ECE 271

Electronic Circuits I

Topic 8
Complementary MOS (CMOS) Logic Design

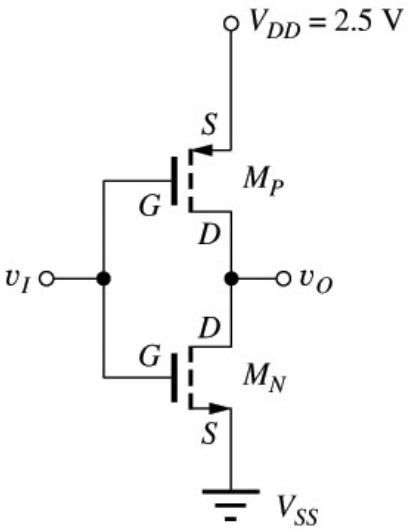
Chapter Goals

- Introduce CMOS logic concepts
- Explore the voltage transfer characteristics of CMOS inverters
- Learn to design basic and complex CMOS logic gates
- Discuss the static and dynamic power in CMOS logic
- Present expressions for dynamic performance of CMOS logic devices
- Present noise margins for CMOS logic
- Introduce design techniques for “cascade buffers”

CMOS Inverter Technology

- Complementary MOS, or CMOS, needs both PMOS and NMOS devices for the logic gates to be realized
- The concept of CMOS was introduced in 1963 by Wanlass and Sah.
- CMOS are more complicated in design and production, thus are more expensive to fabricate
- Have not been widely used until the 1980's as NMOS microprocessors started to dissipating as much as 50 W and more and alternative design technique was needed
- CMOS dominate digital IC design today

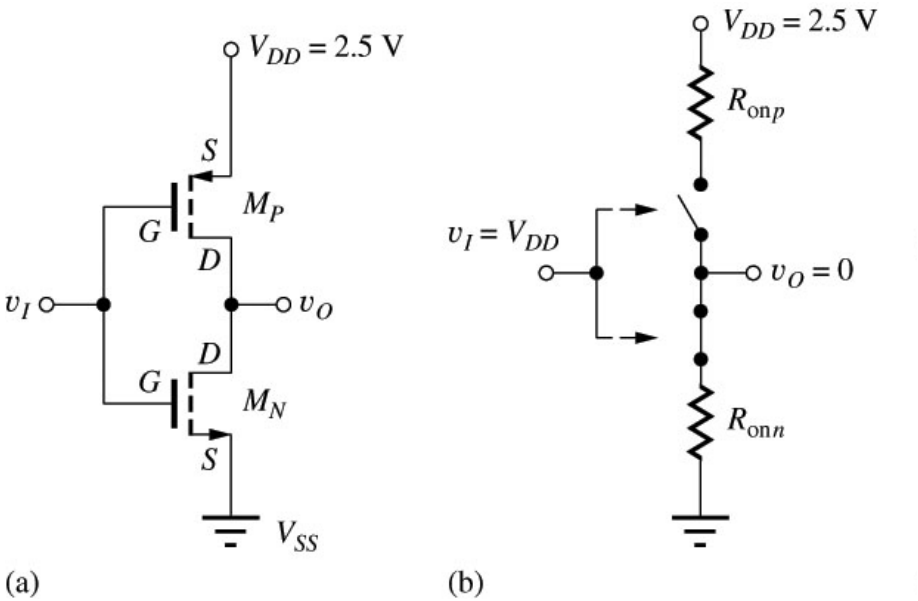
CMOS Inverter



(a)

(a) Circuit schematic for a CMOS inverter

CMOS Inverter



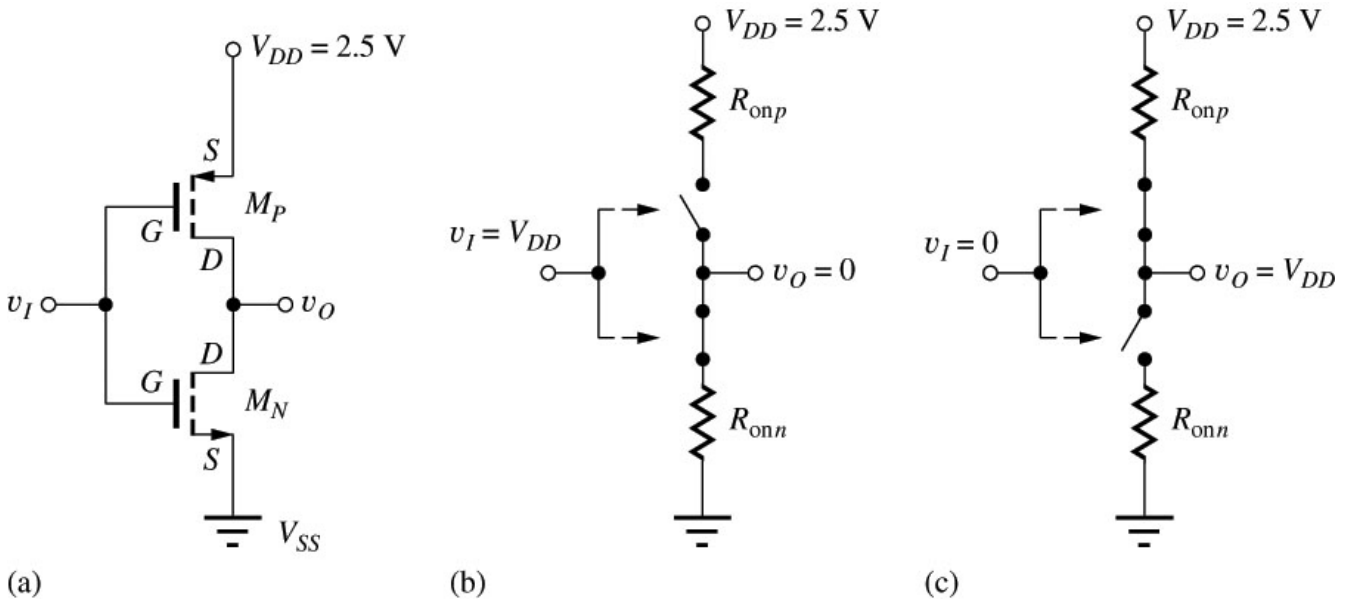
(a) Circuit schematic for a CMOS inverter

(b) Simplified operation model with a high input applied

(

- When v_I is pulled high (to V_{DD}), the PMOS transistor is turned off, while the NMOS device is turned on pulling the output down to V_{SS}

CMOS Inverter



(a) Circuit schematic for a CMOS inverter

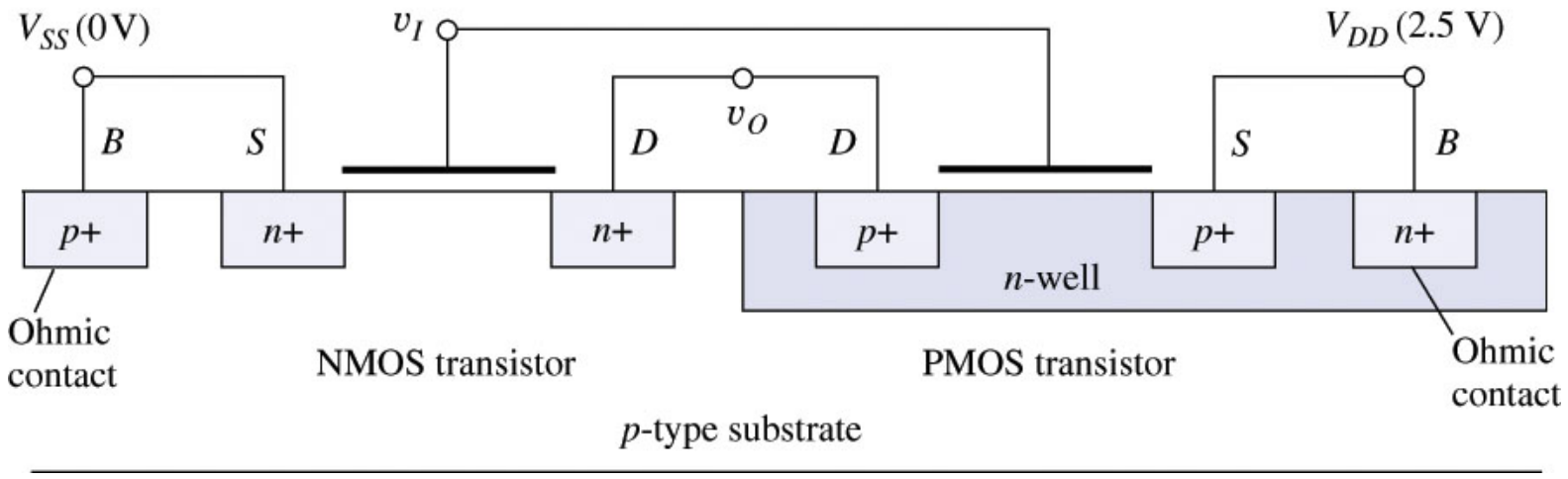
(b) Simplified operation model with a high input applied

(c) Simplified operation model with a low input applied

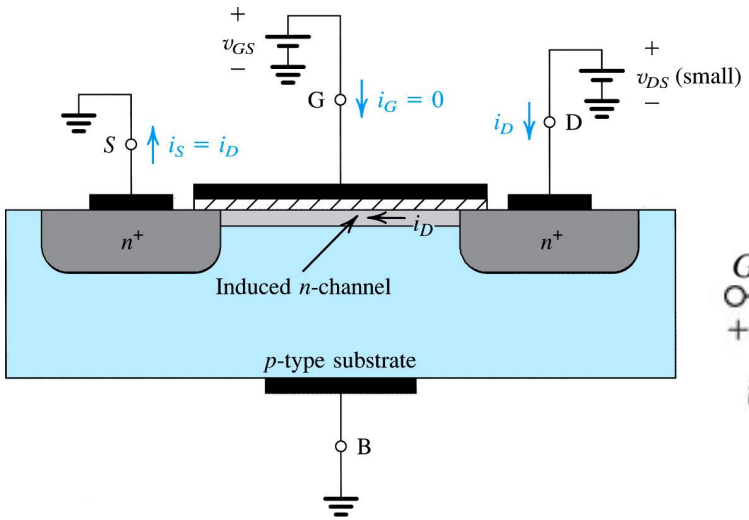
- When v_I is pulled high (to V_{DD}), the PMOS transistor is turned off, while the NMOS device is turned on pulling the output down to V_{SS}
- When v_I is pulled low (to V_{SS}), the NMOS transistor is turned off, while the PMOS device is turned on pulling the output up to V_{DD}

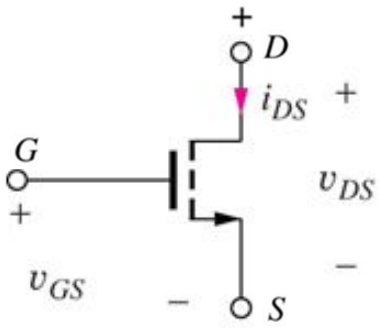
CMOS Inverter Technology

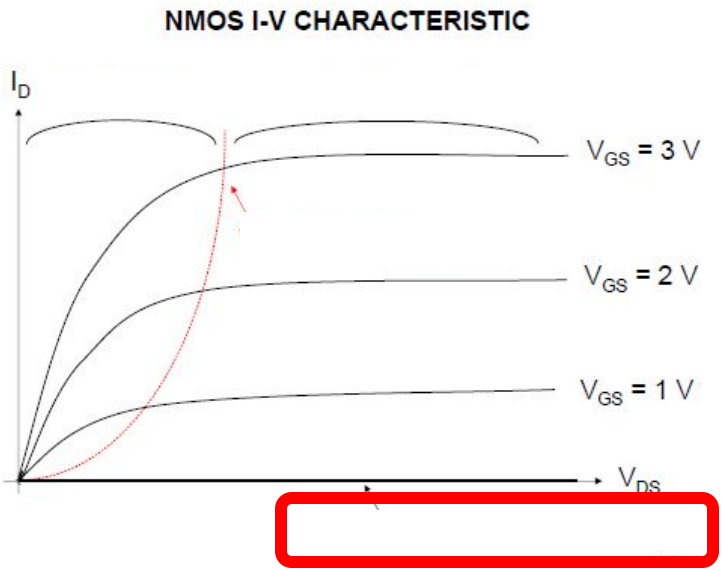
- The CMOS inverter consists of a PMOS device stacked on top on an NMOS device, but they need to be fabricated on the same wafer
- To accomplish this, the technique of "n-well" implantation is needed as shown in this cross-section of a CMOS inverter



