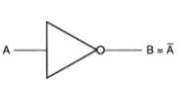
EELE 414 – Introduction to VLSI Design

Module #5 –Inverters

- **Agenda**
  1. Inverters
     - Static Characteristics
     - Switching Characteristics
- **Announcements**
  1. Read Chapters 5, & 7

Inverters

- **Inverters**
  - an inverter is a basic gate that complements the input



Symbol

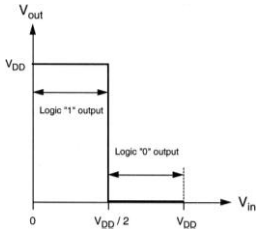
A	B
0	1
1	0

Truth Table

  - we study the invert in order to understand the Static and Dynamic performance
  - once we do this, we can model more complex logic gates as "equivalent inverters" and use the same analysis.

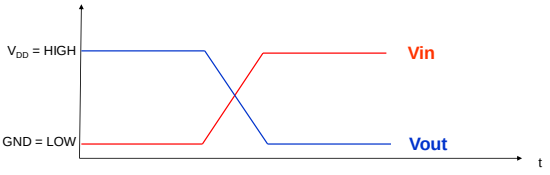
Inverters

- **Inverters**
  - The "Voltage Transfer Characteristics" (VTC) of an ideal inverter



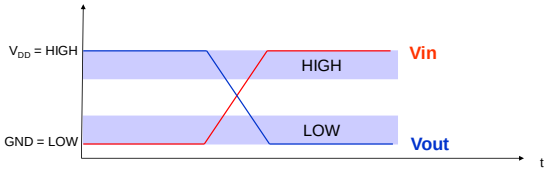
Inverters

- **Inverters**
  - Graphically, this looks like:



Inverters

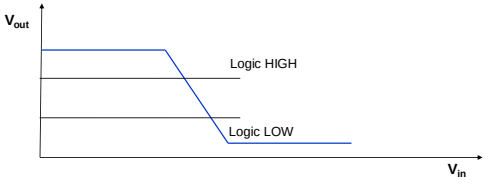
- **Logic Levels**
  - We need to define boundaries when the signal is considered HIGH or LOW
  - these are called the "Logic Levels"



  - what is the logic level in the middle region? It is unknown…

Inverter Static Behavior

- **Static Behavior**
  - "Static" or "DC" refers to the gate's operation when the inputs are NOT changing
  - also called "Steady State"
  - if we plotted Vout vs. Vin of an Inverter, we would get…

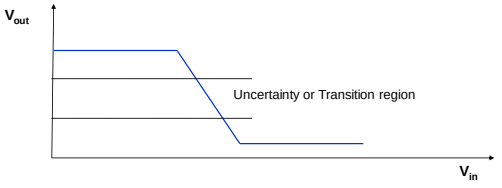


Inverter Static Behavior

- **Static Behavior**

  - the region in the middle is not definitely a HIGH or a LOW because of:

    - Power Supply Variation
    - Process
    - Noise



Inverter Static Behavior

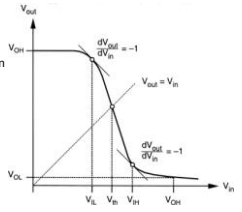
- **DC Specifications**

  - we need to be able to guarantee operation of the gate over all possible conditions
  - the limits on guaranteed operation are called "specifications"
  - Specifications can give limits on the worst case situations
  - Specifications can also give limits on typical situations

Inverter Static Behavior

- **DC Specifications**

  - in a real inverter VTC, the output doesn't switch instantaneously
  - there are two critical points on the real VTC curve which occur when the slope of $V_{out}(V_{in})$= -1
    - **V_{IL}** is the input low voltage which corresponds to an output high voltage with a slope of -1.
    - **V_{IH}** is the input high voltage which corresponds to an output low voltage with a slope of -1
  - other critical points are:
    - **V_{OH}** is the output voltage when the output level is logic "1"
    - **V_{OL}** is the output voltage when the output level is logic "0"
    - **V_{th}** is the point at which V_{out}=V_{in}

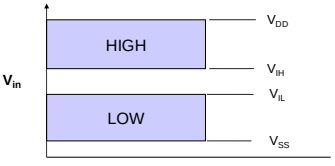


Inverter Static Behavior

- **DC Input Specifications**

  V_{IH} : Minimum input voltage guaranteed to be recognized as a HIGH (aka V_{IHmin})

  V_{IL} : Maximum input voltage guaranteed to be recognized as a LOW (aka V_{ILmax})

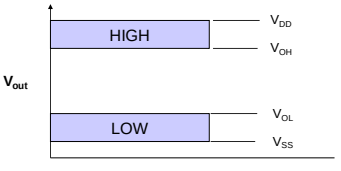


Inverter Static Behavior

- **DC Output Specifications**

  V_{OH} : Minimum output voltage guaranteed when driving a HIGH (aka V_{OHmin})

  V_{OL} : Maximum output voltage guaranteed when driving a LOW (aka V_{OLmax})

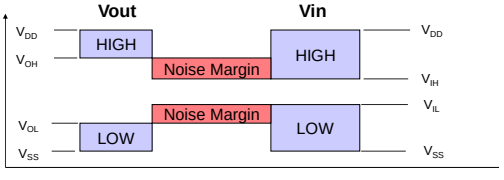


Inverter Static Behavior

- **DC Noise Margins (NM)**

  HIGH State Noise Margin : $(NM_H) = (V_{OH} - V_{IH}) = (V_{OHmin} - V_{IHmin})$

  LOW State Noise Margin : $(NM_L) = (V_{IL} - V_{OL}) = (V_{ILmax} - V_{OLmax})$



Inverter Static Behavior

- **DC Power Specifications**

  - the total DC power dissipated by an IC is given by:

$$P_{DC} = V_{DD} \cdot I_{DC}$$

  - for a given gate, the current drawn will vary depending on the logic level

    Driving a Logic HIGH: $I_{DC1}(V_{in} = low)$

    Driving a Logic LOW: $I_{DC2}(V_{in} = high)$

  - the gate will be in each one of these states 50% of the time

  - if we assume the output voltage will swing from 0 to V_{DD}, we can estimate the average output voltage as $V_{DD}/2$

  - a rough **estimate** of the DC power is: $P_{DC} = \frac{V_{DD}}{2} \cdot [I_{DC}(V_{in} = low) + I_{DC}(V_{in} = high)]$

Inverter Static Behavior

- **Area**

  - as designers, we can adjust the sizes of L and W.

  - we know that there is additional area required to fabricate the MOSFET

    - active regions (surrounding FOX)
    - channel Length (Y)
    - substrate contacts

  - but, as a practical measure, we talk about the area of a circuit as **W·L**

  - while we know this isn't the full area that the device takes, it gives us a standard way to compare the sizes of different layouts.

  - it is widely accepted that the area of a device is **W·L**

Inverter Design

- **Inverter Implementations**

  - now we turn our attention to the circuit level implementation of the inverter

  - there are many ways to create an inverter using MOSFETs

    1) inverter with resistive-load
    2) inverter with enhancement n-Type MOSFET load operating in the linear region
    3) inverter with enhancement n-Type MOSFET load operating in the saturation region
    4) inverter with depletion n-Type MOSFET load
    5) CMOS inverter

  - the most common type of inverter in VLSI is CMOS. This is due to the low static power consumption

  - however, it is worth while to briefly look at other types of inverter implementations in case you use a fab that doesn't have PMOS

  - for example, the Montana Microfabrication Facility (MMF)

    - no N-Well & PMOS

  - BUT, we can still design inverters using different circuit styles.

Resistive-Load Inverter

- **Resistive-Load Inverter**

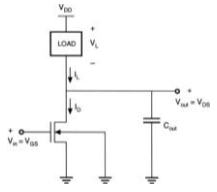
  - this circuit consists of an enhancement-type, N-Channel MOSFET as the *driver*

  - a load resistor is connected between V_{DD} and the Drain (Vout) of the MOSFET

  - the gates that this inverter drives are assumed to be of the same configuration so there is no DC load current looking into their gate terminals.

  - $V_{out} = V_{DS}$

  - $V_{in} = V_{gs}$

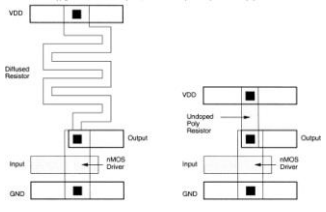


Resistive-Load Inverter

- **Resistive-Load Inverter**

  - we need a relatively high resistor (kΩ) so we can implement the resistor using either a diffused or undoped-poly resistor

  - this resistor takes a large amount of die area to implement



Resistive-Load Inverter

- **Resistive-Load Inverter**

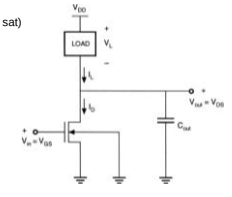
  - we solve for $V_{out}(V_{in})$ using KVL where:

$$V_{out} = V_{DD} - R_L \cdot I_R$$

$$I_R = I_{DS} = \frac{V_{DD} - V_{out}}{R_L}$$

  - we solve for V_{OH} and V_{OL}

    - applying $V_{in}=V_{GS}$=logic "0" or "1"
    - determining the mode of operation (cut-off, linear, sat)
    - creating an equation relating I_{DS}, V_{out}, and V_{in}



Resistive-Load Inverter

- **Resistive-Load Inverter**

  - we solve for V_{IH} and V_{IL}

    - applying $V_{in}=V_{GS}$=logic "0" or "1"
    - determining the mode of operation (cut-off, linear, sat)
    - creating an equation relating I_{DS}, V_{out}, and V_{in}

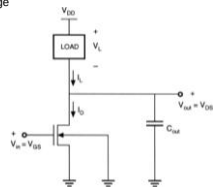
  - remember that VIH and VIL are defined as the input voltage when the output has a slope of -1

  - then we need to:

    - differentiate the equation with respect to V_{in}
    - plug in (dV_{out}/dV_{in})=-1
    - solve for V_{IH} or V_{IL}

