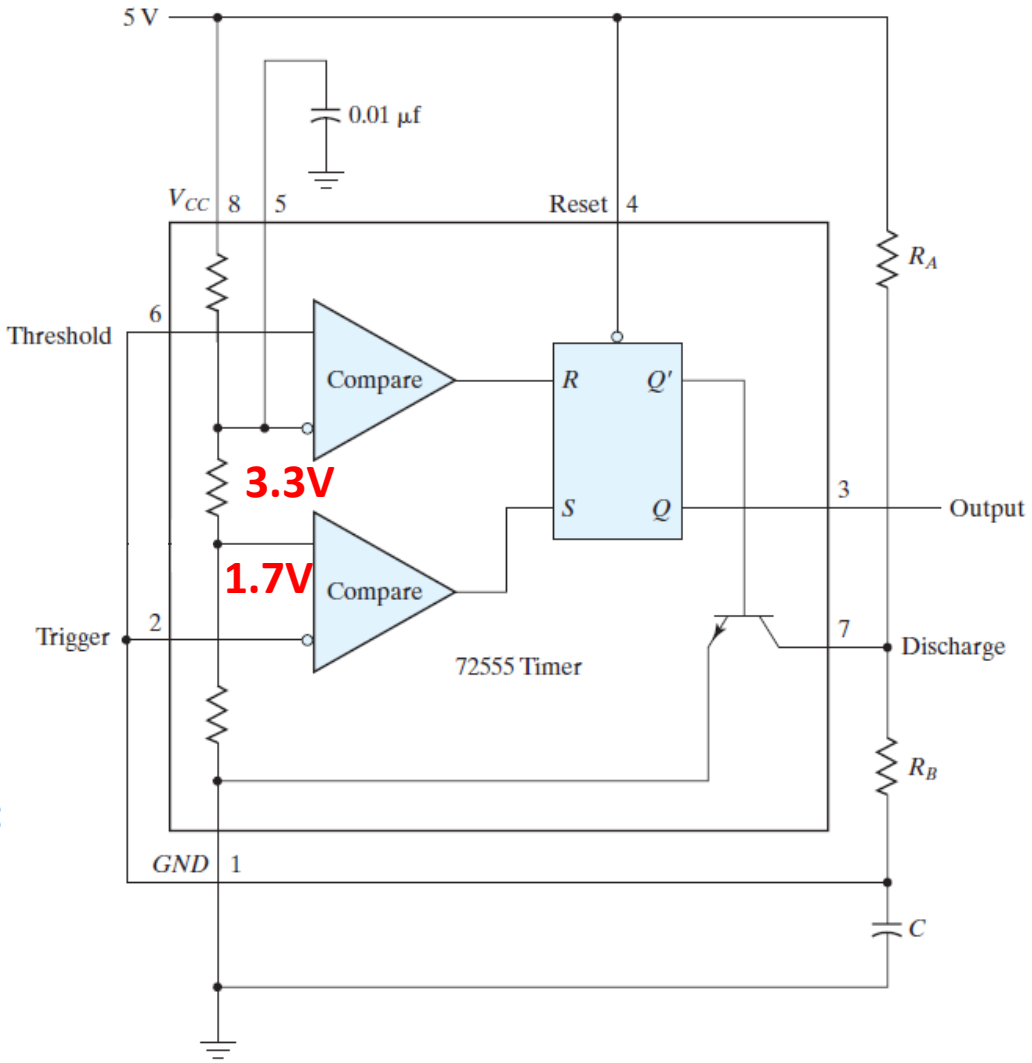
$$T = t_1 + t_2 = 0.693\ (R_A + 2R_B)\ C$$

The frequency of oscillation is:

$$f = \frac{1}{T} = \frac{1.44}{(R_A + 2R_B)\ C}$$

The duty cycle is:

$$D = \frac{R_B}{R_A + 2R_B}$$



https://www.ti.com/lit/ds/symlink/lm555.pdf

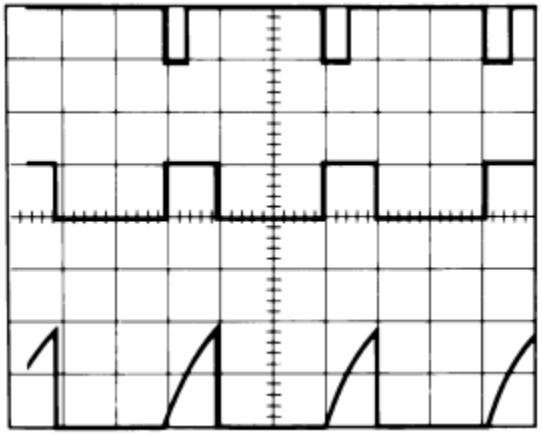
Monostable 555 IC Timer

- Generate pulse of programmable duration for each trigger – can be used for frequency division etc.

