Dynamic Combinational Circuits

- Dynamic circuits
  - Charge sharing, charge redistribution
- Domino logic
- np-CMOS (zipper CMOS)

Dynamic Logic

- Dynamic gates use a clocked pMOS pullup
- Two modes: precharge and evaluate
The Foot

• What if pulldown network is ON during precharge?
• Use series evaluation transistor to prevent fight.

Dynamic Logic

2 phase operation:
• Precharge
• Evaluation
Logical Effort

Inverter      NAND2      NOR2

\[ \begin{align*}
\text{unfooted} & \quad \phi & \quad A & \quad Y \\
& \quad \frac{g_i}{p_i} = 1/3 & \quad A & \quad \frac{g_o}{p_o} = 2/3
\end{align*} \]

\[ \begin{align*}
\text{footed} & \quad \phi & \quad A & \quad Y \\
& \quad \frac{g_i}{p_i} = 2/3 & \quad A & \quad \frac{g_o}{p_o} = 3/3
\end{align*} \]

Krish Chakrabarty

Dynamic Logic

- N+2 transistors for N-input function
  - Better than 2N transistors for complementary static CMOS
  - Comparable to N+1 for ratio-ed logic
- No static power dissipation
  - Better than ratio-ed logic
- Careful design, clock signal $\Phi$ needed

Krish Chakrabarty
**Dynamic Logic: Principles**

- **Precharge**
  \( \Phi = 0 \), \( \text{Out} \) is precharged to \( V_{DD} \) by \( M_p \).
  \( M_e \) is turned off, no dc current flows (regardless of input values)

- **Evaluation**
  \( \Phi = 1 \), \( M_e \) is turned on, \( M_p \) is turned off.
  Output is pulled down to zero depending on the values on the inputs. If not, precharged value remains on \( C_L \).

**Important**: Once \( \text{Out} \) is discharged, it cannot be charged again!
Gate input can make only one transition during evaluation

- Minimum clock frequency must be maintained
- Can \( M_e \) be eliminated?

---

**Example**

\[ V_{DD} \]

\[ \Phi \quad M_p \quad \text{Out} \]

\[ A \quad C \]

\[ \Phi \quad M_e \]

- **Rationale**
- **No Static Power Consumption**
- **Noise Margins small (NM)**
- **Requires Clock**
Dynamic 4 Input NAND Gate

Cascading Dynamic Gates

Internal nodes can only make 0-1 transitions during evaluation period
Monotonicity

- Dynamic gates require *monotonically rising* inputs during evaluation
  - 0 -> 0
  - 0 -> 1
  - 1 -> 1
  - But not 1 -> 0

Monotonicity Woes

- But dynamic gates produce monotonically falling outputs during evaluation
- Illegal for one dynamic gate to drive another!
Reliability Problems — Charge Leakage

(a) Leakage sources

(1) Leakage through reverse-biased diode of the diffusion area
(2) Subthreshold current from drain to source

Minimum Clock Frequency: > 1 MHz

Leakage

- Dynamic node floats high during evaluation
  - Transistors are leaky (I_{OFF} ≠ 0)
  - Dynamic value will leak away over time
  - Formerly miliseconds, now nanoseconds!
- Use keeper to hold dynamic node
  - Must be weak enough not to fight evaluation
Charge Sharing (redistribution)

- Assume: during precharge, A and B are 0, $C_a$ is discharged
- During evaluation, B remains 0 and A rises to 1
- Charge stored on $C_L$ is now redistributed over $C_L$ and $C_a$

\[ C_L V_{DD} = C_L V_{out}(t) + C_a V_X \]

\[ V_X = V_{DD} - V_t, \text{ therefore} \]
\[ \delta V_{out}(t) = V_{out}(t) - V_{DD} = - \frac{C_a}{C_L} (V_{DD} - V_t) \]

Desirable to keep the voltage drop below threshold of pMOS transistor (why?) \( \Rightarrow C_a/C_L < 0.2 \)

Charge Sharing

- Dynamic gates suffer from charge sharing

\[ V_X = V_Y = \frac{C_Y}{C_x + C_Y} V_{DD} \]
Charge Redistribution - Solutions

(a) Static bleeder

(b) Precharge of internal nodes

Secondary Precharge

- Solution: add secondary precharge transistors
  - Typically need to precharge every other node
- Big load capacitance $C_V$ helps as well
Domino Logic

- Follow dynamic stage with inverting static gate
  - Dynamic / static pair is called domino gate
  - Produces monotonic outputs
**Domino Logic - Characteristics**

- Only non-inverting logic
- Very fast - Only 1->0 transitions at input of inverter
- Precharging makes pull-up very fast
- Adding level restorer reduces leakage and charge redistribution problems
- Optimize inverter for fan-out

---

**Domino Optimizations**

- Each domino gate triggers next one, like a string of dominos toppling over
- Gates evaluate sequentially but precharge in parallel
- Thus evaluation is more critical than precharge
- HI-skewed static stages can perform logic
### Dual-Rail Domino

- Domino only performs noninverting functions:
  - AND, OR but not NAND, NOR, or XOR
- Dual-rail domino solves this problem
  - Takes true and complementary inputs
  - Produces true and complementary outputs

<table>
<thead>
<tr>
<th>sig_h</th>
<th>sig_l</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>Precharged</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>‘0’</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>‘1’</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>invalid</td>
</tr>
</tbody>
</table>

**Example: AND/NAND**

- Given $A_h, A_l, B_h, B_l$
- Compute $Y_h = A \times B$, $Y_l = \neg(A \times B)$
- Pulldown networks are conduction complements
Example: XOR/XNOR

- Sometimes possible to share transistors

\[ Y_{l} = A \text{xnor} B \quad A_{l} \quad \phi \quad A_{h} \quad Y_{h} = A \text{xor} B \]

Domino Summary

- Domino logic is attractive for high-speed circuits
  - 1.5 – 2x faster than static CMOS
  - But many challenges:
    - Monotonicity
    - Leakage
    - Charge sharing
    - Noise
- Widely used in high-performance microprocessors
np-CMOS (Zipper CMOS)

- Only 1-0 transitions allowed at inputs of PUN
- Used a lot in the Alpha design

np CMOS Adder

Krish Chakrabarty
# CMOS Circuit Styles - Summary

<table>
<thead>
<tr>
<th>Style</th>
<th>Ratioed</th>
<th>Static Power</th>
<th># transistors</th>
<th>Area (μm²)</th>
<th>Propagation Delay (nsec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Complementary</td>
<td>No</td>
<td>No</td>
<td>8</td>
<td>533</td>
<td>0.61</td>
</tr>
<tr>
<td>Pseudo-NMOS</td>
<td>Yes</td>
<td>Yes</td>
<td>5</td>
<td>288</td>
<td>1.49</td>
</tr>
<tr>
<td>CPL</td>
<td>No</td>
<td>No</td>
<td>14</td>
<td>800</td>
<td>0.75</td>
</tr>
<tr>
<td>Dynamic (NP)</td>
<td>No</td>
<td>No</td>
<td>6</td>
<td>212</td>
<td>0.37</td>
</tr>
</tbody>
</table>

4-input NAND Gate