  - note that the solution will be quadratic (i.e., have two solutions). We pick the logical solution

    i.e., the smaller solution for V_{IL}
    and
    the larger solution for V_{IH}



Resistive-Load Inverter

- **Resistive-Load Inverter**

  - these solutions yield:

$$V_{OH} = V_{DD}$$

$$V_{OL} = V_{DD} - V_{T0} + \frac{1}{k_n \cdot R_L} - \sqrt{\left(V_{DD} - V_{T0} + \frac{1}{k_n \cdot R_L}\right)^2 - \frac{2 \cdot V_{DD}}{k_n \cdot R_L}}$$

$$V_{IH} = V_{TO} + \sqrt{\frac{8}{3} \cdot \frac{V_{DD}}{k_n \cdot R_L}} - \frac{1}{k_n \cdot R_L}$$

$$V_{IL} = V_{T0} + \frac{1}{k_n \cdot R_L}$$

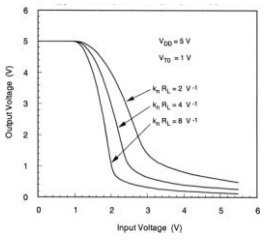
Resistive-Load Inverter

- **Resistive-Load Inverter**

  - notice that these solutions only depend on $k_n \cdot R_L$

  - we have control over W & L, which alters k_n

  - we have control over R_L by altering the shape of the resistor



Active-Load Inverter

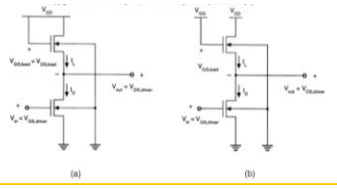
- **Inverter with Enhancement-Type NMOS Load**

  - the resistive-load inverter takes a lot of chip area due to the resistor which makes it impractical for VLSI

  - another way to implement the load is to use an enhancement-type NMOS transistor

  - this gives a load that takes less area

  - this topology can have the load either in the linear or saturation region depending on how it is biased



Active-Load Inverter

- **Inverter with Depletion-Type NMOS Load**

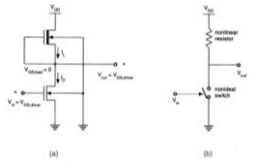
  - the enhancement-type NMOS load has the drawback of a larger DC current when not switching.

  - this power consumption make it less than ideal for VLSI

  - another technique is to use a depletion-type NMOS load

  - this gives a sharper VTC curve and better noise margin

  - however, an additional process step is required to create the depletion-type device



CMOS Inverter

- **CMOS Inverter**

  - the CMOS inverter uses an NMOS and a PMOS transistor in a complementary push/pull configuration

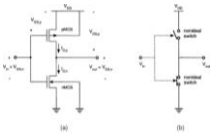
  - for a Logic "1" output, the PMOS=ON and the NMOS=OFF

  - for a Logic "0" output, the PMOS=OFF and the NMOS=ON

  - this configuration has two major advantages:

    1) low static power consumption : due to one MOSFET always being off

    2) a sharp and symmetric VTC profile giving full swing signals (1=V_{DD}, 0=V_{SS})

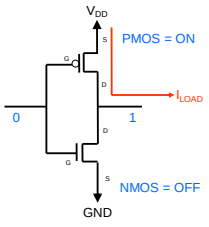


CMOS Inverter

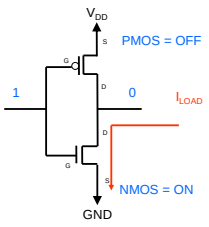
- **CMOS Inverter**

  - basic operation, complementary switches

Input = 0



Input = 1



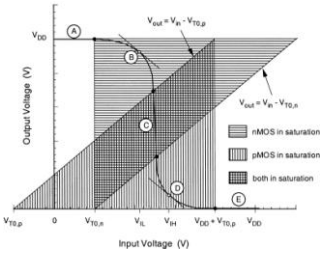
CMOS Inverter

- **CMOS Inverter Static Behavior**

  - let's start the Static Analysis by describing the regions of operation as the Inverter Switches

  - Remember that:

$$V_{GS,n} = V_{in}$$
$$V_{DS,n} = V_{out}$$
$$V_{GS,p} = -(V_{DD} - V_{in})$$
$$V_{DS,p} = -(V_{DD} - V_{out})$$

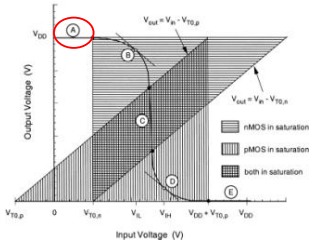


CMOS Inverter

- **CMOS Inverter Static Behavior**

Region A

- let's assume V_{DD}=5v, $V_{T,n}$=1, $V_{T,p}$= -1 (V_{in} = 0v, V_{out} = 5v)
- When V_{in} = 0v, the output is V_{out}=V_{DD}
- the NMOS transistor is OFF since $V_{GS,n} \leq V_{T,n}$ (**cut-off**) i.e., 0 ≤ 1
- the NMOS drain current $I_{D,n}$=0
- the PMOS transistor is ON since $V_{GS,p} \leq V_{T,p}$ i.e., (0-5) ≤ -1
- the PMOS drain current $I_{D,p}$=0 since $I_{D,n}$=$I_{D,p}$
- since $V_{DS,p}$=0v, then the PMOS is in the ***linear*** *region* since: $V_{DS,p} > (V_{GS,p}-V_{T,p})$ i.e., 0 > (0-5) - (-1)

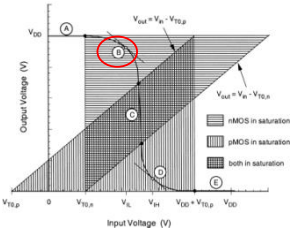


CMOS Inverter

- **CMOS Inverter Static Behavior**

Region B

- Now let's move V_{in} above $V_{T,n}$ but below V_{th} (V_{in} ~= 1v, V_{out} ~=5v)
- the NMOS transistor turns ON since $V_{GS,n} \geq V_{T,n}$, i.e., 1 ≥ 1
- since $V_{DS,n}$ is still near V_{DD}, the NMOS goes directly into ***saturation*** since: $V_{DS,n} > (V_{GS,n}-V_{T,n})$ i.e., (~5) > 1-1
- the PMOS transistor is still ON since $V_{GS,p} \leq V_{T,p}$ i.e., ~(1-5) ≤ -1
- the PMOS is still in the ***linear region*** since: $V_{DS,p} > (V_{GS,p}-V_{T,p})$ i.e., (~5-5) > (1-5) - (-1)

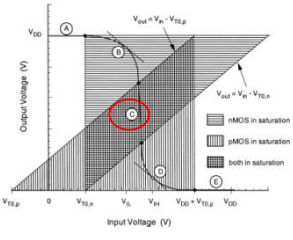


CMOS Inverter

- **CMOS Inverter Static Behavior**

Region C

- Now let's move to where V_{in} = V_{out} (V_{in} ~= 2.5v, V_{out} ~=2.5v)
- This is defined as V_{th}
- the NMOS transistor is ON since $V_{GS,n} \geq V_{T,n}$ i.e., 2.5 ≥ 1
- the NMOS transistor is in ***saturation*** since $V_{DS,n} > (V_{GS,n}-V_{T,n})$ i.e., ~2.5 > (2.5 - 1)
- the PMOS transistor is ON since $V_{GS,p} \leq V_{T,p}$ i.e., (2.5-5) ≤ -1
- the PMOS is in ***saturation*** since: $V_{DS,n} < (V_{GS,n}-V_{T,p})$ i.e., (2.5-5) < (2.5-5) - (-1)

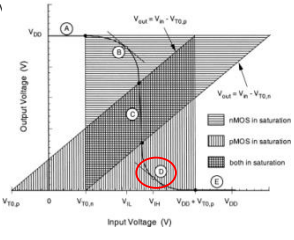


CMOS Inverter

- **CMOS Inverter Static Behavior**

Region D

- Now let's move V_{in} above V_{th} but below (V_{DD}+|$V_{T,p}$|) (V_{in} ~= 4v, V_{out} ~=1v)
- the NMOS transistor is ON since $V_{GS,n} \geq V_{T,n}$ i.e., 4 ≥ 1
- the NMOS transistor is in ***linear*** since $V_{DS,n} < (V_{GS,n}-V_{T,n})$ i.e., ~1 < (4 - 1)
- the PMOS transistor is ON since $V_{GS,p} \leq V_{T,p}$ i.e., (4-5) ≤ -1
- the PMOS is in ***saturation*** since: $V_{DS,n} < (V_{GS,n}-V_{T,p})$ i.e., (1-5) < (4-5) - (-1)

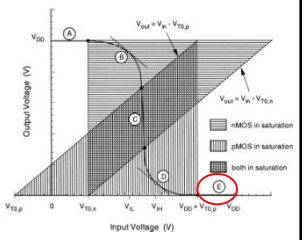


CMOS Inverter

- **CMOS Inverter Static Behavior**

Region E

- Now let's move V_{in} above ($V_{DD}+V_{T,p}$) (V_{in} = 5v, V_{out} = 0v)
- the NMOS transistor is ON since $V_{GS,n} \geq V_{T,n}$ i.e., $5 \geq 1$
- the NMOS transistor is in ***linear*** since $V_{DS,n} < (V_{GS,n}-V_{T,n})$ i.e., ~0 < (5 - 1)
- the PMOS transistor is OFF since $V_{GS,p} \geq V_{T,p}$ i.e., (5-5) $\geq$ -1 (***cut-off***)

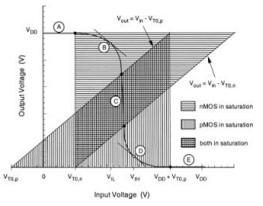


CMOS Inverter

- **CMOS Inverter Static Behavior**

Summary

Region	NMOS	PMOS
A	cut-off	linear
B	saturation	linear
C	saturation	saturation
D	linear	saturation
E	linear	cut-off



CMOS Inverter

- **CMOS Inverter Static Behavior (V_{OH} & V_{OL})**

- Now let's calculate the static operating specifications
- V_{OH} and V_{OL} are trivial since $I_{D,p} = I_{D,n} = 0A$ in both cases
- this condition gives a full output swing across the complementary structure:

$$V_{OH} = V_{DD}$$
$$V_{OL} = V_{SS}$$

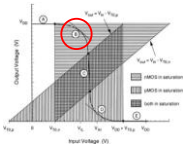
- Note that V_{DD} is typically the power supply and V_{SS} is typically GND.

