CSE 140: Components and Design Techniques for Digital Systems

Lecture 10:
Sequential Networks: Timing and Retiming

CK Cheng
Dept. of Computer Science and Engineering
University of California, San Diego
Timing

- Motivation
- Gate Delay
- Flip-Flop Timing Window
- Two Timing Constraints: shortest and longest timing paths
- Examples
Timing: Motivation

• Clock specifies a precise time for the next state
  – In general, we allocate one clock period for signal propagation between registers. **Goldilocks** timing.

• Too late: Fail to reach for the setup of the next state.

• Too early: Race to disturb the holding of the next state.

• Analysis: Verify the timing of the system.

• Goal: A robust design.
Once upon a time, there was a little girl named Goldilocks. She went for a walk in the forest. Pretty soon, she came upon a house. She knocked and, when no one answered, she walked right in. At the table in the kitchen, there were three bowls of porridge. Goldilocks was hungry. She tasted the porridge from the first bowl. "This porridge is too hot!" she exclaimed. So, she tasted the porridge from the second bowl. "This porridge is too cold," she said. So, she tasted the last bowl of porridge. "Ahhh, this porridge is just right," she said happily and she ate it all up.
Motivation: So far …. 

Combinational

Logic-level analysis
Motivation: This lecture …

- When does our (seemingly logically correct) design go wrong?
- How can we design a circuit that works under real constraints?
- Popular interview question.
A typical sequential network has combinational circuit between registers (R1 to R2).
The registers are synchronized by clocks (CLK1 to CLK2).
Timing is set between clocks (CLK1 and CLK2).
The beauty of the synchronized design is that we need only to take care of the timing of the regions separated by the registers.
For a synchronized digital Moore machine, we need to take care of the timing of the following region(s).

- Between every pair of registers.
- Between i. input and register, and ii. register and output.
Gate Delay: Combinational Logic Timing

I. **Min delay of a gate, also called Contamination delay**: $t_{cd}$
   Minimum time from when an input changes until the output *starts* to change

II. **Max delay of a gate, also called Propagation delay**: $t_{pd}$
    Maximum time from when an input changes until the output *is* guaranteed to reach its final value (i.e., stop changing)
Gate Delay: Combinational Logic Timing

Different input transition causes different delay at output

<table>
<thead>
<tr>
<th>$A_tB_t$</th>
<th>$A_{t+1}B_{t+1}$</th>
<th>$Y$</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>11</td>
<td>1/0</td>
</tr>
<tr>
<td>01</td>
<td>11</td>
<td>1/0</td>
</tr>
<tr>
<td>10</td>
<td>11</td>
<td>1/0</td>
</tr>
<tr>
<td>11</td>
<td>11</td>
<td>1/1</td>
</tr>
</tbody>
</table>
Different path causes different output transition delay.
Speed of light: $\frac{C}{\sqrt{\varepsilon}} \approx 1.5 \times 10^{10} \text{ cm/s}$

For 1 cm, it takes $0.7 \times 10^{-10} \text{ s} = 70\text{ps (picosecond)}$ for the light to reach from one end to the other end.

Chain of buffers: 5-40 times of speed of light.

For 5GHz/s, the clock period is $200\text{ps/cycle}$. 
Combinational Logic: Output timing constraints

I. Contamination delay (shortest): $t_{cd}$
   Minimum time from when an input changes until any output *starts* to change

II. Propagation delay (longest): $t_{pd}$
   Maximum time from when an input changes until the output *or* outputs of a combinational circuit are guaranteed to reach their final value (i.e., stop changing)
Once a flip flop has been ‘built’ we are stuck with its timing characteristics:

- $t_{\text{setup}}$, $t_{\text{hold}}$ timing relation between D and CLK
- $t_{\text{ccq}}$, $t_{\text{pcq}}$ timing relation between CLK and Q

No direct timing relation between input D and output Q
**FF Input Constraints: Set up and hold time**

Setup time $t_{\text{setup}}$ Time *before* the clock edge that data must be stable (i.e. not change)

Setup time violation This occurs if the input signal $D$ does not settle (set up) to the stable value at least $t_{\text{setup}}$ before the clock edge.

Hold time $t_{\text{hold}}$ Time *after* the clock edge that data must be stable

 Hold time violation This occurs if the input signal $D$ does not remain unchanged (hold) for at least $t_{\text{hold}}$ after the clock edge.
FF 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: $t_{ccq} = \text{time after clock edge that } Q \text{ might be unstable (i.e., start changing)}$
Two Timing Constraints

\[ t_{cq} + t_{comb} + t_{setup} \leq T \]
\[ t_{hold} < t_{cq} + t_{comb} \]

- \( t_{cq} \): time from rising edge of clock to Q update (CLK1 \( \Rightarrow \) B)
- \( t_{comb} \): time of combinational logic delay (B \( \Rightarrow \) C)
- \( t_{setup} \): setup time before rising edge of clock (C \( \Rightarrow \) CLK2)
- \( t_{hold} \): hold time after the rising edge of clock
- \( T \): clock period (CLK1 \( \Rightarrow \) CLK2)
Two Timing Constraints