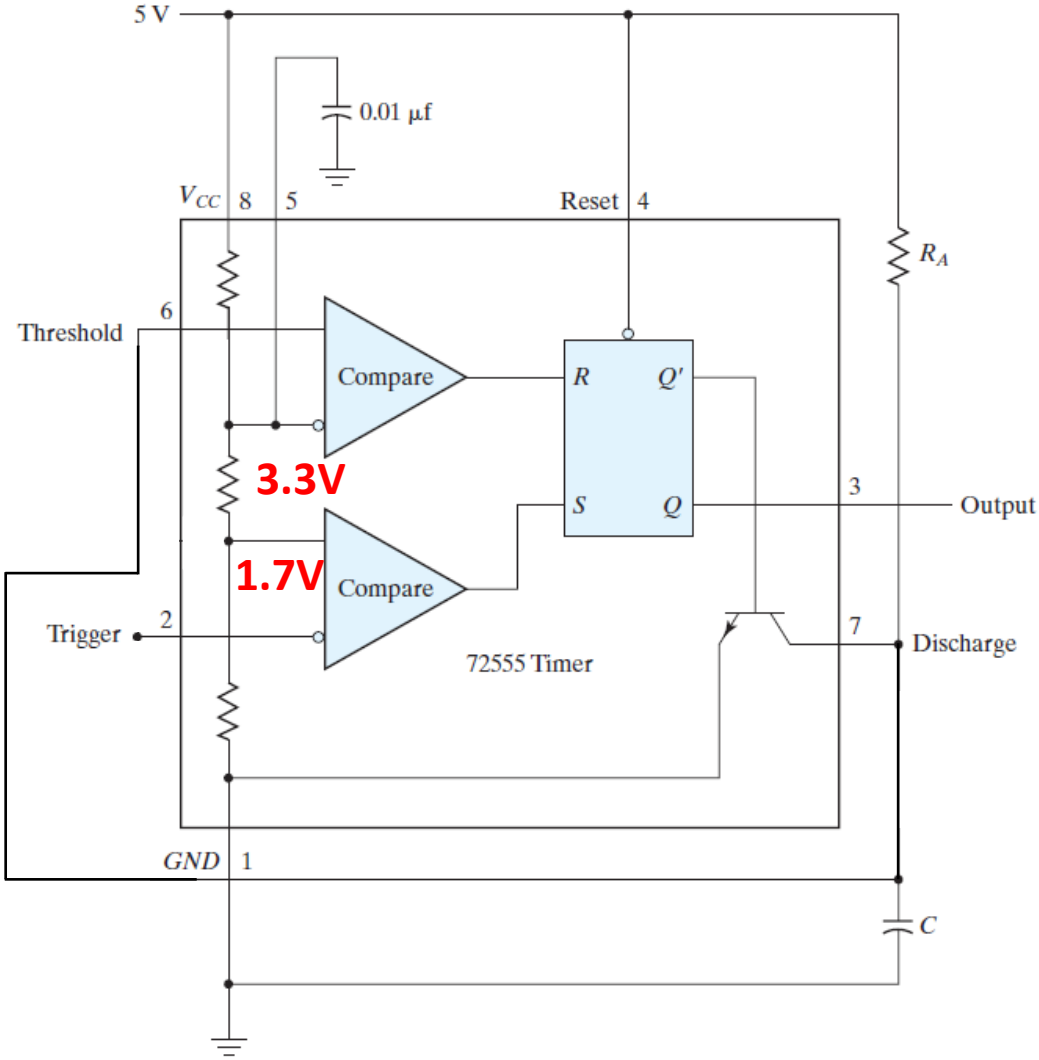
Trigger

O/P

Cap voltage



$$t_A = 1.1R_AC$$



https://www.ti.com/lit/ds/symlink/lm555.pdf

Ring Oscillator (Astable Circuit)

Example 3.3 ASTABLE CIRCUITS

Alyssa P. Hacker encounters three misbegotten inverters who have tied themselves in a loop, as shown in Figure 3.16. The output of the third inverter is *fed back* to the first inverter. Each inverter has a propagation delay of 1 ns. Determine what the circuit does.

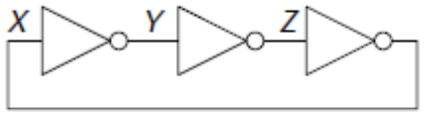


Figure 3.16 Three-inverter loop

Solution: Suppose node X is initially 0. Then $Y = 1$, $Z = 0$, and hence $X = 1$, which is inconsistent with our original assumption. The circuit has no stable states and is said to be *unstable* or *astable*. Figure 3.17 shows the behavior of the circuit. If X rises at time 0, Y will fall at 1 ns, Z will rise at 2 ns, and X will fall again at 3 ns. In turn, Y will rise at 4 ns, Z will fall at 5 ns, and X will rise again at 6 ns, and then the pattern will repeat. Each node oscillates between 0 and 1 with a *period* (repetition time) of 6 ns. This circuit is called a *ring oscillator*.

The period of the ring oscillator depends on the propagation delay of each inverter. This delay depends on how the inverter was manufactured, the power supply voltage, and even the temperature. Therefore, the ring oscillator period is difficult to accurately predict. In short, the ring oscillator is a sequential circuit with zero inputs and one output that changes periodically.