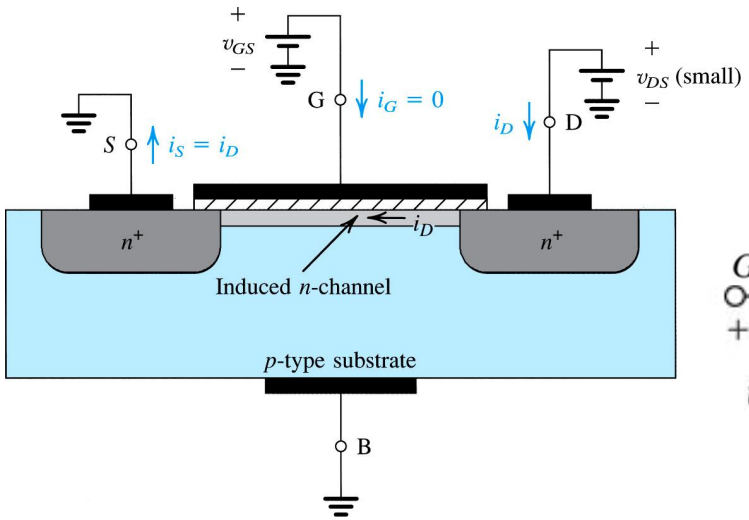
NMOS and PMOS recap

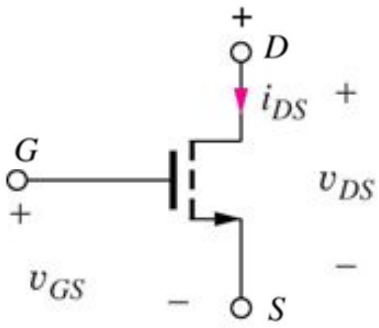


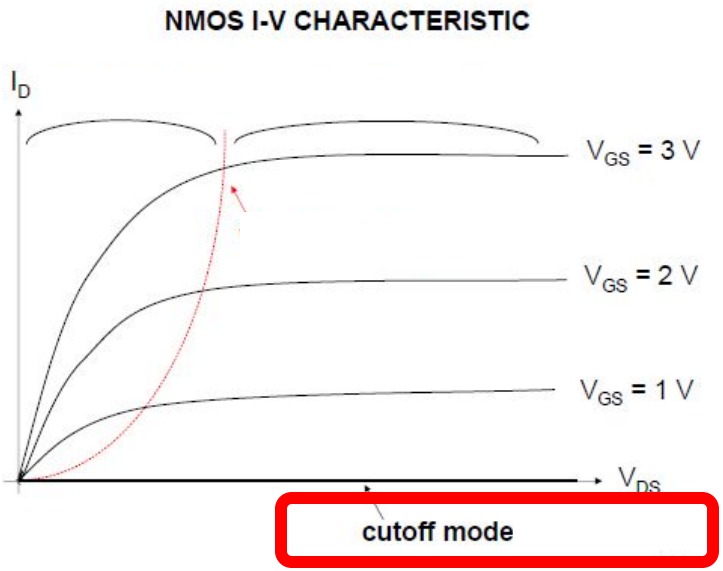




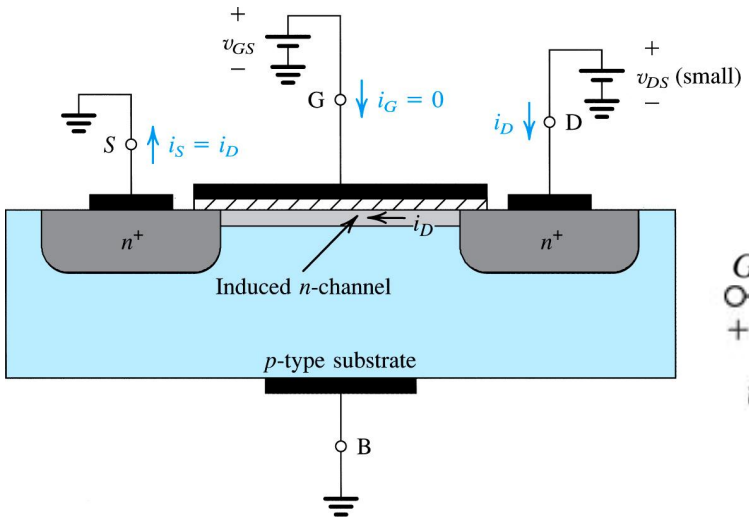
NMOS and PMOS recap

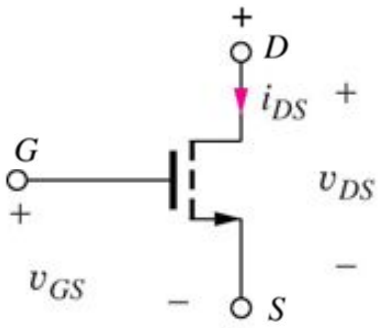


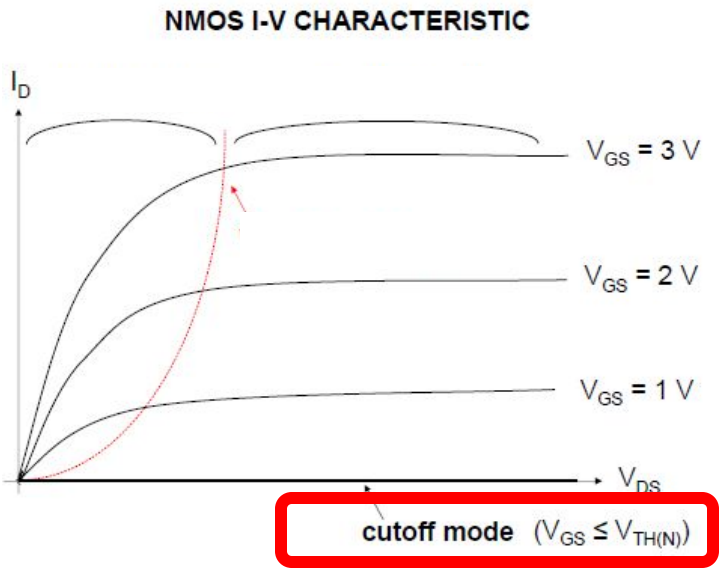




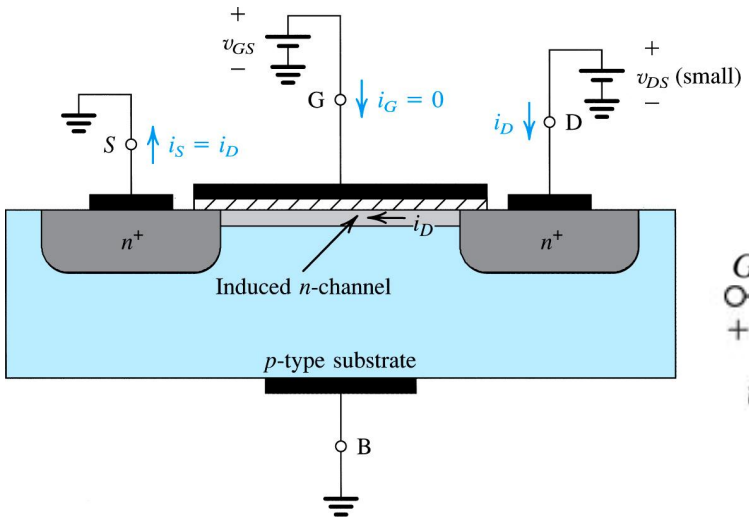
NMOS and PMOS recap

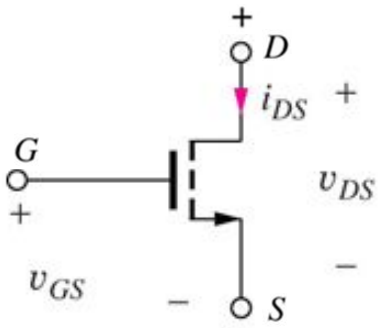


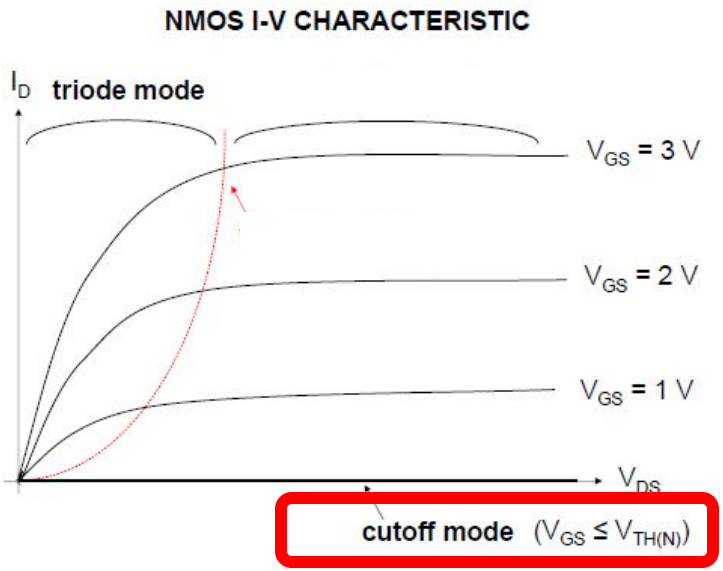




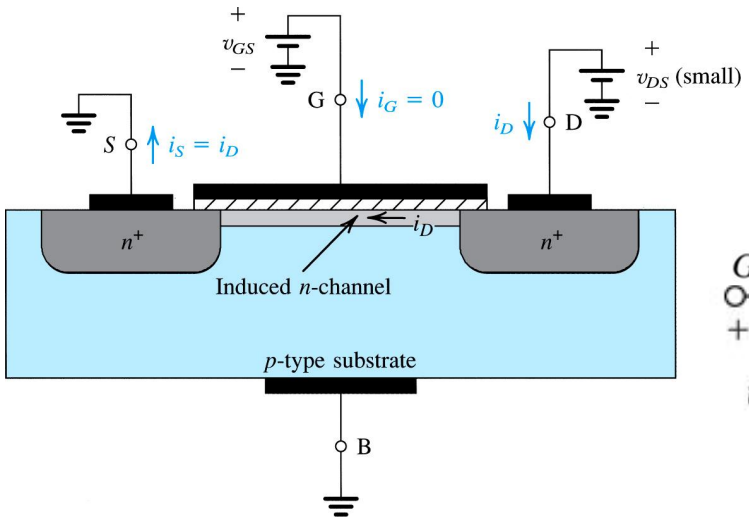
NMOS and PMOS recap

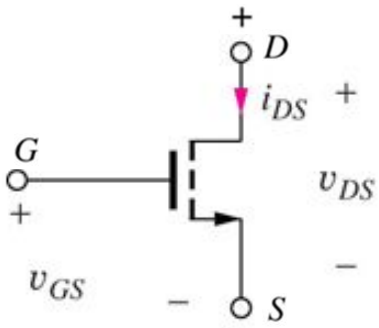


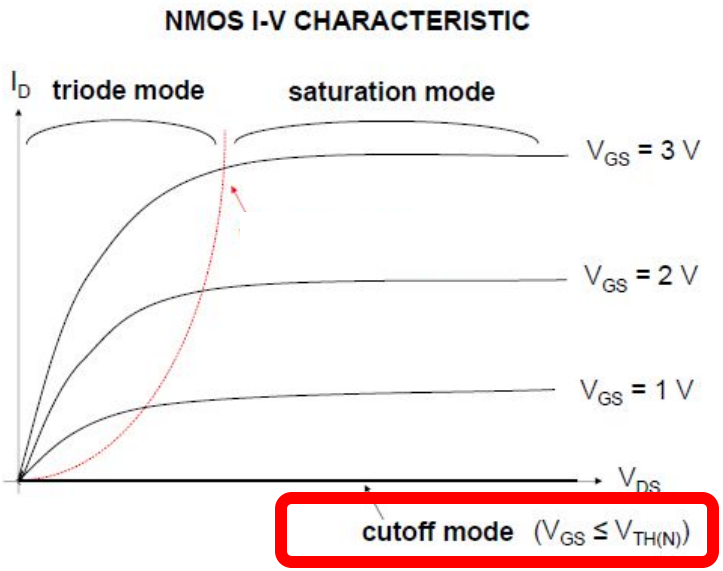




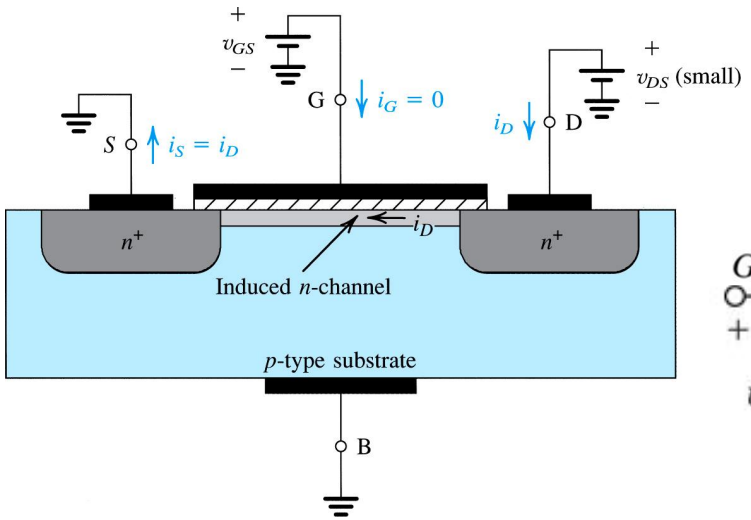
NMOS and PMOS recap

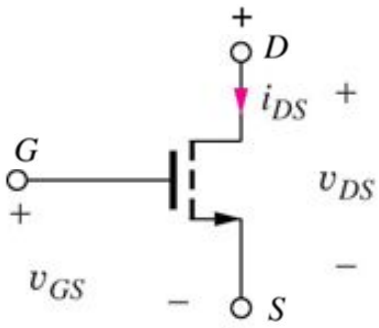


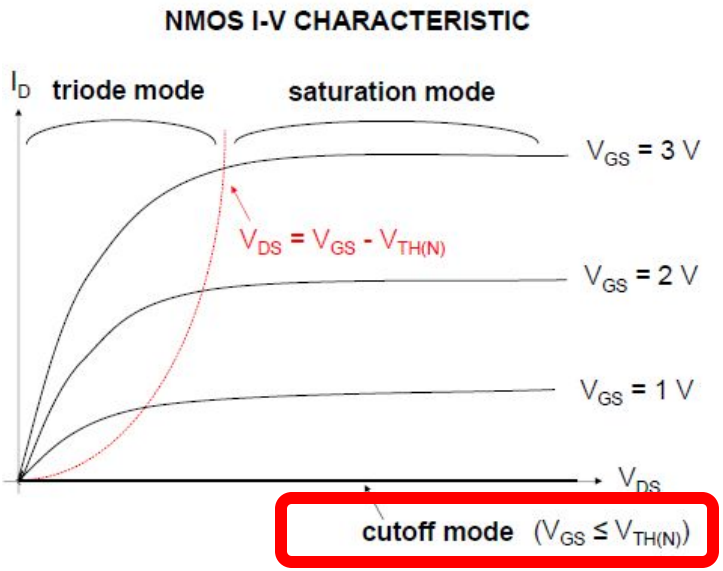




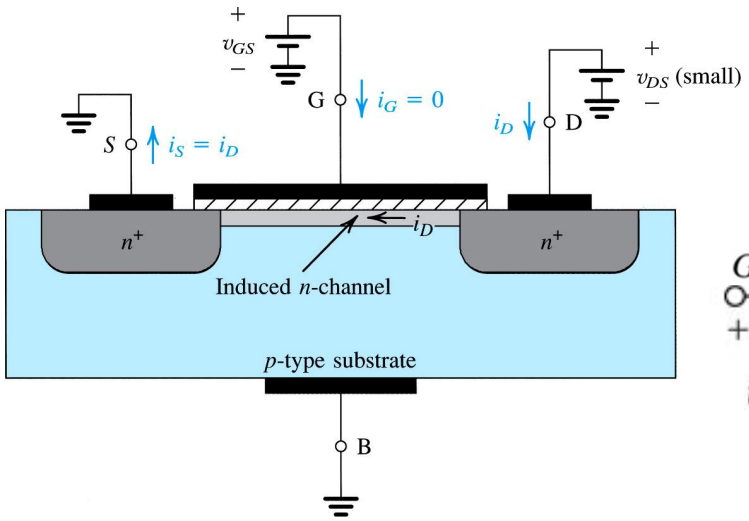
NMOS and PMOS recap

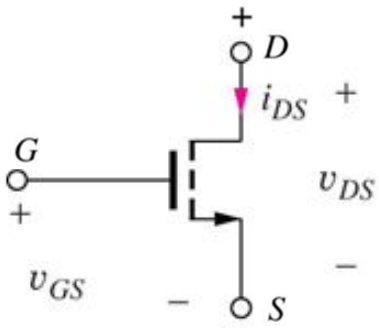


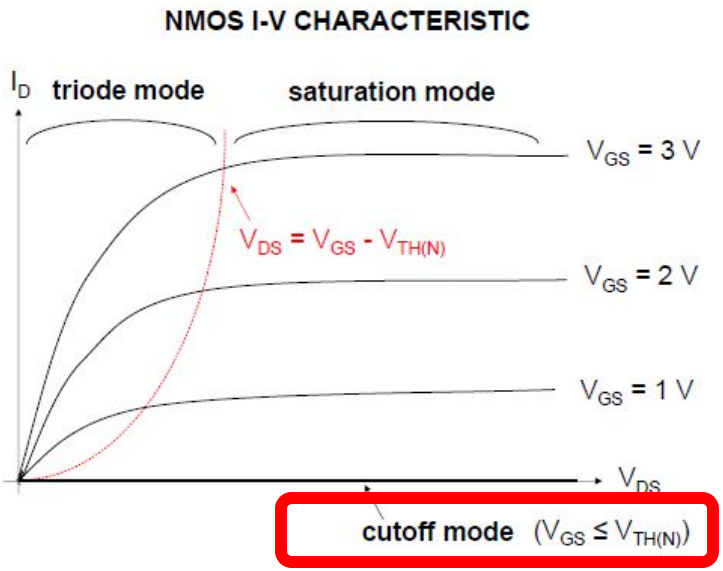


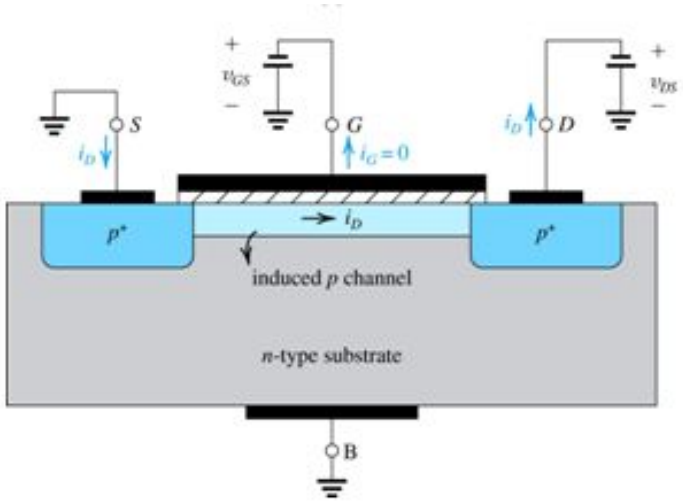


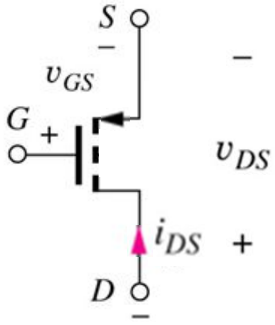
NMOS and PMOS recap



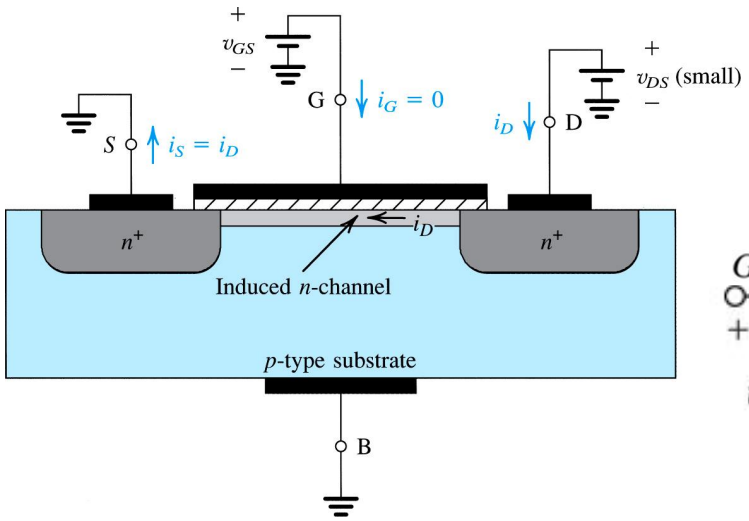


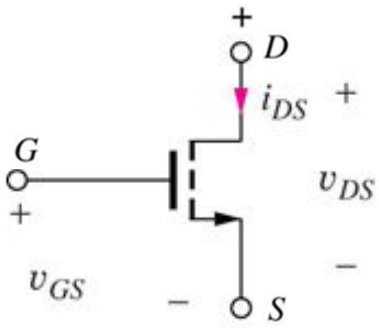


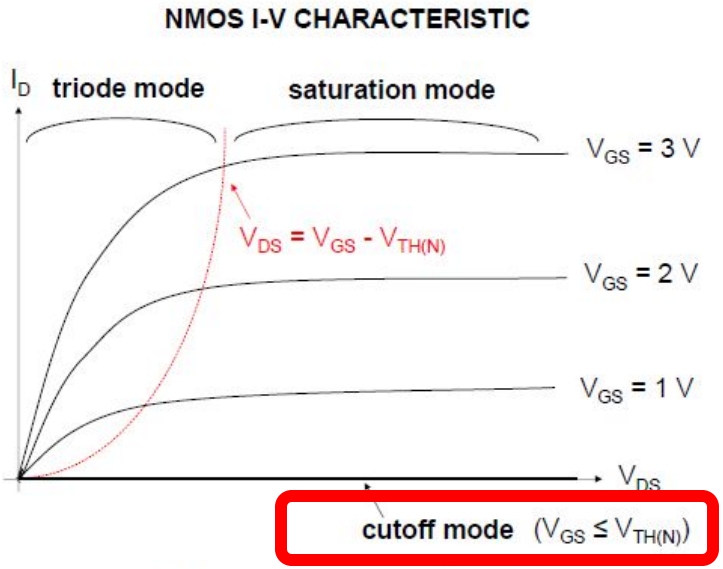


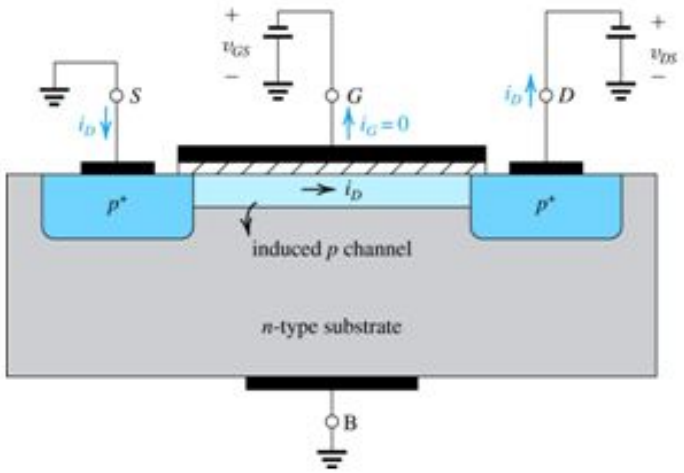


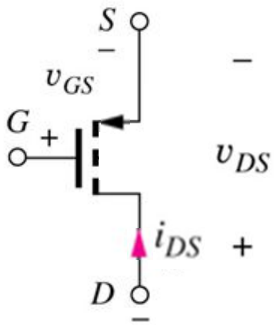
NMOS and PMOS recap

