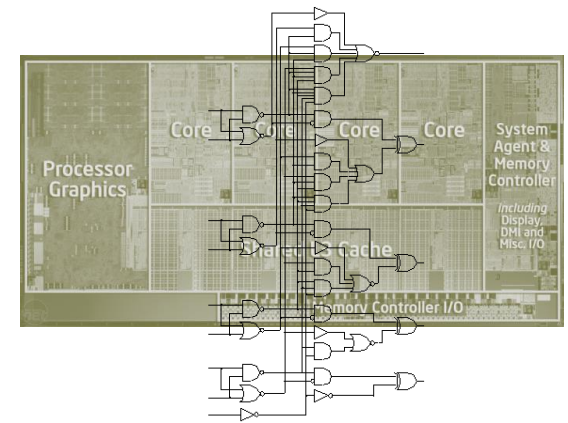




고려대학교
KOREA UNIVERSITY



COSE221, COMP211 Logic Design

Lecture 4. Sequential Logic #3

Prof. Taeweon Suh

Computer Science & Engineering

Korea University

What Determines Clock Speed?

- What is the operating clock frequency of your computer?
 - Why does the atom processor in your netbook run at 1.4GHz?
 - Why does the Core 2 Duo in your notebook run at 2.0GHz?
 - Why can't run at 100GHz or 1000GHz?
 - We are going to answer to this question today



아수스 X61GX-6X015A
인텔 / 코어2듀오(펜린) / T6400
(2.0GHz) / 16인치 / 1366x768 /
2.85Kg / Vista 홈베이직 / 2GB /
250GB / GeForce 9400M / 6cell

평균가 1,191,362원
최저가 1,171,000원



소니 바이오 VGN-G35LN/B
인텔 / 코어2듀오(펜린) / SU9400
(1.4GHz) / 12.1인치 / 1024x768 /
1.191Kg / Vista 비즈니스 / 2GB /
160GB / Intel GMA X4500

평균가 2,370,125원
최저가 2,339,000원



도시바 포테지 M800 PPM81K-
09E01D
인텔 / 펜티엄 듀얼코어 / T4200
(2.0GHz) / 13.3인치 / 1280x800 /
2.01Kg / Vista 홈프리미엄 / 1GB /
250GB / Intel GMA X4500 / 6cell

평균가 1,141,518원
최저가 1,080,000원

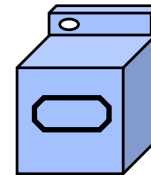
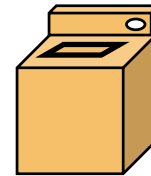
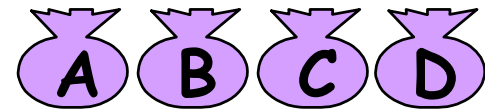


삼성전자 센스 NT-Q310-BA1PA
인텔 / 펜티엄 듀얼코어 / T3400
(2.16GHz) / 13.3인치 / 1280x800 /
2GB / 250GB / DVD레코더 / Intel
GMA X4500 / 6cell

