Digital Circuits

- Basic Logic Gates
- Properties of Logic Families
- Characterization of CMOS Inverter
Digital Circuit Design

- Hierarchical Design
- Basic Logic Gates
- Properties of Logic Families
- Characterization of CMOS Inverter
  - Static CMOS Logic Gates
    - Ratio Logic
  - Propagation Delay
    - Simple analytical models
    - Elmore Delay
  - Sizing of Gates
- Propagation Delay with Multiple Levels of Logic
- Optimal driving of Large Capacitive Loads
- Power Dissipation in Logic Circuits
- Other Logic Styles
- Array Logic
- Ring Oscillators
Hierarchical Digital Design Domains:

- Behavioral:
- Structural:
- Physical

Multiple Levels of Abstraction
Hierarchical Analog Design Domains:

- Behavioral:
- Structural:
- Physical

Bottom Up Design  Top Down Design

Top  Bottom
Hierarchical Digital Design Domains:

**Behavioral**: Describes what a system does or what it should do.

**Structural**: Identifies constituent blocks and describes how these blocks are interconnected and how they interact.

**Physical**: Describes the constituent blocks to both the transistor and polygon level and their physical placement and interconnection.

Multiple representations often exist at any level or sublevel.
Representation of Digital Systems
Standard Approach to Digital Circuit Design

1. Behavioral Description
   - Technology independent
2. RTL Description
   (must verify (1) $\Leftrightarrow$ (2))
3. RTL Compiler
   Registers and Combinational Logic Functions
4. Logic Optimizer
5. Logic Synthesis
   Generally use a standard call library for synthesis
Representation of Digital Systems

Standard Approach to Digital Circuit Design

1. Behavioral Description
   - Technology independent

2. RTL Description
   (must verify (1) ⇔ (2))

3. RTL Compiler
   Registers and Combinational Logic Functions

4. Logic Optimizer

5. Logic Synthesis
   Generally use a standard call library for synthesis
6. Place and Route

(physically locates all gates and registers and interconnects them)

7. Layout Extraction
   • DRC
   • Back Annotation

8. Post Layout simulation

   May necessitate a return to a higher level in the design flow

Logic synthesis, though extensively used, often is not as efficient nor as optimal for implementing some important blocks or some important functions

These applications generally involve transistor level logic circuit design that may combine one or more different logic design styles
Logic Optimization

What is optimized (or minimized) ?

- Number of Gates
- Number or Levels of Logic
- Speed
- Delay
- Power Dissipation
- Area
- Cost
- Peak Current

 Depends Upon What User Is Interested In
Standard Cell Library

- Set of primitive building blocks that have been pre-characterized for dc and high frequency performance
- Generally includes basic multiple-input gates and flip flops
- P-cells often included
- Can include higher-level blocks
  - Adders, multipliers, shift registers, counters,…
- Cell library often augmented by specific needs of a group or customer
Logic Circuit Block Design

Many different logic design styles

- Static Logic Gates
- Complex Logic Gates
- Pseudo NMOS
- Pass Transistor Logic
- Dynamic Logic Gates
  - Domino Logic
  - Zipper Logic
  - Output Prediction Logic

Various logic design styles often combined in the implementation of one logic block
The basic logic gates

\[ X \rightarrow Y \quad Y = \overline{X} \]
\[ X \rightarrow Y \quad Y = X \]
\[ A \rightarrow Y \quad Y = A + B \]
\[ A \rightarrow Y \quad Y = A \cdot B \]
\[ A \rightarrow Y \quad Y = \overline{A + B} \]
\[ A \rightarrow Y \quad Y = \overline{A \cdot B} \]
\[ A \rightarrow Y \quad Y = A \oplus B \]
\[ A \rightarrow Y \quad Y = \overline{A \oplus B} \]
The basic logic gates

\[
\begin{align*}
X & \quad \quad Y \quad \quad Y = \bar{X} \\
X & \quad \quad Y \quad \quad Y = X \\
A & \quad \quad B \quad \quad Y = A + B \\
A & \quad \quad B \quad \quad Y = A \cdot B
\end{align*}
\]

\[
\begin{align*}
A & \quad \quad B \quad \quad Y = \bar{A} + B \\
A & \quad \quad B \quad \quad Y = \bar{A} \cdot B \\
A & \quad \quad B \quad \quad Y = A \oplus B \\
A & \quad \quad B \quad \quad Y = \bar{A} \oplus B
\end{align*}
\]

