Timing Design in Digital Systems

Dr. Paul D. Franzon

Outline
1. Timing design in Synchronous (clocked) Logic
   - Min/Max timing with flip-flops
   - Latch-based design
3. Timing Issues in CMOS circuits
4. Timing verification and SDF

References
2. Smith, Sections 2.4, 13.6, 13.7, 16.4.1

Attachments: Need sample library sheets.
**General Approach to Timing Design**

- In general, all signals start and end in registers every clock period

![D-Flip-flops and Combinational Logic Diagram]

- In general, there is only one clock (in any one sub-block of the chip) and each flip-flop is clocked every cycle
  - If a chip needs multiple clocks, they should share a common root clock (e.g. divide a master clock) and be fed to different sub-designs within the chip
Clock Level Timing

- Example (Revision):

  ![Diagram of clock level timing]

  - Clock
  - In
  - Out

©2004 Dr. Paul D. Franzon, www.ece.ncsu.edu/erl/faculty/paulf.html
**Critical Path**

- Thus, the clock speed is determined by the slowest feasible path between registers in the design
  - *Often referred to as “the critical path”*

Critical path is longer with increased \textit{logic depth} (# gates in series)

\textbf{FIGURE 11-1: Critical Path}
Clock Distribution

- The goal of clock tree is for the clock to arrive at every leaf node at the same time:

- Usually designed after synthesis: Matched buffers; matched capacitance loads
- Some clock skew (difference in arrival times) is unavoidable
  - Clock skew specification is an important parameter
  - Clock tree carefully designed to meet a skew requirement
- Clock jitter (cycle to cycle skew variations) is also important but is treated as being lumped in with the fixed clock skew to make a total skew (+jitter) budget
  - Comes from phase noise in Phase Lock Loop, etc.
Flip-Flop based design

**Edge triggered D-flip-flop**
Q becomes D after clock edge

**Set-up time:**
Data can not change no later than this point before the clock edge.

**Hold time:**
Data can not change during this time after the clock edge.

\[ t_{\text{clock-Q}} \]
Delay on output (Q) changing from positive clock edge
Preventing Set-Up Violations

Set-up violation:
Logic is too slow for the correct logic value to arrive at the inputs to the register on the right before one set-up time before the clock edge

Constraint to prevent this:

\[ t_{clock} > t_{clock-Q-max} + t_{logic-max} + t_{set-up} + t_{skew} \]

- The amount of time required to turn ‘<‘ into ‘=‘ is referred to as **timing slack**
Preventing hold violations

Hold violations occur when race-through is possible.

Constraint to prevent hold violations:

\[ t_{hold} + t_{skew} < t_{clock-Q-min} + t_{logic-min} \]

- Sometimes have to insert additional logic to prevent hold violations.
**Latch Based Design**

**D-latch**
- Q follows D while clock is high
- Value on D when clock goes low is stored on Q

*Set-up and hold times:*
D can not change close to the falling (‘latching’) clock edge.

\[ t_{\text{clock-Q}} \]
Delay from clock going high to Q changing
Latch timing constraints

To prevent set-up violations:

\[ t_{\text{clock}} + t_{\text{clock-high-max}} > t_{\text{clock-Q-max}} + t_{\text{logic-max}} + t_{\text{set-up}} + t_{\text{skew}} \]

Notes:
- The percentage of time the clock is high is referred to as the duty-cycle.
- If part of the following clock-high time is used to allow this logic to be slower, then the logic-block connected to Q2 must be proportionally faster.
- Using the clock-high time like this is called cycle-stealing.
Latch Set Up Violations

Notes:
• The percentage of time the clock is high is referred to as the duty-cycle

• If part of the following clock-high time is used to allow this logic to be slower, then the logic-block connected to Q2 must be proportionally faster

• Using the clock-high time like this is called cycle-stealing

• Normally cycle stealing is not enabled
Latches … Cycle Stealing
...Latch timing constraints

To prevent hold violations:

\[ t_{\text{clock-high-max}} + t_{\text{hold}} + t_{\text{skew}} < t_{\text{clock-Q-min}} + t_{\text{logic-min}} \]

\[ \text{clock} \]

\[ \text{clock'} \]

\[ Q1 \]

\[ D2 \]

Note:
- Hold violations are harder to prevent in latch-based designs
Example:

\[
\begin{align*}
T_{clock-Q} &= 3 : 4 : 5 \\
T_{NOR} &= 1 : 2 : 3 \\
T_{su_{\text{max}}} &= 1 \\
T_{thold\_{\text{max}}} &= 2 \\
T_{skew} &= 1 \text{ ns}
\end{align*}
\]

If this is the critical path, what is the fastest clock frequency?
Is there potential for a hold violation?