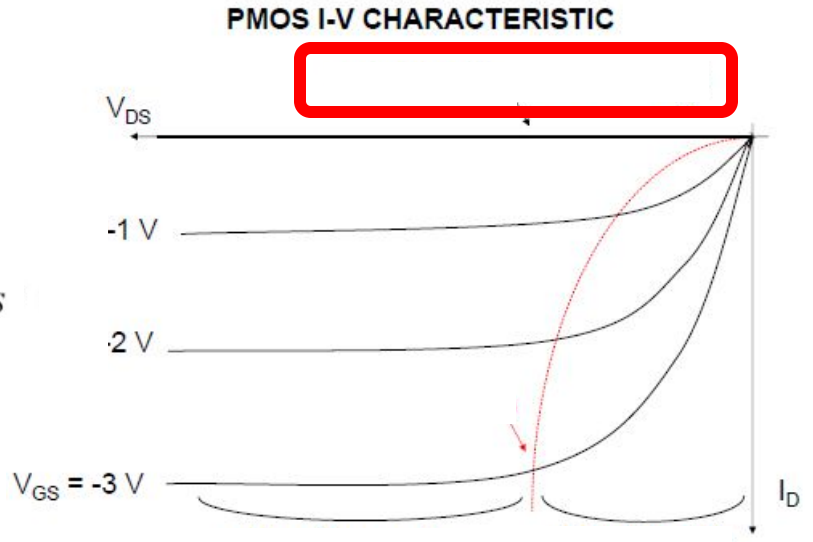




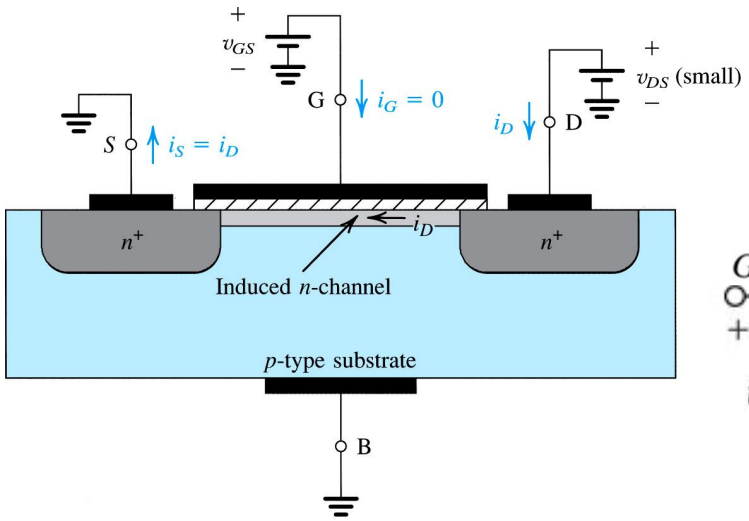


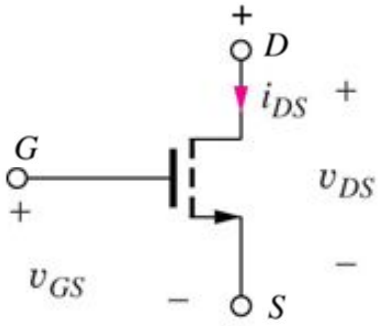


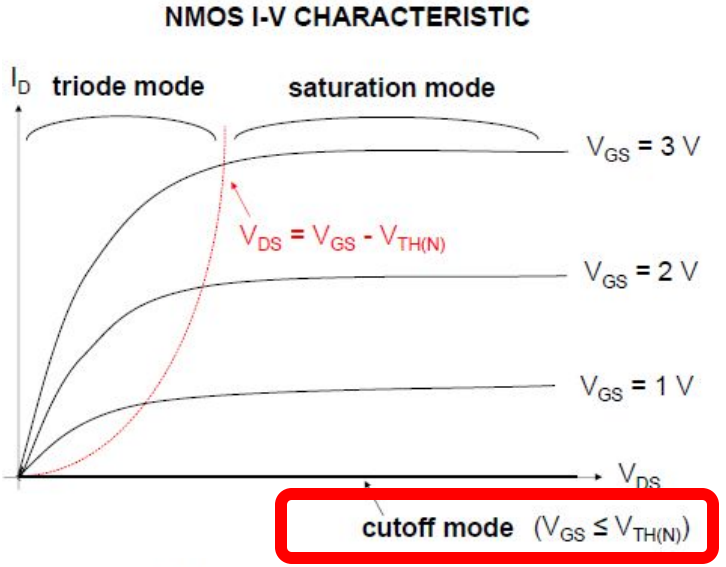


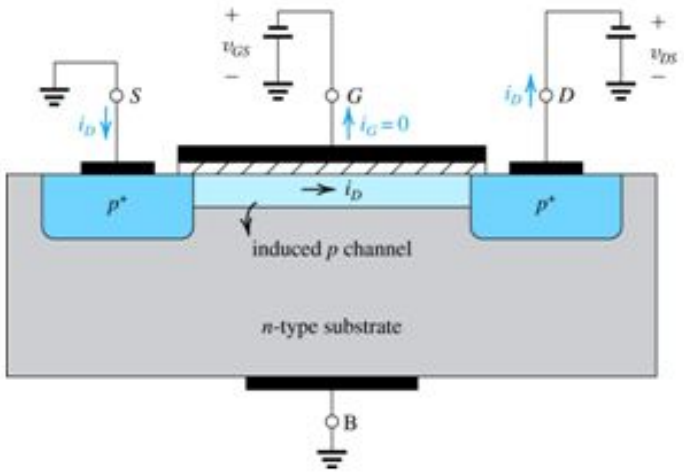


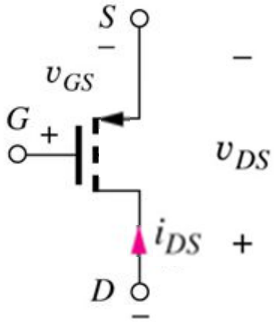
NMOS and PMOS recap

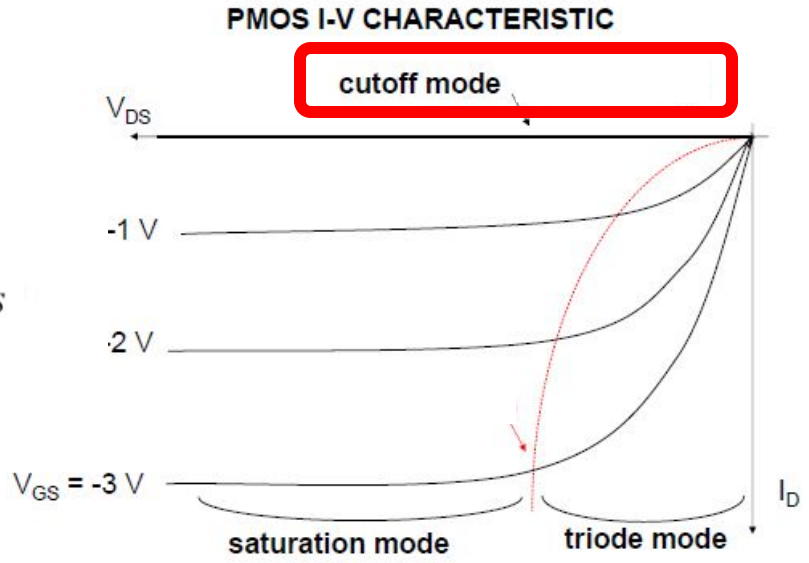




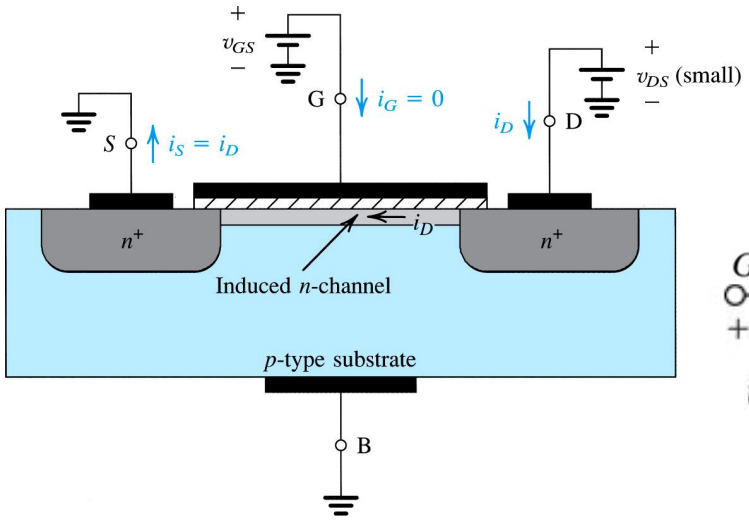


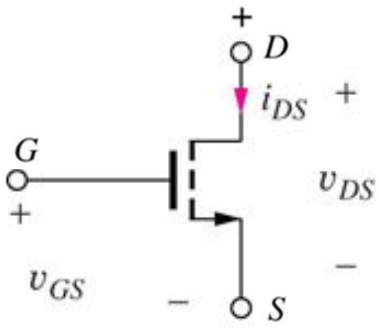


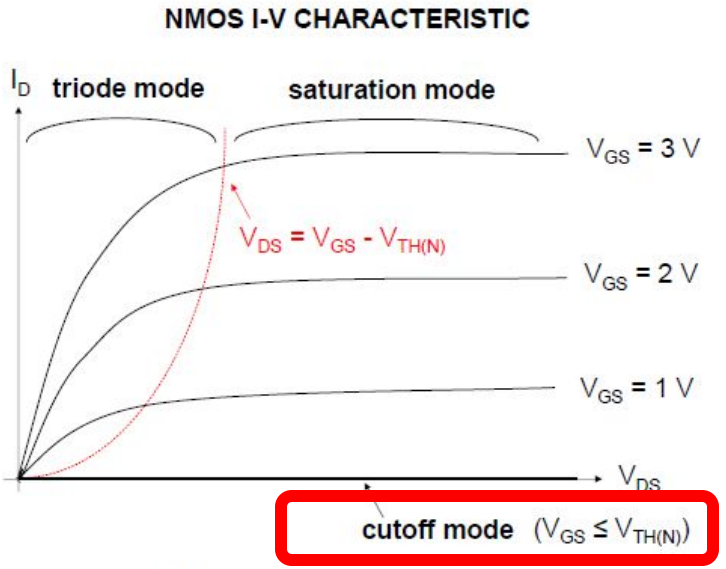


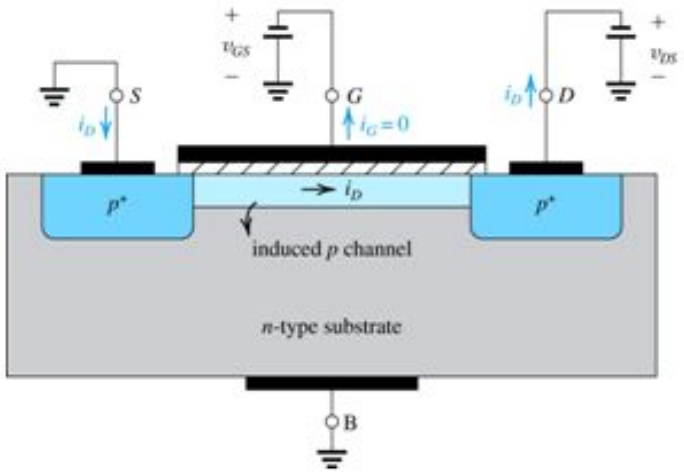


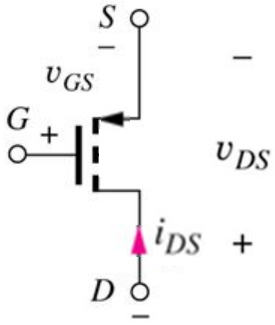
NMOS and PMOS recap

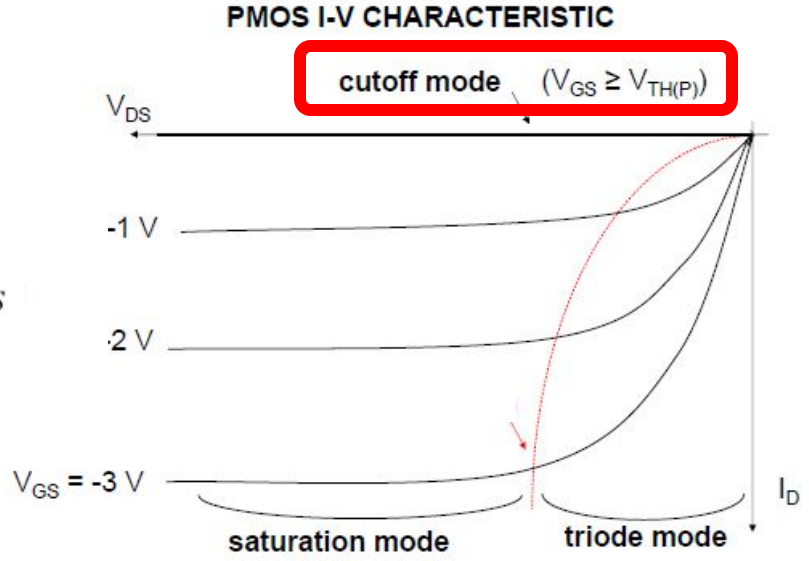




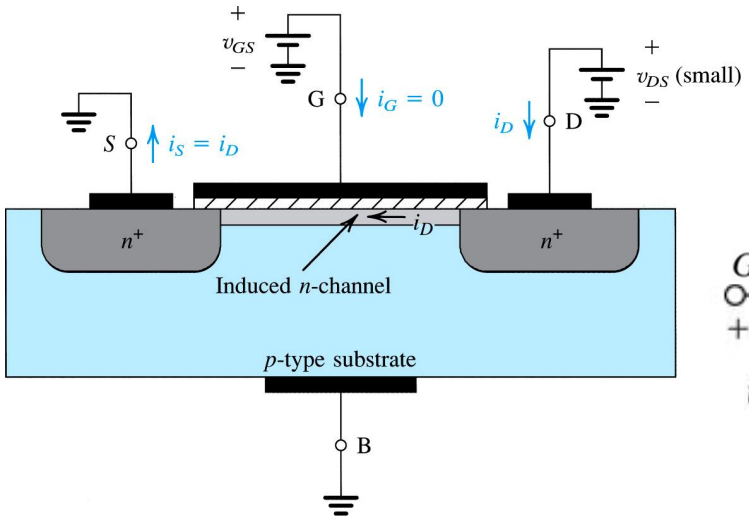


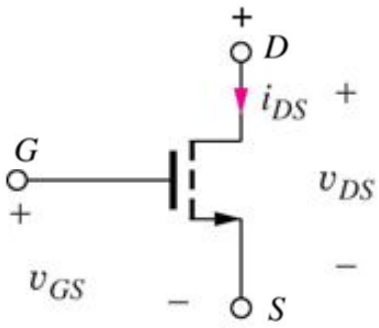


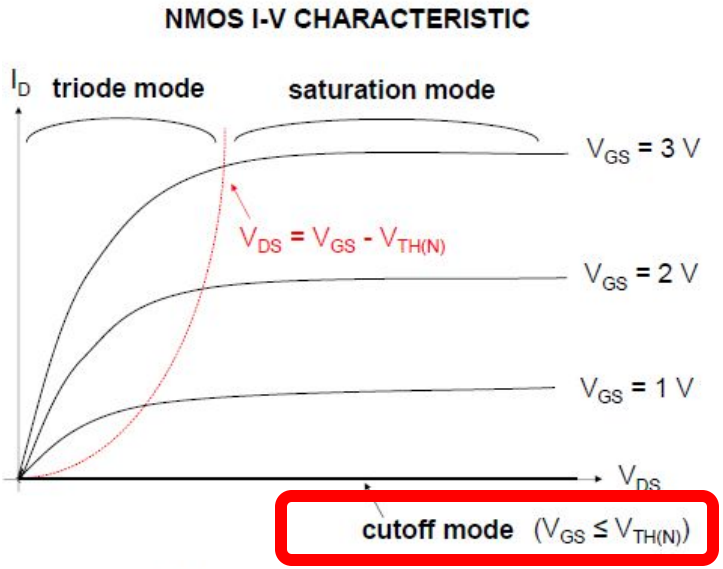


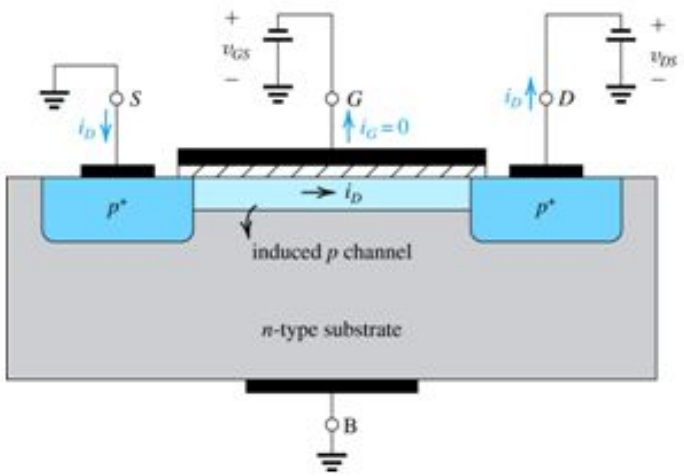


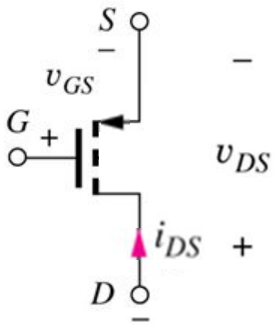
NMOS and PMOS recap

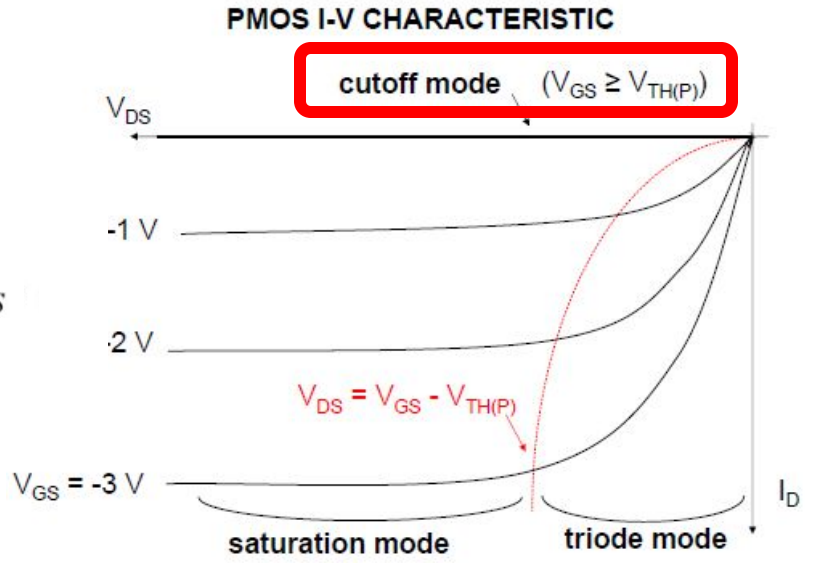




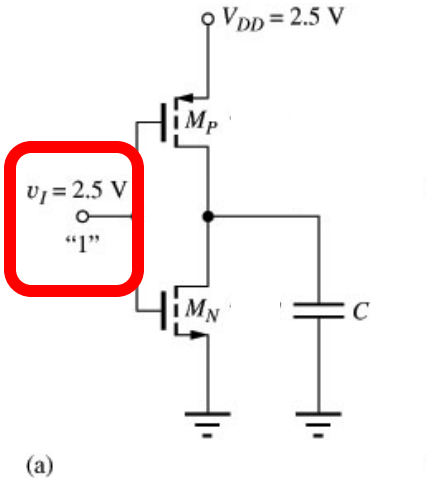








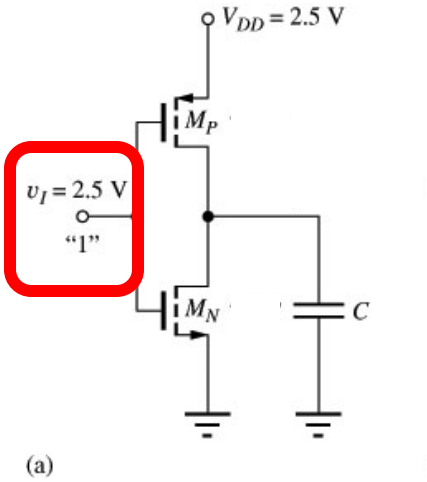
Static States of the CMOS Inverter



$$v_I = 2.5\text{V (1 state)} \Rightarrow \left\{ v_{GS} = \qquad V_{TN} (= \qquad \text{for NMOS} \right\}$$

$$\Rightarrow$$

Static States of the CMOS Inverter

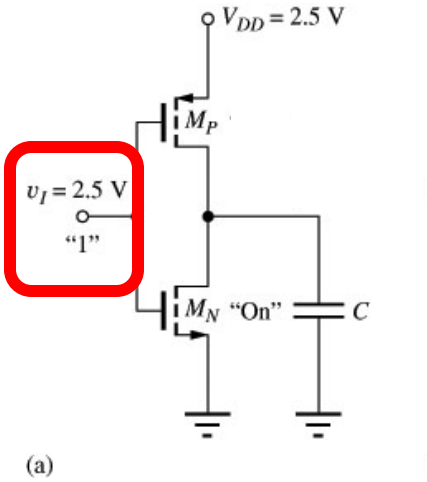


(a)