Setup time constraint
\[ t_{cq} + t_{comb} + t_{setup} \leq T \]

Longest delay from CLK1 to CLK2
\[ max(t_{cq} + t_{comb} + t_{setup}) \leq T \]

Hold time constraint
\[ t_{hold} < t_{cq} + t_{comb} \]

Shortest delay from CLK1 to CLK2
\[ t_{hold} < min(t_{cq} + t_{comb}) \]
Two Timing Constraints

$ t_{cq} + t_{comb} + t_{setup} \leq T $

$ t_{hold} < t_{cq} + t_{comb} $

Too long

Just right

Too short

CLK1

CLK2
PIQ: The timing of which of the following signals can cause a setup-time violation?

A. Signal D arrives too early
B. Signal D arrives too late
C. Clock CLK arrives too late
D. Output Q(t) responds too early
E. None of the above
PIQ: A hold time violation is likely to occur when

A. Signal D changes too early
B. Signal D changes too late
C. Clock CLK arrives too early
D. None of the above
PIQ: A hold time violation is likely to occur when

A. Signal D changes too late
B. Clock CLK arrives too early
C. Clock CLK arrives too late
D. None of the above
An alternate view of the sequential circuit

Combinational

CLK

D1

R1

CLK

Q1

Combinational

D2

R2

CLK
What should happen within a clock cycle for correct functionality?

R1

CLK

D1 Q1

Combinational

D2

R2

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

![Diagram showing the delay between registers](image)

- **CLK**
- **Q1**
- **D2**
- **R1**
- **R2**

Diagram (a) and (b) illustrate the timing and propagation of signals through the circuit elements.
The delay between registers has a **minimum** and **maximum** delay, dependent on the delays of the circuit elements.
PI Q: Suppose CLK rises at $t_1$, what is the maximum delay (from $t_1$) after which $D_2$ reaches a stable value?

A. Setup time of $R_1$+ Propagation delay of $CL$ + Propagation delay of $R_2$

B. Hold time of $R_1$+ Propagation delay of $CL$ + setup time of $R_1$

C. Propagation delay of $R_1$+ Propagation delay of $CL$ + Propagation delay of $R_2$

D. Propagation delay of $R_1$+ Propagation delay of $CL$

E. Propagation delay of $CL$ + Propagation delay of $R_2$
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 $t_{\text{setup}}$ before the clock edge.

Maximum delay, $t_{\text{max}}$ = 

Setup Time Constraint:
Setup Time Constraint

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

PI Q: As a designer, which of the following parameters would you modify to meet the set up time constraint?

A. The clock period, \( T_c \)
B. The prop. delay of R1, \( t_{pcq} \)
C. The prop. delay of CL, \( t_{pd} \)
D. The setup time of R2, \( t_{setup} \)
E. All of the above
Setup Time Constraint

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

PI Q: As a designer, which of the following parameters would you modify to meet the set up time constraint?

A. The clock period, \( T_c \)
B. The prop. delay of R1, \( t_{pcq} \)
C. The prop. delay of CL, \( t_{pd} \)
D. The setup time of R2, \( t_{\text{setup}} \)
E. All of the above
PI Q: Suppose CLK rises at $t_1$, what is the minimum delay (from $t_1$) after which D2 starts to change?

A. Setup time of R1+ propagation delay of CL + propagation of R2
B. Hold time of R1+ propagation time of CL + setup time of R1
C. Hold time of R1+ Contamination delay of CL + Propagation time of R2
D. Contamination delay of R1+ Contamination delay of CL
E. Contamination delay of CL + Contamination delay of R2
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 $t_{\text{hold}}$ after the clock edge.

Minimum delay, $t_{\text{min}}$

Hold Time Constraint:
Hold Time Constraint

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

\[ t_{cd} > t_{\text{hold}} - t_{ccq} \]
Timing Analysis: Example

Timing Characteristics

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

Gates
\[ t_{pd}(g) = 35 \, \text{ps} \]
\[ t_{cd}(g) = 25 \, \text{ps} \]

Setup time constraint:
\[ T_c \geq f_c = 1/T_c = \]

Hold time constraint:
\[ t_{ccq} + t_{pd} > t_{\text{hold}} ? \]
Timing Analysis: Example

FFs
- $t_{ccq} = 30 \text{ ps}$
- $t_{pcq} = 50 \text{ ps}$
- $t_{setup} = 60 \text{ ps}$
- $t_{hold} = 70 \text{ ps}$

Gates
- $t_{pd}(g) = 35 \text{ ps}$
- $t_{cd}(g) = 25 \text{ ps}$

Setup time constraint:
- $T \geq t_{pcq} + t_{pd}(\text{com}) + t_{setup}$
- $= 50 + 105 + 60 = 215 \text{ ps}$
- $f_c = 1/T_c = 4.65 \text{ GHz}$

Hold time constraint:
- $t_{ccq} + t_{cd}(\text{com}) > t_{hold}$?
- $(30 + 25) \text{ ps} > 70 \text{ ps}$?
- No!
Example: Fix Hold Time Violation