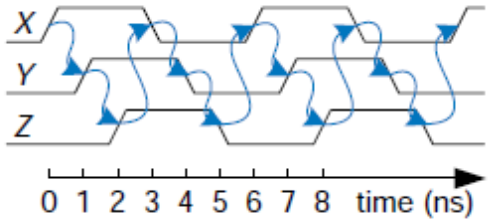


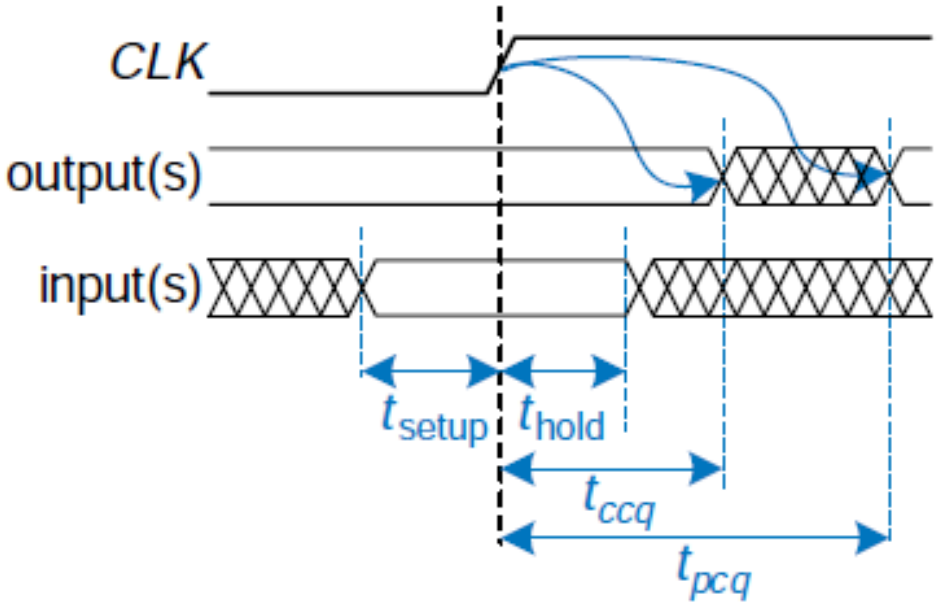
Figure 3.17 Ring oscillator waveforms

Timing constraints in sequential circuits

- Consider a D-flipflop – The D input is sampled at the rising edge of the clock signal.
- What happens if the D-input changes simultaneously as the clock rises?
  - The flipflop output might become a value that is neither 0 nor 1 - undefined logic
  - To avoid this problem, timing constraints are placed for the D-input with respect to the clock edge
  - These are known as “setup time” and “hold time” constraints
- Also, for a real sequential circuit with many flipflops, the clock may not reach all the flip-flops precisely at the same time – this is known as “clock skew”

Concept of *setup* and *hold* times

- t_{ccq} → Clock to Q contamination time – minimum after which output(s) starts to change
- t_{pcq} → Clock to Q propagation time – maximum time by which the output(s) settles down
- The input(s) must stabilize at least t_{setup} time before the rising edge of the clock
- Similarly, input(s) must remain stable for at least t_{hold} time, after the rising edge of the clock.
- $t_{setup} + t_{hold}$ is called the aperture time of the circuit