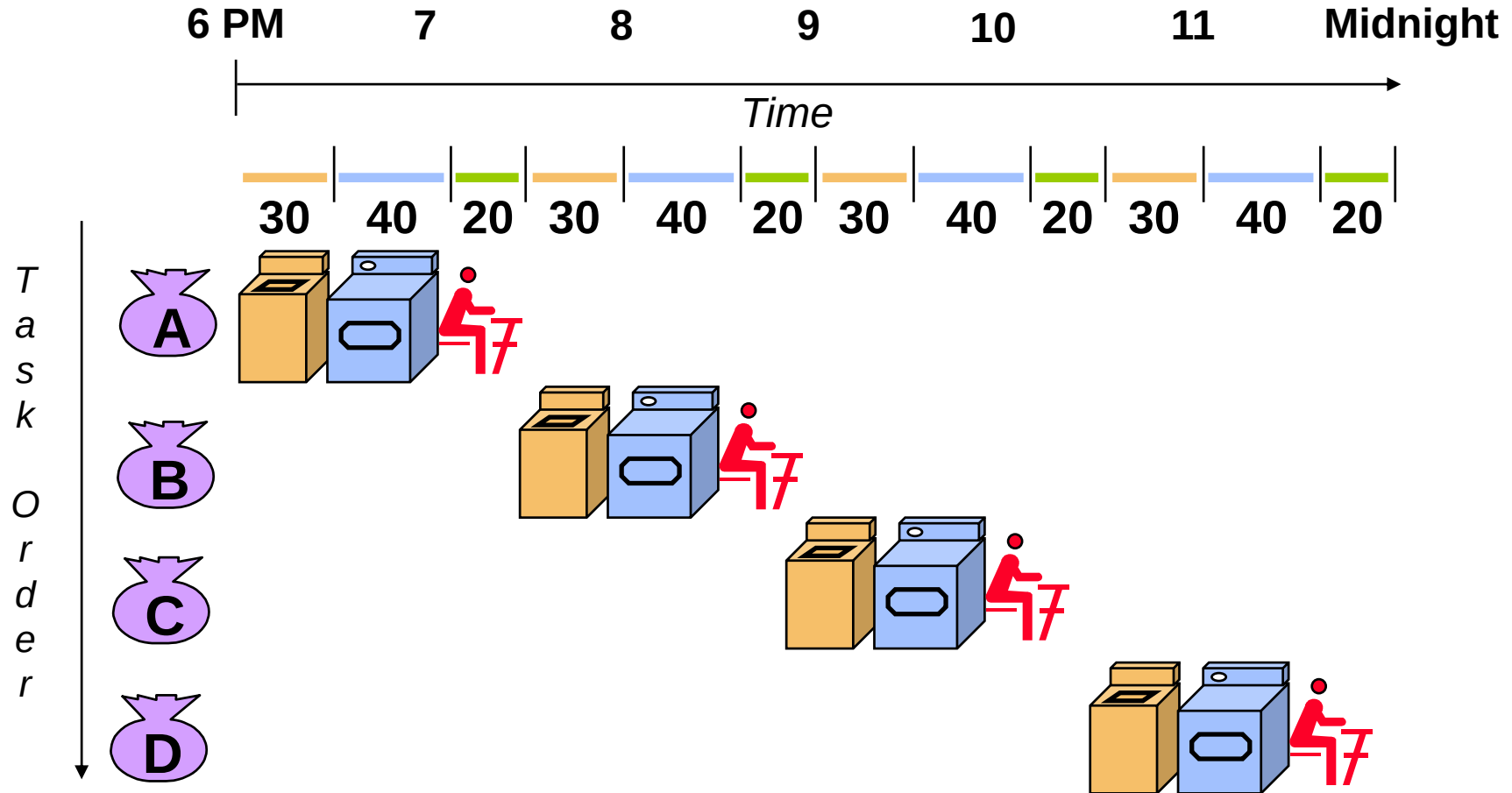
평균가 1,374,950원
최저가 1,247,000원

Response Time and Throughput

- Laundry Example
 - Ann, Brian, Cathy, Dave each have one load of clothes to wash, dry, and fold
 - **Washer** takes 30 minutes
 - **Dryer** takes 40 minutes
 - **Folder** takes 20 minutes