CMOS Inverter

- **CMOS Inverter Static Behavior (V_{IL})**

- V_{IL} is defined as the input voltage that corresponds to the higher of the two output voltages with a slope of -1.
- we know the modes of operation for the transistors in this region:

NMOS = saturation
PMOS = linear



- we also know from KCL that $I_{D,p} = I_{D,n}$
- from this, we can write our first current equation:

$$I_{D,n(sat)} = I_{D,p(lin)}$$
$$\frac{k_n}{2}\cdot(V_{GS,n}-V_{T0,n})^2 = \frac{k_p}{2}\cdot\left[2\cdot(V_{GS,p}-V_{T0,p})\cdot V_{DS,p} - V_{DS,p}^2\right]$$

CMOS Inverter

- **CMOS Inverter Static Behavior (V_{IL}) cont…**

- remembering the relationships between V_{in} & V_{out}, and V_{GS} & V_{DS}:

$$V_{GS,n} = V_{in}$$
$$V_{DS,n} = V_{out}$$
$$V_{GS,p} = -(V_{DD}-V_{in}) = V_{in}-V_{DD}$$
$$V_{DS,p} = -(V_{DD}-V_{out}) = V_{out}-V_{DD}$$

- we can write:

$$\frac{k_n}{2}\cdot(V_{in}-V_{T0,n})^2 = \frac{k_p}{2}\cdot\left[2\cdot(V_{in}-V_{DD}-V_{T0,p})\cdot(V_{out}-V_{DD}) - (V_{out}-V_{DD})^2\right]$$

CMOS Inverter

- **CMOS Inverter Static Behavior (V_{IL}) cont…**

- we are looking for when the derivative of $dV_{out}/dV_{in} = -1$ so we differentiate both sides:

$$\frac{d}{dV_{in}}\left(\frac{k_n}{2}\cdot(V_{in}-V_{T0,n})^2\right) = \frac{d}{dV_{in}}\left(\frac{k_p}{2}\cdot\left[2\cdot(V_{in}-V_{DD}-V_{T0,p})\cdot(V_{out}-V_{DD}) - (V_{out}-V_{DD})^2\right]\right)$$

- the left-hand-side is straight forward to perform a partial derivative on (with respect to Vin), but the right-hand side consists of two products that must be differentiated using the product rule.

Remember the product rule:

$$Z = f(x,y)\cdot g(x,y)$$
$$\frac{dZ}{dx} = \frac{df}{dx}\cdot g(x,y) + \frac{dg}{dx}\cdot f(x,y)$$

CMOS Inverter

- **CMOS Inverter Static Behavior (V_{IL}) cont…**

- let's re-write the RHS as the sum of two products:

$$\frac{d}{dV_{in}}\left(\frac{k_n}{2}\cdot\left(V_{in}-V_{T0,n}\right)^2\right)=\frac{d}{dV_{in}}\left(\frac{k_p}{2}\cdot\left[2\cdot\left(V_{in}-V_{DD}-V_{T0,p}\right)\cdot\left(V_{out}-V_{DD}\right)\right]-\left(V_{out}-V_{DD}\right)\cdot\left(V_{out}-V_{DD}\right)\right]$$

Product #1 Product #2

- the left-hand-side is straight forward to perform a partial derivative on (with respect to Vin),

$$LHS=k_n\cdot\left(V_{in}-V_{T0,n}\right)$$

CMOS Inverter

- **CMOS Inverter Static Behavior (V_{IL}) cont…**

- now let's perform a partial derivative on the RHS (with respect to V_{in}) using the product rule:

$$RHS=\frac{d}{dV_{in}}\left(\frac{k_p}{2}\cdot\left[2\cdot\left(V_{in}-V_{DD}-V_{T0,p}\right)\cdot\left(V_{out}-V_{DD}\right)\right]-\left(V_{out}-V_{DD}\right)\cdot\left(V_{out}-V_{DD}\right)\right]$$

$\frac{df}{dx}\cdot g(x,y)+\frac{dg}{dx}\cdot f(x,y)$ $\frac{df}{dx}\cdot g(x,y)+\frac{dg}{dx}\cdot f(x,y)$

$$RHS=\frac{k_p}{2}\cdot\left[2\cdot\left(\left(1\right)\cdot\left(V_{out}-V_{DD}\right)+\left(V_{in}-V_{DD}-V_{T0,p}\right)\cdot\left(\frac{dVout}{dVin}\right)\right)-\left(\left(\frac{dVout}{dVin}\right)\cdot\left(V_{out}-V_{DD}\right)+\left(V_{out}-V_{DD}\right)\cdot\left(\frac{dVout}{dVin}\right)\right)\right]$$

$$RHS=\frac{k_p}{2}\cdot\left[2\cdot\left(\left(1\right)\cdot\left(V_{out}-V_{DD}\right)+\left(V_{in}-V_{DD}-V_{T0,p}\right)\cdot\left(\frac{dVout}{dVin}\right)\right)-\left(2\cdot\left(V_{out}-V_{DD}\right)\cdot\left(\frac{dVout}{dVin}\right)\right)\right]$$ Combine two like expressions

$$RHS=k_p\left[\left(\left(V_{out}-V_{DD}\right)+\left(V_{in}-V_{DD}-V_{T0,p}\right)\cdot\left(\frac{dVout}{dVin}\right)\right)-\left(\left(V_{out}-V_{DD}\right)\cdot\left(\frac{dVout}{dVin}\right)\right)\right]$$ Pull out 2

$$RHS=k_p\cdot\left[\left(V_{in}-V_{DD}-V_{T0,p}\right)\cdot\left(\frac{dVout}{dVin}\right)+\left(V_{out}-V_{DD}\right)-\left(V_{out}-V_{DD}\right)\cdot\left(\frac{dVout}{dVin}\right)\right]$$ Rearrange expression

CMOS Inverter

- **CMOS Inverter Static Behavior (V_{IL}) cont…**

- now our complete differentiated expression is:

$$k_n\cdot\left(V_{in}-V_{T0,n}\right)=k_p\cdot\left[\left(V_{in}-V_{DD}-V_{T0,p}\right)\cdot\left(\frac{dVout}{dVin}\right)+\left(V_{out}-V_{DD}\right)-\left(V_{out}-V_{DD}\right)\cdot\left(\frac{dVout}{dVin}\right)\right]$$

- let's plug in the condition we're solving for (dV_{out}/dV_{in} = -1)

$$k_n\cdot\left(V_{in}-V_{T0,n}\right)=k_p\left[\left(V_{in}-V_{DD}-V_{T0,p}\right)\cdot\left(-1\right)+\left(V_{out}-V_{DD}\right)-\left(V_{out}-V_{DD}\right)\cdot\left(-1\right)\right]$$
$$k_n\cdot\left(V_{in}-V_{T0,n}\right)=k_p\left[-V_{in}+V_{DD}+V_{T0,p}+V_{out}-V_{DD}+V_{out}-V_{DD}\right]$$
$$k_n\cdot\left(V_{in}-V_{T0,n}\right)=k_p\cdot\left(2\cdot V_{out}-V_{in}+V_{T0,p}-V_{DD}\right)$$

- then we can substitute V_{in}=V_{IL}:

$$k_n\cdot\left(V_{IL}-V_{T0,n}\right)=k_p\cdot\left(2\cdot V_{out}-V_{IL}+V_{T0,p}-V_{DD}\right)$$

CMOS Inverter

- **CMOS Inverter Static Behavior (V_{IL}) cont…**

- let's rearrange to solve for V_{IL}:

$$k_n\cdot\left(V_{IL}-V_{T0,n}\right)=k_p\cdot\left(2\cdot V_{out}-V_{IL}+V_{T0,p}-V_{DD}\right)$$

$$k_n\cdot V_{IL}-k_n\cdot V_{T0,n}=k_p\cdot 2\cdot V_{out}-k_p\cdot V_{IL}+k_p\cdot V_{T0,p}-k_p\cdot V_{DD}$$ Multiply through by k_n & k_p

$$k_n\cdot V_{IL}+k_p\cdot V_{IL}=k_p\cdot 2\cdot V_{out}+k_p\cdot V_{T0,p}+k_n\cdot V_{T0,n}-k_p\cdot V_{DD}$$ Arrange V_{IL} terms on LHS

$$V_{IL}\cdot\left(k_n+k_p\right)=k_p\cdot 2\cdot V_{out}+k_p\cdot V_{T0,p}+k_n\cdot V_{T0,n}-k_p\cdot V_{DD}$$ Pull out V_{IL} of LHS

$$V_{IL}=\frac{k_p\cdot 2\cdot V_{out}+k_p\cdot V_{T0,p}+k_n\cdot V_{T0,n}-k_p\cdot V_{DD}}{k_n+k_p}$$ Bring (kn+kp) to RHS

CMOS Inverter

- **CMOS Inverter Static Behavior (V_{IL}) cont…**

- to make this a little simpler, let's divide the top and bottom of the RHS by k_p:

$$V_{IL}=\frac{\frac{k_p}{k_p}\cdot 2\cdot V_{out}+\frac{k_p}{k_p}\cdot V_{T0,p}+\frac{k_n}{k_p}\cdot V_{T0,n}-\frac{k_p}{k_p}\cdot V_{DD}}{\frac{k_n}{k_p}+\frac{k_p}{k_p}}$$

- let's define k_R as the ratio of:

$$k_R=\frac{k_n}{k_p}$$

CMOS Inverter

- **CMOS Inverter Static Behavior (V_{IL}) cont…**

- substituting in k_R, we get our final expression for V_{IL}:

$$V_{IL}=\frac{2\cdot V_{out}+V_{T0,p}-V_{DD}+k_R\cdot V_{T0,n}}{1+k_R}$$

NOTE:

- this still depends on V_{out}. This means to get a numerical solution, we must solve this together with our expression relating the drain currents:

$$\frac{k_n}{2}\cdot\left(V_{IL}-V_{T0,n}\right)^2=\frac{k_p}{2}\cdot\left[2\cdot\left(V_{IL}-V_{DD}-V_{T0,p}\right)\cdot\left(V_{out}-V_{DD}\right)-\left(V_{out}-V_{DD}\right)^2\right]$$

this gives us two expressions and two unknowns (V_{IL} and V_{out})

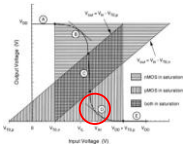
CMOS Inverter

- **CMOS Inverter Static Behavior (V_{IH})**

- V_{IH} is defined as the input voltage that corresponds to the lower of the two output voltages with a slope of -1.

- we know the modes of operation for the transistors in this region:

NMOS = linear
PMOS = saturation



- we also know from KCL that $I_{D,p} = I_{D,n}$

- from this, we can write our first current equation:

$$I_{D,n(lin)} = I_{D,p(sat)}$$

$$\frac{k_n}{2}\cdot\left[2\cdot\left(V_{GS,n}-V_{T0,n}\right)\cdot V_{DS,n}-V_{DS,n}^2\right]=\frac{k_p}{2}\cdot\left(V_{GS,p}-V_{T0,p}\right)^2$$

CMOS Inverter

- **CMOS Inverter Static Behavior (V_{IH}) cont…**

- remembering the relationships between V_{in} & V_{out}, and V_{GS} & V_{DS}:

$$V_{GS,n} = V_{in}$$
$$V_{DS,n} = V_{out}$$
$$V_{GS,p} = -\left(V_{DD}-V_{in}\right) = V_{in}-V_{DD}$$
$$V_{DS,p} = -\left(V_{DD}-V_{out}\right) = V_{out}-V_{DD}$$

- we can write:

$$\frac{k_n}{2}\cdot\left[2\cdot\left(V_{in}-V_{T0,n}\right)\cdot V_{out}-V_{out}^2\right]=\frac{k_p}{2}\cdot\left(V_{in}-V_{DD}-V_{T0,p}\right)^2$$

CMOS Inverter

- **CMOS Inverter Static Behavior (V_{IH}) cont…**

- we are looking for when the derivative of $dV_{out}/dV_{in} = -1$ so we differentiate both sides:

$$\frac{d}{dV_{in}}\left(\frac{k_n}{2}\cdot\left[2\cdot\left(V_{in}-V_{T0,n}\right)\cdot V_{out}-V_{out}^2\right]\right)=\frac{d}{dV_{in}}\left(\frac{k_p}{2}\cdot\left(V_{in}-V_{DD}-V_{T0,p}\right)^2\right)$$

- once again, we have a situation where we are differentiating an expression that contains product terms we use the product rule again:

$$Z = f(x,y)\cdot g(x,y)$$
$$\frac{dZ}{dx}=\frac{df}{dx}\cdot g(x,y)+\frac{dg}{dx}\cdot f(x,y)$$

CMOS Inverter

- **CMOS Inverter Static Behavior (V_{IH}) cont…**

- let's expand the product terms in the LHS:

$$\frac{d}{dV_{in}}\left(\frac{k_n}{2}\cdot\left[2\cdot\left(V_{in}-V_{T0,n}\right)\cdot V_{out}-\left(V_{out}\right)\cdot\left(V_{out}\right)\right]\right)=\frac{d}{dV_{in}}\left(\frac{k_p}{2}\cdot\left(V_{in}-V_{DD}-V_{T0,p}\right)^2\right)$$

Product #1 Product #2

- the right-hand-side is straight forward to perform a partial derivative on (with respect to Vin),

$$RHS = k_p\cdot\left(V_{in}-V_{DD}-V_{T0,n}\right)$$

CMOS Inverter

- **CMOS Inverter Static Behavior (V_{IH}) cont…**

- now let's perform a partial derivative on the LHS (with respect to V_{in}) using the product rule:

$$LHS=\frac{d}{dV_{in}}\left(\frac{k_n}{2}\cdot\left[2\cdot\left(V_{in}-V_{T0,n}\right)\cdot V_{out}-\left(V_{out}\right)\cdot\left(V_{out}\right)\right]\right)$$

$$\frac{df}{dx}\cdot g(x,y)+\frac{dg}{dx}\cdot f(x,y) \qquad \frac{df}{dx}\cdot g(x,y)+\frac{dg}{dx}\cdot f(x,y)$$

$$LHS=\frac{k_n}{2}\cdot\left[2\cdot\left(\left(1\right)\cdot V_{out}+\left(V_{in}-V_{T0,n}\right)\cdot\left(\frac{dV_{out}}{dV_{in}}\right)\right)-\left(\left(\frac{dV_{out}}{dV_{in}}\right)\cdot\left(V_{out}\right)+\left(V_{out}\right)\cdot\left(\frac{dV_{out}}{dV_{in}}\right)\right)\right]$$

$$LHS=\frac{k_n}{2}\cdot\left[2\cdot\left(\left(1\right)\cdot V_{out}+\left(V_{in}-V_{T0,n}\right)\cdot\left(\frac{dV_{out}}{dV_{in}}\right)\right)-\left(2\cdot\left(V_{out}\right)\cdot\left(\frac{dV_{out}}{dV_{in}}\right)\right)\right]$$ Combine two like expressions

$$LHS=k_n\cdot\left[\left(\left(1\right)\cdot V_{out}+\left(V_{in}-V_{T0,n}\right)\cdot\left(\frac{dV_{out}}{dV_{in}}\right)\right)-\left(\left(V_{out}\right)\cdot\left(\frac{dV_{out}}{dV_{in}}\right)\right)\right]$$ Pull out 2

$$LHS=k_n\cdot\left[\left(V_{in}-V_{T0,n}\right)\cdot\left(\frac{dV_{out}}{dV_{in}}\right)+V_{out}-V_{out}\cdot\left(\frac{dV_{out}}{dV_{in}}\right)\right]$$ Rearrange expression

CMOS Inverter

- **CMOS Inverter Static Behavior (V_{IH}) cont…**

- now our complete differentiated expression is:

$$k_n\cdot\left[\left(V_{in}-V_{T0,n}\right)\cdot\left(\frac{dV_{out}}{dV_{in}}\right)+V_{out}-V_{out}\cdot\left(\frac{dV_{out}}{dV_{in}}\right)\right]=k_p\cdot\left(V_{in}-V_{DD}-V_{T0,p}\right)$$

- let's plug in the condition we're solving for ($dV_{out}/dV_{in} = -1$)

$$k_n\cdot\left[\left(V_{in}-V_{T0,n}\right)\cdot\left(-1\right)+V_{out}-V_{out}\cdot\left(-1\right)\right]=k_p\cdot\left(V_{in}-V_{DD}-V_{T0,p}\right)$$
$$k_n\cdot\left[-V_{in}+V_{T0,n}+V_{out}+V_{out}\right]=k_p\cdot\left(V_{in}-V_{DD}-V_{T0,p}\right)$$
$$k_n\cdot\left(-V_{in}+V_{T0,n}+2\cdot V_{out}\right)=k_p\cdot\left(V_{in}-V_{DD}-V_{T0,p}\right)$$

- then we can substitute $V_{in}=V_{IH}$:

$$k_n\cdot\left(-V_{IH}+V_{T0,n}+2\cdot V_{out}\right)=k_p\cdot\left(V_{IH}-V_{DD}-V_{T0,p}\right)$$

CMOS Inverter

- **CMOS Inverter Static Behavior (V_{IH}) cont…**

  - let's rearrange to solve for V_{IH}:

$$k_n \cdot \left(-V_{IH} + V_{T0,n} + 2 \cdot V_{out}\right) = k_p \cdot \left(V_{IH} - V_{DD} - V_{T0,p}\right)$$

$$-k_n \cdot V_{IH} + k_n \cdot V_{T0,n} + 2 \cdot k_n \cdot V_{out} = k_p \cdot V_{IH} - k_p \cdot V_{DD} - k_p \cdot V_{T0,p}$$ Multiply through by k_n & k_p

$$-k_n \cdot V_{IH} - k_p \cdot V_{IH} = -k_p \cdot V_{DD} - k_p \cdot V_{T0,p} - k_n \cdot V_{T0,n} - 2 \cdot k_n \cdot V_{out}$$ Arrange V_{in} terms on LHS

$$-V_{IH} \cdot \left(k_n + k_p\right) = -k_p \cdot V_{DD} - k_p \cdot V_{T0,p} - k_n \cdot V_{T0,n} - 2 \cdot k_n \cdot V_{out}$$ Pull out V_{in} of LHS

$$V_{IH} \cdot \left(k_n + k_p\right) = k_p \cdot V_{DD} + k_p \cdot V_{T0,p} + k_n \cdot V_{T0,n} + 2 \cdot k_n \cdot V_{out}$$ Multiply both sides by -1

$$V_{IH} = \frac{-k_p \cdot V_{DD} + k_p \cdot V_{T0,p} + k_n \cdot V_{T0,n} + 2 \cdot k_n \cdot V_{out}}{k_n + k_p}$$ Bring (kn+kp) to RHS

CMOS Inverter

- **CMOS Inverter Static Behavior (V_{IH}) cont…**

  - again, let's divide the top and bottom of the RHS by k_p:

$$V_{IH} = \frac{\frac{k_p}{k_p} \cdot V_{DD} + \frac{k_p}{k_p} \cdot V_{T0,p} + \frac{k_n}{k_p} \cdot V_{T0,n} + 2 \cdot \frac{k_n}{k_p} \cdot V_{out}}{\frac{k_n}{k_p} + \frac{k_p}{k_p}}$$

  - remember that we defined k_R as:

$$k_R = \frac{k_n}{k_p}$$

CMOS Inverter

- **CMOS Inverter Static Behavior (V_{IH}) cont…**

  - substituting in k_R, we get our final expression for V_{IH}:

$$V_{IH} = \frac{V_{DD} + V_{T0,p} + k_R \cdot V_{T0,n} + 2 \cdot k_R \cdot V_{out}}{1 + k_R}$$

$$V_{IH} = \frac{V_{DD} + V_{T0,p} + k_R \cdot \left(2 \cdot V_{out} + V_{T0,n}\right)}{1 + k_R}$$

NOTE:

  - this expression again depends on V_{out}. This means to get a numerical solution, we must solve this together with our expression relating the drain currents (where V_{in}=V_{IH}):

$$\frac{k_n}{2} \cdot \left[2 \cdot \left(V_{IH} - V_{T0,n}\right) \cdot V_{out} - V_{out}^2\right] = \frac{k_p}{2} \cdot \left(V_{IH} - V_{DD} - V_{T0,p}\right)^2$$

this gives us two expressions and two unknowns (V_{IH} and V_{out})