Add buffers to the short paths:

Timing Characteristics

*FFs*
- \( t_{ccq} = 30 \text{ ps} \)
- \( t_{pcq} = 50 \text{ ps} \)
- \( t_{setup} = 60 \text{ ps} \)
- \( t_{hold} = 70 \text{ ps} \)

*Gates*
- \( t_{pd}(g) = 35 \text{ ps} \)
- \( t_{cd}(g) = 25 \text{ ps} \)

Setup time constraint:

\[ T \geq \]

Hold time constraint:

\[ t_{ccq} + t_{pd} > t_{hold} ? \]
Example: Fix Hold Time Violation

Add buffers to the short paths:

- **FFs**
  - $t_{ccq} = 30$ ps
  - $t_{pcq} = 50$ ps
- **Gates**
  - $t_{pd}(g) = 35$ ps
  - $t_{cd}(g) = 25$ ps

Setup time constraint:

$T = 50 + 105 + 60 = 215$ ps

$f_c = 1/T = 4.65$ GHz

Hold time constraint:

$t_{ccq} + t_{cd}(com) > t_{hold}$ ?

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

The clock doesn’t arrive at all registers at the same time. The difference between two clock edges is skew.

- **Skew as Noise**: Caused by process variation, voltage fluctuation, crosstalks (PVC). Examine the worst case to guarantee that the timing is right.
- **Designated Skew**: Make skew by design to improve the performance.
Time Constraint with Clock Skew (Noise)

In the worst case, the CLK2 is:

- Earlier than CLK1 for setup time
- Later than CLK1 for hold time.

\[ T_c \geq t_{pcq} + t_{pd(\text{com})} + t_{\text{setup}} + t_{\text{skew}} \]

\[ t_{ccq} + t_{cd(\text{com})} > t_{\text{hold}} + t_{\text{skew}} \]
Timing Analysis with Clock Skew: Example

Timing Characteristics

\[ t_{ccq} = 30 \text{ ps} \]

\[ t_{pcq} = 50 \text{ ps} \]

\[ t_{\text{setup}} = 60 \text{ ps} \]

\[ t_{\text{hold}} = 70 \text{ ps} \]

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

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

\[ t_{\text{skew}} = 50 \text{ ps} \]

**Setup time constraint:**

\[ T_c \geq 265 \text{ ps} \]

\[ f_c = \frac{1}{T_c} = 3.77 \text{ GHz} \]

Without skew we got \( f_c = 4.65 \text{ GHz} \)
Time Constraint with Clock Skew: Example

- In the worst case for setup time, CLK2 is later than CLK1

\[ t_{ccq} + t_{cd}(\text{com}) > t_{hold} + t_{skew} \]
Clock Skew: Example

Add buffers to the short paths:

![Circuit Diagram]

Timing Characteristics

- \( t_{ccq} = 30 \text{ ps} \)
- \( t_{pcq} = 50 \text{ ps} \)
- \( t_{setup} = 60 \text{ ps} \)
- \( t_{hold} = 70 \text{ ps} \)
- \( t_{pd}(g) = 35 \text{ ps} \)
- \( t_{cd}(g) = 25 \text{ ps} \)
- \( t_{skew} = 50 \text{ ps} \)

Hold time constraint:

\[
\begin{align*}
\text{Hold time constraint:} & \\
\Rightarrow \quad t_{ccq} + t_{cd}(\text{com}) & > t_{hold} + t_{skew} \\
(30 + 50) & > (70 + 50) \text{ ps ?}
\end{align*}
\]
Retiming with **Designated Skew**

- CLK2 is later than CLK1 by $t_{skew}$

\[
\begin{align*}
T & \geq t_{pcq} + t_{pd}\text{(com)} + t_{setup} + t_{skew} \\
t_{ccq} + t_{cd}\text{(com)} & > t_{hold} + t_{skew}
\end{align*}
\]

**Skew as noise (worst case)**

**Designated skew**

\[
\begin{align*}
T & \geq t_{pcq} + t_{pd}\text{(com)} + t_{setup} - t_{skew} \\
t_{ccq} + t_{cd}\text{(com)} & > t_{hold} + t_{skew}
\end{align*}
\]
Retiming: Example

\[ T_c \geq t_{pcq} + t_{pd}(\text{com}) + t_{\text{setup}} - t_{\text{skew}} \]
\[ t_{ccq} + t_{cd}(\text{com}) \geq t_{\text{hold}} + t_{\text{skew}} \]
\[ T_c \geq 50 + 105 + 60 - t_{\text{skew}} \]
\[ 30 + 50 \geq 70 + t_{\text{skew}} \]

iClicker: The minimum clock period T can be:
A. 195
B. 205
C. 215
D. None of the above
Timing and Retiming

• Retiming: Adjust the clock skew so that the clock period can be reduced.
• Add a few more examples on timing and retiming.
Conclusion

• Clock to Clock: Range of shortest and longest paths
• Design revision and retiming to adjust the constraints
• Research: Variation aware designs

Extra materials: