Outline for Today's Lecture

• Sequential Design Timing
• Metastability
• Synchronizer
Timing

- Flip-flop samples $D$ at clock edge
- $D$ must be stable when sampled
- Similar to a photograph, $D$ must be stable around clock edge
- If not, metastability can occur
Input Timing Constraints

- **Setup time:** $t_{\text{setup}} = \text{time before clock edge data must be stable (i.e. not changing)}$
- **Hold time:** $t_{\text{hold}} = \text{time after clock edge data must be stable}$
- **Aperture time:** $t_a = \text{time around clock edge data must be stable} \quad (t_a = t_{\text{setup}} + t_{\text{hold}})$
Output Timing Constraints

- **Propagation delay**: $t_{pcq} = \text{time after clock edge that the output } Q \text{ is guaranteed to be stable (i.e., to stop changing)}$
- **Contamination delay**: $t_{ccq} = \text{time after clock edge that } Q \text{ might be unstable (i.e., start changing)}$
Dynamic Discipline

- Synchronous sequential circuit inputs must be stable during aperture (setup and hold) time around clock edge
  - Specifically, inputs must be stable
    - at least $t_{\text{setup}}$ before the clock edge
    - at least until $t_{\text{hold}}$ after the clock edge
Dynamic Discipline

• The delay between registers has a **minimum** and **maximum** delay, dependent on the delays of the circuit elements

![Diagram of circuit elements](a)

![Waveforms](b)
Setup Time Constraint

- Depends on the maximum delay from register R1 through combinational logic to R2
- The input to register R2 must be stable at least $t_{\text{setup}}$ before clock edge

\[ T_c \geq \]
Setup Time Constraint

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

\[
T_c \geq t_{pcq} + t_{pd} + t_{\text{setup}}
\]

\[
t_{pd} \leq \]

\[\text{ENGR 303} \]
Setup Time Constraint

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

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

$(t_{pcq} + t_{\text{setup}})$: sequencing overhead
Hold Time Constraint

- Depends on the **minimum** delay from register R1 through the combinational logic to R2
- The input to register R2 must be stable for at least $t_{\text{hold}}$ after the clock edge
Hold Time Constraint

- Depends on the **minimum** delay from register R1 through the combinational logic to R2
- The input to register R2 must be stable for at least \( t_{\text{hold}} \) after the clock edge

\[
t_{\text{hold}} < t_{\text{ccq}} + t_{cd}
\]

\( t_{cd} > \)
Hold Time Constraint

- Depends on the **minimum** delay from register R1 through the combinational logic to R2.
- The input to register R2 must be stable for at least $t_{\text{hold}}$ after the clock edge.

\[ t_{\text{hold}} < t_{\text{ccq}} + t_{\text{cd}} \]
\[ t_{\text{cd}} > t_{\text{hold}} - t_{\text{ccq}} \]

ENGR 303
Timing Analysis

Timing Characteristics

- $t_{ccq} = 30$ ps
- $t_{pcq} = 50$ ps
- $t_{setup} = 60$ ps
- $t_{hold} = 70$ ps
- $t_{pd} = 35$ ps
- $t_{cd} = 25$ ps

Setup time constraint:

$T_c \geq \quad \text{Hold time constraint:}$

$t_{ccq} + t_{cd} > t_{hold}$?
Timing Analysis

Timing Characteristics

\[ t_{ccq} = 30 \text{ ps} \]
\[ t_{pcq} = 50 \text{ ps} \]
\[ t_{setup} = 60 \text{ ps} \]
\[ t_{hold} = 70 \text{ ps} \]

\[ t_{pd} = 35 \text{ ps} \]
\[ t_{cd} = 25 \text{ ps} \]

Setup time constraint:
\[ T_{c} \geq (50 + 105 + 60) \text{ ps} = 215 \text{ ps} \]
\[ f_{c} = 1/T_{c} = 4.65 \text{ GHz} \]

Hold time constraint:
\[ t_{ccq} + t_{cd} > t_{hold} ? \]
\[ (30 + 25) \text{ ps} > 70 \text{ ps} ? \text{ No!} \]
Timing Analysis

Add buffers to the short paths:

CLK
- A
- B
- C
- D

CLK

CLK

X'

Y'

X

Y

X

Y

Add buffers to the short paths:

CLK
- A
- B
- C
- D

CLK

CLK

X'

Y'

X

Y

X

Y

Timing Characteristics

\[ t_{ccq} = 30 \text{ ps} \]
\[ t_{pcq} = 50 \text{ ps} \]
\[ t_{setup} = 60 \text{ ps} \]
\[ t_{hold} = 70 \text{ ps} \]
\[ t_{pd} = 35 \text{ ps} \]
\[ t_{cd} = 25 \text{ ps} \]

Setup time constraint:

\[ T_c \geq \]

Hold time constraint:

\[ t_{ccq} + t_{cd} > t_{hold} ? \]
Timing Analysis

Add buffers to the short paths:

Timing Characteristics

- $t_{ccq} = 30$ ps
- $t_{pcq} = 50$ ps
- $t_{setup} = 60$ ps
- $t_{hold} = 70$ ps
- $t_{pd} = 35$ ps
- $t_{cd} = 25$ ps

Setup time constraint:

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

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

Hold time constraint:

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

$$ (30 + 50) \text{ ps} > 70 \text{ ps} \ ? \ \text{Yes!} $$
Clock Skew

- The clock doesn’t arrive at all registers at the same time
- **Skew**: difference between two clock edges
- Perform **worst case analysis** to guarantee dynamic discipline is not violated for any register – many registers in a system!
Setup Time Constraint with Skew

- In the worst case, CLK2 is earlier than CLK1
Setup Time Constraint with Skew

- In the worst case, CLK2 is earlier than CLK1

\[ T_c \geq t_{pcq} + t_{pd} + t_{setup} + t_{skew} \]

\[ t_{pd} \leq \]
Setup Time Constraint with Skew

- In the worst case, CLK2 is earlier than CLK1

\[ T_c \geq t_{pcq} + t_{pd} + t_{setup} + t_{skew} \]
\[ t_{pd} \leq T_c - (t_{pcq} + t_{setup} + t_{skew}) \]
Hold Time Constraint with Skew

• In the worst case, CLK2 is later than CLK1

$t_{ccq} + t_{cd} >$
Hold Time Constraint with Skew

- In the worst case, CLK2 is later than CLK1

$t_{ccq} + t_{cd} > t_{hold} + t_{skew}$

$t_{cd} >$
Hold Time Constraint with Skew

- In the worst case, CLK2 is later than CLK1

\[ t_{ccq} + t_{cd} > t_{hold} + t_{skew} \]

\[ t_{cd} > t_{hold} + t_{skew} - t_{ccq} \]
Asynchronous (for example, user) inputs might violate the dynamic discipline.

Violating the Dynamic Discipline

- Case I
- Case II
- Case III
Metastability

- **Bistable devices**: two stable states, and a metastable state between them
- **Flip-flop**: two stable states (1 and 0) and one metastable state
- If flip-flop lands in metastable state, could stay there for an undetermined amount of time
Synchronizers

- **Asynchronous inputs are inevitable** (user interfaces, systems with different clocks interacting, etc.)
- **Synchronizer goal**: make the probability of failure (the output $Q$ still being metastable) low
- Synchronizer cannot make the probability of failure 0