CMOS Inverter

- **CMOS Inverter Static Behavior**

  - we now have all of the critical voltages to describe the Noise Margins of the Inverter:

$$V_{OL} = 0$$

$$V_{OH} = V_{DD}$$

$$V_{IL} = \frac{2 \cdot V_{out} + V_{T0,p} - V_{DD} + k_R \cdot V_{T0,n}}{1 + k_R}$$

$$V_{IH} = \frac{V_{DD} + V_{T0,p} + k_R \cdot \left(2 \cdot V_{out} + V_{T0,n}\right)}{1 + k_R}$$

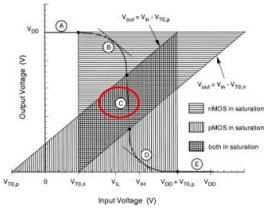
CMOS Inverter Threshold

- **CMOS Inverter Static Behavior (V_{th})**

  - one of the most important static parameters of a CMOS inverter is the switching Threshold

  - Remember that V_{th} is defined as when V_{in}=V_{out}

  - this occurs in Region C of the VTC where both transistors are in ***saturation***



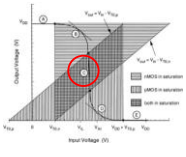
CMOS Inverter Threshold

- **CMOS Inverter Static Behavior (V_{th}) cont…**

  - since we know the modes of operation we can write KCL to get our current equation:

$$I_{D,n(sat)} = -I_{D,p(sat)}$$

$$\frac{k_n}{2} \cdot \left(V_{GS,n} - V_{T0,n}\right)^2 = -\frac{k_p}{2} \cdot \left(V_{GS,p} - V_{T0,p}\right)^2$$



CMOS Inverter Threshold

- **CMOS Inverter Static Behavior (V_{th}) cont…**

  - remembering the relationships between V_{in} & V_{out}, and V_{GS} & V_{DS}:

$$V_{GS,n} = V_{in}$$
$$V_{DS,n} = V_{out}$$
$$V_{GS,p} = -(V_{DD} - V_{in}) = V_{in} - V_{DD}$$
$$V_{DS,p} = -(V_{DD} - V_{out}) = V_{out} - V_{DD}$$

  - we can write:

$$\frac{k_n}{2}\cdot(V_{in} - V_{T0,n})^2 = \frac{k_p}{2}\cdot(V_{in} - V_{DD} - V_{T0,p})^2$$

CMOS Inverter Threshold

- **CMOS Inverter Static Behavior (V_{th}) cont…**

  - now we solve for V_{in}

$$\frac{k_n}{2}\cdot(V_{in} - V_{T0,n})^2 = \frac{k_p}{2}\cdot(V_{in} - V_{DD} - V_{T0,p})^2$$

  - let's walk through the steps of this solution:

$$k_n\cdot(V_{in} - V_{T0,n})^2 = k_p\cdot(V_{in} - V_{DD} - V_{T0,p})^2$$ Multiply both sides by 2

$$\sqrt{k_n\cdot(V_{in} - V_{T0,n})^2} = \sqrt{k_p\cdot(V_{in} - V_{DD} - V_{T0,p})^2}$$ Take square root of both sides

$$\sqrt{k_n}\cdot\sqrt{(V_{in} - V_{T0,n})^2} = \sqrt{k_p}\cdot\sqrt{(V_{in} - V_{DD} - V_{T0,p})^2}$$ Break square roots into equivalent products

$$\sqrt{k_n}\cdot(V_{in} - V_{T0,n}) = -\sqrt{k_p}\cdot(V_{in} - V_{DD} - V_{T0,p})$$ Simplify the square roots. Choosing the correct solution for Vin under the square gives a negative.

$$(V_{in} - V_{T0,n}) = -\frac{\sqrt{k_p}}{\sqrt{k_n}}\cdot(V_{in} - V_{DD} - V_{T0,p})$$ Divide by sqrt(k_n)

CMOS Inverter Threshold

- **CMOS Inverter Static Behavior (V_{th}) cont…**

  - continuing…

$$V_{in} - V_{T0,n} = -\sqrt{\frac{k_p}{k_n}}\cdot(V_{in} - V_{DD} - V_{T0,p})$$ Simplify the square root ratio

$$V_{in} - V_{T0,n} = -\sqrt{\frac{k_p}{k_n}}\cdot(V_{in} - (V_{DD} + V_{T0,p}))$$ Group the V_{DD}+$V_{T0,p}$ terms

$$V_{in} - V_{T0,n} = -\sqrt{\frac{k_p}{k_n}}\cdot V_{in} + \sqrt{\frac{k_p}{k_n}}\cdot(V_{DD} + V_{T0,p})$$ Multiply through the sqrt(kp/kn) term

$$V_{in} + \sqrt{\frac{k_p}{k_n}}\cdot V_{in} = V_{T0,n} + \sqrt{\frac{k_p}{k_n}}\cdot(V_{DD} + V_{T0,p})$$ Rearrange terms to get Vin on the LHS

$$V_{in}\cdot\left(1 + \sqrt{\frac{k_p}{k_n}}\right) = V_{T0,n} + \sqrt{\frac{k_p}{k_n}}\cdot(V_{DD} + V_{T0,p})$$ Pull out Vin of LHS

CMOS Inverter Threshold

- **CMOS Inverter Static Behavior (V_{th}) cont…**

  - continuing…

$$V_{in} = \frac{V_{T0,n} + \sqrt{\frac{k_p}{k_n}}\cdot(V_{DD} + V_{T0,p})}{\left(1 + \sqrt{\frac{k_p}{k_n}}\right)}$$ Divide both sides by (1+sqrt(kp/kn))

$$V_{in} = \frac{V_{T0,n} + \sqrt{\frac{1}{k_R}}\cdot(V_{DD} + V_{T0,p})}{\left(1 + \sqrt{\frac{1}{k_R}}\right)}$$ Substitute in kr=sqrt(kn/kp)

$$V_{th} = \frac{V_{T0,n} + \sqrt{\frac{1}{k_R}}\cdot(V_{DD} + V_{T0,p})}{\left(1 + \sqrt{\frac{1}{k_R}}\right)}$$ Replace Vin with Vth

CMOS Inverter Threshold

- **CMOS Inverter Static Design**

  - notice that V_{DD}, $V_{T0,p}$, and $V_{T0,n}$ are constants for a given system:

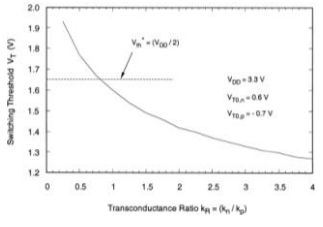
  - this means that the only thing that effects the switching threshold is k_R

$$V_{th} = \frac{V_{T0,n} + \sqrt{\frac{1}{k_R}}\cdot(V_{DD} + V_{T0,p})}{\left(1 + \sqrt{\frac{1}{k_R}}\right)}$$

CMOS Inverter Threshold

- **CMOS Inverter Static Design**

  - we actually have control over k_R by altering the Widths and Lengths of the transistors:



CMOS Inverter Threshold

- **CMOS Inverter Static Design**

  - let's relate the threshold voltage's sensitivity to k_R:

$$V_{th} = \frac{V_{T0,n} + \sqrt{\frac{1}{k_R}} \cdot (V_{DD} + V_{T0,p})}{\left(1 + \sqrt{\frac{1}{k_R}}\right)}$$ Original expression for Vth

$$V_{th}\left(1 + \sqrt{\frac{1}{k_R}}\right) = V_{T0,n} + \sqrt{\frac{1}{k_R}} \cdot (V_{DD} + V_{T0,p})$$ Multiply both sides by (1+sqrt(1/kR))

$$V_{th} + V_{th} \cdot \sqrt{\frac{1}{k_R}} = V_{T0,n} + \sqrt{\frac{1}{k_R}} \cdot (V_{DD} + V_{T0,p})$$ Multiply Vth through

$$V_{th} - V_{T0,n} = \sqrt{\frac{1}{k_R}} \cdot (V_{DD} + V_{T0,p}) - V_{th} \cdot \sqrt{\frac{1}{k_R}}$$ Rearrange terms to get kR on RHS

$$V_{th} - V_{T0,n} = \sqrt{\frac{1}{k_R}} \cdot (V_{DD} + V_{T0,p} - V_{th})$$ Pull out (1+sqrt(1/kR))

$$\sqrt{\frac{1}{k_R}} = \frac{V_{th} - V_{T0,n}}{V_{DD} + V_{T0,p} - V_{th}}$$ Divide to get kR term alone

CMOS Inverter Threshold

- **CMOS Inverter Static Design**

  - rearranging the expression to get in terms of k_R, we get:

$$k_R = \frac{k_n}{k_p} = \left(\frac{V_{DD} + V_{T0,p} - V_{th}}{V_{th} - V_{T0,n}}\right)^2$$

  - an Ideal inverter puts the switching threshold directly in the middle of the voltage swing:

$$v_{th,ideal} = \frac{V_{DD}}{2}$$

  - if we plug in the ideal threshold voltage, we get:

$$\left(\frac{k_n}{k_p}\right)_{ideal} = \left(\frac{V_{DD} + V_{T0,p} - 0.5 \cdot V_{DD}}{0.5 \cdot V_{DD} - V_{T0,n}}\right)^2 = \left(\frac{0.5 \cdot V_{DD} + V_{T0,p}}{0.5 \cdot V_{DD} - V_{T0,n}}\right)^2$$

CMOS Inverter Threshold

- **CMOS Inverter Static Design**

  - in most processes, $V_{T0,n} = |V_{T0,p}|$.