Sequential Laundry

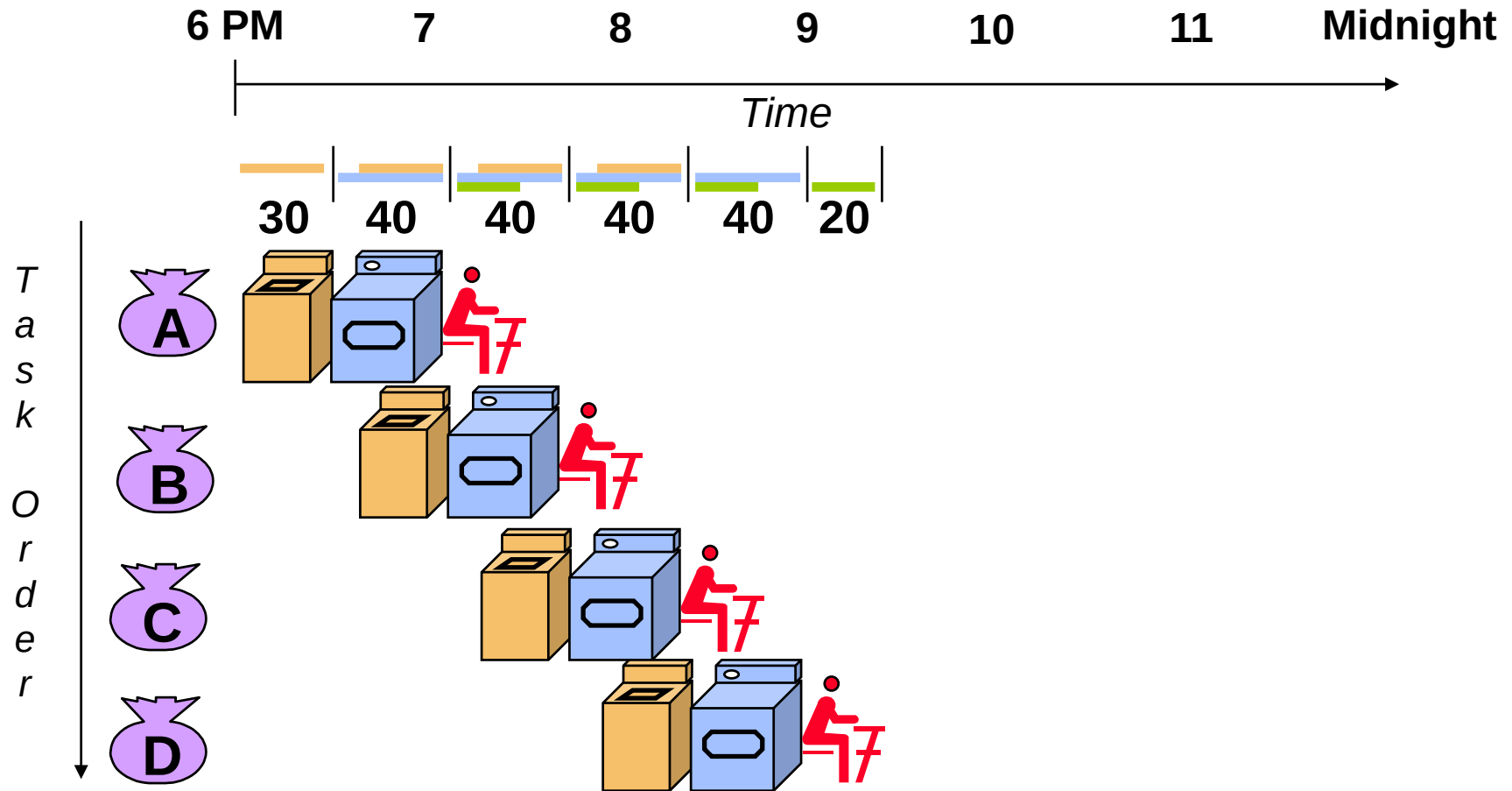


- Response time: 90 mins
- Throughput: 0.67 tasks / hr (= 90mins/task, 6 hours for 4 loads)

Response time (Execution time, Latency): Time between the start and the completion of a task

Throughput: Total number of tasks done in a given time

Pipelined Laundry



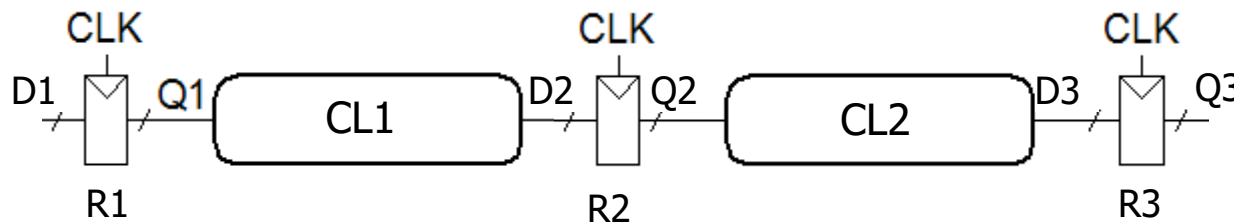
- Response time: 90 mins
- Throughput: 1.14 tasks / hr (= 52.5 mins/task, 3.5 hours for 4 loads)

Response time (Execution time, Latency): Time between the start and the completion of a task