\[
\begin{align*}
\text{Setup:} & \\
\text{Hold:}
\end{align*}
\]
CMOS Drive Strength

Revision: CMOS transistors operating in the linear region:

\[ I_{ds} = \beta ((V_{gs} - V_t)V_{ds} - V_{ds}^2 / 2 \]

where \( \beta = (\mu \varepsilon t_{ox})(W / L) \)

where \( W \) is the transistor width, and \( L \) is the channel length

i.e. To a first approximation,

\[ I_{ds} \approx V_{ds} / R_{on} \]

\[ R_{on} \approx 1 / \beta (V_{GS} - V_T) \]

Thus, delay in CMOS circuits depends largely on \( W / L \) of the drive transistor and the capacitance of the load it is driving.

That capacitance consists of:

\[ \tau = R_{on} C_{load} \]

• Input gates of cells being driven, and
• Capacitance of wiring
Using timing approximations in the datasheet, what is the maximum clock frequency for this circuit (ignore wire load, Tskew):

\[ T_{clock-Q} \]

\[ T_{clock-Q} = \max (1.12 + 1.59 \times CL, 1.09 + 1.32 \times CL) \]

\[ Q \rightarrow A1: \quad CL_{max} = 0.0371 + 0.0117 \text{ pF} = 0.0488 \text{ pF} \]
\[ T_{cp-Q_{max}} = \max (1.12 + 1.59 \times 0.0488, 1.09 + 1.32 \times 0.0488) \]
\[ = \max (1.2, 1.15) = 1.2 \text{ ns} \]

\[ Q \rightarrow A2: \quad T_{cp-Q_{max}} = 1.2 \text{ ns} \]
### Example

**Tlogic:**

<table>
<thead>
<tr>
<th>a1→y</th>
<th>10→01</th>
<th>PD</th>
<th>TR</th>
<th>0.25 × load</th>
<th>1 × load</th>
<th>4 × load</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.749</td>
<td>1.68</td>
<td>5.07</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>a1→y</th>
<th>01→10</th>
<th>PD</th>
<th>TR</th>
<th>0.503 × 3.27 × CL</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>1.15</td>
<td>2.91</td>
<td>10.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.502</td>
<td>1.25</td>
<td>3.40</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>a0→y</th>
<th>10→01</th>
<th>PD</th>
<th>TR</th>
<th>0.371 × 3.21 × CL</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>0.920</td>
<td>1.87</td>
<td>5.61</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.536</td>
<td>1.50</td>
<td>4.85</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>a0→y</th>
<th>01→10</th>
<th>PD</th>
<th>TR</th>
<th>0.505 × 2.12 × CL</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>1.19</td>
<td>2.84</td>
<td>10.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.600</td>
<td>1.31</td>
<td>3.43</td>
</tr>
</tbody>
</table>

**CL:**

<table>
<thead>
<tr>
<th>Pin Capacitance (fF)</th>
<th>NOR2</th>
</tr>
</thead>
<tbody>
<tr>
<td>best</td>
<td>typical</td>
</tr>
<tr>
<td>A0</td>
<td>16.5</td>
</tr>
<tr>
<td>A1</td>
<td>15.9</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Pin Capacitance (fF)</th>
<th>DFF</th>
</tr>
</thead>
<tbody>
<tr>
<td>pin</td>
<td>best</td>
</tr>
<tr>
<td>CP</td>
<td>14.7</td>
</tr>
<tr>
<td>D</td>
<td>10.4</td>
</tr>
<tr>
<td>Q</td>
<td>5.04</td>
</tr>
<tr>
<td>QBAR</td>
<td>11.4</td>
</tr>
</tbody>
</table>

\[
CL = 0.0136 + 0.0184 = 0.032 \text{ pF}
\]

From A0: Tlogic = max \((0.503 + 3.27 \times 0.032, 0.371 + 3.21 \times 0.032)\)

\[= 0.61 \text{ ns}\]

From A1: Tlogic = max \((0.420 + 2.17 \times 0.032, 0.505 + 2.12 \times 0.032)\)

\[= 0.57 \text{ ns}\]
### Example

**Tsu**

<table>
<thead>
<tr>
<th>Special Timing Information</th>
<th>best, 5.5V, -55°C</th>
<th>typical, 5V, 25°C</th>
<th>worst, 4.5V, 125°C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Setup-time on D</td>
<td>0.3</td>
<td>0.6</td>
<td>1.4</td>
</tr>
<tr>
<td>Hold-time on D</td>
<td>0.1</td>
<td>0.05</td>
<td>0.01</td>
</tr>
<tr>
<td>Minimum-pulse-width below CP</td>
<td>0.2</td>
<td>0.3</td>
<td>0.9</td>
</tr>
<tr>
<td>Minimum-pulse-width high on CP</td>
<td>0.08</td>
<td>0.2</td>
<td>0.6</td>
</tr>
<tr>
<td>Minimum-period on CP</td>
<td>0.4</td>
<td>0.8</td>
<td>2.2</td>
</tr>
<tr>
<td>Maximum-fall-time on CP</td>
<td>4</td>
<td>39</td>
<td>3.8×10⁻²</td>
</tr>
</tbody>
</table>