  - since $V_{T0,p}$ is negative, then our Transconductance ratio looks like:

$$\left(\frac{k_n}{k_p}\right)_{ideal} = \left(\frac{0.5 \cdot V_{DD} + V_{T0,p}}{0.5 \cdot V_{DD} - V_{T0,n}}\right)^2 = 1$$

CMOS Inverter Threshold

- **CMOS Inverter Static Design**

  - remembering the expression for k_n/k_p, we can see that C_{ox} will not have an effect:

$$k_R = \frac{k_n}{k_p} = \frac{\mu_n \cdot C_{ox} \cdot \left(\frac{W}{L}\right)_n}{\mu_p \cdot C_{ox} \cdot \left(\frac{W}{L}\right)_p} = \frac{\mu_n \cdot \left(\frac{W}{L}\right)_n}{\mu_p \cdot \left(\frac{W}{L}\right)_p}$$

  - since for an ideal (symmetric) inverter, we have:

$$\left(\frac{k_n}{k_p}\right)_{ideal} = 1 = \frac{\mu_n \cdot \left(\frac{W}{L}\right)_n}{\mu_p \cdot \left(\frac{W}{L}\right)_p}$$

CMOS Inverter Threshold

- **CMOS Inverter Static Design**

  - we can rearrange to see that for a symmetrical inverter:

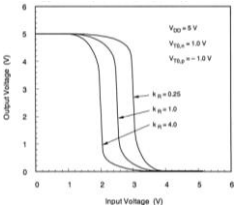
$$\frac{\left(\frac{W}{L}\right)_p}{\left(\frac{W}{L}\right)_n} = \frac{\mu_n}{\mu_p}$$

  - this means that for the given electron mobility of a process, we can size the PMOS and NMOS transistors in order to move the switching threshold to $V_{DD}/2$

CMOS Inverter Threshold

- **CMOS Inverter Static Design**

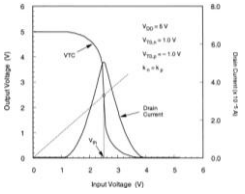
  - sizing of the transistor can have a large impact on the noise margins and sensitivity of the inverter



CMOS Inverter Power

- **CMOS Inverter Power**
  - CMOS inverters don't draw a significant amount of current when they are static (except for leakage)
  - however, when they switch, a path forms between V_{DD} and GND that goes through both transistors
  - the peak current comes at V_{in}=V_{out} when both devices are in saturation
  - this current is also called *short circuit* since VDD has a low resistance path to GND
  - we typically say that the majority of current consumed in a CMOS inverter is *dynamic*

$$I_{D(\max)} = \frac{k_n}{2}\cdot(V_{GS,n} - V_{T0,n})^2 = \frac{k_p}{2}\cdot(V_{GS,p} - V_{T0,p})^2$$

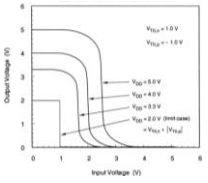


CMOS Inverter Power Supply Scaling

- **CMOS Inverter Power Supply**
  - in theory, we can reduce the power supply voltage to the point at which:

$$V_{DD,\min} = V_{T0,n} + |V_{T0,p}|$$

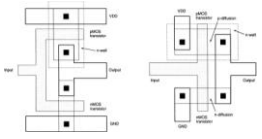
  - below this minimum amount, the inverter will exhibit *hysteresis*
  - in reality, other noise sources in the system require us to select the supply voltage so that the Noise Margins are enough to ensure robust operation.
  - most modern designs select VDD to be ~5xV_{T0}



CMOS Inverter Area

- **CMOS Inverter Area**
  - there are many different ways to layout a CMOS inverter
  - typically, the PMOS electron mobility is lower than the NMOS electron mobility
  - this means that when we design a *symmetric* inverter, the PMOS device is larger
  - we typically make the Lengths of the NMOS and PMOS devices equal (usually L_{min} of the process)
  - this leaves simply the ratio of W_p to W_n as the only design parameters:

$$\frac{W_p}{W_n} = \frac{\mu_n}{\mu_p}$$

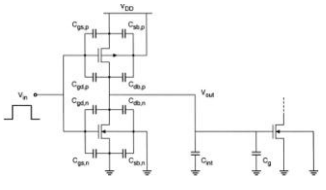


CMOS Switching Characteristics

- **CMOS Switching Characteristics**
  - we studied the DC (or Static) characteristics of the CMOS inverter
  - we learned how to calculate: V_{OH}, V_{OL}, V_{IL}, V_{IH}, V_{th}, NM_L, NM_H,
  - we learned that we can modify some of these parameters using the W/L ratios of the inverter
  - specifically, we say that the V_{th} is solely dependant on W/L and is usually the most important and most commonly controlled parameter
  - we now turn to the Switching (or AC or Dynamic) behavior of the inverter
  - the switching characteristics give us how *fast* the circuit will run
  - when designing, we must meet both DC and AC specs

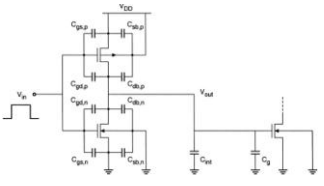
CMOS Switching Characteristics

- **CMOS Switching Characteristics**
  - in an AC analysis, we need to consider the capacitance in the circuit
  - note that the parasitic inductance tends to be small enough to be ignored (for now!)
  - we consider an inverter that is driving another CMOS device or multiple CMOS devices in parallel



CMOS Switching Characteristics

- **CMOS Switching Characteristics**
  - there are 4 main groups of capacitance in the circuit
    1) Driver's Oxide Capacitance
    2) Driver's Junction Capacitance
    3) Interconnect Capacitance
    4) Receiver Oxide Capacitance



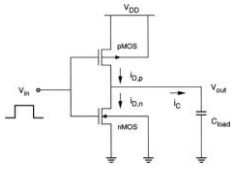
CMOS Switching Characteristics

- **CMOS Switching Characteristics**

  - we know that all of these capacitances vary as the dimensions of the inverter are altered and for various interconnect configurations

  - in order to get a feel for how the capacitance effects performance, we assume that we can *lump* all of the capacitances into a fixed load capacitance (C_{load})

$$C_{load} = C_{gd,n} + C_{gd,p} + C_{db,n} + C_{db,p} + C_{int} + C_g$$



CMOS Switching Characteristics

- **CMOS Switching Characteristics**

  - in this expression we **eliminate** some of the capacitances:

    $C_{sb,n}$, $C_{sb,p}$: There is no voltage change from $V_{sb,n}$ or $V_{sb,p}$ so there is no net capacitance

    $C_{gs,n}$, $C_{gs,p}$: Since these are connected between V_{in} and V_{DD}/V_{SS}, the *input* drives these capacitances. It is not part of the capacitance that the device *output* drives.

  - this expression does include the interconnect and gate capacitance of the circuits that this inverter is driving

$$C_{load} = C_{gd,n} + C_{gd,p} + C_{db,n} + C_{db,p} + C_{int} + C_g$$

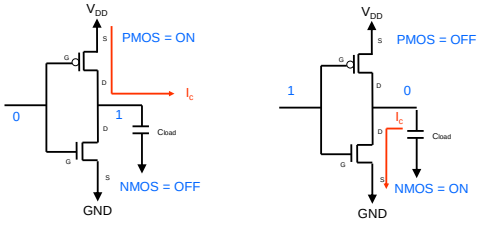
Oxides of Driver — Junctions of Driver — Interconnect — Oxide of Receiver

CMOS Switching Characteristics

- **CMOS Switching Characteristics**

  - the speed of the device describes how fast we can charge or discharge the load capacitor

$$i_C = C\frac{dV}{dt}$$



CMOS Switching Characteristics

- **Delay Time Definition**

  - the delay is the time it takes to switch from the steady state level to the 50% level

$$V_{50\%} = V_{OL} + \frac{1}{2}\cdot(V_{OH} - V_{OL}) = \frac{1}{2}\cdot(V_{OH} + V_{OL})$$

$\tau_{PHL} = t_1 - t_0$

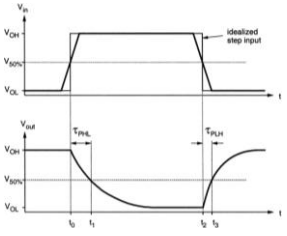
$\tau_{PLH} = t_3 - t_2$

- Note that in CMOS:

  $V_{OH}=V_{DD}$

  $V_{OL}=V_{SS}$

  So $V_{50\%}=V_{DD}/2$



CMOS Switching Characteristics

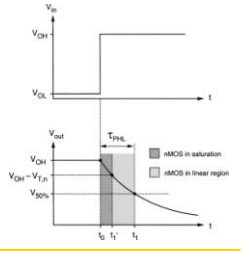
- **Delay Time Derivation (τ_{PHL})**

  - The current that is used to discharge C_{load} is dictated by the region of operation that the NMOS is in.

  - There are two distinct regions of operation that the NMOS operates in during the transition:

1) V_{OH} to $(V_{OH} - V_{T,n})$ NMOS in Saturation

2) $(V_{OH} - V_{T,n})$ to $V_{50\%}$ NMOS in Linear



CMOS Switching Characteristics

- **Delay Time Derivation (τ_{PHL})**

"Differential Equation Method"

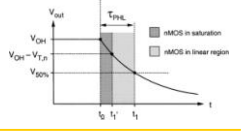
  - we can re-arrange the current expression in the capacitor to be:

$$i_C = -i_{D,n} = C_{load}\frac{dV_{out}}{dt}$$

$$dt = -C_{load}\frac{dV_{out}}{i_{D,n}}$$

  - now we can integrate to solve for *dt*

  - we need to perform two integrals, one for each of the two regions of operation



CMOS Switching Characteristics

- **Delay Time Derivation (τ_{PHL})**

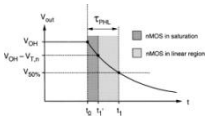
"Differential Equation Method"

- For the *saturation* region, our integral is:

$$t_1' - t_0 = \int_{t=t_0}^{t=t_1'} dt = -C_{load} \int_{V_{out}=V_{OH}}^{V_{out}=V_{OH}-V_{T,n}} \left(\frac{1}{i_{D,n-sat}} \right) dV_{out}$$

- For the *linear* region, our integral is:

$$t_1 - t_1' = \int_{t=t_1'}^{t=t_0} dt = -C_{load} \int_{V_{out}=V_{OH}-V_{T,n}}^{V_{out}=V_{OL}} \left(\frac{1}{i_{D,n-lin}} \right) dV_{out}$$



- The delay is simply the sum of these two solutions:

$$\tau_{PHL} = (t_1' - t_0) + (t_1 - t_1')$$

CMOS Switching Characteristics

- **Delay Time Derivation (τ_{PHL})**

"Differential Equation Method"

- evaluating these integrals and adding the two delays together, we get:

$$\tau_{PHL} = \frac{C_{load}}{k_n (V_{OH} - V_{T,n})} \cdot \left[\frac{2 \cdot V_{T,n}}{V_{OH} - V_{T,n}} + \ln\left(\frac{4(V_{DD} - V_{T,n})}{V_{OH} + V_{OL}} - 1 \right) \right]$$

- we can simplify this further by substituting in $V_{OH}=V_{DD}$ and $V_{OL}=0$

$$\tau_{PHL} = \frac{C_{load}}{k_n (V_{DD} - V_{T,n})} \cdot \left[\frac{2 \cdot V_{T,n}}{V_{DD} - V_{T,n}} + \ln\left(\frac{4(V_{DD} - V_{T,n})}{V_{DD}} - 1 \right) \right]$$

CMOS Switching Characteristics

- **Delay Time Derivation (τ_{PLH})**

"Differential Equation Method"

- we can follow the same process to find τ_{PLH} using the current equations for the PMOS:

$$\tau_{PLH} = \frac{C_{load}}{k_p (V_{DD} - |V_{T,p}|)} \cdot \left[\frac{2 \cdot |V_{T,p}|}{V_{DD} - |V_{T,p}|} + \ln\left(\frac{4(V_{DD} - |V_{T,p}|)}{V_{DD}} - 1 \right) \right]$$

- these solutions are accurate from the standpoint that we use the exact current in the transistors in our derivation of delay.