Throughput: Total number of tasks done in a given time

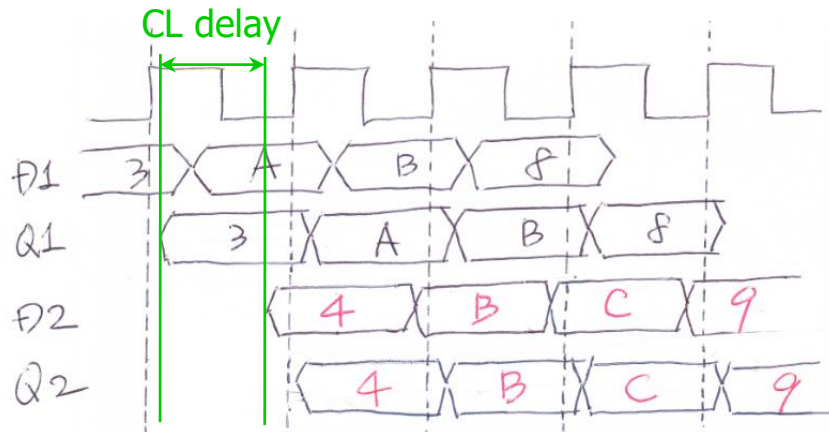
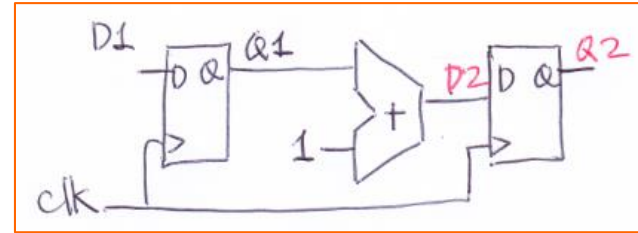
Synchronous Sequential Circuit

- As studied, the synchronous sequential circuits are composed of
 - Flip-flops
 - Combinational logic between flip-flops
- **Pipelining in CPU is also implemented this way**
 - We are going to talk deep about this in computer architecture course next semester



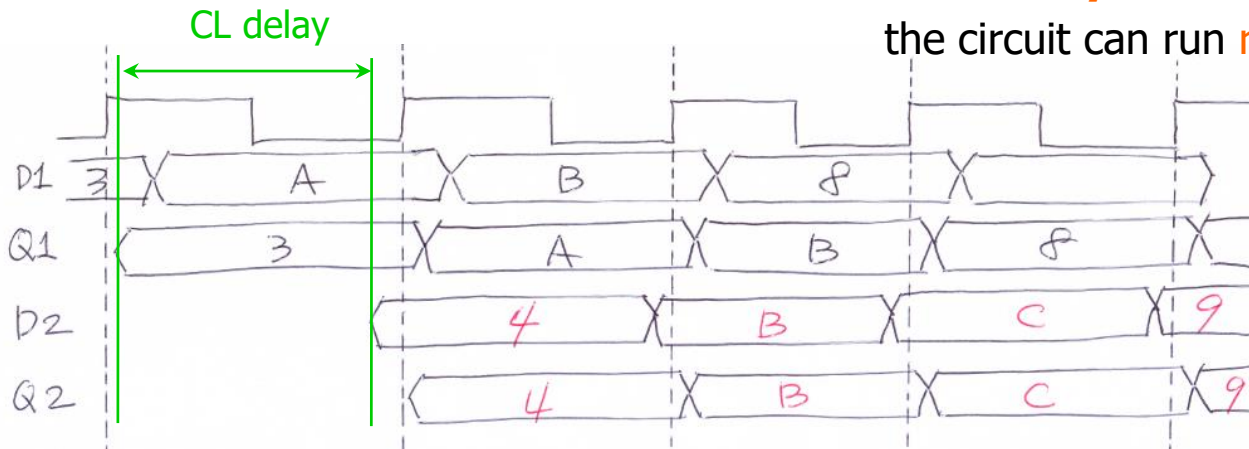
A Short Answer

- Suppose that the synchronous sequential circuit performs addition to the input
 - CL increments its input (Q1) by 1
 - So, we want to get $D1+1$ after **1 clock cycle**



