Basic Logic Circuits
Complete Logic Family
Other Logic Styles
  − complex logic gates
  − pass transistor logic
Improved Device Models
Review from Last Time

The key patents that revolutionized the electronics field:

Jack Kilby (34 years old at invention)  
patent: 3,138,743  
Filed  Feb 6, 1959  
Issued  June 23, 1964

Robert Noyce (31 years old at invention)  
patent: 2,981,877  
Filed  July 30, 1959  
Issued  April 25, 1961
Basic Logic Circuits

- Semiconductor and Fabrication Technology
- Device Operation and Models
- CAD Tools
- Circuit Structures and Circuit Design
- Circuit Structures and Circuit Design
- Device Operation and Models
Basic Logic Circuits

• Will present a brief description of logic circuits based upon simple models and qualitative description of processes
• Will later discuss process technology needed to develop better models
• Will even later provide more in-depth discussion of logic circuits based upon better device models
Models of Devices

• Several models of the electronic devices will be introduced throughout the course
  – Complexity
  – Accuracy
  – Insight
  – Application

• Will use the simplest model that can provide acceptable results for any given application
MOS Transistor

Qualitative Discussion of n-channel Operation

Designer always works with top view
Complete Symmetry in construction between Drain and Source
MOS Transistor
Qualitative Discussion of n-channel Operation

Behavioral Description of Operation of n-channel MOS Transistors Created for use in Basic Digital Circuits

If $V_{GS}$ is large, short circuit exists between drain and source

If $V_{GS}$ is small, open circuit exists between drain and source
Boolean/Continuous Notation:

- Voltage Axis is Continuous between 0V and $V_{DD}$
- Boolean axis is discrete with only two points

Most logic circuits characterized by the relationship between the Boolean input/output variables though these correspond to voltage intervals on the continuous voltage axis
MOS Transistor
Qualitative Discussion of n-channel Operation

Equivalent Circuit for n-channel MOSFET

- Source assumed connected to (or close to) ground
- Boolean G at gate is relative to ground potential

This is the first model we have for the n-channel MOSFET!
MOS Transistor MODEL

Equivalent Circuit for n-channel MOSFET

Mathematically:

\[ I_D = 0 \quad \text{if } V_{GS} \text{ is low} \]
\[ V_{DS} = 0 \quad \text{if } V_{GS} \text{ is high} \]
MOS Transistor
Qualitative Discussion of p-channel Operation

Complete Symmetry in construction between Drain and Source
Behavioral Description of Operation of p-channel transistors created for use in basic digital circuits

If $V_{GS}$ is large (and negative), short circuit exists between drain and source

If $V_{GS}$ is small (near 0), open circuit exists between drain and source

MOS Transistor
Qualitative Discussion of p-channel Operation

p-channel MOSFET
MOS Transistor
Qualitative Discussion of p-channel Operation

This is the first model we have for the p-channel MOSFET!

Source assumed connected to (or close to) positive $V_{DD}$ and Boolean G at gate is relative to ground potential.

Equivalent Circuit for p-channel MOSFET
MOS Transistor MODEL

Equivalent Circuit for p-channel MOSFET

Mathematically:

\[ I_D = 0 \quad \text{if} \quad V_G \quad \text{is high} \quad (|V_{GSp}| \quad \text{is small})\]

\[ V_{DS} = 0 \quad \text{if} \quad V_G \quad \text{is low} \quad (|V_{GSp}| \quad \text{is large})\]
MOS Transistor
Comparison of Operation

Source assumed connected to (or close to) ground

Source assumed connected to (or close to) positive $V_{DD}$ and Boolean $G$ at gate is relative to ground
Logic Circuits

Circuit Behaves as a Boolean Inverter
Logic Circuits

Inverter

Truth Table

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

A = 1
B = 0
C

V_{DD}
Logic Circuits

\[
\begin{align*}
A &= 0 \\
B &= 1 \\
C &= 0
\end{align*}
\]
Logic Circuits

A = 1
B = 1
C
V_{DD}

A = 1
B = 1
C
V_{DD}

C = 0
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

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>
Logic Circuits

Approach can be extended to arbitrary number of inputs

n-input NOR gate

n-input NAND gate
Complete Logic Family

Family of $n$-input NOR gates forms a complete logic family

Family of $n$-input NAND gates forms a complete logic family

Having both NAND and NOR gates available is a luxury

Can now implement any combinational logic function !!

If add one flip flop, can implement any Boolean system !!

Flip flops easy to design but will discuss sequential logic systems later
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
PU network comprised of p-channel device and “tries” to pull B to VDD when conducting
PD network comprised of n-channel device and “tries to pull B to GND when conducting
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 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

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 these 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 key properties of these Static CMOS Gates
1. What are $V_H$ and $V_L$?
   $V_H=V_{DD}$, $V_L=0$ (too good to be true?)
2. What is the power dissipation?
   $P_H=P_L=0$ (too good to be true?)
3. How fast are these logic circuits?
   $t_{HL}=t_{LH}=0$ (too good to be true?)

These 3 properties are inherent in all Boolean circuits that have 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

\[
\begin{align*}
X_1 &\quad X_2 &\quad \ldots &\quad X_n &\quad Y \\
&\quad &\quad &\quad &\quad V_{DD}
\end{align*}
\]

n-input NAND gate

\[
\begin{align*}
X_1 &\quad X_2 &\quad \ldots &\quad X_n &\quad Y \\
&\quad &\quad &\quad &\quad V_{DD}
\end{align*}
\]
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

\[ V_H = V_{DD}, \quad V_L = 0 \]
\[ P_H = P_L = 0 \]
\[ t_{HL} = t_{LH} = 0 \]
Nomenclature

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

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

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

\[ \text{20 transistors and 5 levels of logic} \]
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

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

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

6 transistors and 2 levels of logic
Example 2: XOR Function

\[ Y = A \oplus B \]

A widely-used 2-input Gate

Static CMOS implementation

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

22 transistors 5 levels of logic

Delays unacceptable (will show later) and device count is too large!
Example 3: \[ Y = (A \cdot B) + (C \cdot D) \]

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:

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)

$(Y \text{ and } \overline{Y} \text{ often expressed in either SOP or POS form})$
XOR in Complex Logic Gates

\[ Y = A \oplus B \]

Will express \( \bar{Y} \) and \( Y \) in standard SOP or POS form
**XOR in Complex Logic Gates**

\[ Y = A \oplus B \]

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

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

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

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

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

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

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

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:

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

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).
Pass Transistor Logic

Even simpler pass transistor logic implementations are possible

Requires only 1 transistor (and a resistor).

Will see later that the area of a single practical resistor for this circuit may be comparable to that needed for hundreds or even thousands of transistors.
Pass Transistor Logic

- May be able to replace resistor with transistor (one of several ways shown)

- But high logic level can not be determined with existing device model (or even low logic level for circuit on right)

- Power dissipation can not be determined with existing device model for circuit on right

Better device model is needed (Power? Signal Swing? Speed?)
Pass Transistor Logic

6 transistors, 1 resistor, two levels of logic

(the 4 transistors in the two inverters are not shown)
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