Static CMOS Circuits

- Conventional (ratio-less) static CMOS
  - Covered so far
- Ratio-ed logic (depletion load, pseudo nMOS)
- Pass transistor logic
Example 1

module mux(input s, d0, d1, 
    output y);

    assign y = s ? d1 : d0;
endmodule

1) Sketch a design using AND, OR, and NOT gates.
Example 1

module mux(input s, d0, d1,
           output y);

    assign y = s ? d1 : d0;
endmodule

1) Sketch a design using AND, OR, and NOT gates.
Example 2

2) Sketch a design using NAND, NOR, and NOT gates. Assume \( \sim S \) is available.
Example 2

2) Sketch a design using NAND, NOR, and NOT gates. Assume \( \sim S \) is available.
Bubble Pushing

- Start with network of AND / OR gates
- Convert to NAND / NOR + inverters
- Push bubbles around to simplify logic
  - Remember DeMorgan’s Law
Example 3

3) Sketch a design using one compound gate and one NOT gate. Assume \( \sim S \) is available.
Example 3

3) Sketch a design using one compound gate and one NOT gate. Assume \( \sim S \) is available.
Compound Gates

• Logical Effort of compound gates

unit inverter

\[ Y = \overline{A} \]

AOI21

\[ Y = A \overline{B} + C \]

AOI22

\[ Y = A \overline{B} + C \overline{D} \]

Complex AOI

\[ Y = A \overline{(B + C)} + D \overline{E} \]

\[ g_A = \frac{6}{3} \]
\[ g_B = \frac{6}{3} \]
\[ g_C = \frac{5}{3} \]
\[ p = \frac{7}{3} \]

\[ g_A = \frac{6}{3} \]
\[ g_B = \frac{6}{3} \]
\[ g_C = \frac{6}{3} \]
\[ g_D = \frac{6}{3} \]
\[ p = \frac{12}{3} \]

\[ g_A = \frac{5}{3} \]
\[ g_B = \frac{8}{3} \]
\[ g_C = \frac{8}{3} \]
\[ g_D = \frac{8}{3} \]
\[ g_E = \frac{8}{3} \]
\[ p = \frac{16}{3} \]
Example 4

• The multiplexer has a maximum input capacitance of 16 units on each input. It must drive a load of 160 units. Estimate the delay of the NAND and compound gate designs.

\[ H = \frac{160}{16} = 10 \]
\[ B = 1 \]
\[ N = 2 \]
NAND Solution

\[ Y = \overline{D0 \cdot S} + \overline{D1 \cdot S} \]
NAND Solution

\[ P = 2 + 2 = 4 \]
\[ G = \left(\frac{4}{3}\right) \times \left(\frac{4}{3}\right) = \frac{16}{9} \]
\[ F = GBH = \frac{160}{9} \]
\[ \hat{f} = \sqrt[\text{N}]{F} = 4.2 \]
\[ D = N\hat{f} + P = 12.4\tau \]
Compound Solution
Compound Solution

\[ P = 4 + 1 = 5 \]
\[ G = (6 / 3) \oplus (1) = 2 \]
\[ F = GBH = 20 \]
\[ \hat{f} = \sqrt[4.5]{F} = 4.5 \]
\[ D = N\hat{f} + P = 14\tau \]
Example 5

- Annotate your designs with transistor sizes that achieve this delay.

Homework exercise!
Input Order

- Our parasitic delay model was too simple
  - Calculate parasitic delay for Y falling
    - If A arrives latest?
    - If B arrives latest?
Input Order

• Our parasitic delay model was too simple
  – Calculate parasitic delay for Y falling
    • If A arrives latest? $2\tau$
    • If B arrives latest? $2.33\tau$
Inner & Outer Inputs

- *Outer* input is closest to rail (B)
- *Inner* input is closest to output (A)

- If input arrival time is known
  - Connect latest input to inner terminal
Asymmetric Gates

- Asymmetric gates favor one input over another
- Ex: suppose input A of a NAND gate is most critical
  - Use smaller transistor on A (less capacitance)
  - Boost size of noncritical input
  - So total resistance is same
- \( g_A = \frac{10}{9} \)
- \( g_B = 2 \)
- \( g_{\text{total}} = g_A + g_B = \frac{28}{9} \)
- Asymmetric gate approaches \( g = 1 \) on critical input
- But total logical effort goes up
Symmetric Gates

• Inputs can be made perfectly symmetric
Skewed Gates

- Skewed gates favor one edge over another
- Ex: suppose rising output of inverter is most critical
  - Downsize noncritical nMOS transistor

\[
g_u = \frac{1}{2} \quad \text{HI-skew} \quad \text{inverter}
\]

\[
g_d = \frac{1}{2} \quad \text{unkskewed inverter} \quad (\text{equal rise resistance})
\]

\[
g_d = \frac{1}{2} \quad \text{unkskewed inverter} \quad (\text{equal fall resistance})
\]