If **CL delay $\cong 0.4$ ns**,
the circuit can run **roughly** at **2.5 GHz**

If **CL delay $\cong 0.8$ ns**,
the circuit can run **roughly** at **1.25 GHz**



A Long Answer

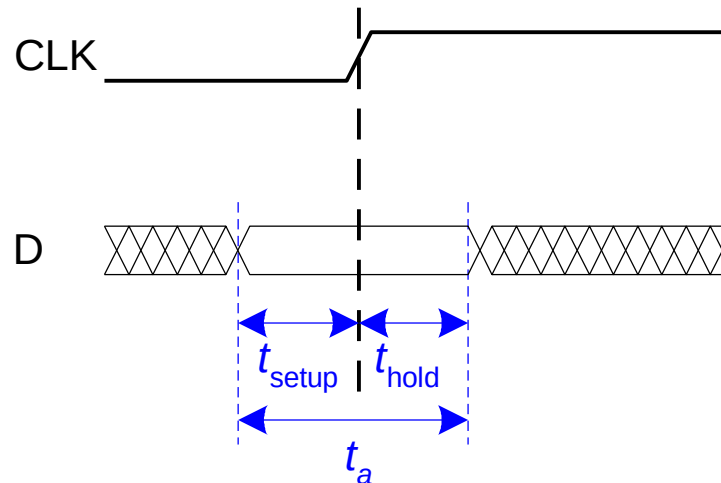
- Let's talk a little deep about what contributes to the delay
- Consequently what determines the clock frequency of the synchronous sequential circuit

Timing

- Flip-flop samples its input (D) at the rising (or falling) edge of the clock
 - **Input data in D must be stable** when it is sampled
 - Similar to a photograph, input data must be stable around the clock edge
 - If input data is changing when it is sampled, metastability can occur
 - For detailed explanation on metastability, read the section 3.5.4 in the textbook (page 143)

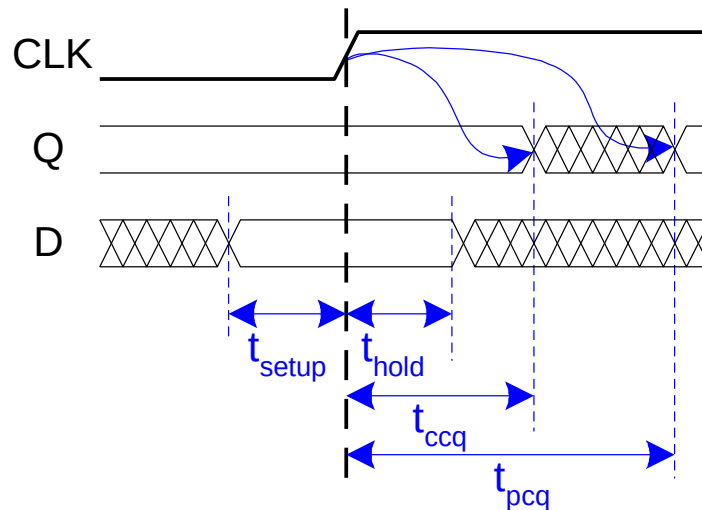
Input Timing Constraints

- Setup time
 - t_{setup} = time **before** the clock edge that data must be stable
- Hold time
 - t_{hold} = time **after** the clock edge that data must be stable
- Aperture time
 - t_a = time around clock edge that data must be stable ($t_{\text{setup}} + t_{\text{hold}}$)