Question: How many basic one and two input gates exist and how many of these are useful?
The basic logic gates

The set of NOR gates is complete
Any combinational logic function can be realized with only multiple-input NOR gates

The set of NAND gates is complete
Any combinational logic function can be realized with only multiple-input NOR gates

Performance of the BASIC gates is critical!
The basic logic gates

A gate logic family can be formed based upon a specific design style for implementing logic functions

Many different gate logic family types exist

NMOS, PMOS, CMOS, TTL, ECL, RTL, DCTL,…

Substantial differences in performance from one family type to another

Power, Area, Noise Margins, ….
The basic logic gates

The characteristics of any gate logic family can be expressed rather simply in terms of the characteristics of the basic inverter in that logic family.
The basic logic gates

It suffices to characterize the inverter of a logic family and then express the performance of other gates in that family in terms of the performance of the inverter.
The basic logic gates

It suffices to characterize the inverter of a logic family and then express the performance of other gates in that family in terms of the performance of the inverter.

What characteristics are required and desirable for an inverter to form the basis for a useful logic family?
What restrictions are there on the designer for building Boolean circuits?

• None !!!!

• It must “work” as expected

• Designer is Master of the silicon !
Desirable and/or Required Logic Family Characteristics

What are the desired characteristics of a logic family?
Desirable and/or Required Logic Family Characteristics

1. High and low logic levels must be uniquely distinguishable (even in a long cascade)
2. Capable of driving many loads (good fanout)
3. Fast transition times (but in some cases, not too fast)
4. Good noise margins (low error probabilities)
5. Small die area
6. Low power consumption
7. Economical process requirements
Desirable and/or Required Logic Family Characteristics

8. Minimal noise injection to substrate
9. Low leakage currents
10. No oscillations during transitions
11. Compatible with synthesis tools
12. Characteristics do not degrade too much with temperature
13. Characteristics do not vary too much with process variations

Are some of these more important than others?
Desirable and/or Required Logic
Family Characteristics

8. Minimal noise injection to substrate
9. Low leakage currents
10. No oscillations during transitions
11. Compatible with synthesis tools
12. Characteristics do not degrade too much with temperature
13. Characteristics do not vary too much with process variations

Are some of these more important than others?

Yes! – must have well-defined logic levels for circuits to even function as logic
Desirable and/or Required Logic Family Characteristics

Are some of these more important than others?

Yes! – must have well-defined logic levels for circuits to even function as logic.

What properties of an inverter are necessary for it to be useful for building a logic family?

What are the logic levels for a given inverter of a given logic family?
What are the logic levels for a given inverter of a given logic family?

\[ V_H = ? \quad V_L = ? \]

Can we legislate them?

- Some authors choose to simply define a value for them
- Simple and straightforward approach
- But what if the circuit does not interpret them the same way they are defined!!
What are the logic levels for a given inverter of for a given logic family?

\[ V_H = ? \quad V_L = ? \]

**Can we legislate them?**

- Some authors choose to define them based upon specific features of inverter
- Analytical expressions may be complicated
- But what if the circuit does not interpret them the same way they are defined!!
What are the logic levels for a given inverter of for a given logic family?

\[ V_H = ? \quad V_L = ? \]

Ask the inverter how it will interpret logic levels
- The circuit will interpret them the way they are defined !!
- Analytical expressions may be complicated
- How is this determination made?
Ask the inverter how it will interpret logic levels

\[ \text{VIN} \rightarrow \text{VOUT} \]

\[ V_H=? \]

\[ V_L=? \]

Consider a very long cascade of inverters

Apply a large voltage at the input (alternatively a small input could be used)

w.l.o.g. assume an even number of inverters in chain indicated

\[ \text{VLARGE} \rightarrow \text{V?} \rightarrow \text{V?} \rightarrow \cdots \rightarrow \text{V?} \rightarrow \text{V?} \rightarrow \cdots \rightarrow \text{V?} \]

even number of stages
Ask the inverter how it will interpret logic levels

\[ V_{\text{IN}} \rightarrow \text{INVERTER} \rightarrow V_{\text{OUT}} \]

\[ V_H=? \]

\[ V_L=? \]