$$v_I = 2.5\text{V (1 state)} \Rightarrow \left\{ v_{GS} = 2.5 > V_{TN} (= 0.6) \text{ for NMOS} \right\}$$

$$\Rightarrow$$

Static States of the CMOS Inverter

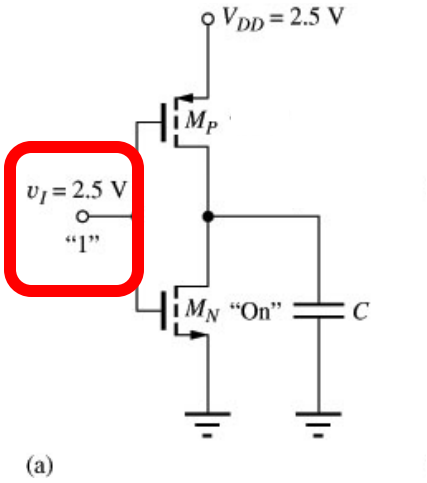


(a)

$$v_I = 2.5\text{V (1 state)} \Rightarrow \left\{ v_{GS} = 2.5 > V_{TN} (= 0.6) \text{ for NMOS} \right\}$$

$$\Rightarrow \left\{ M_N \text{ is "On"} \right\} \Rightarrow$$

Static States of the CMOS Inverter

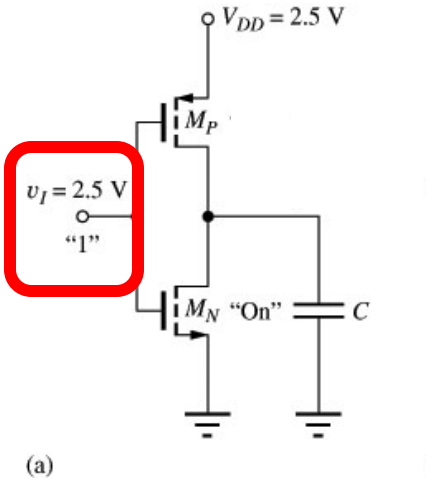


(a)

$$v_I = 2.5\text{V (1 state)} \Rightarrow \left\{ \begin{array}{ll} v_{GS} = 2.5 > V_{TN} (= 0.6) & \text{for NMOS} \\ v_{GS} = \quad V_{TP} (= & \text{for PMOS} \end{array} \right\}$$

$$\Rightarrow \left\{ M_N \text{ is "On"} \right\} \Rightarrow$$

Static States of the CMOS Inverter

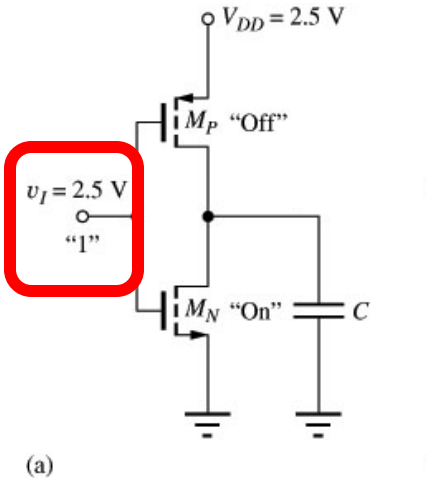


(a)

$$v_I = 2.5\text{V (1 state)} \Rightarrow \left\{ \begin{array}{l} v_{GS} = 2.5 > V_{TN} (= 0.6) \text{ for NMOS} \\ v_{GS} = 0 > V_{TP} (= -0.6) \text{ for PMOS} \end{array} \right\}$$

$$\Rightarrow \left\{ M_N \text{ is "On"} \right\} \Rightarrow$$

Static States of the CMOS Inverter

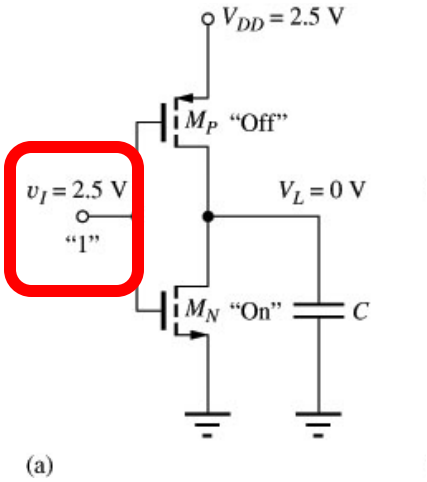


(a)

$$v_I = 2.5\text{V (1 state)} \Rightarrow \left\{ \begin{array}{l} v_{GS} = 2.5 > V_{TN} (= 0.6) \text{ for NMOS} \\ v_{GS} = 0 > V_{TP} (= -0.6) \text{ for PMOS} \end{array} \right\}$$

$$\Rightarrow \left\{ \begin{array}{l} M_N \text{ is "On"} \\ M_P \text{ is "Off"} \end{array} \right\} \Rightarrow$$

Static States of the CMOS Inverter

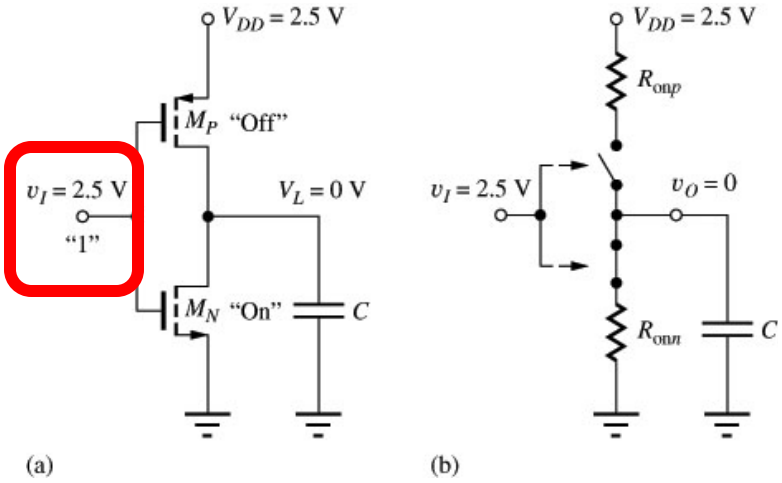


(a)

$$v_I = 2.5\text{V (1 state)} \Rightarrow \left\{ \begin{array}{l} v_{GS} = 2.5 > V_{TN} (= 0.6) \text{ for NMOS} \\ v_{GS} = 0 > V_{TP} (= -0.6) \text{ for PMOS} \end{array} \right\}$$

$$\Rightarrow \left\{ \begin{array}{l} M_N \text{ is "On"} \\ M_P \text{ is "Off"} \end{array} \right\} \Rightarrow v_O = 0 = V_L \text{ (0 state)}$$

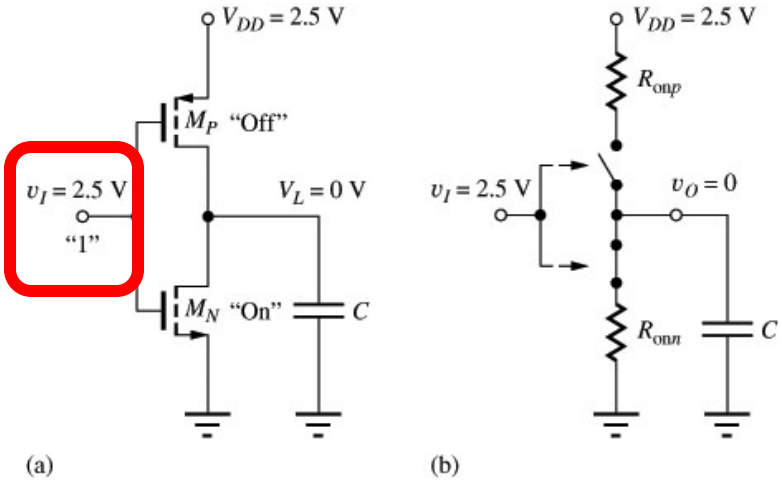
Static States of the CMOS Inverter



$$v_I = 2.5\text{V (1 state)} \Rightarrow \begin{Bmatrix} v_{GS} = 2.5 > V_{TN}(= 0.6) \text{ for NMOS} \\ v_{GS} = 0 > V_{TP}(= -0.6) \text{ for PMOS} \end{Bmatrix}$$

$$\Rightarrow \begin{Bmatrix} M_N \text{ is "On"} \\ M_P \text{ is "Off"} \end{Bmatrix} \Rightarrow v_O = 0 = V_L \text{ (0 state)}$$

Static States of the CMOS Inverter

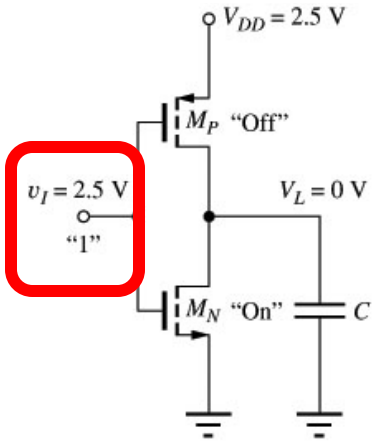


$$v_I = 2.5\text{V (1 state)} \Rightarrow \begin{Bmatrix} v_{GS} = 2.5 > V_{TN}(= 0.6) \text{ for NMOS} \\ v_{GS} = 0 > V_{TP}(= -0.6) \text{ for PMOS} \end{Bmatrix}$$

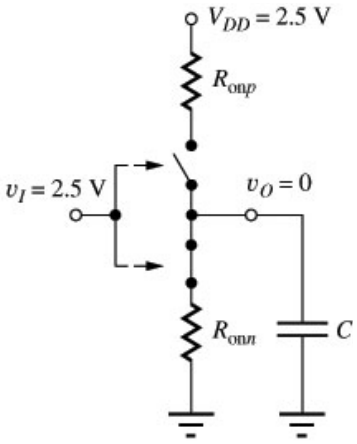
$$\Rightarrow \begin{Bmatrix} M_N \text{ is "On"} \\ M_P \text{ is "Off"} \end{Bmatrix} \Rightarrow v_O = 0 = V_L \text{ (0 state)}$$

- The capacitor discharges through R_{onN}, current exists only during discharge, **no dc current exists**.

Static Characteristics of the CMOS Inverter



(a)

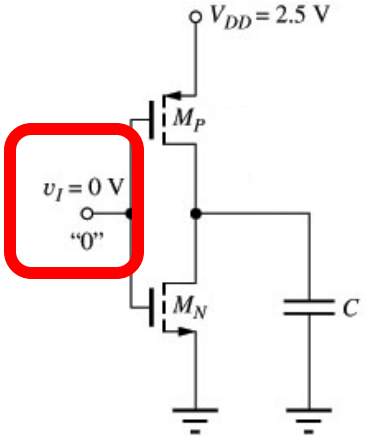


(b)

$$v_I = 2.5\text{V (1 state)} \Rightarrow \left\{ \begin{array}{l} v_{GS} = 2.5 > V_{TN} (= 0.6) \text{ for NMOS} \\ v_{GS} = 0 > V_{TP} (= -0.6) \text{ for PMOS} \end{array} \right\}$$

$$\Rightarrow \left\{ \begin{array}{l} M_N \text{ is "On"} \\ M_P \text{ is "Off"} \end{array} \right\} \Rightarrow v_O = 0 = V_L \text{ (0 state)}$$

- The capacitor discharges through R_{onN}, current exists only during discharge, **no dc current exists.**

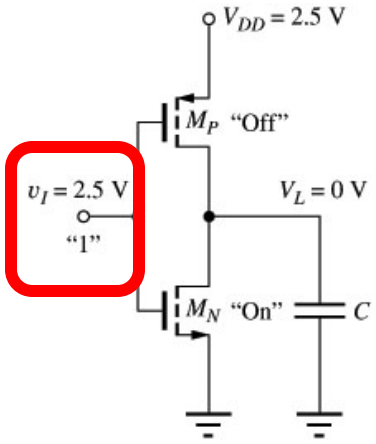


(c)

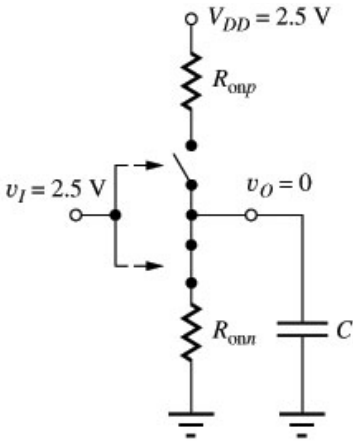
$$v_I = 0\text{V (0 state)} \Rightarrow \left\{ \begin{array}{l} v_{GS} = \quad V_{TN} \quad \text{for NMOS} \end{array} \right\}$$

$$=$$

Static Characteristics of the CMOS Inverter



(a)

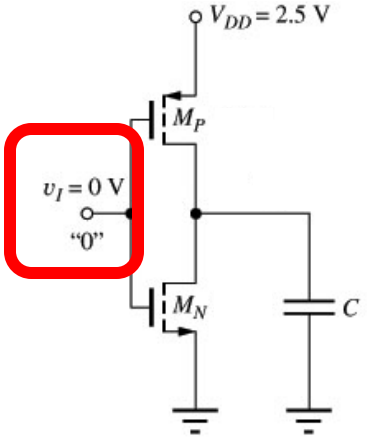


(b)

$$v_I = 2.5\text{V (1 state)} \Rightarrow \left\{ \begin{array}{l} v_{GS} = 2.5 > V_{TN} (= 0.6) \text{ for NMOS} \\ v_{GS} = 0 > V_{TP} (= -0.6) \text{ for PMOS} \end{array} \right\}$$

$$\Rightarrow \left\{ \begin{array}{l} M_N \text{ is "On"} \\ M_P \text{ is "Off"} \end{array} \right\} \Rightarrow v_O = 0 = V_L \text{ (0 state)}$$

- The capacitor discharges through R_{onN}, current exists only during discharge, **no dc current exists.**

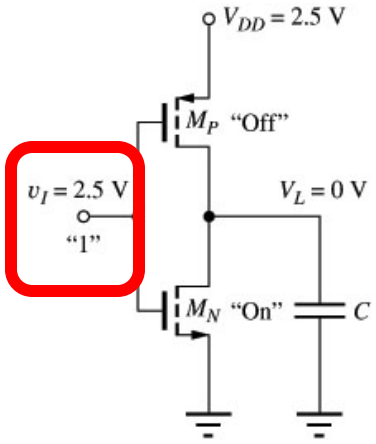


(c)

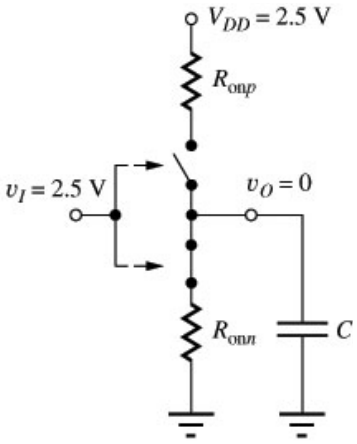
$$v_I = 0\text{V (0 state)} \Rightarrow \left\{ \begin{array}{l} v_{GS} = 0 < V_{TN} (= 0.6) \text{ for NMOS} \end{array} \right\}$$

$$\Rightarrow$$

Static Characteristics of the CMOS Inverter



(a)

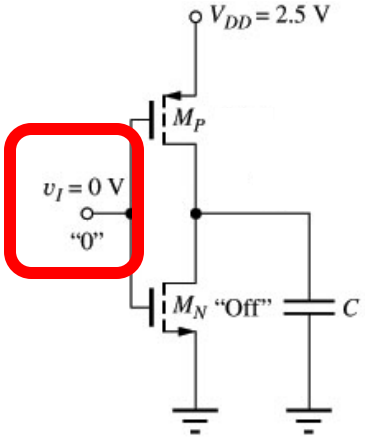


(b)

$$v_I = 2.5\text{V (1 state)} \Rightarrow \left\{ \begin{array}{l} v_{GS} = 2.5 > V_{TN}(= 0.6) \text{ for NMOS} \\ v_{GS} = 0 > V_{TP}(= -0.6) \text{ for PMOS} \end{array} \right\}$$

$$\Rightarrow \left\{ \begin{array}{l} M_N \text{ is "On"} \\ M_P \text{ is "Off"} \end{array} \right\} \Rightarrow v_O = 0 = V_L \text{ (0 state)}$$

- The capacitor discharges through R_{onN}, current exists only during discharge, **no dc current exists.**

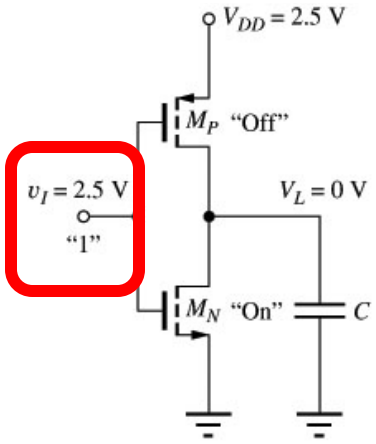


(c)

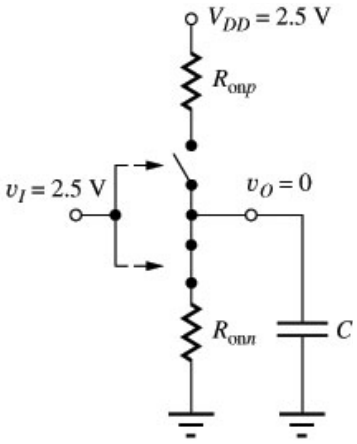
$$v_I = 0\text{V (0 state)} \Rightarrow \left\{ \begin{array}{l} v_{GS} = 0 < V_{TN}(= 0.6) \text{ for NMOS} \end{array} \right\}$$

$$\Rightarrow \left\{ M_N \text{ is "Off"} \right\} \Rightarrow$$

Static Characteristics of the CMOS Inverter



(a)

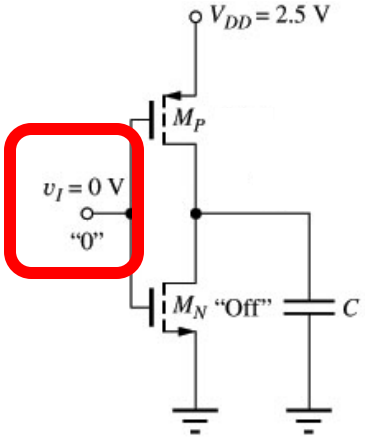


(b)

$$v_I = 2.5\text{V (1 state)} \Rightarrow \begin{Bmatrix} v_{GS} = 2.5 > V_{TN}(= 0.6) \text{ for NMOS} \\ v_{GS} = 0 > V_{TP}(= -0.6) \text{ for PMOS} \end{Bmatrix}$$

$$\Rightarrow \begin{Bmatrix} M_N \text{ is "On"} \\ M_P \text{ is "Off"} \end{Bmatrix} \Rightarrow v_O = 0 = V_L \text{ (0 state)}$$

- The capacitor discharges through R_{onN}, current exists only during discharge, **no dc current exists.**