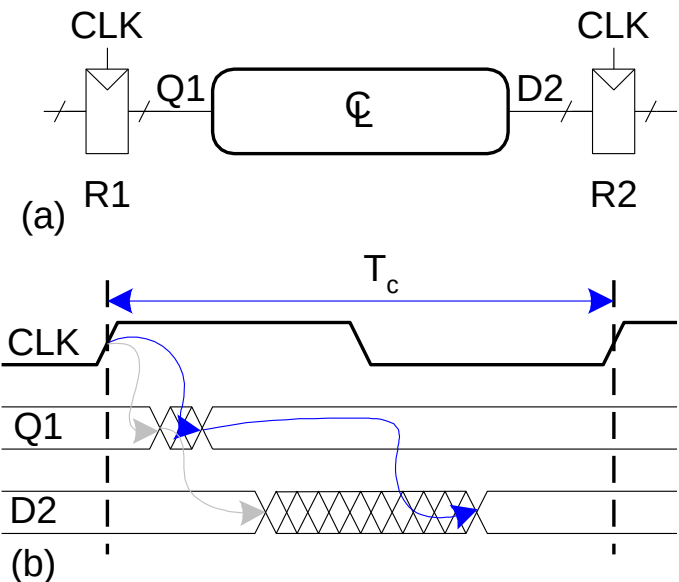
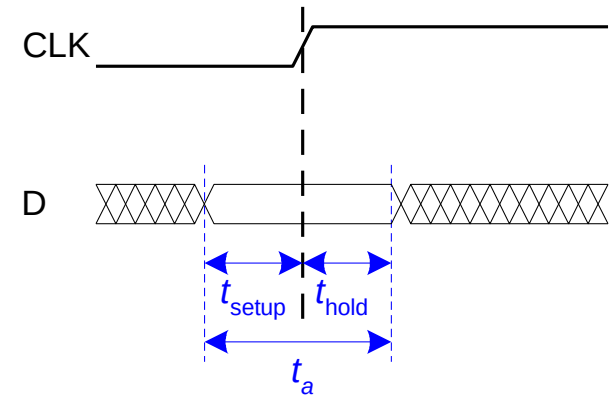
Output Timing of Flip-Flop

- Propagation delay
 - t_{pcq} = time after clock edge that the output Q is guaranteed to be stable (i.e., to stop changing)
- Contamination delay
 - t_{ccq} = time after clock edge that Q might be unstable (i.e., **start changing**)
- Output timing is determined depending on the flip-flop implementation



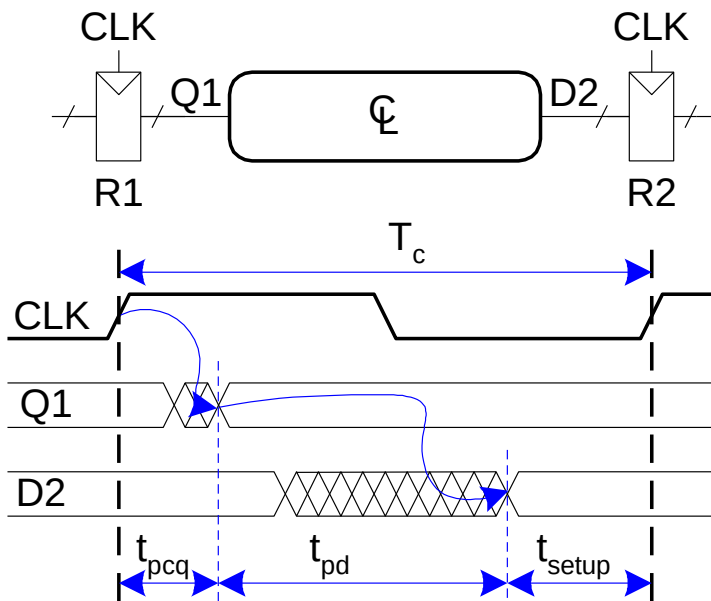
Dynamic Discipline

- The input to a synchronous sequential circuit must be stable during the aperture (setup and hold) time around the clock edge
- The input must be stable
 - at least t_{setup} before the clock edge
 - at least until t_{hold} after the clock edge
- The delay between registers has a **minimum** and **maximum** delay, dependent on the delays of the circuit elements



Setup Time Constraint

- The setup time constraint depends on the **maximum** delay from register R1 through the combinational logic
- The input to register R2 must be stable at least t_{setup} before the clock edge