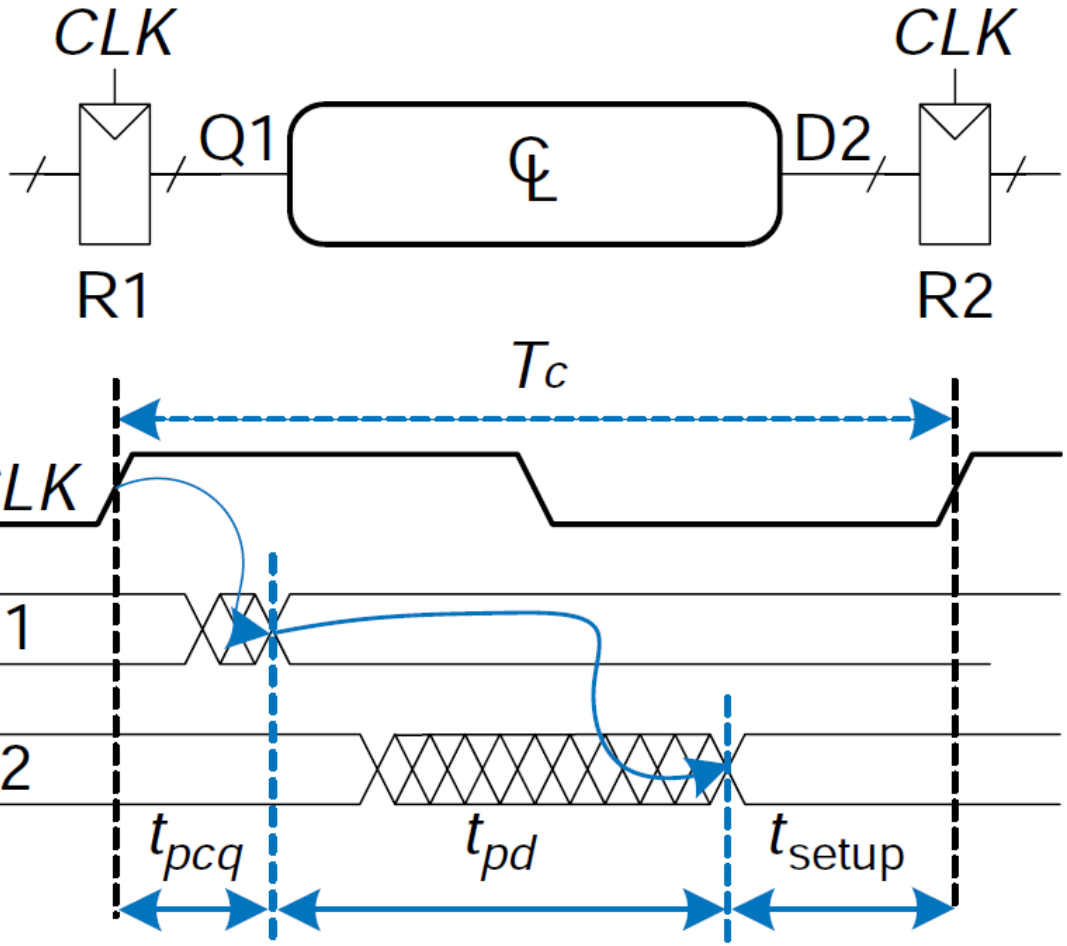
Example of *setup* time constraint

- With respect to R2, all the transitions must settle at least setup time before the next clock edge