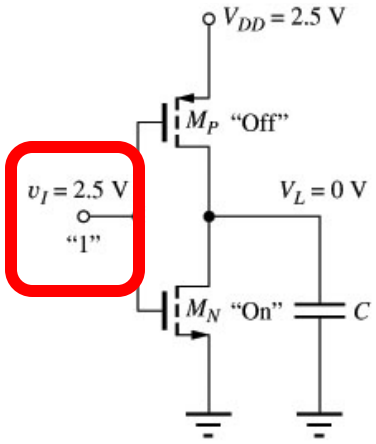


(c)

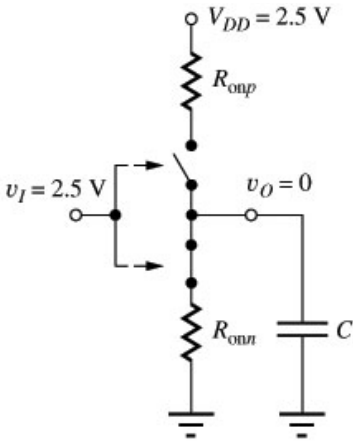
$$v_I = 0\text{V (0 state)} \Rightarrow \begin{Bmatrix} v_{GS} = 0 < V_{TN}(= 0.6) \text{ for NMOS} \\ v_{GS} = \quad V_{TP} \quad \text{for PMOS} \end{Bmatrix}$$

$$\Rightarrow \begin{Bmatrix} M_N \text{ is "Off"} \end{Bmatrix} \Rightarrow$$

Static Characteristics of the CMOS Inverter



(a)

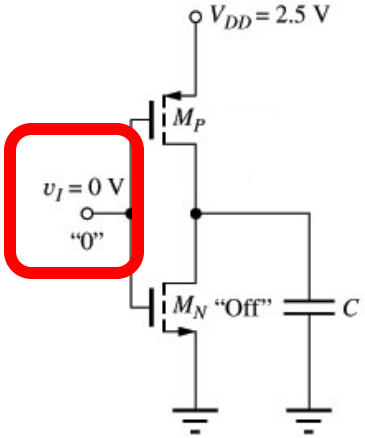


(b)

$$v_I = 2.5\text{V (1 state)} \Rightarrow \left\{ \begin{array}{l} v_{GS} = 2.5 > V_{TN}(= 0.6) \text{ for NMOS} \\ v_{GS} = 0 > V_{TP}(= -0.6) \text{ for PMOS} \end{array} \right\}$$

$$\Rightarrow \left\{ \begin{array}{l} M_N \text{ is "On"} \\ M_P \text{ is "Off"} \end{array} \right\} \Rightarrow v_O = 0 = V_L \text{ (0 state)}$$

- The capacitor discharges through R_{onN}, current exists only during discharge, **no dc current exists.**

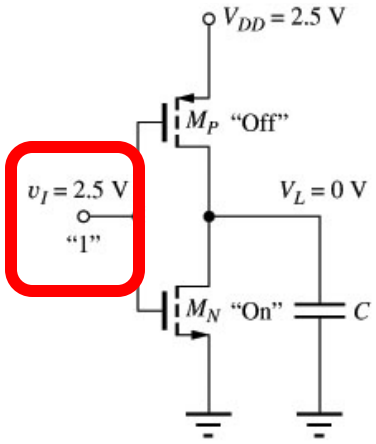


(c)

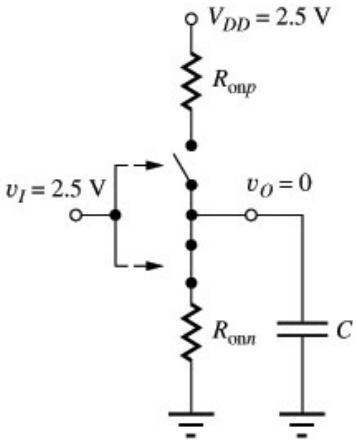
$$v_I = 0\text{V (0 state)} \Rightarrow \left\{ \begin{array}{l} v_{GS} = 0 < V_{TN}(= 0.6) \text{ for NMOS} \\ v_{GS} = -2.5 < V_{TP}(= -0.6) \text{ for PMOS} \end{array} \right\}$$

$$\Rightarrow \left\{ M_N \text{ is "Off"} \right\} \Rightarrow$$

Static Characteristics of the CMOS Inverter



(a)

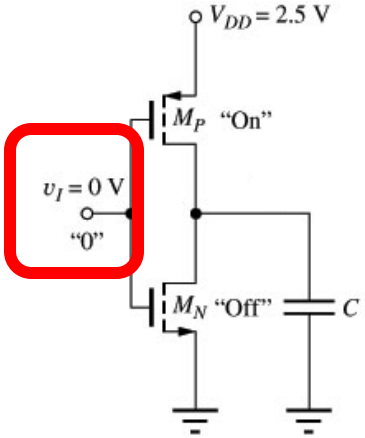


(b)

$$v_I = 2.5\text{V (1 state)} \Rightarrow \begin{Bmatrix} v_{GS} = 2.5 > V_{TN}(= 0.6) \text{ for NMOS} \\ v_{GS} = 0 > V_{TP}(= -0.6) \text{ for PMOS} \end{Bmatrix}$$

$$\Rightarrow \begin{Bmatrix} M_N \text{ is "On"} \\ M_P \text{ is "Off"} \end{Bmatrix} \Rightarrow v_O = 0 = V_L \text{ (0 state)}$$

- The capacitor discharges through R_{onN}, current exists only during discharge, **no dc current exists.**

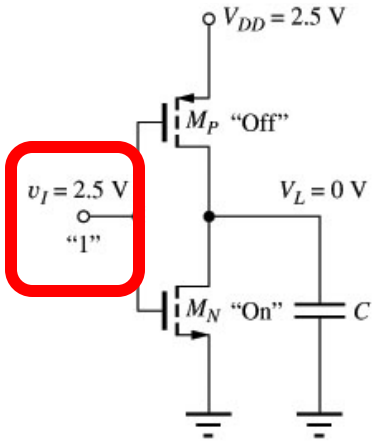


(c)

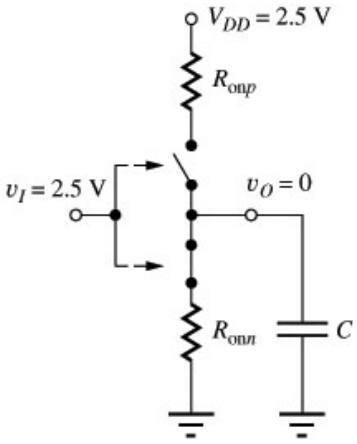
$$v_I = 0\text{V (0 state)} \Rightarrow \begin{Bmatrix} v_{GS} = 0 < V_{TN}(= 0.6) \text{ for NMOS} \\ v_{GS} = -2.5 < V_{TP}(= -0.6) \text{ for PMOS} \end{Bmatrix}$$

$$\Rightarrow \begin{Bmatrix} M_N \text{ is "Off"} \\ M_P \text{ is "On"} \end{Bmatrix} \Rightarrow$$

Static Characteristics of the CMOS Inverter



(a)

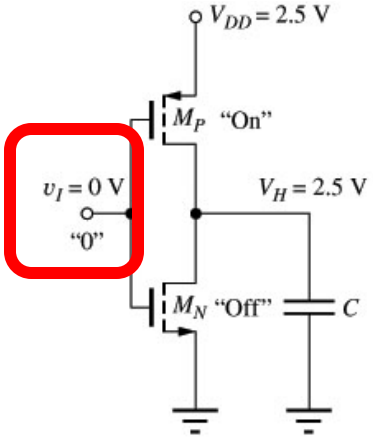


(b)

$$v_I = 2.5\text{V (1 state)} \Rightarrow \begin{Bmatrix} v_{GS} = 2.5 > V_{TN}(= 0.6) \text{ for NMOS} \\ v_{GS} = 0 > V_{TP}(= -0.6) \text{ for PMOS} \end{Bmatrix}$$

$$\Rightarrow \begin{Bmatrix} M_N \text{ is "On"} \\ M_P \text{ is "Off"} \end{Bmatrix} \Rightarrow v_O = 0 = V_L \text{ (0 state)}$$

- The capacitor discharges through R_{onN}, current exists only during discharge, **no dc current exists.**

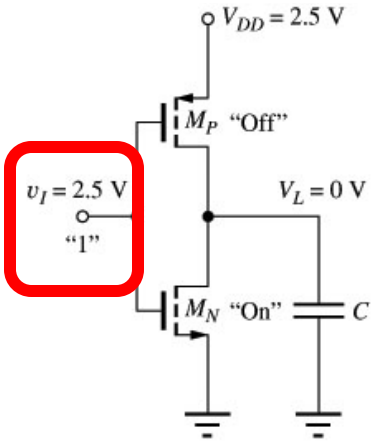


(c)

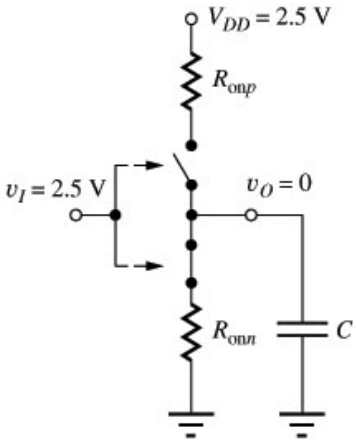
$$v_I = 0\text{V (0 state)} \Rightarrow \begin{Bmatrix} v_{GS} = 0 < V_{TN}(= 0.6) \text{ for NMOS} \\ v_{GS} = -2.5 < V_{TP}(= -0.6) \text{ for PMOS} \end{Bmatrix}$$

$$\Rightarrow \begin{Bmatrix} M_N \text{ is "Off"} \\ M_P \text{ is "On"} \end{Bmatrix} \Rightarrow v_O = V_{DD} = V_H \text{ (1 state)}$$

Static Characteristics of the CMOS Inverter



(a)

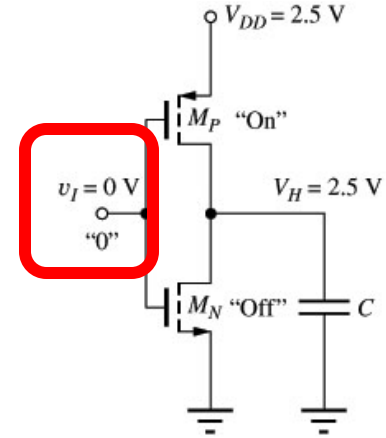


(b)

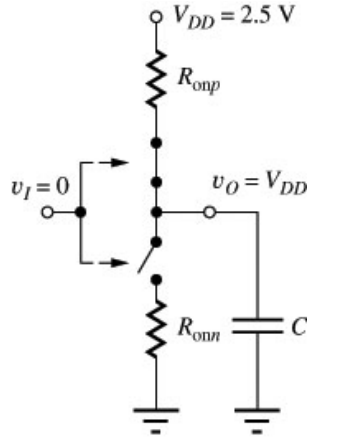
$$v_I = 2.5\text{V (1 state)} \Rightarrow \begin{Bmatrix} v_{GS} = 2.5 > V_{TN}(= 0.6) \text{ for NMOS} \\ v_{GS} = 0 > V_{TP}(= -0.6) \text{ for PMOS} \end{Bmatrix}$$

$$\Rightarrow \begin{Bmatrix} M_N \text{ is "On"} \\ M_P \text{ is "Off"} \end{Bmatrix} \Rightarrow v_O = 0 = V_L \text{ (0 state)}$$

- The capacitor discharges through R_{onN}, current exists only during discharge, **no dc current exists.**



(c)

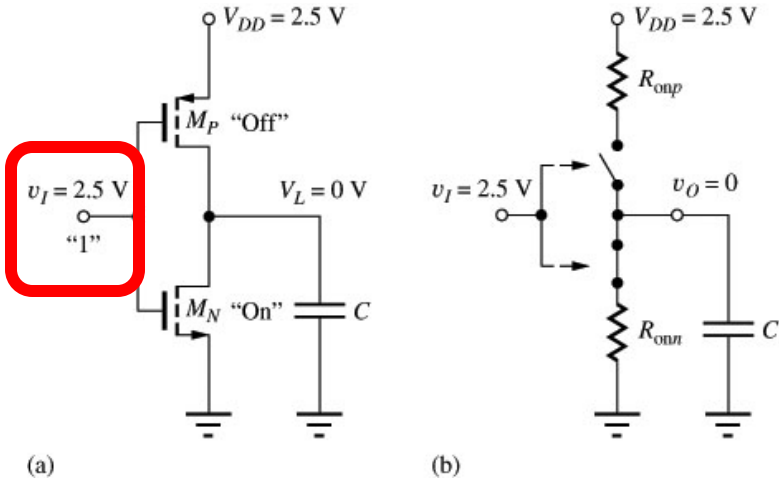


(d)

$$v_I = 0\text{V (0 state)} \Rightarrow \begin{Bmatrix} v_{GS} = 0 < V_{TN}(= 0.6) \text{ for NMOS} \\ v_{GS} = -2.5 < V_{TP}(= -0.6) \text{ for PMOS} \end{Bmatrix}$$

$$\Rightarrow \begin{Bmatrix} M_N \text{ is "Off"} \\ M_P \text{ is "On"} \end{Bmatrix} \Rightarrow v_O = V_{DD} = V_H \text{ (1 state)}$$

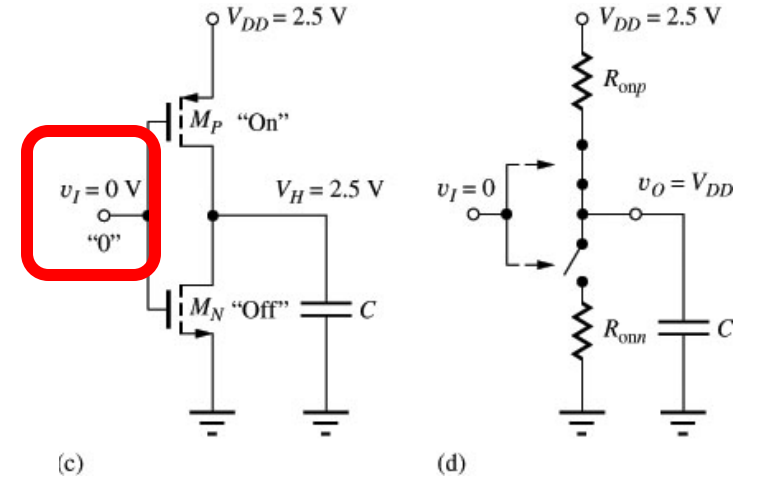
Static Characteristics of the CMOS Inverter



$$v_I = 2.5\text{V (1 state)} \Rightarrow \begin{Bmatrix} v_{GS} = 2.5 > V_{TN}(=0.6) \text{ for NMOS} \\ v_{GS} = 0 > V_{TP}(=-0.6) \text{ for PMOS} \end{Bmatrix}$$

$$\Rightarrow \begin{Bmatrix} M_N \text{ is "On"} \\ M_P \text{ is "Off"} \end{Bmatrix} \Rightarrow v_O = 0 = V_L \text{ (0 state)}$$

- The capacitor discharges through R_{onN}, current exists only during discharge, **no dc current exists.**



$$v_I = 0\text{V (0 state)} \Rightarrow \begin{Bmatrix} v_{GS} = 0 < V_{TN}(=0.6) \text{ for NMOS} \\ v_{GS} = -2.5 < V_{TP}(=-0.6) \text{ for PMOS} \end{Bmatrix}$$

$$\Rightarrow \begin{Bmatrix} M_N \text{ is "Off"} \\ M_P \text{ is "On"} \end{Bmatrix} \Rightarrow v_O = V_{DD} = V_H \text{ (1 state)}$$

- The capacitor charges through R_{onP}, current exists only during charging, **no dc current exists.**

CMOS Inverter – building VTC

- To better understand what's happening in inverter and to get the complete model we need to build the VTC - voltage transfer characteristics.

CMOS Inverter – building VTC

- To better understand what's happening in inverter and to get the complete model we need to build the VTC - voltage transfer characteristics.
- To construct VTC we use the **load line** method for different load states.

CMOS Inverter – building VTC

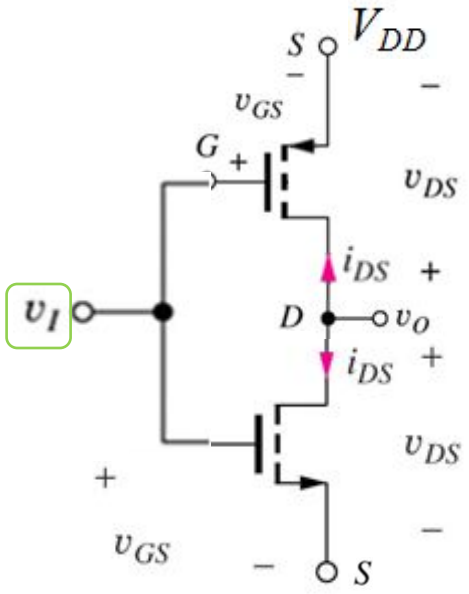
- To better understand what's happening in inverter and to get the complete model we need to build the VTC - voltage transfer characteristics.
- To construct VTC we use the **load line** method for different load states.
- However, since the load in this case is also a nonlinear transistor, the "load line" approach will consist of superimposing *I-V* characteristics of NMOS and PMOS transistors.

CMOS Inverter – building VTC

- To better understand what's happening in inverter and to get the complete model we need to build the VTC - voltage transfer characteristics.
- To construct VTC we use the **load line** method for different load states.
- However, since the load in this case is also a nonlinear transistor, the "load line" approach will consist of superimposing *I-V* characteristics of NMOS and PMOS transistors.
- It requires that the *I-V* curves of the NMOS and PMOS devices are transformed onto a common coordinate set.