- Calculate logical effort by comparing to unskewed inverter with same effective resistance on that edge.
  - \( g_u = \)
  - \( g_d = \)
Skewed Gates

- Skewed gates favor one edge over another
- Ex: suppose rising output of inverter is most critical
  - Downsize noncritical nMOS transistor

\[
\text{HI-skew inverter} \quad \text{unskewed inverter (equal rise resistance)} \quad \text{unskewed inverter (equal fall resistance)}
\]

\[
\begin{array}{c}
A \quad \text{Y} \\
\downarrow \quad 1/2 \\
\end{array}
\quad \begin{array}{c}
A \quad \text{Y} \\
\downarrow \quad 1 \\
\end{array}
\quad \begin{array}{c}
A \quad \text{Y} \\
\downarrow \quad 1/2 \\
\end{array}
\]

- Calculate logical effort by comparing to unskewed inverter with same effective resistance on that edge.
  - \( g_u = \frac{2.5}{3} = \frac{5}{6} \)
  - \( g_d = \frac{2.5}{1.5} = \frac{5}{3} \)
HI- and LO-Skew

- Def: Logical effort of a skewed gate for a particular transition is the ratio of the input capacitance of that gate to the input capacitance of an unskewed inverter delivering the same output current for the same transition.

- Skewed gates reduce size of noncritical transistors
  - HI-skew gates favor rising output (small nMOS)
  - LO-skew gates favor falling output (small pMOS)

- Logical effort is smaller for favored direction
- But larger for the other direction
Catalog of Skewed Gates

Inverter

- **Unskewed**
  - $g_u = 1$
  - $g_d = 1$
  - $g_{avg} = 1$

- **Hi-skew**
  - $g_u = 5/6$
  - $g_d = 5/3$
  - $g_{avg} = 5/4$

- **Lo-skew**
  - $g_u = 4/3$
  - $g_d = 2/3$
  - $g_{avg} = 1$

NAND2

- **Unskewed**
  - $g_u = 4/3$
  - $g_d = 4/3$
  - $g_{avg} = 4/3$

- **Hi-skew**
  - $g_u = 5/3$
  - $g_d = 5/3$
  - $g_{avg} = 5/3$

- **Lo-skew**
  - $g_u = 5/3$
  - $g_d = 5/3$
  - $g_{avg} = 5/3$

NOR2

- **Unskewed**
  - $g_u = 1$
  - $g_d = 1$
  - $g_{avg} = 1$

- **Hi-skew**
  - $g_u = 4/3$
  - $g_d = 4/3$
  - $g_{avg} = 4/3$

- **Lo-skew**
  - $g_u = 5/3$
  - $g_d = 5/3$
  - $g_{avg} = 5/3$
Catalog of Skewed Gates

Inverter

Unskewed

\[ g_u = 1 \]
\[ g_d = 1 \]
\[ g_{avg} = 1 \]

HI-skew

\[ g_u = \frac{5}{6} \]
\[ g_d = \frac{5}{3} \]
\[ g_{avg} = \frac{5}{4} \]

LO-skew

\[ g_u = \frac{4}{3} \]
\[ g_d = \frac{2}{3} \]
\[ g_{avg} = 1 \]

NAND2

\[ g_u = \frac{4}{3} \]
\[ g_d = \frac{4}{3} \]
\[ g_{avg} = \frac{4}{3} \]

NOR2

\[ g_u = \frac{5}{3} \]
\[ g_d = \frac{5}{3} \]
\[ g_{avg} = \frac{5}{3} \]
Catalog of Skewed Gates

Inverter

- **Unskewed**
  - \( g_u = 4/3 \)
  - \( g_d = 2/3 \)
  - \( g_{avg} = 1 \)

- **HI-skew**
  - \( g_u = 1 \)
  - \( g_d = 1 \)
  - \( g_{avg} = 1 \)

- **LO-skew**
  - \( g_u = 1 \)
  - \( g_d = 1 \)
  - \( g_{avg} = 1 \)

NAND2

- **Unskewed**
  - \( g_u = 1 \)
  - \( g_d = 1 \)
  - \( g_{avg} = 1 \)

- **HI-skew**
  - \( g_u = 4/3 \)
  - \( g_d = 4/3 \)
  - \( g_{avg} = 4/3 \)

- **LO-skew**
  - \( g_u = 2 \)
  - \( g_d = 1 \)
  - \( g_{avg} = 3/2 \)

NOR2

- **Unskewed**
  - \( g_u = 5/3 \)
  - \( g_d = 5/3 \)
  - \( g_{avg} = 5/3 \)

