ESE370: Circuit-Level Modeling, Design, and Optimization for Digital Systems

Day 25: November 7, 2011
Registers

Clocking Discipline
• Follow discipline of combinational logic broken by registers
• Compute
  – From state elements
  – Through combinational logic
  – To new values for state elements
• As long as clock cycle long enough,
  – Will get correct behavior

Previously…

Today
• Clocking: Register Designs
  – Dynamic
  – Static

Alternate Registers

Gate-Latch-Register
• Transistor Count?
• Total Transistor Width?
• Capacitive load on data input?
• Capacitive load on clock?

How does this work as a register?
Compare Gate-Latch-Register

- Transistor Count?
- Total Transistor Width?
- Load on input?
- Load on Clock(s)?

Weaknesses?

- Hold value on capacitance ("Dynamic")
  - Not actively driven
  - Easily upset by noise
  - Will leak away eventually
  - Sets lower bound on clock frequency
  - Cannot "gate off" clock when not in use
- Not drive to rail
  - Less noise margin
  - More static leakage – PMOS not completely off

How Improve?

- Remember weaknesses:
  - Holds value dynamically
  - Not driven to rail

What is the difference?

Transmission Gates

- Idea: use both NMOS/PMOS in parallel
  - NMOS passes the strong 0
  - PMOS passes the strong 1
  - Pass gates that swing full rail
- Often used in mux
Transmission Gate Mux

In1 In2 S
\( \overline{S} \)
Out

In1 In2 S
\( \overline{S} \)
Out

How is this different from a static mux?

How Improve?

- Remember weaknesses:
  - Holds value dynamically
  - Not driven to rail

Level Restorer (“Staticizer”)

Without level restorer

With level restorer

Level Restore

- What issue arises here?
Level Restore

- What issue arises here?

Register with Level Restore

Static Register

- Transistors?
- Total width?
- Clock load?
- Input load?

Static Register

Comparison

<table>
<thead>
<tr>
<th></th>
<th>Gate Latch</th>
<th>Dynamic Latch</th>
<th>Static Latch</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Area</strong></td>
<td>Large</td>
<td>Small</td>
<td>Moderate</td>
</tr>
<tr>
<td><strong>Input Cap</strong></td>
<td>Small</td>
<td>Moderate</td>
<td>Moderate</td>
</tr>
<tr>
<td><strong>Delay</strong></td>
<td>Slow</td>
<td>Fast</td>
<td>Fast</td>
</tr>
<tr>
<td><strong>Dynamic</strong></td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td><strong>Full Rail</strong></td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Typical Static Register

Advantages:
- Static
- Full Rail
- Fast

Isolation inverters:
- Input Cap
- Input/Output Noise
- State Node Noise
Admin

• Midterm Review tonight at 8pm (Moore 204)
• Tuesday: Andre away (no office hour)
• Wednesday: Midterm
  – No lecture
  – Midterm 7-9pm in Towne 303
  – New Project out
• Thursday: <nothing> (read project)
• Friday: Class (Andre)

Ideas

• Registers can be implemented more compactly with pass transistor-based designs