VLSI Digital Systems Design

Circuit and Physical Design of Fully-Complementary CMOS Logic Gates

Fan-in

- Fan-in = number of inputs to a gate
- Examples:
  1. Nand $g_1$ (outg1, in1, in2, in3, in4); fan-in = 4
  2. Nor $g_2$ (outg2, in1, in2);  

Fan-out

- Fan-out = number of inputs that a gate's output drives
- In units of minimum-sized inverters
- Examples:
  - Nand $g_1$ (outg1, a, b, c, d);
  - Nor g2 (outg2, a, e);
  - Not g3 (a, f);  

Large Fan-in Slows Gate

- Resistance of series transistors in pull-up network or pull-down network additive
- Two transistors in series will double the rise or fall time compared to single transistor

Worst-case delay time

- $T_{df}$ = worst-case fall delay time
- $T_{df} = t_{\text{internal-f}} + k \cdot t_{\text{output-f}}$
- $T_{dr}$ = worst-case rise delay time
- $T_{dr} = t_{\text{internal-r}} + k \cdot t_{\text{output-r}}$
- where $k$ = fan-out
  - in units of minimum-sized inverters

Gate Delays

<table>
<thead>
<tr>
<th>Gate</th>
<th>$t_{\text{internal-f}}$</th>
<th>$t_{\text{output-f}}$</th>
<th>$t_{\text{internal-r}}$</th>
<th>$t_{\text{output-r}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>INV</td>
<td>0.08</td>
<td>1.70</td>
<td>0.08</td>
<td>2.10</td>
</tr>
<tr>
<td>ND2</td>
<td>0.20</td>
<td>3.10</td>
<td>0.15</td>
<td>2.10</td>
</tr>
<tr>
<td>ND4</td>
<td>0.68</td>
<td>5.70</td>
<td>0.25</td>
<td>2.10</td>
</tr>
<tr>
<td>ND8</td>
<td>2.44</td>
<td>9.99</td>
<td>0.38</td>
<td>2.20</td>
</tr>
<tr>
<td>NR2</td>
<td>0.14</td>
<td>1.75</td>
<td>0.25</td>
<td>4.10</td>
</tr>
</tbody>
</table>
8-input And, Case 1

- Nand stage1 (s1out, a, b, c, d, e, f, g, h);
- Not stage2 (s2out, s1out);

8-input And, Case 2

- Still 2 stages
- Less fan-in

8-input And, Case 3

- Twice as many stages
- Minimal fan-in

8-input And, Comparison

<table>
<thead>
<tr>
<th>Case</th>
<th>Stage</th>
<th>Stage</th>
<th>Stage</th>
<th>Stage</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td></td>
<td>fall</td>
<td>rise</td>
<td>fall</td>
<td>rise</td>
<td>Delay</td>
</tr>
</tbody>
</table>

- 1 2.44 2.18 0.00 0.00 4.62
- 2 0.68 4.35 0.00 0.00 5.03
- 3 0.20 0.25 0.20 2.18 2.83

Transistor Sizing

1. Start with minimum-sized devices
2. Determine critical paths
3. Increase the gate size along critical paths

Timing Rules of Thumb, Page 1

- Use Nands
- Avoid Nors
- Output rise and fall time depend on input rise and fall times
  - Keep edges sharp throughout each critical path
Timing Rules of Thumb, Page 2

- Keep fan-out low
  - Below 5-10
  - Use minimal-sized gates to minimize the load on gates with high fan-out
  - Drive high fan-out nets with inverters

- Keep fan-in low
  - Unless low power or low area is more important

---

Layout Layer Key

- **aa** = poly (input)
- **bb** = poly (input)
- **cc** = poly (internal)
- **mm** = metal (internal)
- **dd** = Vdd (metal)
- **ss** = Vss (metal)
- **nn** = n-diffusion
- **pp** = p-diffusion
- **--** = gate
- **** = contact

---

Vertical-Transistor Inverter Layout

- **dd** = Vdd
- **pp**
- **aa**
- **cc**
- **mm**
- **nn**
- **ss**
- **ssssssssssss** = Vss

---

Horizontal-Transistor Inverter Layout

- **dd** = Vdd
- **pp**
- **aa**
- **cc**
- **mm**
- **nn**
- **ss**
- **ssssssssssss** = Vss

---

Horizontal-Transistor Inv Layout w Central Metal Pass-Through

- **dd** = Vdd
- **pp**
- **aa**
- **cc**
- **mm**
- **nn**
- **ss**
- **ssssssssssssss** = Vss

---

Horizontal-Transistor Inv Layout w Top & Bottom Metal Pass-Thru

- **dd** = Vdd
- **pp**
- **aa**
- **cc**
- **mm**
- **nn**
- **ss**
- **ssssssssssssss** = Vss
Large-Transistor Inverter Layout

Back-to-Back-Transistor Inverter Layout

Advantages of Back-to-Back-Transistor Inverter Layout
- Drain area does not increase in size much
  - Drain capacitance does not increase much
- Transistor gain doubled

Advantages of Doughnut-Transistor Inverter Layout
- Drain area does not increase in size much
  - Drain capacitance does not increase much
- Transistor gain almost quadrupled
- Also called “round transistor” connection

“Line of Diffusion” Layout
- Single row of p-transistors above single row of n-transistors
- Aligned at common gate connections
- Transistors form line of diffusion intersected by polysilicon gate connections
- All complementary gates can be designed this way
- Most simple gates can be designed without a break in the diffusion
Complex Gate Layout

- Convert circuit to two graphs
  1. p-transistor pull-up network
  2. n-transistor pull-down network
- Each graph is the dual of the other
- Vertices = source-drain connections
- Edges = transistors that connect vertices
  - Labeled with gate signal name for that transistor

Euler Path

- If two edges are adjacent in one of the graphs
  - They share a common source-drain connection
  - They can be connected by abutment
- If there is an Euler path
  - If there is a sequence of edges
    - containing all edges in the p-graph and the n-graph
    - that have identical labeling
  - Then the gate can be designed with no breaks

“Line of Diffusion” Layout Algorithm

1. Find all Euler paths that cover the n-graph and the p-graph
2. Find an Euler path for the n-graph and an Euler path for the p-graph that have identical labeling
   - Labeling = ordering of the gate labels on each vertex
3. If not found, break the gate in the minimum number of places to achieve step 2 by separate Euler paths

Stacked Transistor Layout

- Input signal applied to gates of multiple transistors
- Transistors stacked on appropriate gate signal
- Used for cascaded gates
  - Xnor
- Variation in distance between power lines makes standard cell layout more difficult
  - C.f. “line of diffusion” layout

Xnor Logic

\[
\text{a} \quad \text{b} \quad \text{z} = \text{a} \oplus \text{b}
\]