- **HI-skew**
  - \( g_u = 3/2 \)
  - \( g_d = 3 \)
  - \( g_{avg} = 9/4 \)

- **LO-skew**
  - \( g_u = 2 \)
  - \( g_d = 1 \)
  - \( g_{avg} = 3/2 \)
Asymmetric Skew

- Combine asymmetric and skewed gates
  - Downsize noncritical transistor on unimportant input
  - Reduces parasitic delay for critical input

![Asymmetric Skew Diagram]
Best P/N Ratio

- We have selected P/N ratio for unit rise and fall resistance ($\mu = 2-3$ for an inverter).
- Alternative: choose ratio for least average delay
- Ex: inverter
  - Delay driving identical inverter
    - $t_{pdf} =$
    - $t_{pdr} =$
    - $t_{pd} =$
    - Differentiate $t_{pd}$ w.r.t. $P$
    - Least delay for $P =$
Best P/N Ratio

- We have selected P/N ratio for unit rise and fall resistance ($\mu = 2-3$ for an inverter).
- Alternative: choose ratio for least average delay
- Ex: inverter
  - Delay driving identical inverter
  - $t_{pdf} = (P+1)$
  - $t_{pdr} = (P+1)(\mu/P)$
  - $t_{pd} = (P+1)(1+\mu/P)/2 = (P + 1 + \mu + \mu/P)/2$
  - Differentiate $t_{pd}$ w.r.t. $P$
  - Least delay for $P = \sqrt{\mu}$
P/N Ratios

- In general, best P/N ratio is sqrt of that giving equal delay.
  - Only improves average delay slightly for inverters
  - But significantly decreases area and power

<table>
<thead>
<tr>
<th></th>
<th>Inverter</th>
<th>NAND2</th>
<th>NOR2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fastest P/N Ratio</td>
<td>![Inverter Diagram]</td>
<td>![NAND2 Diagram]</td>
<td>![NOR2 Diagram]</td>
</tr>
<tr>
<td></td>
<td>$g_u = 1.15$</td>
<td>$g_u = 4/3$</td>
<td>$g_u = 2$</td>
</tr>
<tr>
<td></td>
<td>$g_d = 0.81$</td>
<td>$g_d = 4/3$</td>
<td>$g_d = 1$</td>
</tr>
<tr>
<td></td>
<td>$g_{avg} = 0.98$</td>
<td>$g_{avg} = 4/3$</td>
<td>$g_{avg} = 3/2$</td>
</tr>
</tbody>
</table>
Observations

• For speed:
  – NAND vs. NOR
  – Many simple stages vs. fewer high fan-in stages
  – Latest-arriving input

• For area and power:
  – Many simple stages vs. fewer high fan-in stages
Combinational vs. Sequential Logic

(a) Combinational

Output = f(In)

(b) Sequential

Output = f(In, Previous In)
Static CMOS Circuit (Review)

At every point in time (except during the switching transients) each gate output is connected to either $V_{DD}$ or $V_{ss}$ via a low-resistive path.

The outputs of the gates assume at all times the value of the Boolean function, implemented by the circuit (ignoring, once again, the transient effects during switching periods).

This is in contrast to the *dynamic* circuit class, which relies on temporary storage of signal values on the capacitance of high impedance circuit nodes.
Static CMOS (Review)

$V_{DD}$

$In1$
$In2$
$In3$

PUN
PMOS Only

$F = \overline{G}$

$V_{SS}$

$In1$
$In2$
$In3$

PDN
NMOS Only

PUN and PDN are Dual Networks
High noise margins:

$V_{OH}$ and $V_{OL}$ are at $V_{DD}$ and $GND$, respectively.

No static power consumption:

There never exists a direct path between $V_{DD}$ and $V_{SS}$ ($GND$) in steady-state mode.

Comparable rise and fall times:

(under the appropriate scaling conditions)
Influence of Fan-In and Fan-Out on Delay

Fan-Out: Number of Gates Connected
Every fanout (output) adds two gate capacitances (pMOS and nMOS)

FanIn: Quadratic Term due to:
1. Resistance Increasing
2. Capacitance Increasing

\[ t_p = a_1 FI + a_2 FI^2 + a_3 FO \]
Fast Complex Gate-Design Techniques

- **Transistor Sizing:**
  As long as Fan-out Capacitance dominates

- **Progressive Sizing:**

  ![Diagram](Image)

  \[ M_1 > M_2 > M_3 > M_N \]
Fast Complex Gate - Design Techniques

- Transistor Ordering

(a) (b)
Fast Complex Gate - Design Techniques

- Improved Logic Design
Ratioed Logic

Resistive Load

\( V_{DD} \)

\( R_L \)