Consider a very long cascade of inverters

Apply a large voltage at the input (alternatively a small input could be used)

w.l.o.g. assume an even number of inverters in chain indicated

If logic levels are to be maintained, the voltage at the end of this even number of stages must be \( V_H \), that of the next must be \( V_L \), the next \( V_H \), etc. until the end of the cascade is reached
Ask the inverter how it will interpret logic levels

\[ V_{\text{IN}} \rightarrow \frac{V_{\text{OUT}}}{V_H=? \quad V_L=?} \]

VIN VOUT
VLARGE
VH VHVL

\[ V_{\text{LARGE}} \rightarrow \cdots \rightarrow \frac{V_{\text{OUT}}}{V_H \quad V_L \quad V_H \quad \cdots} \]

\[ S_1 \quad S_2 \quad S_3 \]

VH VH VH
Ask the inverter how it will interpret logic levels

$V_H = ?$

$V_L = ?$
Ask the inverter how it will interpret logic levels

- Two inverter loop
- Very useful circuit!
The two-inverter loop

SRAM Cell

S₂ often eliminated (shorted)
The two-inverter loop

S\_2 \text{ often eliminated (shorted)}

Standard 6-transistor SRAM Cell
Ask the inverter how it will interpret logic levels

Thus, consider the inverter pair

Thus, consider the inverter pair

$V_{IN}$  $V_{OUT}$  $V'_{OUT}$
Ask the inverter how it will interpret logic levels

$V_{IN} \rightarrow V_{OUT} \rightarrow V'_{OUT}$

Inverter pair

$V_H$ and $V_L$ will be on the intersection of the transfer characteristics of the inverter pair (IPTC) and the $V'_{OUT}=V_{IN}$ line

$V_H$ and $V_L$ often termed the “1” and “0” states
Ask the inverter how it will interpret logic levels

V\textsubscript{IN} \rightarrow V\textsubscript{OUT} \rightarrow V'\textsubscript{OUT}

Inverter pair

When $V'\text{OUT}=V\text{IN}$, $V_H$ and $V_L$ are stable operating points, $V_{\text{TRIP}}$ is a quasi-stable operating point.

Observe: slope of IPTC is greater than 1 at $V_{\text{TRIP}}$ and less than 1 at $V_H$ and $V_L$. 
Observation

When $V_{\text{OUT}} = V_{\text{IN}}$ for the inverter, $V'_{\text{OUT}}$ is also equal to $V_{\text{IN}}$. Thus the intersection point for $V_{\text{OUT}} = V_{\text{IN}}$ in the inverter transfer characteristics (ITC) is also an intersection point for $V'_{\text{OUT}} = V_{\text{IN}}$ in the inverter-pair transfer characteristics (IPTC).
Logic Family Characteristics

What properties of an inverter are necessary for it to be useful for building a two-level logic family?

What are the logic levels for a given inverter of for a given logic family?
Logic Family Characteristics

What properties of an inverter are necessary for it to be useful for building a two-level logic family?

The inverter-pair transfer characteristics must have three unique intersection points with the $V'_{OUT} = V_{IN}$ line.

What are the logic levels for a given inverter of for a given logic family?

The two extreme intersection points of the inverter-pair transfer characteristics with the $V'_{OUT} = V_{IN}$ line.
Logic Family Characteristics

The inverter-pair transfer characteristics must have three unique intersection points with the \( V'_{\text{OUT}} = V_{\text{IN}} \) line.

The two extreme intersection points of the inverter-pair transfer characteristics with the \( V'_{\text{OUT}} = V_{\text{IN}} \) line are \( V_H \) and \( V_L \).

Can we legislate \( V_H \) and \( V_L \) for a logic family?
Logic Family Characteristics

What properties of an inverter are necessary for it to be useful for building a two-level logic family?