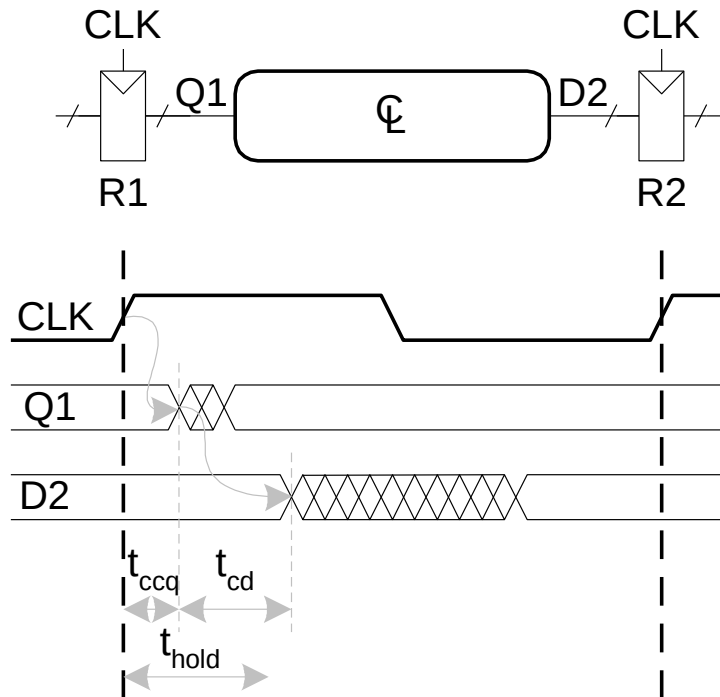


$$T_c \geq t_{pcq} + t_{pd} + t_{\text{setup}}$$

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

Hold Time Constraint

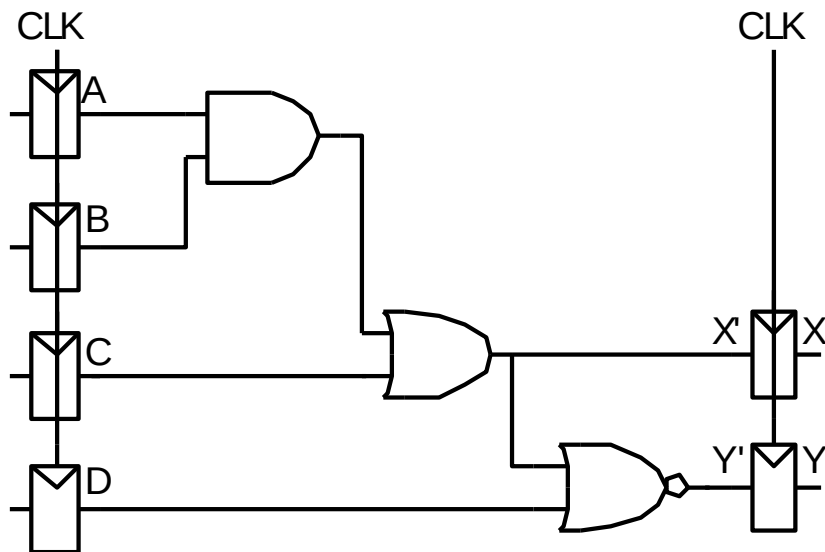
- The hold time constraint depends on the **minimum** delay from register R1 through the combinational logic
- The input to register R2 must be stable for at least t_{hold} after the clock edge



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

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

Timing Analysis Example



Timing Characteristics

$$t_{ccq} = 30 \text{ ps}$$

$$t_{pcq} = 50 \text{ ps}$$

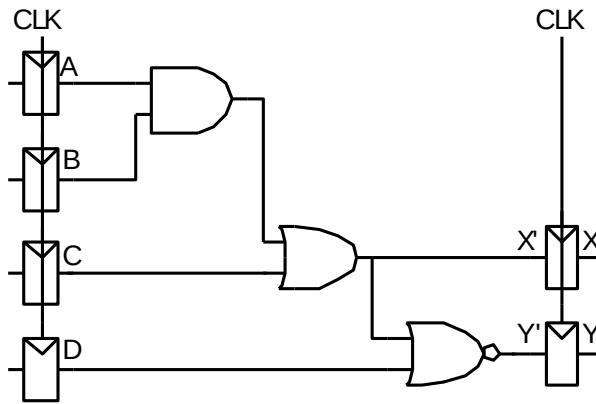
$$t_{\text{setup}} = 60 \text{ ps}$$

$$t_{\text{hold}} = 70 \text{ ps}$$

$$t_{pd} = 35 \text{ ps}$$

$$t_{cd} = 25 \text{ ps}$$

Timing Analysis Example



Timing Characteristics

$$t_{ccq} = 30 \text{ ps}$$

$$t_{pcq} = 50 \text{ ps}$$

$$t_{\text{setup}} = 60 \text{ ps}$$

$$t_{\text{hold}} = 70 \text{ ps}$$

per gate

$$t_{pd} = 35 \text{ ps}$$

$$t_{cd} = 25 \text{ ps}$$

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

Setup time constraint:

