Overview of Chapter 4

• Types of Sequential Circuits
• Storage Elements
  ■ Latches
  ■ Flip-Flops
• Sequential Circuit Analysis
  ■ State Tables
  ■ State Diagrams
• Sequential Circuit Design
  ■ Specification
  ■ Assignment of State Codes
  ■ Implementation
  ■ HDL Representation
Latch and Flip-Flop Triggering

So far, the latches we have talked about are "clocked" with an input pulse. Here are some possible waveforms:

- Positive Clock Pulse
- Negative Clock Pulse
- \( W = \) Pulse Width
- Clock Period = Time between referenced edges.
- Reference level is generally 50%.
- Rise and Fall times may be important as well.

System Level Clocking

Consider a system comprised of ranks of latches or flip-flops connected by logic:

If the Clock Period is TOO SHORT, some data changes will not propagate through the network.
If the Clock Pulse Width is TOO LONG, some data will propagate through the second rank of latches!
Master-Slave Flip-Flop

One way to solve the Clocking Problem is with a master-slave organization:

The complement of the clock is used to change the outputs.
Now outputs change on C' only.
Problem: One's catching in Master.
Problem: Instability in Master.
Another solution: Use D-FF's or Edge Triggering

Edge Triggered Flip-Flops

Edge triggered Flip-Flops are sensitive to a small window for data changes around the time of a clock edge.
- Setup Time: The time required for input data to be stable before the clock edge.
- Hold Time: The time data must remain stable after the clock edge.
Flip-Flop Characteristic Tables

The Characteristic Tables:
- Show current inputs.
- Show current state implicitly.
- Predict flip-flop state AFTER CLOCKING.

Clocking conditions are:
- Positive level triggered.
- Negative level triggered.
- Positive edge triggered.
- Negative edge triggered.

NOTE: Proper clocking or flip-flop operation may be subject to conditions such as:
- Set-up and hold times are met.
- Simultaneous SR changes disallowed.

Characteristic Tables

<table>
<thead>
<tr>
<th>J</th>
<th>K</th>
<th>Q(t+1)</th>
<th>Comment</th>
<th>S</th>
<th>R</th>
<th>Q(t+1)</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>Q(t)</td>
<td>No change</td>
<td>0</td>
<td>0</td>
<td>Q(t)</td>
<td>No change</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>Clear Q</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>Clear Q</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>Set Q</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>Set Q</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>Q'(t)</td>
<td>Complement Q</td>
<td>1</td>
<td>1</td>
<td>???</td>
<td>Indeterminate</td>
</tr>
<tr>
<td>T</td>
<td></td>
<td>Q(t+1)</td>
<td>Comment</td>
<td>D</td>
<td></td>
<td>Q(t+1)</td>
<td>Comment</td>
</tr>
<tr>
<td>0</td>
<td></td>
<td>Q(t)</td>
<td>No change</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>Clear Q</td>
</tr>
<tr>
<td>1</td>
<td></td>
<td>Q'(t)</td>
<td>Complement Q</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>Set Q</td>
</tr>
</tbody>
</table>
### J-K Master Slave Flip Flop

Two SR Latches driven by inverted clocks form a master-slave configuration.

Input logic forms the JK logic transition:
- Set (master) = J·Qs'
- Reset (master) = K·Qs

Master Set is possible if the slave Qs is currently "0" any time the clock CP is high!

Master Reset is possible if the slave Qs is currently "1" any time the clock CP is high!

This is referred to as One's Catching.

### Master Slave Symbols

Master-Slave Flip Flops are denoted by a line near the outputs.

This highlights the fact that the slave changes AFTER the master clocking condition is deasserted.
Flip-Flop Conventions

A Bubble near a clock input denotes an active low assertion.
A Triangle near the clock input denotes edge sensitive.
A L-Shaped Line near the output denotes Master/Slave.

Propagation Delay

Logic gate, Latch and Flip-Flop timing parameters:
TPLH: Propagation time low-to-high -- the time required for an output to transition from a low logic level to a high logic level from an input event (usually clock).
TPHL: Propagation time high-to-low -- the time required for an output to transition from a high logic level to a low logic level from an input event (usually clock).
Clock Skew: Difference in clock arrival times at different flip-flops.
Tsu: Set-up time: Time data must be stable at FFs before the clock.
Minimum clock period is set by:
MAX(TPHL, TPLH) + LogicDelay + Tsu + 2x(Clock Skew)
Hold Time and Clock Skew constrain the minimum logic plus flip-flop delay.
Calculating Clock Frequency

Given the network below, assume signal A is changing from "1" to "0":

Clock Period =
\[ t_{PHL} \text{ (Flip-Flop)} + 3 \times t_{PHL} \text{ (Logic)} + t_{su} \text{ (Flip-Flop)} \]

Frequency = \[ \frac{1}{\text{(Clock Period)}} \]

We usually pick \[ \text{MAX}(t_{PHL}, T_{PLH}) \].

Calculating Clock Frequency (Continued)

Given the network below, assume signal A is changing from "0" to "1":

Clock Period =
\[ t_{PLH} \text{ (Flip-Flop)} + 3 \times t_{PLH} \text{ (Logic)} + t_{su} \text{ (Flip-Flop)} \]

Frequency = \[ \frac{1}{\text{(Clock Period)}} \]

We usually pick \[ \text{MAX}(t_{PHL}, T_{PLH}) \].