The inverter-pair transfer characteristics must have three unique intersection points with the $V'_{\text{OUT}} = V_{\text{IN}}$ line.

What other properties of the inverter are desirable?

Reasonable separation between $V_H$ and $V_L$ (enough separation so that noise does not cause circuit to interpret level incorrectly)

$$V_{\text{TRIP}} \approx \frac{V_H + V_L}{2}$$ (to provide adequate noise immunity and process insensitivity)
What happens near the quasi-stable operating point?

$S_2$ closed and $X=Y=V_{TRIP}$
What happens near the quasi-stable operating point?

S₂ closed and X=Y=V_{TRIP}

If X decreases even very slightly, will move to the X=0, Y=1 state (very fast)

If X increases even very slightly, will move to the X=1, Y=0 state (very fast)
End of Lecture 34
What if the inverter pair had the following transfer characteristics?
What if the inverter pair had the following transfer characteristics?

- Multiple levels of logic
- Every intersection point with slope $<1$ is a stable point
- Every intersection point with slope $>1$ is a quasi-stable point
What are the transfer characteristics of the static CMOS inverter pair?

Consider first the inverter

\[ V_{IN} \stackrel{\text{INV}}{\longrightarrow} V' \]
Transfer characteristics of the static CMOS inverter
Transfer characteristics of the static CMOS inverter
(Neglect λ effects)

Case 1   \( M_1 \) triode, \( M_2 \) cutoff

\[
I_{D1} = \mu_n C_{\text{oxn}} \frac{W_1}{L_1} \left( V_{IN} - V_{Tn} - \frac{V_{OUT}}{2} \right) V_{OUT}
\]

\( I_{D2} = 0 \)

Equating \( I_{D1} \) and \(-I_{D2}\) we obtain:

\[
0 = \mu_n C_{\text{oxn}} \frac{W_1}{L_1} \left( V_{IN} - V_{Tn} - \frac{V_{OUT}}{2} \right) V_{OUT}
\]

It can be shown that setting the first product term to 0 will not verify, thus

\[
V_{OUT} = 0
\]

valid for:

\[
V_{GS1} \geq V_{Tn} \quad V_{DS1} < V_{GS1} - V_{Tn} \quad V_{GS2} \geq V_{Tp}
\]

thus, valid for:

\[
V_{IN} \geq V_{Tn} \quad V_{OUT} < V_{IN} - V_{Tn} \quad V_{IN} - V_{DD} \geq V_{Tp}
\]
\[ V_{GS1} \geq V_{Tn} \quad V_{DS1} < V_{GS1} - V_{Tn} \quad V_{GS2} \geq V_{Tp} \]

\[ V_{IN} \geq V_{Tn} \quad V_{OUT} < V_{IN} - V_{Tn} \quad V_{IN} - V_{DD} \geq V_{Tp} \]
Transfer characteristics of the static CMOS inverter
(Neglect \( \lambda \) effects)

Case 1 \( M_1 \) triode, \( M_2 \) cutoff

\[ V_{\text{OUT}} = 0 \]
Transfer characteristics of the static CMOS inverter
(Neglect $\lambda$ effects)

Case 1  $M_1$ triode, $M_2$ cutoff

$V_{\text{OUT}} = 0$

![Graph showing the transfer characteristics of a static CMOS inverter with cases defined for $V_{\text{IN}}$, $V_{\text{OUT}}$, $V_{\text{DD}}$, $V_{\text{Tn}}$, and $V_{\text{Tp}}$.]
Transfer characteristics of the static CMOS inverter
(Neglect \( \lambda \) effects)
Partial solution:
Transfer characteristics of the static CMOS inverter
(Neglect λ effects)

Case 2   $M_1$ triode, $M_2$ sat
Transfer characteristics of the static CMOS inverter
(Neglect λ effects)

Case 1  \( M_1 \) triode, \( M_2 \) cutoff

\( V_{\text{OUT}} = 0 \)
Transfer characteristics of the static CMOS inverter
(Neglect $\lambda$ effects)
Partial solution:
Transfer characteristics of the static CMOS inverter
(Neglect $\lambda$ effects)
Partial solution:

The rest of the transfer characteristics will be obtained in the next lecture