\[
T_{\text{clock}} > T_{\text{c}} - Q_{\text{max}} + T_{\text{logic max}} + T_{\text{su max}} \\
= 1.2 + 0.61 + 1.4 \\
= 3.21 \text{ ns} \\
F_{\text{clock}} < 311 \text{ MHz}
\]
Timing in Design

- The ASIC vendor will develop delay models for each cell
- Best/worst models developed as transistor $\beta$ changes with process, temperature, etc.
- Models also developed for delay as a function of capacitive load (fan-out)
- Timing models are stored as part of the cell library
- Timing Verification is done with a combination of the following:
  - Logic simulation using actual delays from the timing library
    - Dynamic timing analysis
  - Static timing analysis:
    - A static timing analyzer traces through the actual logic and calculates slowest and fastest delay for each true path through the logic
      - Sometimes, however, it can not tell the difference between true and false (not possible) paths
        - Often the designer has to identify false paths
    - It outputs a timing report for critical (Slowest) and other paths
Timing in ASIC Design Flow

Standard Delay Format (SDF) files:

- Tools communicate timing information using SDF files
  - Specifies timing information as calculated from:
    - ASIC vendor timing models (Synopsys outputs as SDF file based on this information)
    - Wiring delay calculations after place and route
- The SDF file is then used by the logic simulator or static timing verifier to verify that the design will not have any setup or hold violations

Timing Closure

- = Ensuring no timing problems after “Place and Route”
  - More than simple back-annotation needed
- Becoming more difficult with increasing clock frequency and finer lithography
  - Interconnect RC delay becoming large compared with logic delay
  - Current synthesis tools are “interconnect blind”
  - New classes of tools emerging (Physical compilers)
  - New planning tools needed in the future
Good and Bad Timing Design Practices

- **Good General Practices:**
  - In each module, use only one clock and one edge.
  - Pay attention to the amount of logic specified in long logic paths during design.
  - Only use the globally distributed clock(s).

- **Bad Design Practices:**
  - Feeding back combinational logic on itself
    - Specifying a latch!
  - Using multiple clocks, multiple edges
  - Gating the clock or generating local clocks
**Exercise**

- With a flip-flop based design, what is the minimum clock period?
- Is there potential for a hold violation?
- If \( t\text{\_clock\_high} \) is 50% of the clock period ("50% duty cycle"), and cycle stealing is enabled, what is potentially the fastest possible latch-based design?

![Diagram](image)

**Flip Flop**: \[ T\text{\_clock\_Q} > T\text{\_clock\_Q\_max} + T\text{\_logic\_max} + T\text{\_su} + T\text{\_skew} \]

\[ = 2 + 7 + 1 + 1 = 11 \text{ ns} \]

\( 90 \text{ MHz} \)

Is \( \text{thold} + \text{tskew} < T\text{\_clock\_Q\_min} + \text{tlogic\_min} \) \[ 2 + 1 < 1 + 6 \]

No hold violation

**Latch**: \[ T\text{\_clock\_Q} = 1-2 \text{ ns} \]

\[ T\text{\_su} = 1 \text{ ns} \]

\[ T\text{\_hold} = 2 \text{ ns} \]

\[ T\text{\_clock} > 7.3 \text{ ns} \]

\[ 136 \text{ MHz} \]
Summary

- What determines the maximum clock frequency?

- What is a hold violation?

- Why do we prefer flip-flop designs over using latches?

- What tool is used to check timing at design closure?
Remember

- Methodology for purposes of ECE 520
  - If at all possible one-edge of one clock
  - If you need multiple clocks, they must have a common root and be related by factors of 2
    - E.g. Root clock: $T_{clock} = 5$ ns
    - This is OK:
      - $T_{clock}$, $T_{clock10}$, $T_{clock20}$
      - $T_{clock10} = 10$ ns ($T_{clock} \times 2$)
      - $T_{clock20} = 20$ ns ($T_{clock} \times 4$)
    - This is NOT OK
      - $T_{clock}$, $T_{clock15}$, $T_{clock17}$
      - $T_{clock15} = 15$ ns ($T_{clock} \times 3$)
      - $T_{clock17} = 17$ ns (??)