- these are still *estimates* and don't include *channel-length-modulation* or small-geometry effects

CMOS Switching Characteristics

- **Delay Time Derivation (τ_{PHL})**

"Average Current Method"

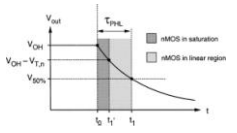
- a simpler technique to estimate the delay is to use the *average current* in the capacitor during the transition.

- this is accomplished by solving for the current at the beginning of the transition and the current at the end of the transition and then averaging the two.

- at the beginning of the High-to-Low transition, the NMOS is in *saturation*

- at the end of the High-to-Low transition, the NMOS is in the *linear region*

$$\tau_{PHL} = \frac{C_{load} \cdot \Delta V_{HL}}{I_{avg,HL}} = \frac{C_{load} \cdot \Delta V_{HL}}{\frac{1}{2} \cdot [i_{D-sat} + i_{D-lin}]}$$



CMOS Switching Characteristics

- **Delay Time Derivation (τ_{PHL} & τ_{PLH})**

"Average Current Method"

- we can write the expression in terms of the voltages at V_{in} ($V_{gs,n}$) and V_{out} ($V_{ds,n}$):

$$\tau_{PHL} = \frac{C_{load} \cdot \Delta V_{HL}}{\frac{1}{2} \cdot [i_{D-sat}(V_{in} = V_{OH}, V_{out} = V_{OH}) + i_{D-lin}(V_{in} = V_{OH}, V_{out} = V_{50\%})]}$$

$$\tau_{PLH} = \frac{C_{load} \cdot \Delta V_{LH}}{\frac{1}{2} \cdot [i_{D-sat}(V_{in} = V_{OL}, V_{out} = V_{OL}) + i_{D-lin}(V_{in} = V_{OL}, V_{out} = V_{50\%})]}$$

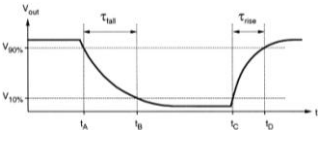
- this technique tends to be faster and easier to use than the *differential equation* method.

CMOS Switching Characteristics

- **Rise & Fall Time Definitions (τ_{rise} & τ_{fall})**

- rise time (τ_{rise}) is the time it takes to transition from $V_{10\%}$ to $V_{90\%}$

- fall time (τ_{fall}) is the time it takes to transition from $V_{90\%}$ to $V_{10\%}$



- we can use either the (1) *differential equation* or the (2) *average current* technique to solve for these

- in these transitions, the transistors again operate in both the *saturation* and *linear* regions

- the only difference is that the limits of the transition are $V_{10\%}$ and $V_{90\%}$

CMOS Switching Characteristics

- **Non-ideal Inputs**
  - in all of these derivations, we have assumed a perfect step input.
  - if the input is not a perfect step (i.e., it has a finite delay or rise time), it will increase the delay of the gate
  - we can use an RMS estimation to account for the non-ideal input:

$$\tau^2_{PHL(actual)} = \tau^2_{PHL(in\_step\_input)} + \tau^2_{PLH(of\_input)}$$

$$\tau^2_{PLH(actual)} = \tau^2_{PLH(in\_step\_input)} + \tau^2_{PHL(of\_input)}$$

  - we can also estimate the delay of the input if we are only given its rise/fall time by using:

$$\tau_{PLH} = \frac{\tau_{rise}}{2}$$

$$\tau_{PHL} = \frac{\tau_{fall}}{2}$$

CMOS Switching Characteristics

- **Non-ideal Inputs**
  - we can apply this technique to the rise and fall times also:

$$\tau^2_{rise(actual)} = \tau^2_{rise(in\_step\_input)} + \tau^2_{fall(of\_input)}$$

$$\tau^2_{fall(actual)} = \tau^2_{fall(in\_step\_input)} + \tau^2_{rise(of\_input)}$$

CMOS Switching Characteristics

- **Designing for Constraints**
  - when we begin a design, we typically start with specification
  - we then size the transistors to achieve the desired performance
  - we saw how the sizes of the transistor effect the DC specs, specifically V_{th}
  - we also need to size the transistors so that for a given load capacitance, the gate can achieve a designed delay or rise/fall time.
  - we can use the expressions for delay and rise/fall time that we derived to calculate the necessary transistor sizes.

CMOS Switching Characteristics

- **Designing for Constraints**
  - the *average current* method is the simplest technique to use:

$$\tau_{PHL} = \frac{C_{load} \cdot \Delta V_{HL}}{\frac{1}{2} \cdot \left[i_{D-sat}(V_{in} = V_{OH}, V_{out} = V_{OH}) + i_{D-lin}(V_{in} = V_{OH}, V_{out} = V_{50\%}) \right]}$$

$$\tau_{PLH} = \frac{C_{load} \cdot \Delta V_{LH}}{\frac{1}{2} \cdot \left[i_{D-sat}(V_{in} = V_{OL}, V_{out} = V_{OL}) + i_{D-lin}(V_{in} = V_{OL}, V_{out} = V_{50\%}) \right]}$$

  - in this expression, we can insert our timing spec in for τ_{PHL} or τ_{PLH}
  - the RHS of the expression must evaluate to be less than or equal to the timing spec

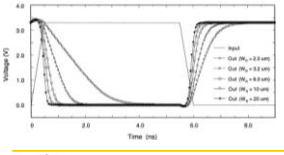
CMOS Switching Characteristics

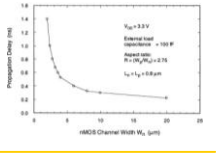
- **Designing for Constraints**
  - in the timing expression, notice that k_n and k_p are parameters under our control
  - these parameters are in the denominator of the timing expression, meaning that as k_n and k_p increase, the delay of the circuit will decrease.
  - this means that *larger = faster*
  - we typically leave the lengths of the NMOS and PMOS transistors equal to each other
  - we also typically set the lengths to the smallest possible dimension for a given process.
  - this gives us the highest transconductance for a given *Length* and also minimizes the area.
  - a given design process consists of the following steps:

1) set L_n=L_p=L_{min}
2) find the W_p/W_nratio that will yield the desired V_{th}
3) find the minimum values for W_p and W_n to achieve timing
4) combine the minimum sizes and the W_p/W_nratio to select final sizes
5) round up the dimensions to give additional margin and standard sizes (i.e., 4.927um rounds up to 5um)

CMOS Switching Characteristics

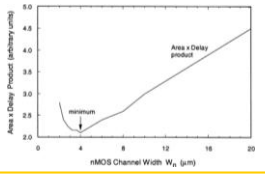
- **Area vs. Delay**
  - we've seen that *larger = faster* for a given inverter
  - however, we have made an assumption that the load capacitance is independent of transistor size
  - we know what a portion of the load capacitance comes from the driver oxide and driver junctions
  - this means that as the inverter gets larger, so does the capacitance
  - this leads to a point of *diminishing returns* with regards to reducing delay





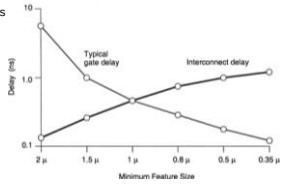
CMOS Switching Characteristics

- **Area vs. Delay**
  - we can look at the **Area X Delay Product** to gauge the *quality* of a design with regards to efficient area usage.
  - typically we will see an inflection point which indicates the point at which increasing the size of the transistors to decrease delay is out-weighed by the negative impact of increasing the area used on the silicon.
  - if a timing specifications requires an excessively large sized gate, it typically means that the process is not sufficient to meet timing.



CMOS Switching Characteristics

- **Interconnect**
  - one of the components in the load capacitance is the interconnect.
  - the interconnect refers to the polysilicon and metal layers that are used to connect the gates together.
  - as sizes on-chip shrink, we've seen that the scaling of interconnect is a big problem because the delay actually increases as you get smaller.
  - in addition, the delay scales quadradically with length meaning that intra-module traces and global interconnect can create significant timing challenges.
  - in modern processes, the delay of the interconnect is actually more than the switching delay of the transistors.



