Digital Systems – A preview

Static CMOS Gates
Other Logic Styles
Improved Device Models
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 ....
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:

$V_H = V_{DD}, \quad V_L = 0V$
If the random variable \( x \) in Normally distributed with mean \( \mu \) and standard deviation \( \sigma \), then \( y = \frac{x - \mu}{\sigma} \) is also a random variable that is Normally distributed with mean 0 and standard deviation of 1.
The random part of many parameters of microelectronic circuits is often assumed to be Normally distributed and experimental observations confirm that this assumption provides close agreement between theoretical and experimental results.

The mapping $y = \frac{x - \mu}{\sigma}$ is often used to simplify the statistical characterization of the random parameters in microelectronic circuits.
The Six-Sigma Challenge

Two-sided capability:

Long-term Capability
Tails are 6.8 parts in a million

Short-term Capability
Tail is 2 parts in a billion

Six Sigma Performance is Very Good !!!
MOS Transistor
Qualitative Discussion of n-channel Operation

Review from Last Time

Equivalent Circuit for n-channel MOSFET

This is the first model we have for the n-channel MOSFET!
MOS Transistor
Qualitative Discussion of p-channel Operation

Complete Symmetry in construction between Drain and Source
Review from Last Time

MOS Transistor
Qualitative Discussion of p-channel Operation

Equivalent Circuit for p-channel MOSFET

This is the first model we have for the p-channel MOSFET!
MOS Transistor
Comparison of Operation

![Diagram of MOS Transistor Operation]

- **G = 0**
- **G = 1**

**Gate**

**Drain**

**Source**
Logic Circuits

Review from Last Time

Circuit Behaves as a Boolean Inverter
Logic Circuits

Inverter

Truth Table

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

Review from Last Time

Truth Table

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>B</td>
<td>C</td>
</tr>
<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

NAND Gate

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>
Other logic circuits

• Other methods for designing logic circuits exist
• Insight will be provided on how other logic circuits evolve
• Several different types of logic circuits are often used simultaneously in any circuit design
Pull-up and Pull-down Networks

PU network comprised of p-channel device
PD network comprised of n-channel device
One and only one of these networks is conducting at the same time
Pull-up and Pull-down Networks

PU network comprised of p-channel devices
PD network comprised of n-channel devices
One and only one of these networks is conducting at the same time
Pull-up and Pull-down Networks

PU network comprised of p-channel device
PD network comprised of n-channel device
One and only one of these networks is conducting at the same time
Pull-up and Pull-down Networks

In these circuits, the PUN and PDN have the 3 interesting characteristics

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

What are \( V_H \) and \( V_L \)?
What is the power dissipation?
How fast are these logic circuits?
What are $V_H$ and $V_L$?
What is the power dissipation?
How fast are these logic circuits?

Consider the inverter

Use switch-level model for MOS devices
What are $V_H$ and $V_L$?
What is the power dissipation?
How fast are these logic circuits?

Consider the inverter

Use switch-level model for MOS devices

$V_H = V_{DD}$
$V_L = 0$
$I_D = 0$ thus $P_H = P_L = 0$
$t_{HL} = t_{LH} = 0$ (too good to be true?)
Pull-up and Pull-down Networks

For these circuits, the PUN and PDN have 3 interesting characteristics

Three key characteristics of Static CMOS Gates
1. PU network comprised of p-channel devices
2. PD network comprised of n-channel devices
3. One and only one of these networks is conducting at the same time

What are $V_H$ and $V_L$?
$V_H = V_{DD}$, $V_L = 0$ (too good to be true?)

What is the power dissipation?
$P_H = P_L = 0$ (too good to be true?)

How fast are these logic circuits?
$t_{HL} = t_{LH} = 0$ (too good to be true?)