$$T_c \geq t_{pcq} + t_{pd} + t_{\text{setup}}$$

- If the clock is fixed, CL must be re-designed to lower delay

$$t_{pd} \leq T_c - (t_{pcq} + t_{\text{setup}})$$



- Setup time constraint also known as max-delay constraint

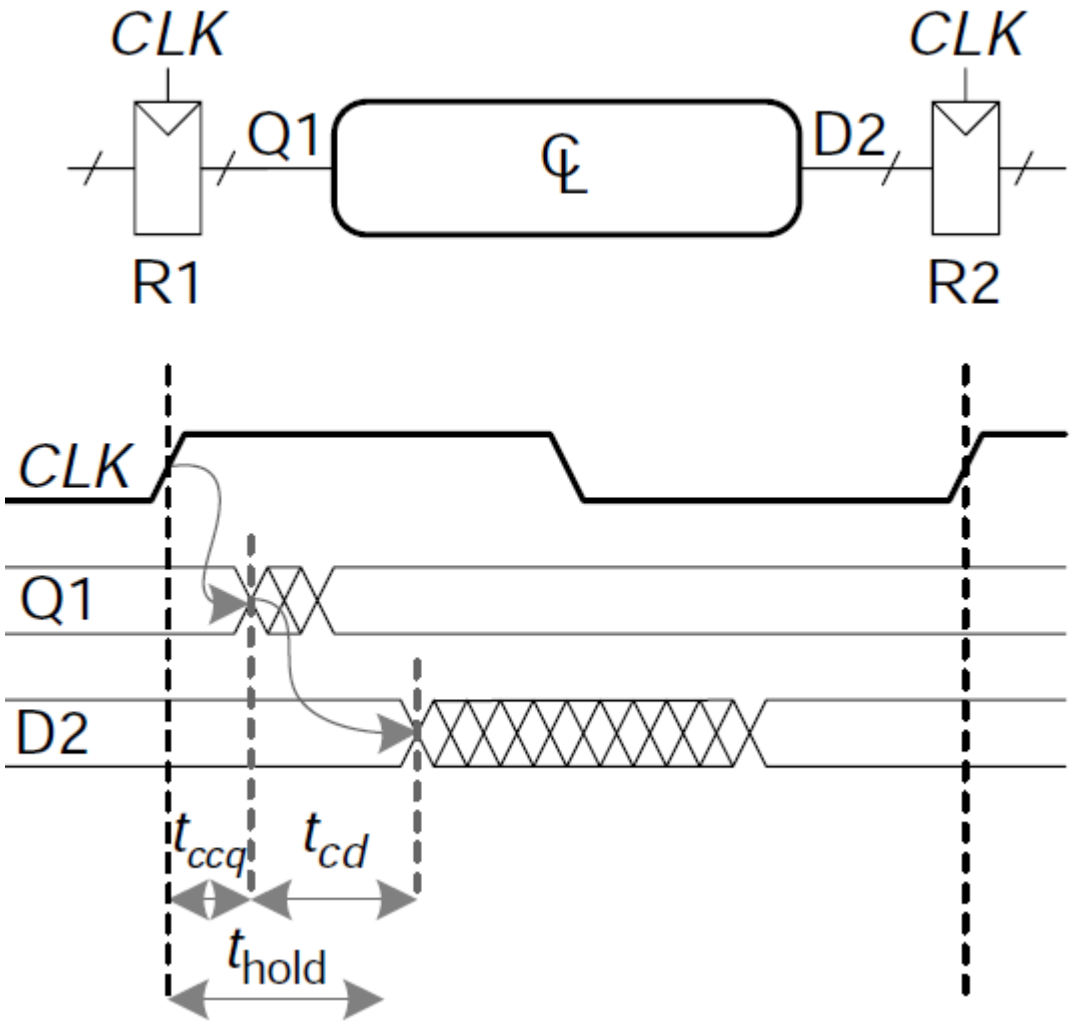
Example of *hold* time constraint

- To satisfy the hold time requirement for R2, we need,

$$t_{ccq} + t_{cd} \geq t_{\text{hold}}$$

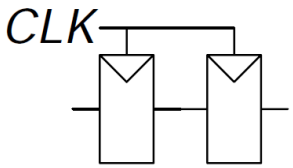
$$t_{cd} \geq t_{\text{hold}} - t_{ccq}$$

- This is also known as *min-delay constraint*
- Setup and hold times together defines the maximum and minimum delay required through the CL block



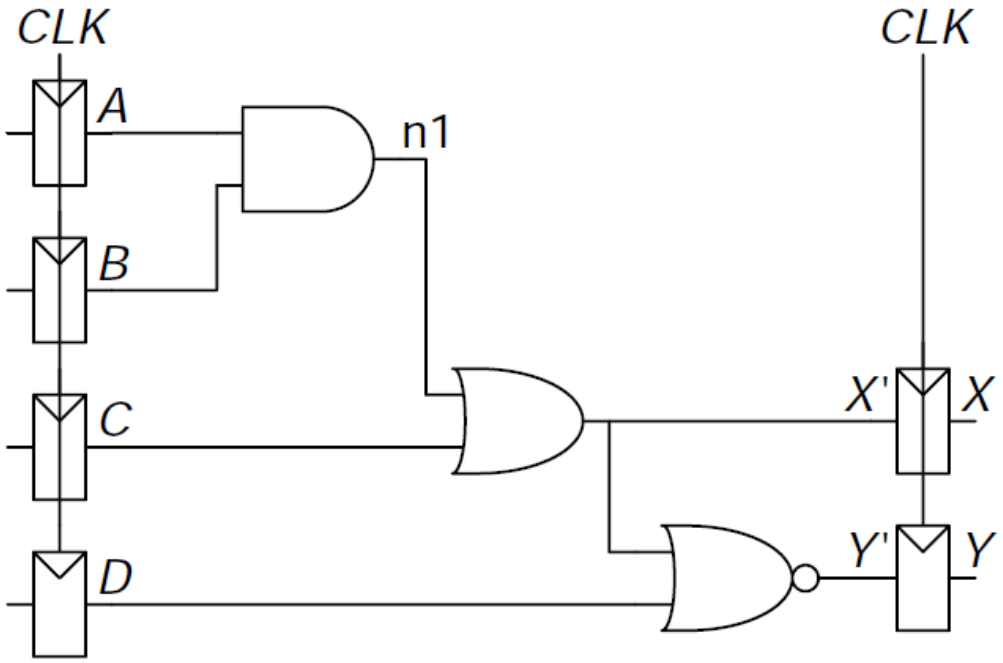
Directly connected flip-flop chain

- No CL block in between requires $t_{\text{hold}} \leq t_{ccq}$
- A reliable flip-flop is preferred to have a hold time shorter than its contamination delay. Often the hold times of flip-flops are designed to be zero.
- Nevertheless, hold time constraints are critically important. If they are violated, the only solution is to increase the contamination delay through the logic, which requires redesigning the circuit. Unlike setup time constraints, they cannot be fixed by adjusting the clock period.



Example Timing Analysis #1

- What is the maximum speed of the circuit?
- Is there hold time violation?