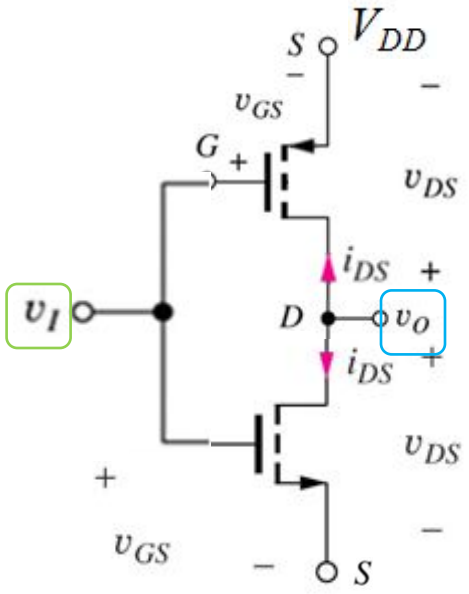
CMOS Inverter – building VTC

- To better understand what's happening in inverter and to get the complete model we need to build the VTC - voltage transfer characteristics.
- To construct VTC we use the **load line** method for different load states.
- However, since the load in this case is also a nonlinear transistor, the "load line" approach will consist of superimposing *I-V* characteristics of NMOS and PMOS transistors.
- It requires that the *I-V* curves of the NMOS and PMOS devices are transformed onto a common coordinate set.
- We select the input voltage V_I



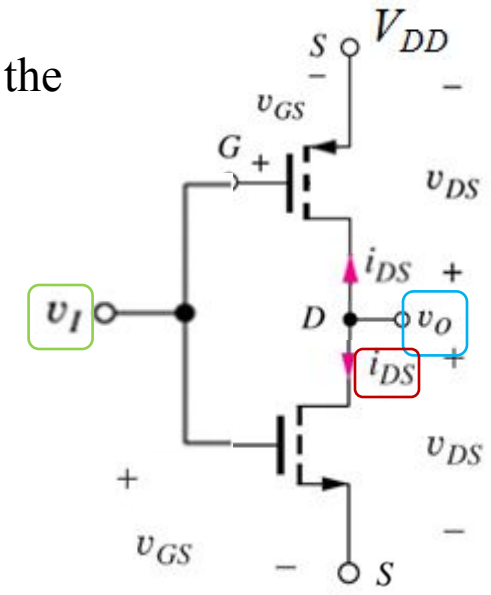
CMOS Inverter – building VTC

- To better understand what's happening in inverter and to get the complete model we need to build the VTC - voltage transfer characteristics.
- To construct VTC we use the **load line** method for different load states.
- However, since the load in this case is also a nonlinear transistor, the "load line" approach will consist of superimposing *I-V* characteristics of NMOS and PMOS transistors.
- It requires that the *I-V* curves of the NMOS and PMOS devices are transformed onto a common coordinate set.
- We select the input voltage V_I, the output voltage V_O



CMOS Inverter – building VTC

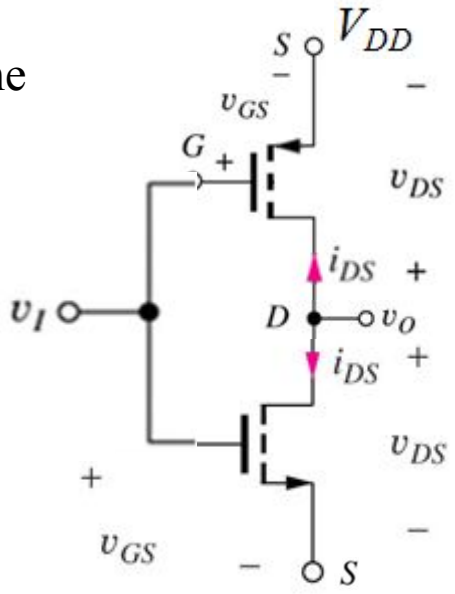
- To better understand what's happening in inverter and to get the complete model we need to build the VTC - voltage transfer characteristics.
- To construct VTC we use the **load line** method for different load states.
- However, since the load in this case is also a nonlinear transistor, the "load line" approach will consist of superimposing *I-V* characteristics of NMOS and PMOS transistors.
- It requires that the *I-V* curves of the NMOS and PMOS devices are transformed onto a common coordinate set.
- We select the input voltage V_I, the output voltage V_O, and the NMOS drain current I_{DN} as the variables of choice.



CMOS Inverter – building VTC

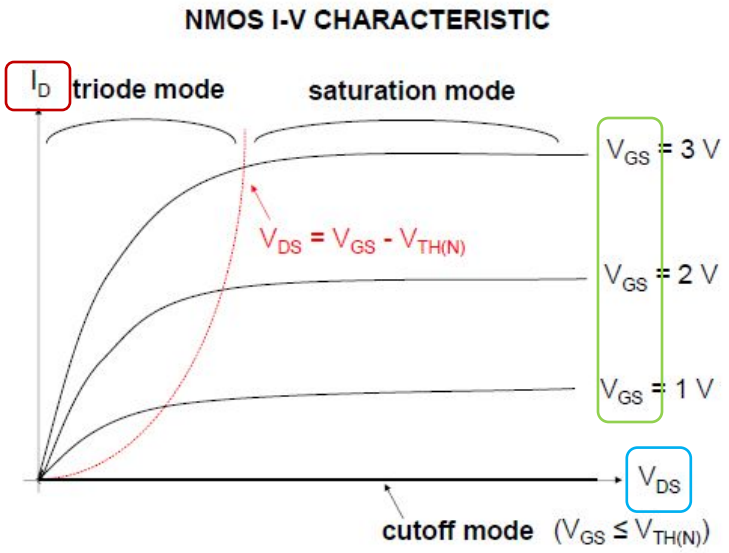
- To better understand what's happening in inverter and to get the complete model we need to build the VTC - voltage transfer characteristics.
- To construct VTC we use the **load line** method for different load states.
- However, since the load in this case is also a nonlinear transistor, the "load line" approach will consist of superimposing *I-V* characteristics of NMOS and PMOS transistors.
- It requires that the *I-V* curves of the NMOS and PMOS devices are transformed onto a common coordinate set.
- We select the input voltage V_I, the output voltage V_O, and the NMOS drain current I_{DSN} as the variables of choice.
- The PMOS *I-V* relationship can be transformed as follows:

$$I_{DSp} = -I_{DSn}$$
$$V_{GSn} = V_I; \quad V_{GSp} = V_I - V_{DD}$$
$$V_{DSn} = V_O; \quad V_{DSp} = V_O - V_{DD}$$



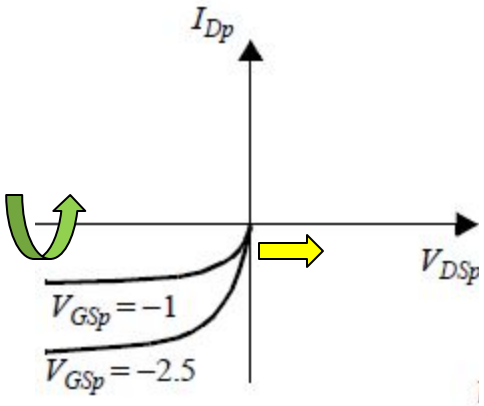
CMOS Inverter – building VTC

- The *I-V* curves for NMOS are already plotted in the selected coordinate set V_{in}, V_{out} and I_{DN}, so no change is needed.



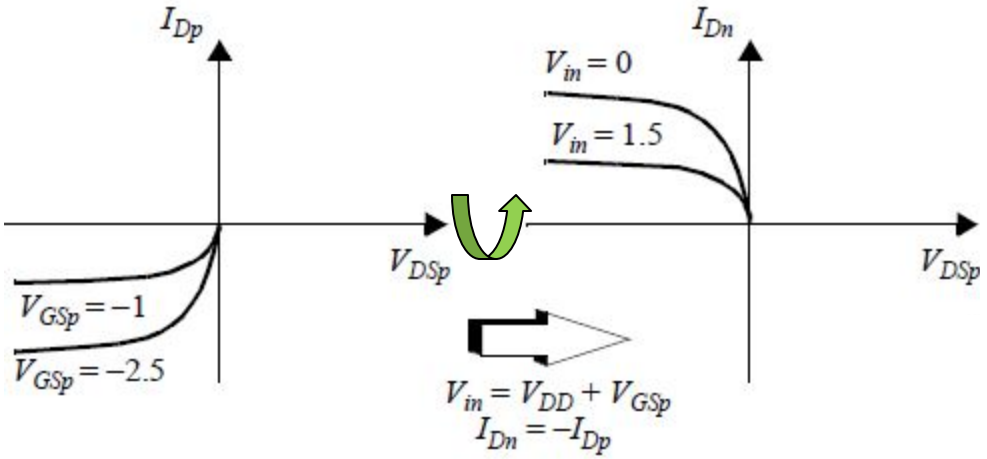
CMOS Inverter – building VTC

- The *I-V* curves for NMOS are already plotted in the selected coordinate set V_{in}, V_{out} and I_{DN}, so no change is needed.
- The load-line curves of the PMOS device are obtained by a mirroring around the *x*-axis and a horizontal shift over V_{DD}.



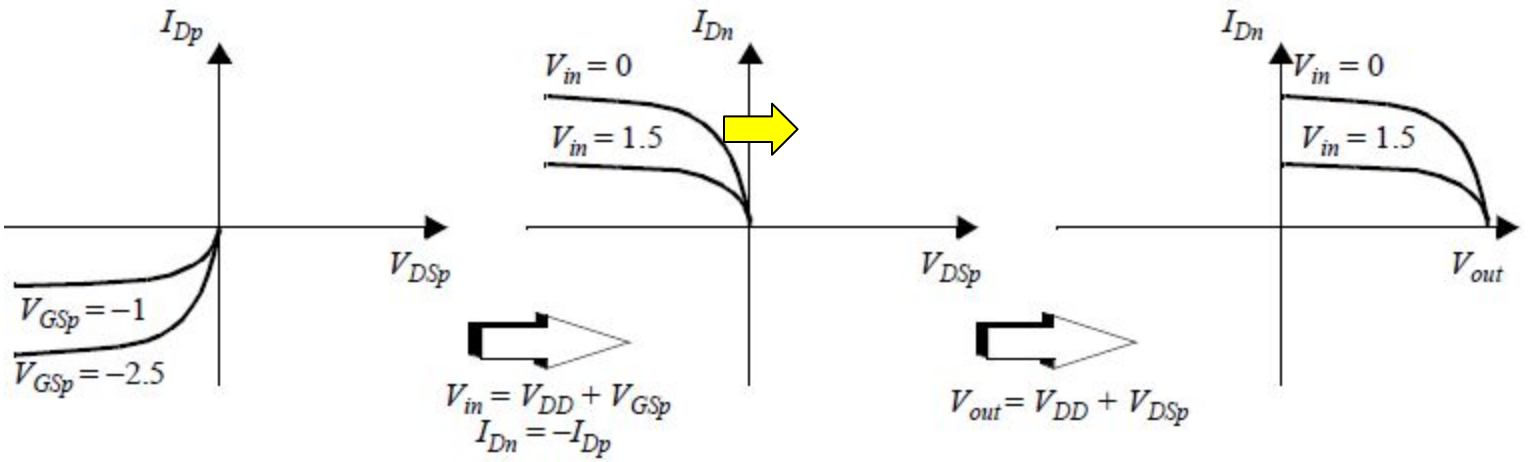
CMOS Inverter – building VTC

- The *I-V* curves for NMOS are already plotted in the selected coordinate set V_{in}, V_{out} and I_{DN}, so no change is needed.
- The load-line curves of the PMOS device are obtained by a mirroring around the *x*-axis and a horizontal shift over V_{DD}.
- This procedure is outlined below, where the subsequent steps to adjust the original PMOS *I-V* curves to the common coordinate set V_{in}, V_{out} and I_{Dn} are illustrated (in this example $V_{DD} = 2.5$V.



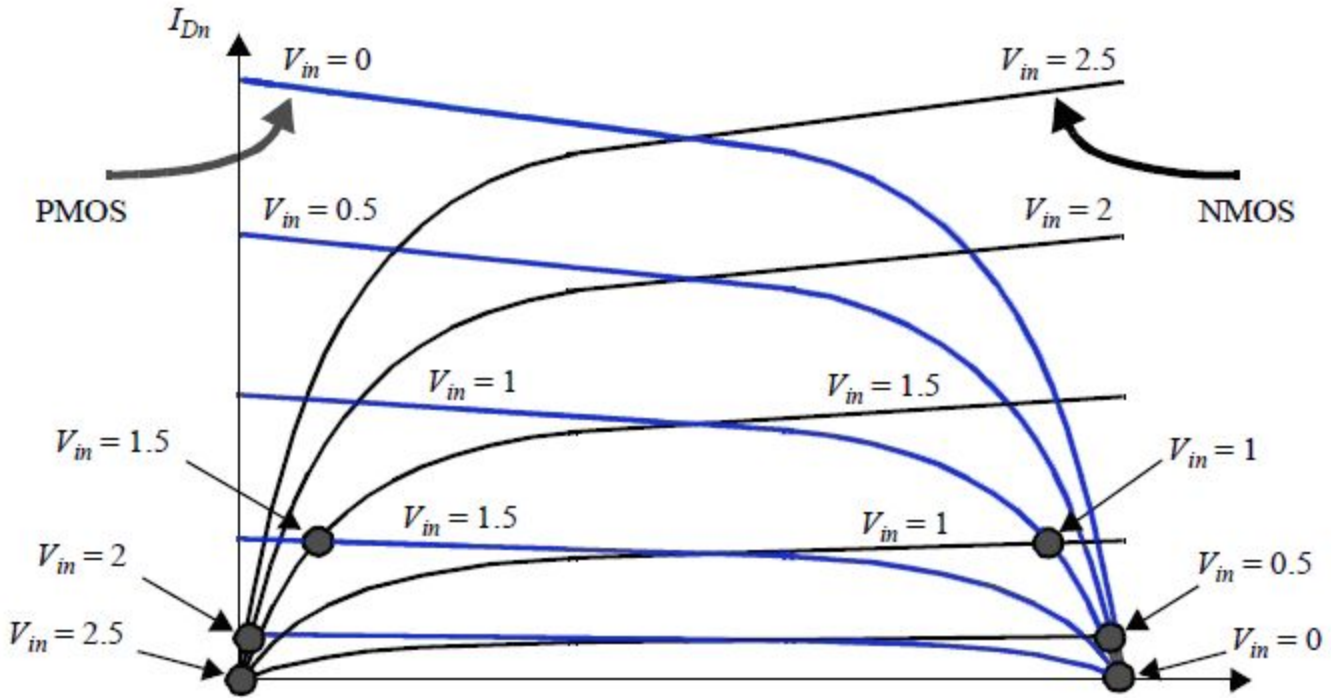
CMOS Inverter – building VTC

- The *I-V* curves for NMOS are already plotted in the selected coordinate set V_{in}, V_{out} and I_{DN}, so no change is needed.
- The load-line curves of the PMOS device are obtained by a mirroring around the *x*-axis and a horizontal shift over V_{DD}.
- This procedure is outlined below, where the subsequent steps to adjust the original PMOS *I-V* curves to the common coordinate set V_{in}, V_{out} and I_{Dn} are illustrated (in this example $V_{DD} = 2.5$V.



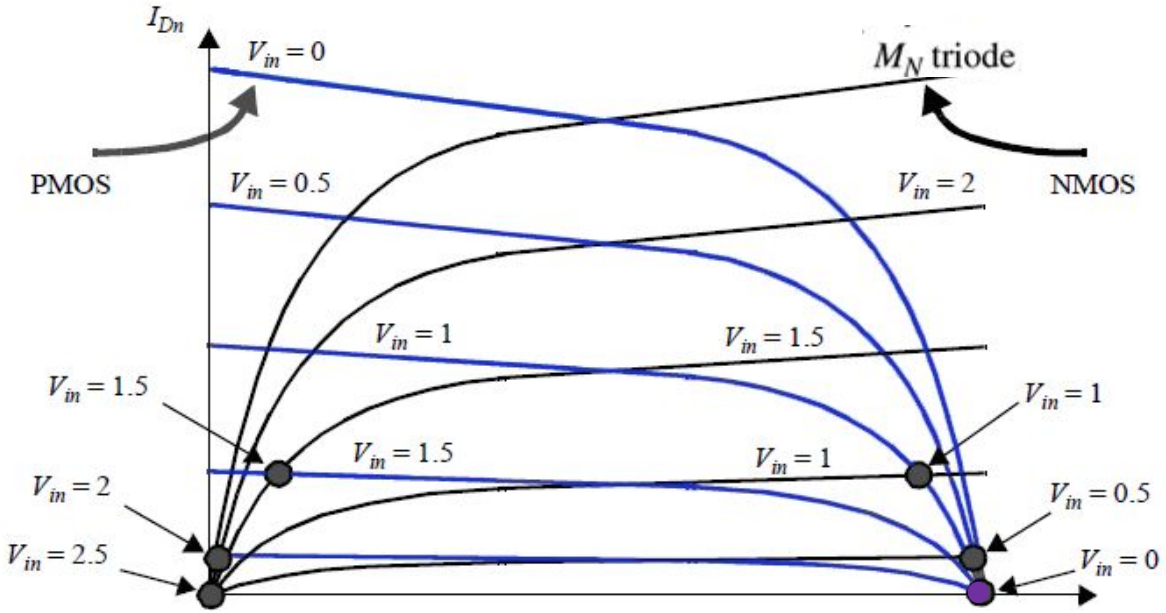
CMOS Inverter – building VTC

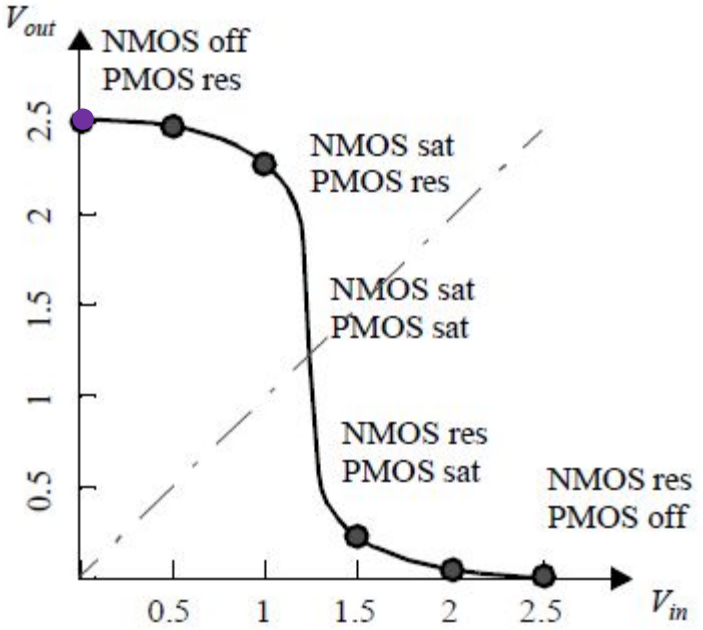
- Now we can superimpose the load curves of the PMOS on the *IV* curves of the NMOS.
- For a dc operating points to be valid, the currents through the NMOS and PMOS devices must be equal. Graphically, this means that the dc points must be located at the intersection of corresponding load lines.
- Some of those points (for *Vin* = 0, 0.5, 1, 1.5, 2, and 2.5 V) are marked on the graph.



CMOS Inverter – building VTC

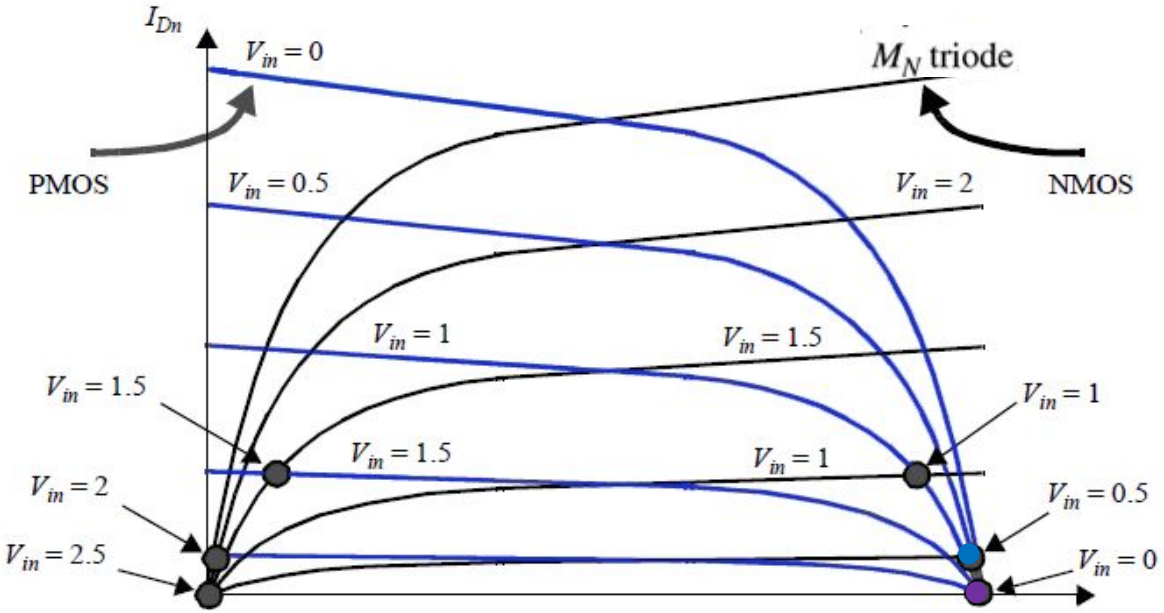
- All operating points are located either at the high or low output levels.
- The VTC of the inverter hence exhibits a very narrow transition zone.
- This results from the high gain during the switching transient, when both NMOS and PMOS are simultaneously on, and in saturation.
- In that operation region, a small change in the input voltage results in a large output variation.