These 3 properties are inherent in Boolean circuits with these 3 characteristics
Pull-up and Pull-down Networks

Three key characteristics of Static CMOS Gates
1. PU network comprised of p-channel devices
2. PD network comprised of n-channel devices
3. One and only one of these networks is conducting at the same time

Three properties of Static CMOS Gates (based upon simple switch-level model)
1. \( V_H = V_{DD}, V_L = 0 \) (too good to be true?)
2. \( P_H = P_L = 0 \) (too good to be true?)
3. \( t_{HL} = t_{LH} = 0 \) (too good to be true?)

These 3 properties are inherent in Boolean circuits with these 3 characteristics
Pull-up and Pull-down Networks

Concept can be extended to arbitrary number of inputs

n-input NOR gate

n-input NAND gate
Pull-up and Pull-down Networks

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

Concept can be extended to arbitrary number of inputs

n-input NOR gate

n-input NAND gate
Pull-up and Pull-down Networks

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

\[ V_{H} = V_{DD}, \quad V_{L} = 0 \]
\[ P_{H} = P_{L} = 0 \]
\[ t_{HL} = t_{LH} = 0 \]
In this class, logic circuits that are implemented by interconnecting multiple-input NAND and NOR gates will be referred to as “Static CMOS Logic”.

Since the set of NAND gates is complete, any combinational logic function can be realized with the NAND circuit structures considered thus far.

Since the set NOR gates is complete, any combinational logic function can be realized with the NOR circuit structures considered thus far.

Many logic functions are realized with “Static CMOS Logic” and this is probably the dominant design style used today!
Example 1:

How many transistors are required to realize the function

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

in a basic CMOS process if static NAND and NOR gates are used? Assume A, B and C are available.
Example 1:

How many transistors are required to realize the function

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

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 1:

How many transistors are required to realize the function

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

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

Solution (alternative): From basic Boolean Manipulations

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

\[ F = \overline{A} \cdot (1 + C) + B = \overline{A} + B \]

8 transistors and 3 levels of logic
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 (alternative):

From basic Boolean Manipulations

\[ F = \overline{A} \cdot (1 + C) + B = \overline{A} + B \]

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

6 transistors and 2 levels of logic
Example 2

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

Standard Static CMOS Implementation

3 levels of Logic

16 Transistors if Basic CMOS Gates are Used
Example 3: XOR Function

A widely-used 2-input Gate

Static CMOS implementation

\[ Y = A \oplus B \]

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

22 transistors    5 levels of logic

Delays unacceptable and device count is too large!
Consider again Example 2:

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

**Standard Static CMOS Implementation**

3 levels of Logic

16 Transistors if Basic CMOS Gates are Used

Can the same Boolean functionality be obtained with less transistors?
Observe:

![Logic Gate Diagram]

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

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

**Termed a “Complex Logic Gate” implementation**

Some authors term this a “compound gate”
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 has Characteristics 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 Gate Design Strategy:

1. Implement $\overline{Y}$ 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

$$\begin{align*}
Y &= A \oplus B \\
Y &= AB + \overline{AB} \\
\overline{Y} &= (AB + \overline{AB}) \\
\overline{Y} &= \overline{AB} \cdot \overline{\overline{B}} \\
\overline{Y} &= (\overline{A} + B) \cdot (A + \overline{B})
\end{align*}$$
XOR in Complex Logic Gates

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

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

12 transistors and 2 levels of logic

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

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

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

Multiple PU and PD networks can be used

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

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

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

If PUN and PDN satisfy the characteristics:

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

Properties of PU/PD logic of this type (with simple switch-level model):

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

\[ \begin{align*}
A & \quad \text{AND} \quad Y \\
B & \quad \text{OR} \quad Y
\end{align*} \]

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

\[ Y = A \cdot B \]

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

\[ Y = A \oplus B \]

6 transistors, 1 resistor, two levels of logic
Pass Transistor Logic

\[ Y = A \oplus B \]

2 transistors, 1 resistor, one level of logic
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
End of Lecture 5