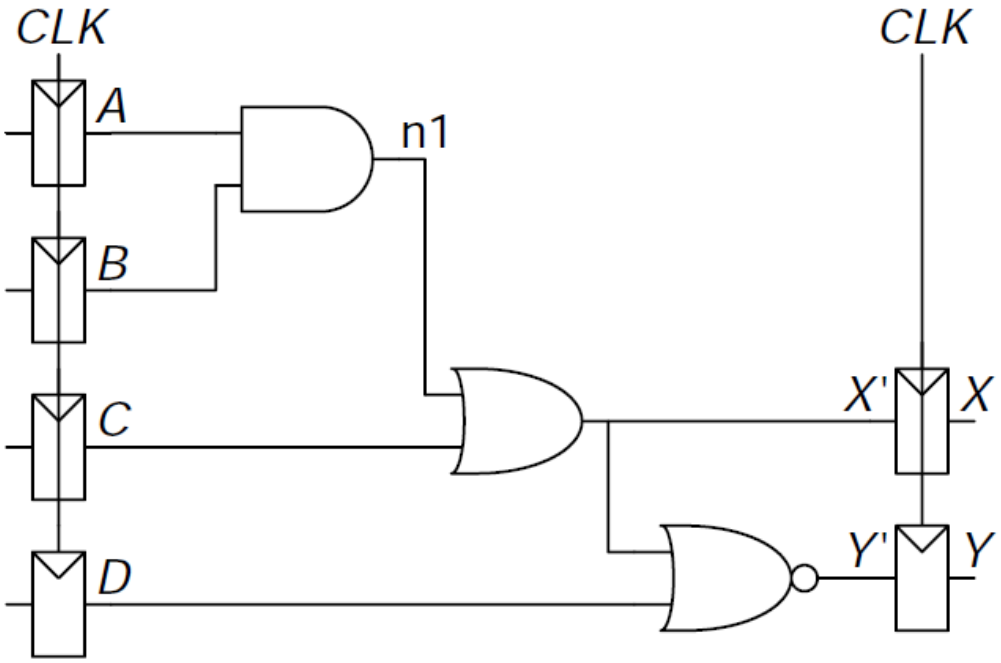


All FFs → $t_{ccq} = 30\ ps$ $t_{pcq} = 80\ ps$

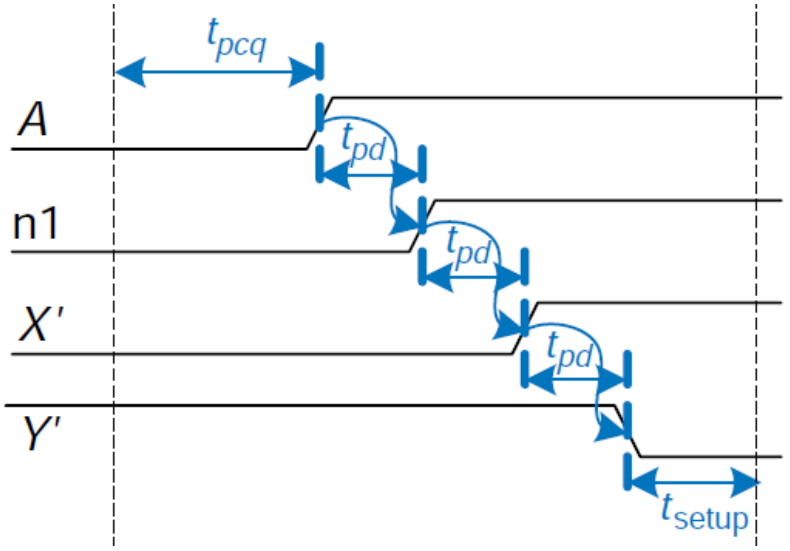
$t_{setup} = 50\ ps; t_{hold} = 60\ ps$

All gates → $t_{cd} = 25\ ps; t_{pd} = 40\ ps$

Example Timing Analysis #1 … contd.



- Find the critical path:



All FFs → $t_{ccq} = 30\ ps \quad t_{pcq} = 80\ ps$

$t_{setup} = 50\ ps; t_{hold} = 60\ ps$

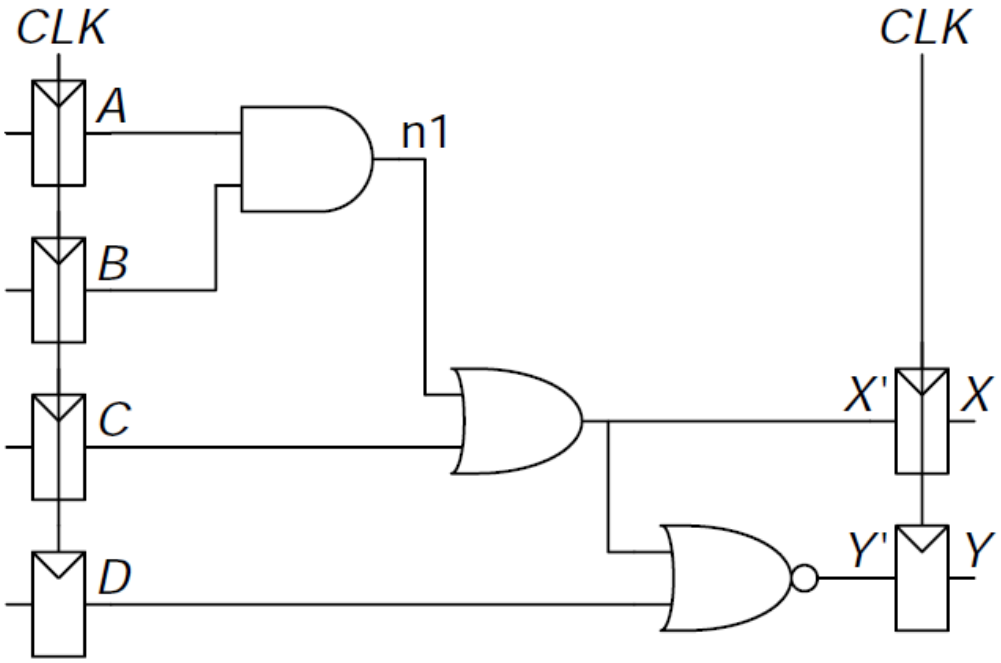
All gates → $t_{cd} = 25\ ps; t_{pd} = \ \ 40\ ps$