CMOS Switching Characteristics

- **Interconnect Modeling**
  - modeling of the interconnect describes the equivalent circuits we use to describe the electrical behavior of the materials.
  - the type of model we use is a trade-off between accuracy and simulation time
  - we typciall use 1 of the 3 following models:

	Typical Uses
1) Lumped Capacitance	inter-module
2) RC network	intra-module and global
3) Transmission Line	global and off-chip

CMOS Switching Characteristics

- **Interconnect Modeling**
  - we choose the appropriate model based on the rise/fall time of the driver relative to the *prop delay* of the interconnect
  - the *prop delay* (t_{prop}) is the time it takes for the wave to travel down the length of the interconnect:
  - the velocity of a wave in a dielectric is given by:

$$v = \frac{c}{\varepsilon_r}$$

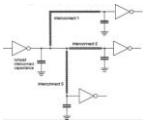
  - the *prop delay* can then be given by:

$$t_{prop} = \frac{length}{v}$$

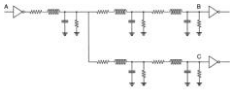
CMOS Switching Characteristics

- **Interconnect Modeling**
  - we move between a *lumped* (C or RC) and a *distributed* (transmission line) model as follows:

"Lumped" $\tau_{rise} < 2.5 \cdot \left(\frac{l}{v}\right)$



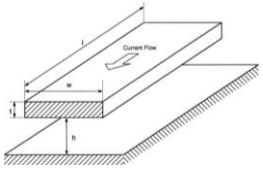
"Distributed" $\tau_{rise} \geq 2.5 \cdot \left(\frac{l}{v}\right)$



CMOS Switching Characteristics

- **Interconnect Resistance**
  - resistance is based on the geometry and materials of the interconnect

$$R = \frac{\rho \cdot l}{A} = R_S \cdot (\#\_of\_squares)$$

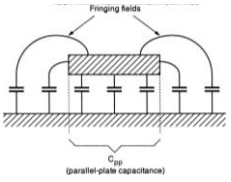


CMOS Switching Characteristics

- **Interconnect Capacitance**

  - capacitance depends on the surface area of the conductor, the insulating materials between the conductors, and the distance between the conductors.

$$C = \frac{\varepsilon \cdot A}{t}$$



CMOS Switching Characteristics

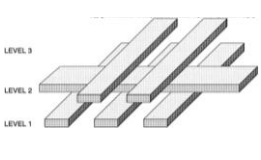
- **Interconnect Capacitance**

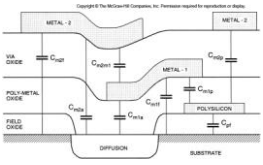
  - interconnect modeling becomes a complex problem due to the 3D geometries present on-chip

  - we typically take a *guess* at the capacitance of the interconnect for initial simulations.

  - once we start physically laying out our design, we can use the CAD tool to *extract* the actual capacitance and back annotate it into our simulation.

  - we then run a new simulation with accurate capacitance models to verify timing is still met post-layout.





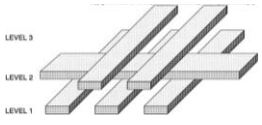
CMOS Switching Characteristics

- **Interconnect Capacitance**

  - *Cross-talk* refers to the noise that is generated on a line due to capacitive coupling from neighboring lines that are switching.

  - as geometries get smaller, lines are closer together so capacitance goes up.

  - we can reduce cross-talk by separating the traces or inserting ground lines between the signals, but this takes area.



CMOS Switching Characteristics

- **Elmore Delay**

  - when we model interconnect using RC networks, it doesn't take many branches in the net before the KVL/KCL solution for the delay gets complex.

  - *Elmore Delay* is a technique to estimate the overall delay between two nodes of an RC network tree.

  - in Elmore Delay, we find the equivalent RC network of the path between two nodes by:

    - summing the delay of each segment in our path-of-interest

    - we construct a set of RC networks as seen by our path-of-interest and then sum them together

    - we walk through the series resistance in our path-of-interest.

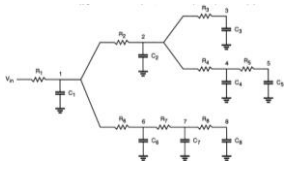
    - for each resistor node in our path of interest, we include RC's in our expression as follows:

      - C's NOT in our path-of-interest are included as ($R_{seg} \cdot C_x$)
      - C's that ARE in our path-of-interest can't be seen if they are on the far side of a resistor in our path-of-interest
      - as we get to the end of our path-of-interest, we can see all of the downstream Capacitances past our end-node.

CMOS Switching Characteristics

- **Elmore Delay**

  - example: Find the expression for the equivalent RC from V_{in} to node 7:



$$\begin{aligned}\tau_{D7} = &(R_1)\cdot(C_1)+(R_1)\cdot(C_2)+(R_1)\cdot(C_3)+(R_1)\cdot(C_4)+(R_1)\cdot(C_5)\\ &+(R_1+R_6)\cdot(C_6)\\ &+(R_1+R_6+R_7)\cdot(C_7)+(R_1+R_6+R_7)\cdot(C_8)\end{aligned}$$

CMOS Switching Characteristics

- **Dynamic Power Consumption**

  - in theory, a CMOS gate does not consume any Static Power because the NMOS and PMOS transistors are in the cut-off regions when driving V_{OH} or V_{OL}

  - we know what there is leakage current in cut-off, however to the first order we neglect it.

  - the majority of the power is due to the charging and discharging of C_{load}

  - this is called *Dynamic Power* because it is AC in nature and only occurs when the gate switches

  - this current is described as:

$$i_C = C_{load}\frac{dV_{out}}{dt}$$

  - since the current consumed is proportional to the number of times that the gate switches, we need to make an assumption to the number of times per second that V_{out} switches

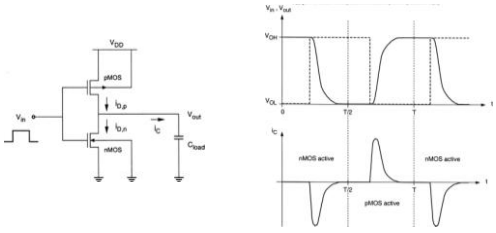
  - since we have a binary system, we can assume that the output will be a '0' 50% of the time and a '1' 50% of the time.

  - we can model the voltage on V_{out} as a periodic square wave

CMOS Switching Characteristics

- **Dynamic Power Consumption**

  - current will be drawn from V_{DD} and sunk into V_{SS} during a transition



CMOS Switching Characteristics

- **Dynamic Power Consumption**

  - assuming a periodic input and output waveform, the average power dissipated by a device over one period is given as:

$$P_{avg} = \frac{1}{T}\int_0^T v(t)\cdot i(t)dt$$

  - we split up the period into two sections:

    0 →T/2 V_{in} transitions from a 0 to a 1, the NMOS discharges C_{load}

    T/2 →T V_{in} transitions from a 1 to a 0, the PMOS charges C_{load}

CMOS Switching Characteristics

- **Dynamic Power Consumption**

  - we can now re-write our average power expression as:

$$P_{avg} = \frac{1}{T}\left[\int_0^{T/2} V_{out}\cdot\left(-C_{load}\frac{dV_{out}}{dt}\right)dt + \int_{T/2}^{T}(V_{DD}-V_{out})\cdot\left(C_{load}\frac{dV_{out}}{dt}\right)dt\right]$$

$$P_{avg} = \frac{1}{T}\left[\left(-C_{load}\frac{V_{out}^2}{2}\right)\Big|_0^{T/2} + \left(V_{DD}\cdot V_{out}\cdot C_{load} - C_{load}\frac{V_{out}^2}{2}\right)\Big|_{T/2}^{T}\right]$$

$$P_{avg} = \frac{1}{T}\cdot C_{load}\cdot V_{DD}^2$$

$$P_{avg} = f\cdot C_{load}\cdot V_{DD}^2$$

CMOS Switching Characteristics

- **Dynamic Power Consumption**

  - a more qualitative view of this power consumption is as follows:

    Capacitance is defined as:

$$C = \frac{Q}{V}$$

    Each cycle, the average current in the capacitor is:

$$I_{AVG} = \frac{Q}{T} = \frac{C\cdot V}{T}$$

    Power is I·V, which gives:

$$P_{AVG} = V_{AVG}\cdot I_{AVG} = V\cdot\frac{C\cdot V}{T} = \frac{1}{T}\cdot C_{load}\cdot V_{out}^2 = f\cdot C_{load}\cdot V_{out}^2$$

CMOS Switching Characteristics

- **Power Delay Product (PDP)**

  - another *quality* measure of a design is the PDP

  - this is a measure of the energy required to switch logic levels in a given period.

  - qualitatively, Power x Time is:

$$P_{avg} = \frac{1}{time}\cdot C_{load}\cdot V_{DD}^2$$

$$P_{avg}\cdot\tau = C_{load}\cdot V_{DD}^2$$

CMOS Switching Characteristics

- **Power Delay Product (PDP)**

  - as the delay goes down, the power goes up.

  - the power going up is due to the increase in intrinsic junction capacitance of the driver.

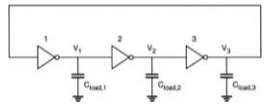
  - the delay reaches an ~asymptotic limit as the size is increased.

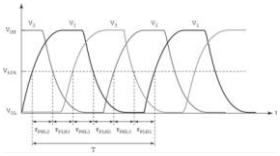
  - the power increases as the size is increased.

  - looking at the PDP can give an estimate of when you are optimally sized to deliver energy in the most effective manner.

Ring Oscillator

- **Ring Oscillator**
  - if we connect a chain of inverters in a loop and have an ODD number of inverters, the circuit is inherently unstable.
  - the circuit will *oscillate* between a 0 and 1 indefinitely.
  - the frequency of the oscillation depends on the gate delay of the inverter.
  - this type of circuit is commonly used to test the device delay of a given process.
  - this can also be used to create a clock.
  - the clock frequency of the ring oscillator is not typically controlled tight enough to be used as the system clock.





$$f = \frac{1}{2 \cdot n_{inv} \cdot \tau_{inv}}$$

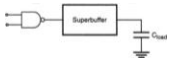
Super Buffer

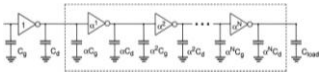
- **Super Buffer**
  - off-chip capacitances are typically an order of magnitude larger than on-chip capacitances.
  - VLSI gates that are used in logic circuitry are sized to drive other gates of comparable size.
  - if these smaller gates are connected to a much larger load capacitance, they are not sized optimally.
  - a gate can typically not drive a capacitance that has a much larger capacitance than its own junction capacitance.
  - a *super buffer* is a circuit that consists of a series of gates, each with a increasingly larger size and drive strength.
  - we define the relative size of each subsequence gate using the *optimal sizing factor* (α)
  - we start with a typical logic gate and then design a subsequent stage that is larger by a factor of α
  - we continue to add stages until the final capacitance that is to be driven (C_{load}) is a factor of α larger than the last state of the *super buffer*.
  - we define the number of stages in the *super buffer* as **N**

Super Buffer

- **Super Buffer**
  - we define C_g as the gate capacitance of the next state.
  - we define C_d as the output drain capacitance of current stage.
  - we can derive an expression to find the total delay as a function of (N, α).
  - this expression gives us a relationship between N and α.
  - we can differentiate this expression to find the optimal scaling factor:

$$\alpha(\ln \alpha - 1) = \frac{C_d}{C_g}$$





Super Buffer

- **Super Buffer**
  - we first find α.
  - this defines how much larger each subsequent stage is relative to its driving stage
  - we continue to add stages until the final C_g that can be driven is C_{load}