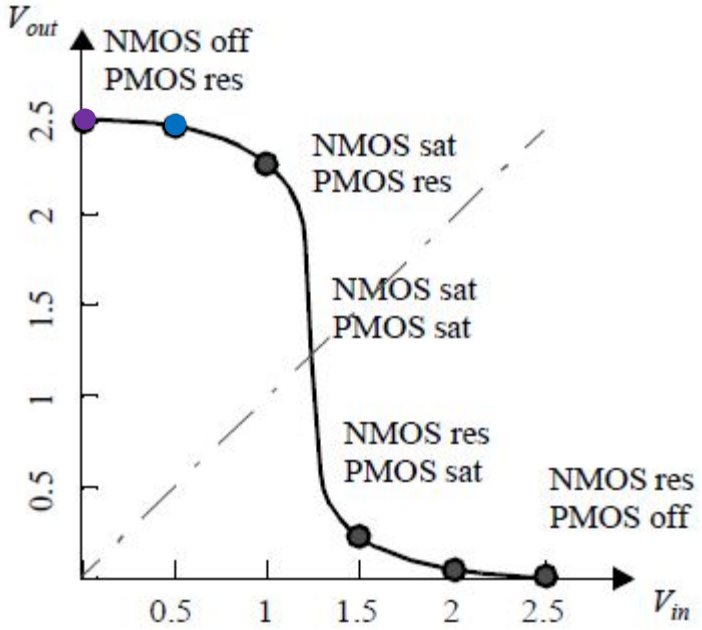




CMOS Inverter – building VTC

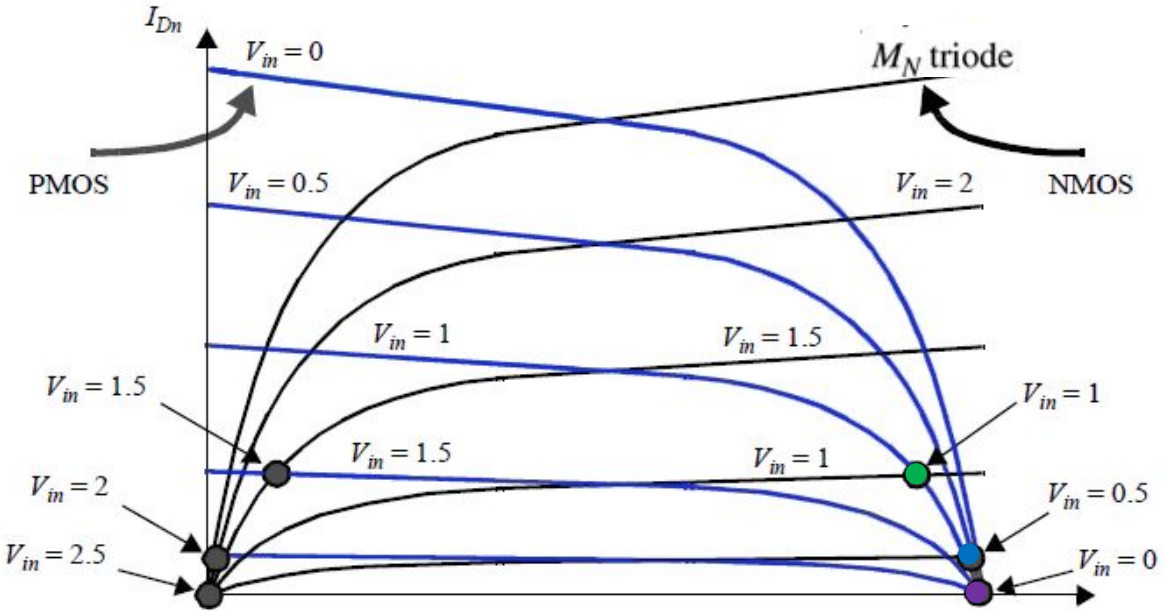
- All operating points are located either at the high or low output levels.
- The VTC of the inverter hence exhibits a very narrow transition zone.
- This results from the high gain during the switching transient, when both NMOS and PMOS are simultaneously on, and in saturation.
- In that operation region, a small change in the input voltage results in a large output variation.

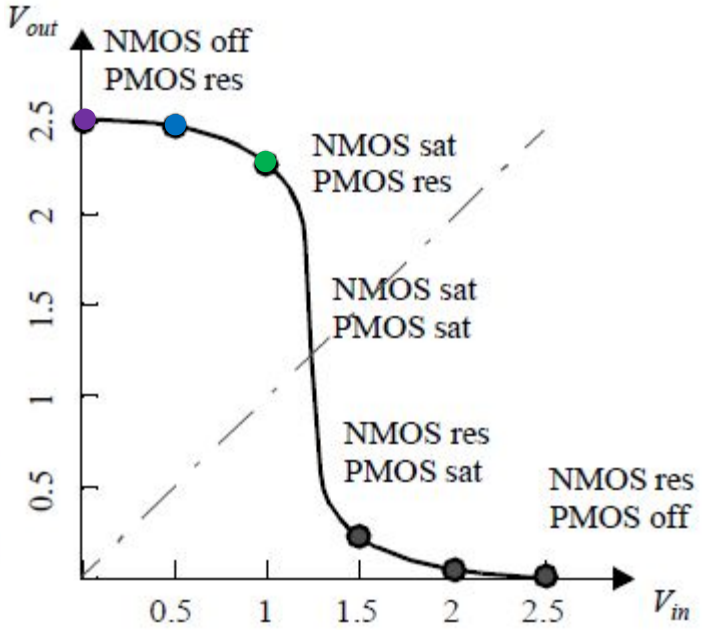




CMOS Inverter – building VTC

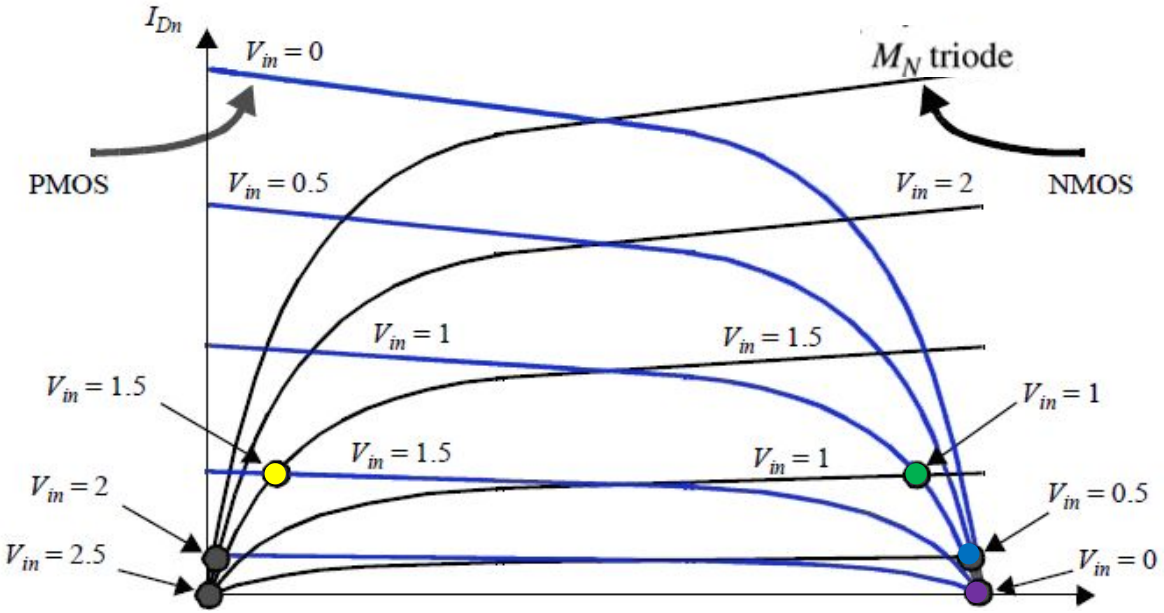
- All operating points are located either at the high or low output levels.
- The VTC of the inverter hence exhibits a very narrow transition zone.
- This results from the high gain during the switching transient, when both NMOS and PMOS are simultaneously on, and in saturation.
- In that operation region, a small change in the input voltage results in a large output variation.

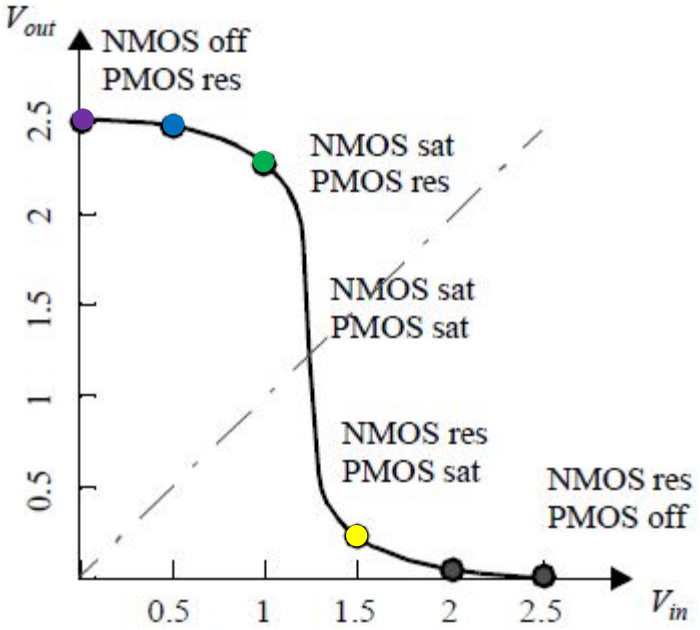




CMOS Inverter – building VTC

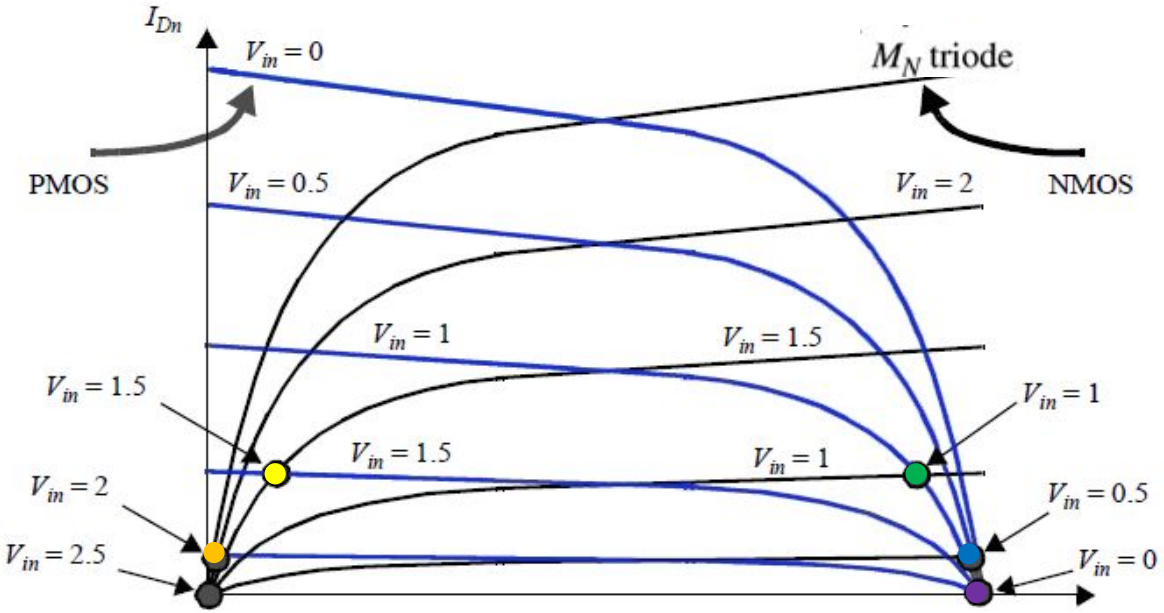
- All operating points are located either at the high or low output levels.
- The VTC of the inverter hence exhibits a very narrow transition zone.
- This results from the high gain during the switching transient, when both NMOS and PMOS are simultaneously on, and in saturation.
- In that operation region, a small change in the input voltage results in a large output variation.

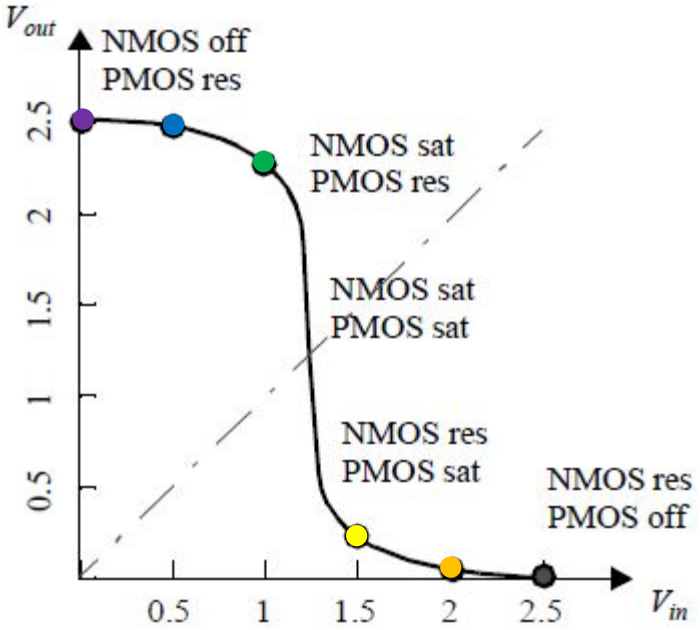




CMOS Inverter – building VTC

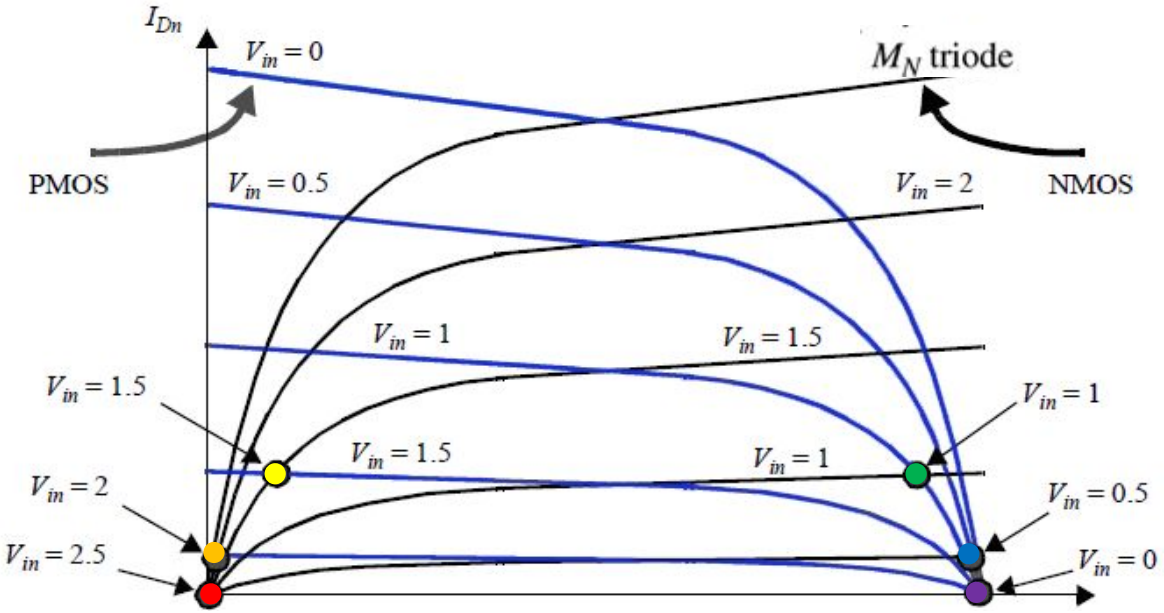
- All operating points are located either at the high or low output levels.
- The VTC of the inverter hence exhibits a very narrow transition zone.
- This results from the high gain during the switching transient, when both NMOS and PMOS are simultaneously on, and in saturation.
- In that operation region, a small change in the input voltage results in a large output variation.