$$T_c \geq t_{pcq} + 3\,t_{pd} + t_{\text{setup}}$$

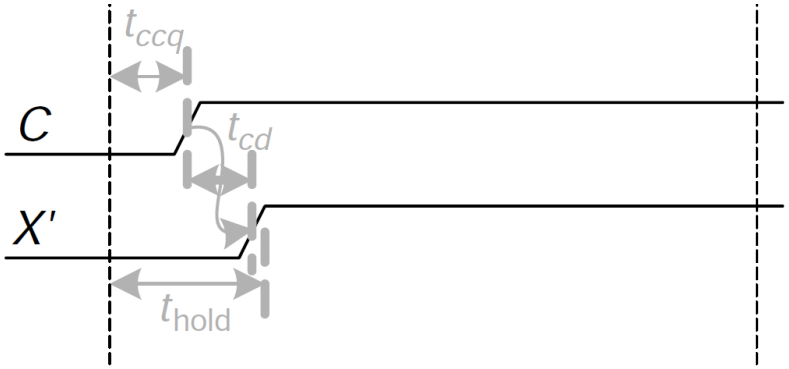
$$= 80 + 3 \times 40 + 50 = 250 \text{ ps}$$

$$\therefore f_c = 1/T_c = 4 \text{ GHz}.$$

Example Timing Analysis #1 … contd.



- Find the short path:



$t_{ccq} + t_{cd} = 30 + 25 = 55$ ps

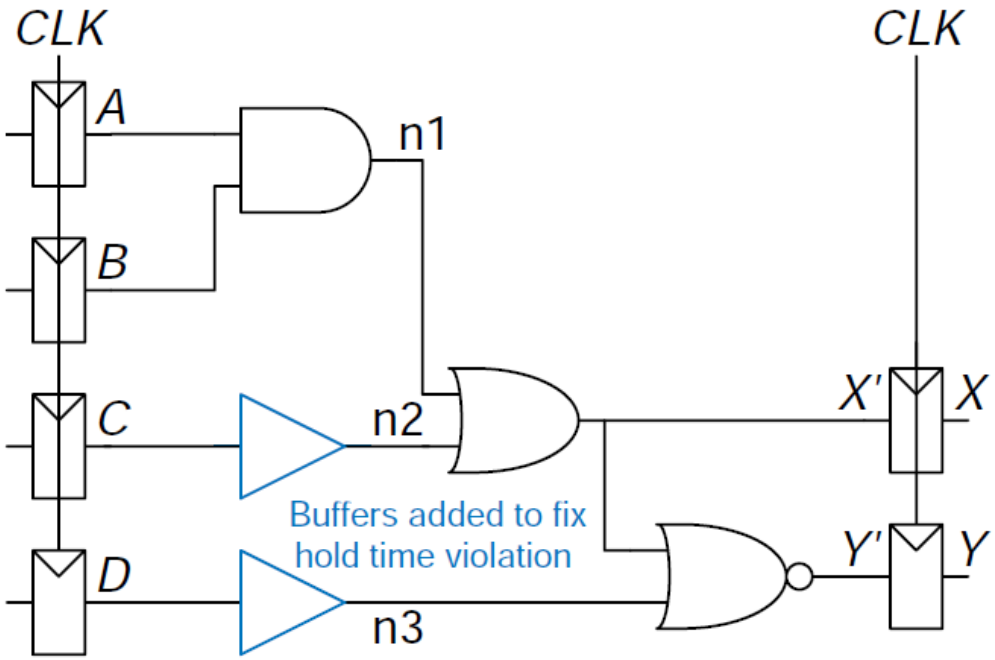
All FFs → $t_{ccq} = 30\ ps \quad t_{pcq} = 80\ ps$

$t_{setup} = 50\ ps; t_{hold} = 60\ ps$

There is hold time violation!

All gates → $t_{cd} = 25\ ps; t_{pd} = \ \ 40\ ps$

Example Timing Analysis #1 … contd.



All FFs → $t_{ccq} = 30\ ps \quad t_{pcq} = 80\ ps$

$t_{setup} = 50\ ps; t_{hold} = 60\ ps$

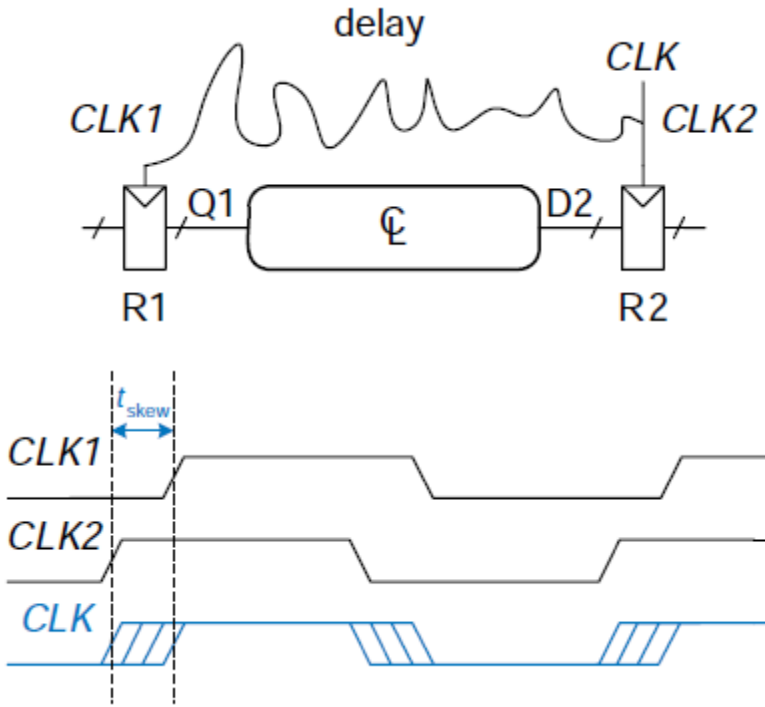
All gates → $t_{cd} = 25\ ps; t_{pd} = \ \ 40\ ps$