\( V_{SS} \)

(a) resistive load

Depletion Load

\( V_{DD} \)

\( V_T < 0 \)

\( V_{SS} \)

(b) depletion load NMOS

PMOS Load

\( V_{SS} \)

\( V_{SS} \)

(c) pseudo-NMOS

Goal: to reduce the number of devices over complementary CMOS

Careful design needed!
Ratioed Logic

- \( V_{OH} = V_{DD} \)

\[
V_{OL} = \frac{R_{PDN}}{R_L + R_{PDN}} V_{DD}
\]

Desired: \( R_L \gg R_{PDN} \) (to keep noise margin low)

\[
t_{PLH} = 0.69 R_L C_L
\]

Problems:
1) Static power dissipation
2) Difficult to implement a large resistor, e.g. 40k\(\Omega\) resistor (typical value) needs 3200 \(\mu^2\) of n-diff, i.e. 1,000 transistors!
Depletion Load

- Depletion-mode transistor has negative threshold
- On if $V_{GS} = 0$
- Body effect may be a problem!

PMOS Load

pseudo-NMOS
Pseudo-nMOS

- No problems due to body effect
- N-input gate requires only N+1 transistors
- Each input connects to only a single transistor, presenting smaller load to preceding gate
- Static power dissipation (when output is zero)
- Asymmetric rise and fall times

Example: Suppose minimal-sized gate consumes 1 mW of static power.

100,000 gate-circuit: 50 W of static power (plus dynamic power)!
(half the gates are in low-output state)

- Effective only for small subcircuits where speed is important, eg address decoders in memories
Pseudo-NMOS NAND Gate

\[ C_{L,\text{pseudo}} = 0.5 \ C_{L,\text{CMOS}} \ (\text{Fan-out of 1}) \]
Pass-Transistor Logic

- No static consumption

Is this transmission gates necessary?

Need a low impedance path to ground when B = 0
Pass-Transistor Based Multiplexer

$$F = AS + BS$$
Transmission Gate XOR

6 transistors only!

*Case 1:*
\[ B = 1, \text{M3/M4 turned off, } F = \overline{AB} \]

*Case 2:*
\[ B = 0, \text{M3/M4 turned on, } F = AB \]

F always has a path to \( V_{DD} \) or Gnd, hence low impedance node. If not, node would be *dynamic*, requiring refresh due to charge leakage.
Delay in Transmission Gate Networks

(a) Insert buffers after every m switches
Delay in Transmission Gate Networks

Consider Kirchoff’s Law at node $V_i$

$$\frac{V_{i+1} - V_i}{R_{eq}} + \frac{V_{i-1} - V_i}{R_{eq}} = \frac{C}{R_{eq}} \frac{dV_i}{dt}$$

Therefore, $\frac{dV_i}{dt} = \frac{V_{i+1} + V_{i-1} - 2V_i}{R_{eq}C}$

Propagation delay can be determined using Elmore delay analysis
Delay Optimization

- **Delay of RC chain**

\[ t_p = 0.69 \sum_{k=0}^{n} CR_{eq}^k = 0.69 CR_{eq} \frac{n(n+1)}{2} \]

Delay can be reduced by adding buffers after m stages

\( t_{buf} = \text{delay of a buffer} \)

- **Delay of Buffered Chain**

\[ t_p = 0.69 \left[ \frac{n}{m} CR_{eq} \frac{m(m+1)}{2} \right] + \left( \frac{n}{m} - 1 \right) t_{buf} \]

\[ = 0.69 \left[ CR_{eq} \frac{n(m+1)}{2} \right] + \left( \frac{n}{m} - 1 \right) t_{buf} \]

\[ m_{opt} = 1.7 \sqrt{\frac{t_{buf}}{CR_{eq}}} \]
Transmission Gate Full Adder

- **Setup**
  - Inputs: $A$, $B$, $C_i$
  - Outputs: $A$, $B$, $C_i$
  - Power Supply: $V_{DD}$

- **Sum Generation**
  - Inputs: $A$, $B$, $P$
  - Outputs: $S$
  - Power Supply: $V_{DD}$

- **Carry Generation**
  - Inputs: $A$, $B$, $P$
  - Outputs: $C_o$
  - Power Supply: $V_{DD}$
NMOS Only Logic: Level Restoring Transistor

- Advantage: Full Swing
- Disadvantage: More Complex, Larger Capacitance
- Other approaches: reduced threshold NMOS
Single Transistor Pass Gate with $V_T=0$

Watch out for leakage currents
Complimentary Pass Transistor Logic

(a) Pass-Transistor Network

F = AB

(b) Inverse Pass-Transistor Network

F = A + B

AND/NAND

OR/NOR

EXOR/NEXOR
4 Input NAND in CPL