Administrative Stuff

- Homework #8 due today
- Project phase 3 due on Monday, April 17
  - Report template posted on the web
- Midterm 2 tonight
  - 6:30-8pm, 277 Cory
Class Material

- Last lecture
  - Domino logic
  - Bitline design
- Today’s lecture
  - Power
  - Sequential logic
- Reading
  - Chapter 7

Power Revisited
**Transition Activity and Power**

- Energy consumed in $N$ cycles, $E_N$:
  \[
  E_N = C_L \cdot V_{DD}^2 \cdot n_{0\rightarrow1}
  \]

  $n_{0\rightarrow1}$ – number of 0→1 transitions in $N$ cycles

  \[
  P_{avg} = \lim_{N \to \infty} \frac{E_N}{N} \cdot f = \left( \lim_{N \to \infty} \frac{n_{0\rightarrow1}}{N} \right) \cdot C_L \cdot V_{DD}^2 \cdot f
  \]

  \[
  \alpha_{0\rightarrow1} = \lim_{N \to \infty} \frac{n_{0\rightarrow1}}{N} \cdot f
  \]

  \[
  P_{avg} = \alpha_{0\rightarrow1} \cdot C_L \cdot V_{DD}^2 \cdot f
  \]

**Factors Affecting Transition Activity**

- “Static” component (does not account for timing)
  - Type of Logic Function (NOR vs. XOR)
  - Type of Logic Style (Static vs. Dynamic)
  - Signal Statistics
  - Inter-signal Correlations

- “Dynamic” or timing dependent component
  - Circuit Topology
  - Signal Statistics and Correlations
Type of Logic Function: NOR vs. XOR

Example: Static 2-input NOR Gate

Assume signal probabilities
\[ p_{A=1} = \frac{1}{2} \]
\[ p_{B=1} = \frac{1}{2} \]

Then transition probability
\[ p_{0 \rightarrow 1} = p_{Out=0} \times p_{Out=1} \]
\[ = \frac{3}{4} \times \frac{1}{4} = \frac{3}{16} \]

If inputs switch every cycle
\[ \alpha_{0 \rightarrow 1} = \frac{3}{16} \]

Example: Static 2-input XOR Gate

Assume signal probabilities
\[ p_{A=1} = \frac{1}{2} \]
\[ p_{B=1} = \frac{1}{2} \]

Then transition probability
\[ p_{0 \rightarrow 1} = p_{Out=0} \times p_{Out=1} \]
\[ = \frac{1}{2} \times \frac{1}{2} = \frac{1}{4} \]

If inputs switch in every cycle
\[ \alpha_{0 \rightarrow 1} = \frac{1}{4} \]
**Power Consumption of Dynamic Gates**

- In1, In2, PDN, In3, Me, CLK, Out, CL

Power only dissipated when previous Out = 0

---

**Dynamic Power Consumption is Data Dependent**

Dynamic 2-input NOR Gate

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>Out</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

Assume signal probabilities

\[
P_{A=1} = 1/2 \quad P_{B=1} = 1/2
\]

Then transition probability

\[
P_{0\to1} = P_{\text{out}=0} \times P_{\text{out}=1}
\]

\[= 3/4 \times 1 = 3/4\]

Switching activity always higher in dynamic gates!

\[
P_{0\to1} = P_{\text{out}=0}
\]
**Dynamic CVSL**

Guaranteed transition for every operation!

\[ \alpha_{0 \rightarrow 1} = 1 \]

**Clock**

- Always switches
- Consumes 25-50% of power
- Clock gating commonly employed
**Problem: Reconvergent Fanout**

\[ P(Z = 1) = P(B = 1) \cdot P(X = 1 | B=1) \]

Becomes complex and intractable fast

---

**Inter-Signal Correlations**

- Logic without reconvergent fanout:
  \[ p_{0\rightarrow1} = (1 - p_A p_B) \cdot p_B \]

- Logic with reconvergent fanout:
  \[ P(Z = 1) = p(C=1 | B=1) \cdot p(B=1) \]
  \[ p_{0\rightarrow1} = 0 \]

- Need to use conditional probabilities to model inter-signal correlations
- CAD tools required for such analysis
Glitching in Static CMOS

The result is correct, but there is extra power dissipated.

Example: Chain of NOR Gates
Principles for Power Reduction

- Prime choice: Reduce voltage!
  - Recent years have seen an acceleration in supply voltage reduction
  - Design at very low voltages still open question
  - Reducing thresholds to improve performance increases leakage
- Reduce switching activity
- Reduce physical capacitance

Sequential Logic
**Writing into a Static Latch**

Use the clock as a decoupling signal, that distinguishes between the transparent and opaque states

- Forcing the state (can implement as NMOS-only)
- Converting into a MUX

**Latch Properties**

- **Two phase operation**
  - $\text{Clk} = 1$: transparent
  - $\text{Clk} = 0$: latches data
- **Transparency can cause the data contamination**
  - Often avoided by using edge-triggered registers
**Master-Slave (Edge-Triggered) Register**

Two opposite latches trigger on edge
Also called master-slave latch pair

---

**Master-Slave Register**

Multiplexer-based latch pair
Reduced Clock Load Master-Slave Register

Clk-Q Delay
**Setup Time**

(a) $T_{\text{setup}} = 0.21$ nsec

(b) $T_{\text{setup}} = 0.20$ nsec

**More Precise Setup Time**

(a) $t_D$, $t_H$, $t_{\text{setup}}$, $t_{\text{C}} - Q$

(b) $t_D - C$, $t_{\text{C}} - Q$, $t_D$, $t_{\text{C}}$
Setup-Hold Time Illustrations

Circuit before clock arrival (Setup-1 case)

Hold-1 case
**Other Latches/Registers: C²MOS**

Keepers should be added to staticize

---

**Other Latches/Registers: TSPC**

Positive latch (transparent when CLK= 1)  
Negative latch (transparent when CLK= 0)
Including Logic in TSPC

Example: logic inside the latch

AND latch

TSPC Register

EECS141
Pulse-Triggered Latches

Ways to design an edge-triggered sequential cell:

Master-Slave Latches

Pulse-Triggered Latch

Ways to design an edge-triggered sequential cell:

Pulsed Latches

(a) register

(b) glitch generation

(c) glitch clock
**Pulsed Latches**

Hybrid Latch – Flip-flop (HLFF), AMD K-6 and K-7:

![HLFF Circuit Diagram](image)

**HLFF Timing**

![Timing Graph](image)
Next Lecture

- Sequential logic
- Timing