- Can this modified circuit fix hold time violation without breaking setup time constraints?
- Assume the buffers offer same delays as the other gates.

$$t_{ccq} + 2t_{cd} = 30 + 2 \times 25 = 80 \text{ ps}$$

Clock skew

- In high-speed circuits, the CLK signal does not reach all flip-flops precisely at the same instant
- Also, there may be parts of the circuit with “clock-gating” causing delayed clock signal
- This variation of clock edge across the circuit is called “clock skew”



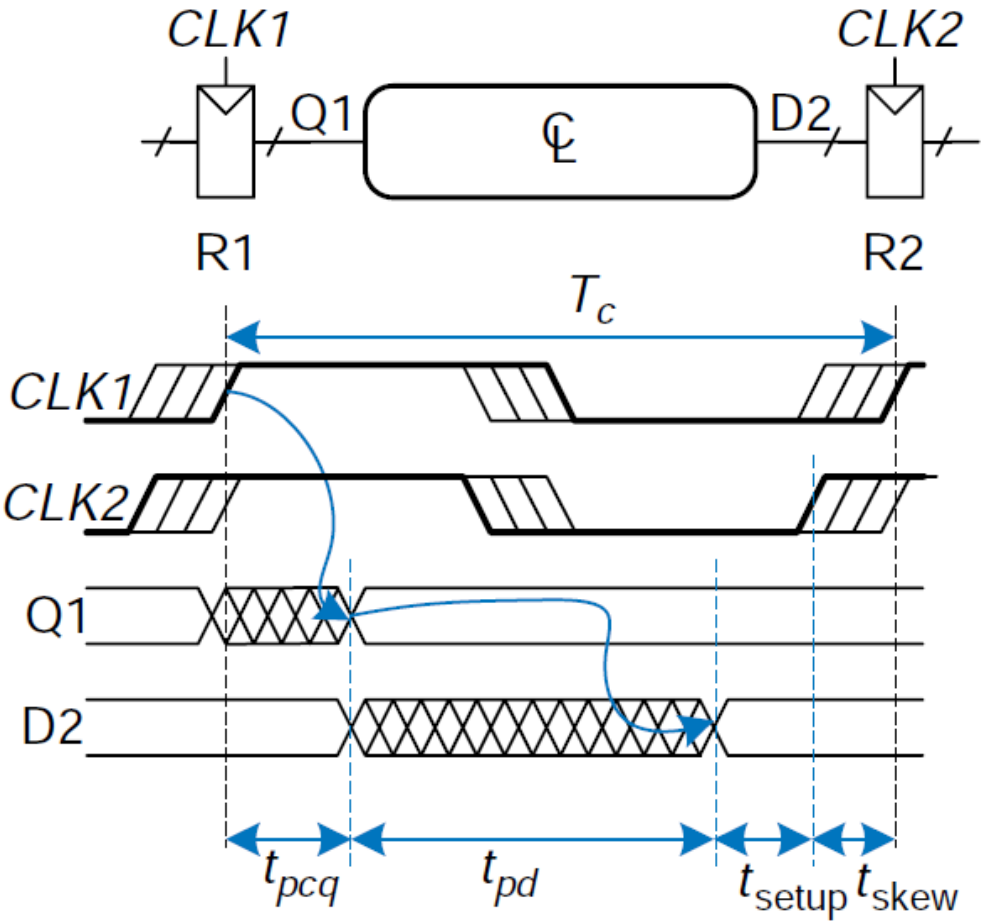
Implications of clock skew on setup time constraint

- Assume R1 gets delayed most clock, R2 gets earliest clock

- Then setup time constraint is modified as:

$$T_c \geq t_{pcq} + t_{pd} + t_{\text{setup}} + t_{\text{skew}}$$

$$t_{pd} \leq T_c - (t_{pcq} + t_{\text{setup}} + t_{\text{skew}})$$



Implications of clock skew on hold time constraint

- In this case, assume R1 gets earliest clock, R2 gets most delayed clock.

- Then the hold time constraint is modified as:

$$t_{ccq} + t_{cd} \geq t_{\text{hold}} + t_{\text{skew}}$$

$$t_{cd} \geq t_{\text{hold}} + t_{\text{skew}} - t_{ccq}$$