$$t_{pd} = 3 \times 35 \text{ ps} = 105 \text{ ps}$$

$$T_c \geq (50 + 105 + 60) \text{ ps} = 215 \text{ ps}$$

$$f_c = 1/T_c = 4.65 \text{ GHz}$$

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

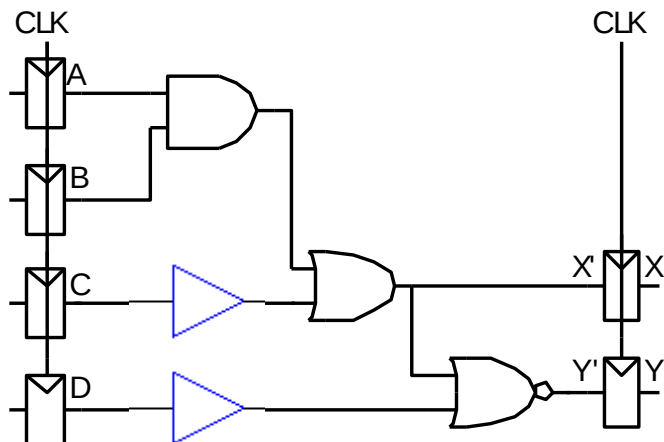
Hold time constraint:

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

$$(30 + 25) \text{ ps} \geq 70 \text{ ps} ? \text{ No!}$$

Fixing Hold Time Violation

Add buffers to the short paths:



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

Setup time constraint:

$$t_{pd} = 3 \times 35 \text{ ps} = 105 \text{ ps}$$

$$T_c \geq (50 + 105 + 60) \text{ ps} = 215 \text{ ps}$$

$$f_c = 1/T_c = 4.65 \text{ GHz}$$

Timing Characteristics

$$t_{ccq} = 30 \text{ ps}$$

$$t_{pcq} = 50 \text{ ps}$$

$$t_{setup} = 60 \text{ ps}$$

$$t_{hold} = 70 \text{ ps}$$

per gate

$$\left[\begin{array}{l} t_{pd} = 35 \text{ ps} \\ t_{cd} = 25 \text{ ps} \end{array} \right]$$

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

Hold time constraint:

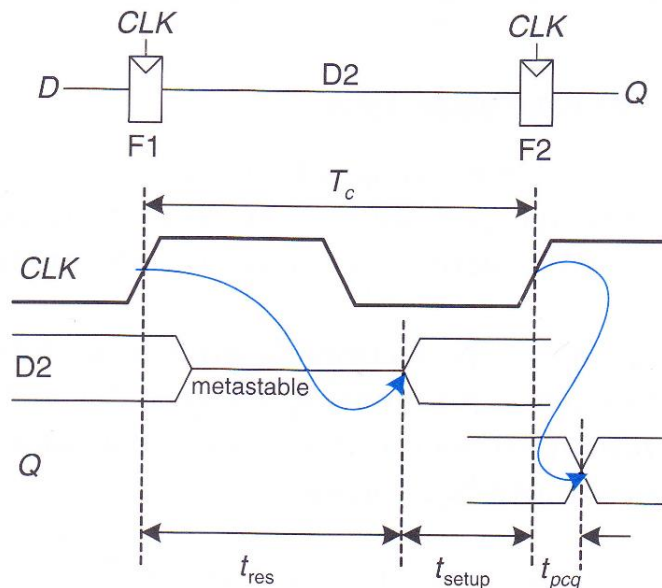
$$t_{ccq} + t_{cd} \geq t_{hold} ?$$

$$(30 + 50) \text{ ps} \geq 70 \text{ ps} ? \text{ Yes!}$$

Backup Slides

Simple Synchronizer

- Every bistable device has a metastable state between the 2 stable states (0 and 1)
- If the input happens to change within the aperture, the resolution time, t_{res} , can be substantially longer
 - Theoretical and experimental analyses have shown that the probability that t_{res} exceeds some arbitrary time t , decreases exponentially with t



$$P(t_{res} > t) = \frac{T_0}{T_c} e^{-\frac{t}{\tau}} \quad (3.23)$$

where T_c is the clock period, and T_0 and τ are characteristic of the flip-flop. The equation is valid only for t substantially longer than t_{pcq} .

Figure 3.51 Simple synchronizer