Digital Systems – A preview
Quiz 4

Determine $V_H$ and $V_L$ at the C output for the Static CMOS NOR gate shown. Use the switch-level model for the transistors introduced in the last lecture.
And the number is ....
And the number is .... 7
Quiz 4

Determine $V_H$ and $V_L$ at the C output for the Static CMOS NOR gate shown. Use the switch-level model for the transistors introduced in the last lecture.

Solution:

$V_H = V_{DD}$, $V_L = 0V$
Review from Last Time

MOS Transistor
Qualitative Discussion of n-channel Operation
Review from Last Time

**MOS Transistor**

Comparison of Operation
Logic Circuits

**Review from Last Time**

Inverter

\[
\begin{array}{c|c|c}
A & B \\
0 & 1 \\
1 & 0 \\
\end{array}
\]
Review from Last Time

Logic Circuits

NOR Gate

Truth Table

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</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>
Logic Circuits

Review from Last Time

Truth Table

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>
Pull-up and Pull-down Networks

1. PU network comprised of p-channel device
2. PD network comprised of n-channel device
3. One and only one of these networks is conducting at the same time

\[ V_H = V_{DD} \quad V_L = 0 \quad (too\ good\ to\ be\ true?) \]
\[ P_H = P_L = 0 \quad (too\ good\ to\ be\ true?) \]
\[ t_{HL} = t_{LH} = 0 \quad (too\ good\ to\ be\ true?) \]

These characteristics are inherent in Boolean circuits with these 3 properties
Review from Last Time

Example 1:
How many transistors are required to realize the function

\[ F = \overline{A \cdot B} + \overline{A \cdot C} \]

in a basic CMOS process if static NAND and NOR gates are used? Assume A, B and C are available.

Solution:

20 transistors and 5 levels of logic
Example: XOR Function

\[ Y = A \bigoplus B \]

A widely-used 2-input Gate

Static CMOS implementation

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

22 transistors  5 levels of logic

Delays unacceptable and device count is too large!
Example

\[ Y = (A \cdot B) + (C \cdot D) \]

Standard Static CMOS Implementation

3 levels of Logic

16 Transistors if Basic CMOS Gates are Used
Observe:

Significant reduction in transistor count and levels of logic for realizing same Boolean function

Termed a “Complex Logic Gate” implementation
Complex Logic Gates

\[ Y = (A \cdot B) + (C \cdot D) \]
Complex Gates

Pull up and pull down network never both conducting

One of the two networks is always conducting
Complex Gates

Nomenclature:

When the logic gate shown is not a multiple-input NAND or NOR gate but satisfies Properties 1, 2, and 3 above, the gate will be referred to as a Complex Logic Gate.

Complex Logic Gates also implement static logic functions and some authors would refer to this as Static CMOS Logic as well but we will make the distinction and refer to this as “Complex Logic Gates”
Complex Gates

Complex Gate Design Strategy:

1. Implement $\overline{X}$ in the PDN

2. Implement $Y$ in the PUN (must complement the input variables since p-channel devices are used)
XOR in Complex Logic Gates

\[ Y = A \oplus B \]
XOR in Complex Logic Gates

\[ Y = A \oplus B \]

\[ Y = AB + \overline{AB} \]

\[ \overline{Y} = (\overline{AB} + \overline{AB}) \]

\[ \overline{Y} = \overline{AB} \cdot \overline{AB} \]

\[ Y = (\overline{A+B}) \cdot (A+B) \]
XOR in Complex Logic Gates

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

\[
Y = (\overline{A} + B) \cdot (A + \overline{B})
\]
XOR in Complex Logic Gates

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

\[ \overline{Y} = (\overline{A} + B) \cdot (A + \overline{B}) \]

12 transistors and 2 levels of logic

Notice a significant reduction in the number of transistors required
XOR in Complex Logic Gates

\[ Y = AB + \overline{AB} \]

\[ \overline{Y} = (\overline{A+B}) \cdot (A+B) \]

Multiple PU and PD networks can be used

\[ \overline{Y} = (\overline{A+B}) \cdot (A+B) \]

\[ = (\overline{A \cdot (A+B)}) + (B \cdot (A+B)) \]

\[ = (\overline{A \cdot B}) + (A \cdot B) \]
Complex Logic Gate Summary:

1. PU network comprised of p-channel device
2. PD network comprised of n-channel device
3. One and only one of these networks is conducting at the same time

Characteristics of PU/PD logic of this type:

- Rail to rail logic swings
- Zero static power dissipation in both Y=1 and Y=0 states
- Arbitrarily fast (too good to be true? will consider again with better model)
Consider \[ Y = A \cdot B \]

Standard CMOS Implementation

- 2 levels of Logic
- 6 Transistors if Basic CMOS Gates are Used

Basic noninverting functions generally require more complexity if basic CMOS gates are used for implementation.
Pass Transistor Logic

Requires only 2 transistors rather than 6 for a standard CMOS gate (and a resistor).

\[ Y = A \cdot B \]
Pass Transistor Logic

\[ Y = A \cdot B \]

Even simpler pass transistor logic implementations are possible

Requires only 1 transistor (and a resistor).
Pass Transistor Logic

Requires only 1 transistor (and a resistor)

- Pass transistor logic can offer significant reductions in complexity for some functions (particularly noninverting)
- Resistor may require more area than several hundred or even several thousand transistors
- Signal levels may not go to $V_{DD}$ or to 0V
- Static power dissipation may not be zero
- Signals may degrade unacceptably if multiple gates are cascaded
- “resistor” often implemented with a transistor to reduce area but signal swing and power dissipation problems still persist
- Pass transistor logic is widely used
Logic Design Styles

• Several different logic design styles are often used throughout a given design (3 considered thus far)
  – Static CMOS
  – Complex Logic Gates
  – Pass Transistor Logic

• The designer has complete control over what is placed on silicon and governed only by cost and performance

• New logic design strategies have been proposed recently and others will likely emerge in the future

• The digital designer needs to be familiar with the benefits and limitations of varying logic styles to come up with a good solution for given system requirements
MOSFET Modeling

• Simple model of MOSFET was developed
• Simple gates designed in CMOS Process were introduced
  – Some have zero power dissipation
  – Some have or appeared to have rail to rail logic voltage swings
  – All appeared to be Infinitely fast
  – Logic levels of some can not be predicted with simple model
  – Simple model is not sufficiently accurate to provide insight relating to some of these properties

• MOSFET modeling strategy
  – hierarchical model structure will be developed
  – generally use simplest model that can be justified
MOS Transistor Models

1. **Switch-Level model**

   ![Diagram of MOS Transistor Models](image)

   **Advantages:**
   - Simple, does not require understanding of semiconductor properties,
   - does not depend upon process, adequate for understanding basic
   - operation of many digital circuits

   **Limitations:**
   - Does not provide timing information (surfaced when looking at static
   - CMOS circuits, and several others that have not yet become apparent from
   - the applications that have been considered) and can not support design of
   - “resistor” used in Pass Transistor Logic
Recall

MOS Transistor
Qualitative Discussion of n-channel Operation

n-channel MOSFET
Recall

**MOS Transistor**

**Qualitative Discussion of n-channel Operation**

For $V_{GS}$ small

For $V_{GS}$ large
MOS Transistor
Qualitative Discussion of n-channel Operation

For $V_{GS}$ small

For $V_{GS}$ large
Improved Switch-Level Model

Switch-level model including gate capacitance and drain resistance

Still neglect bulk connection and connect the bulk capacitance to the source