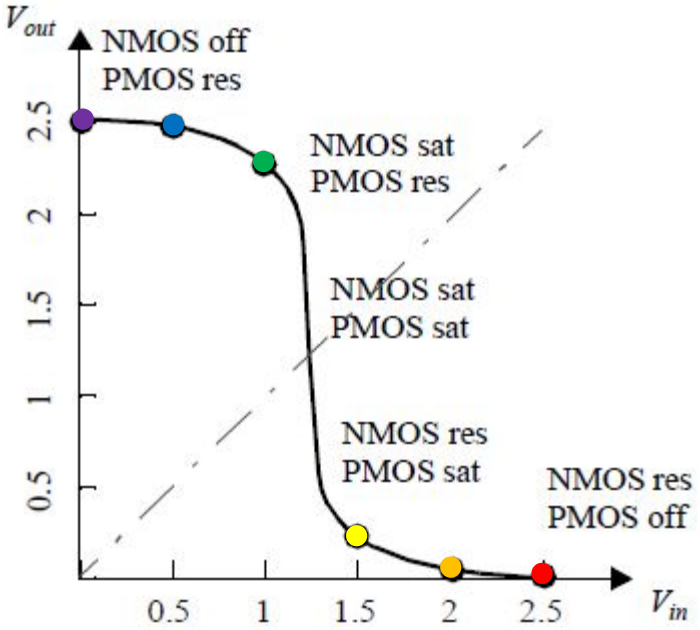




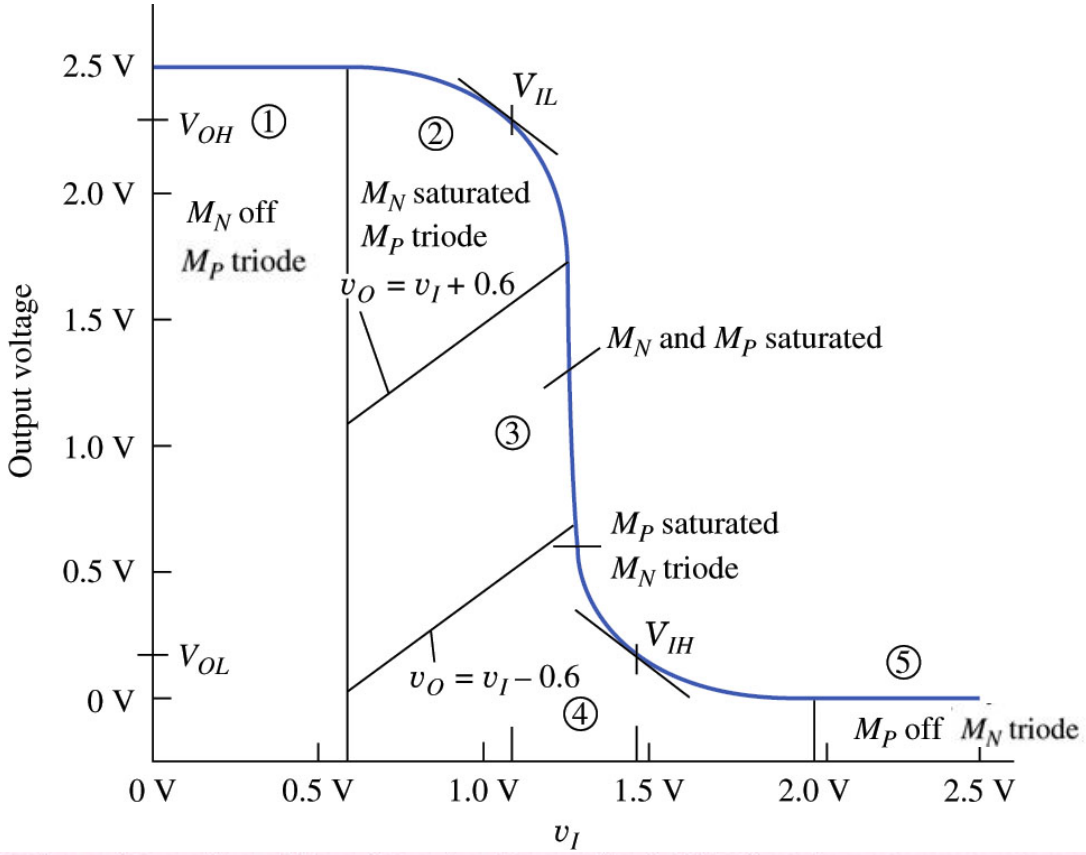
CMOS Inverter – building VTC

- All operating points are located either at the high or low output levels.
- The VTC of the inverter hence exhibits a very narrow transition zone.
- This results from the high gain during the switching transient, when both NMOS and PMOS are simultaneously on, and in saturation.
- In that operation region, a small change in the input voltage results in a large output variation.

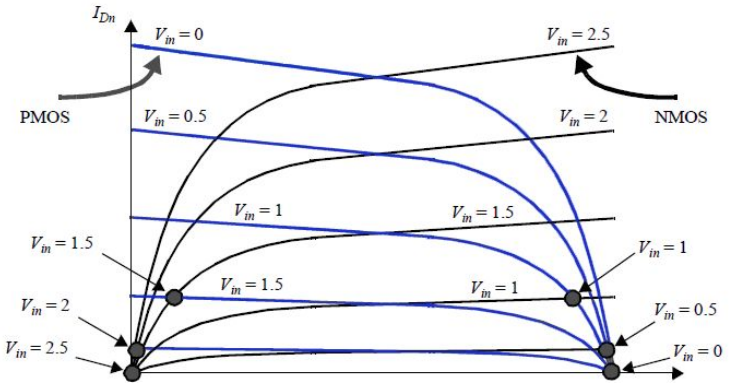




CMOS Voltage Transfer Characteristics

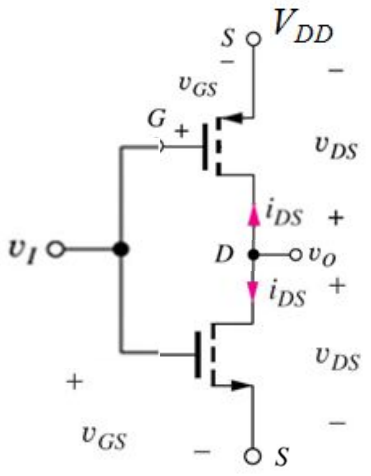


Symmetrical CMOS inverter ($K_p = K_n$).

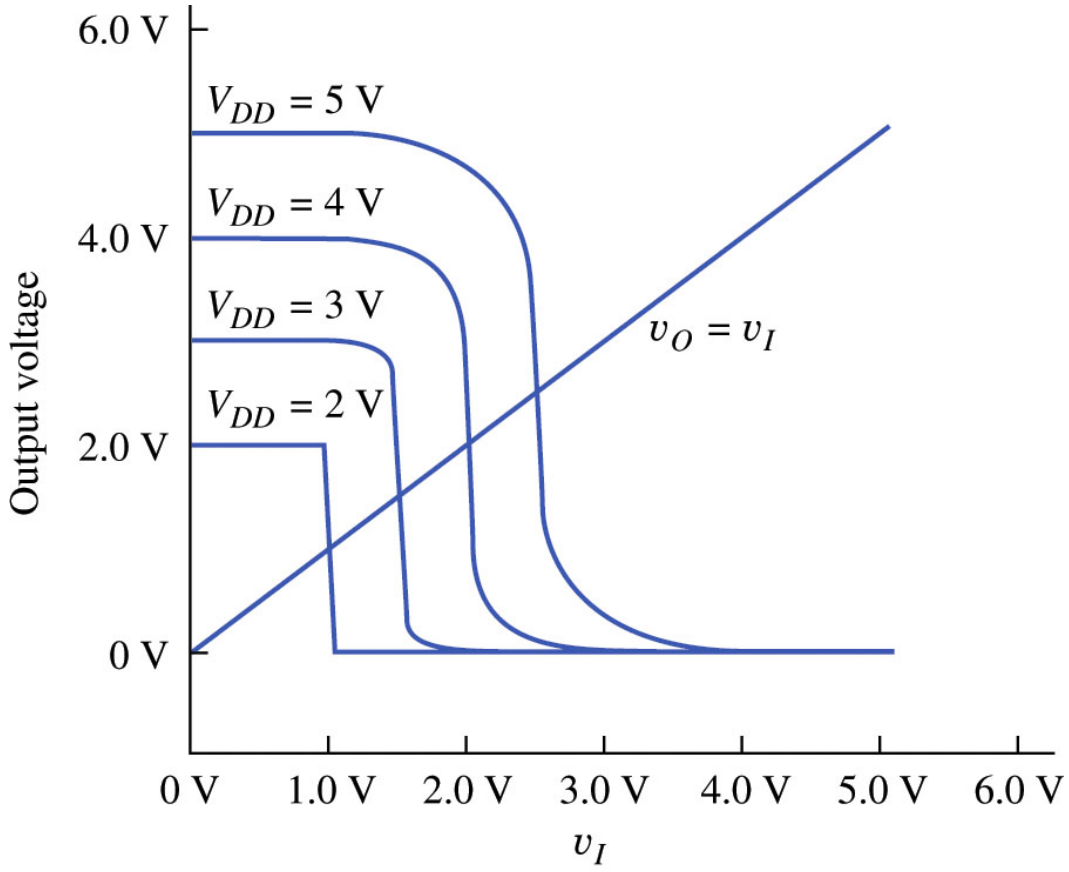


Regions of Operation of Transistors in a Symmetrical CMOS Inverter

REGION	INPUT VOLTAGE v_I	OUTPUT VOLTAGE v_O	NMOS TRANSISTOR	PMOS TRANSISTOR
1	$v_I \le V_{TN}$	$V_H = V_{DD}$	Cutoff	Triode
2	$V_{TN} < v_I \le v_O + V_{TP}$	High	Saturation	Triode
3	$v_I \cong V_{DD}/2$	$V_{DD}/2$	Saturation	Saturation
4	$v_O + V_{TN} < v_I \le (V_{DD} - \lvert V_{TP}\rvert)$	Low	Triode	Saturation
5	$v_I \ge (V_{DD} - \lvert V_{TP}\rvert)$	$V_L = 0$	Triode	Cutoff

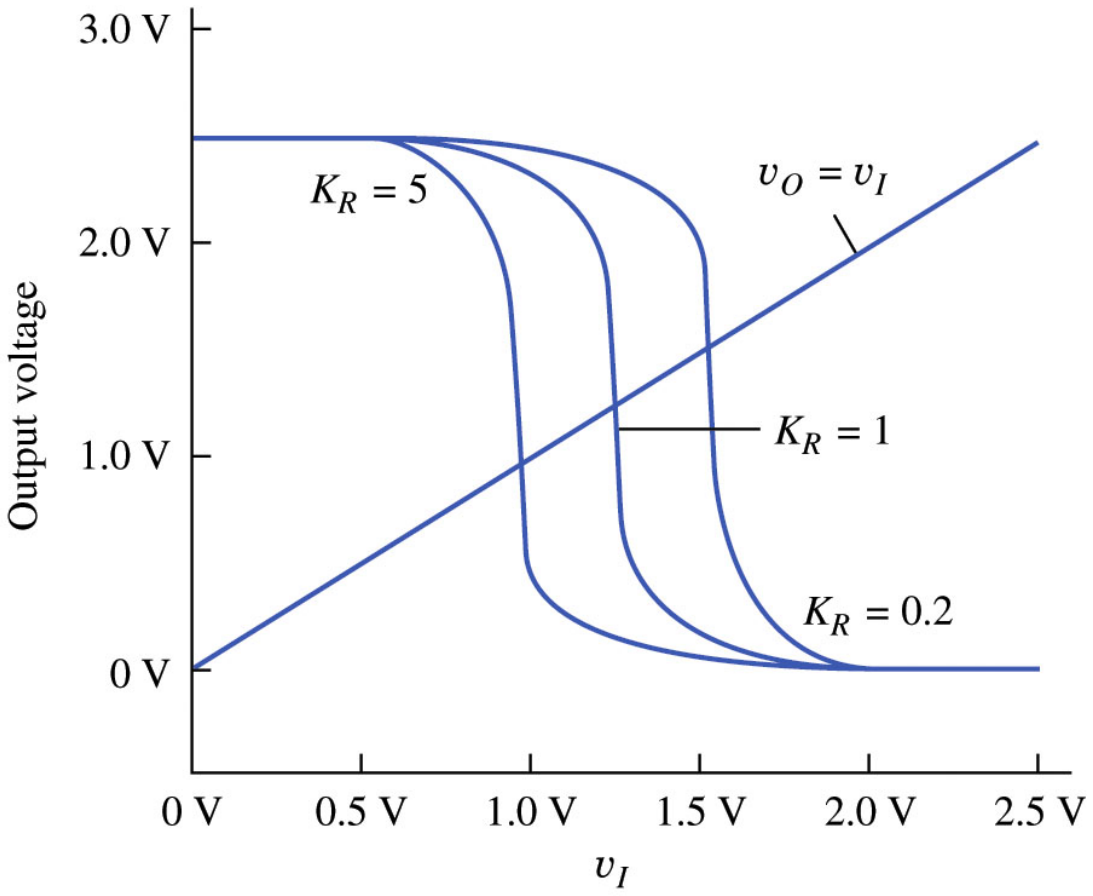


CMOS VTC –Varying V_{DD}



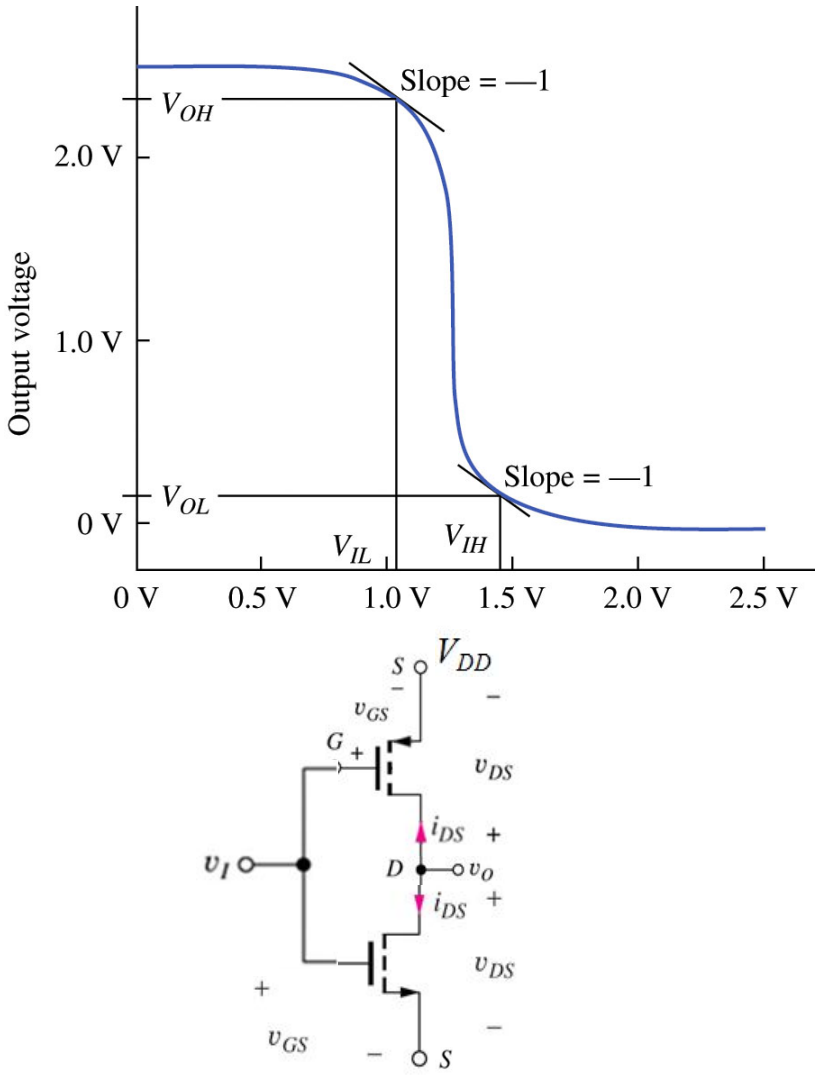
- The simulation results show the changes in VTC of the symmetrical design inverter as V_{DD} is changed
- The transition between V_H and V_L is centered at $V_{DD}/2$ (line $v_O = v_I$)

CMOS VTC –Varying K_N, K_P



- Simulation results show the changes in VTC of the inverter as $K_N/K_P = K_R$ is changed
- For $K_R > 1$ ($K_N > K_P$) the NMOS current drive capability is greater, so the transition region shifts to $v_I < V_{DD}/2$
- For $K_R < 1$ ($K_N < K_P$) the PMOS current drive is greater, and it the transition region shifts toward $v_I > V_{DD}/2$

Noise Margins for the CMOS Inverter



- Noise margins are defined by the points V_{IH} and V_{IL}, at which the slope of VTC is -1.
- For v_I near V_{IH}, V_{DS} is large for PMOS (and small for NMOS PMOS is saturated, NMOS is in triode. Equating currents and using $K_R = K_N/K_P$), we get

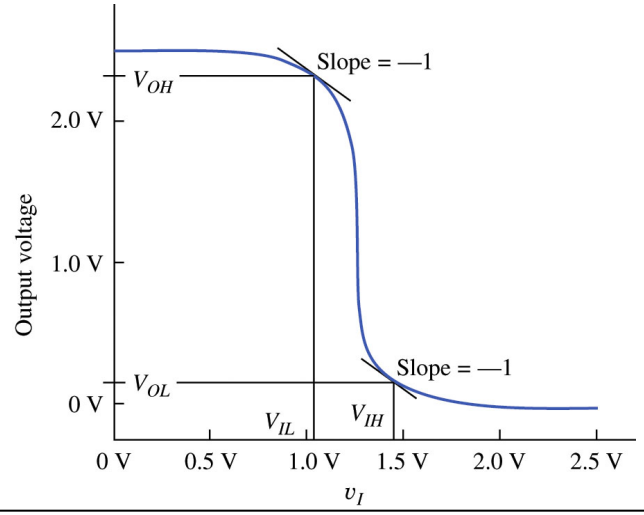
$$K_R(2v_I - 2V_{TN} - v_O)(v_O) = (v_I - V_{DD} - V_{TP})^2$$

$$v_O = (v_I - V_{TN}) \pm \sqrt{(v_I - V_{TN})^2 - \frac{(v_I - V_{DD} - V_{TP})^2}{K_R}}$$

- Taking derivative WRT v_I, and setting it to -1 (quite evolving process) we get
  $\boldsymbol{v_I = V_{IH}}$ and corresponding $\boldsymbol{v_O = V_{OL}}$

- Repeating the process for For v_I near V_{IL} we get
  $\boldsymbol{v_I = V_{IL}}$ and corresponding $\boldsymbol{v_O = V_{OL}}$

Noise Margins for the CMOS Inverter



$$NM_L = V_{IL} - V_{OL}$$

$$NM_H = V_{OH} - V_{IH}$$

---

where $K_R = \frac{K_N}{K_P}$ and

$$V_{IH} = \frac{2K_R(V_{DD} - V_{TN} + V_{TP})}{(K_R - 1)\sqrt{1 + 3K_R}} - \frac{(V_{DD} - K_R V_{TN} + V_{TP})}{K_R - 1}$$

$$V_{OL} = \frac{(K_R + 1)V_{IH} - V_{DD} - K_R V_{TN} - V_{TP}}{2K_R}$$

$$V_{IL} = \frac{2\sqrt{K_R}(V_{DD} - V_{TN} + V_{TP})}{(K_R - 1)\sqrt{K_R + 3}} - \frac{(V_{DD} - K_R V_{TN} + V_{TP})}{K_R - 1}$$

$$V_{OH} = \frac{(K_R + 1)V_{IL} + V_{DD} - K_R V_{TN} - V_{TP}}{2}$$

CMOS Logic Design

- The design of logic gates for CMOC inverter is different from the similar logic design for NMOS inverters that we considered earlier.

WHY?

What is the important difference between

NMOS inverter with load transistor

and

CMOS inverter?

CMOS Logic Design

- The design of logic gates for CMOC inverter is different from the similar logic design for NMOS inverters that we considered earlier.

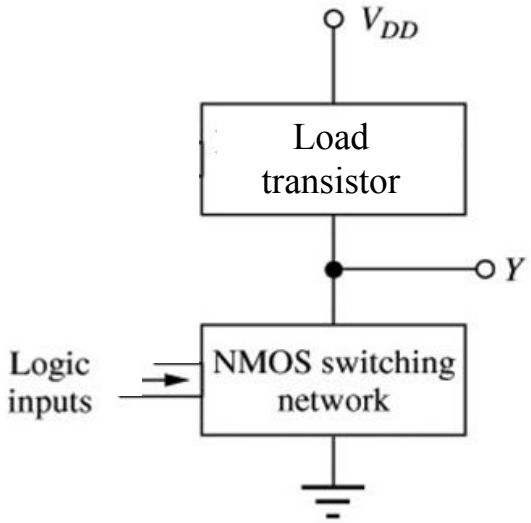
WHY?

What is the important difference between

NMOS inverter with load transistor

and

CMOS inverter?



CMOS Logic Design

- The design of logic gates for CMOC inverter is different from the similar logic design for NMOS inverters that we considered earlier.

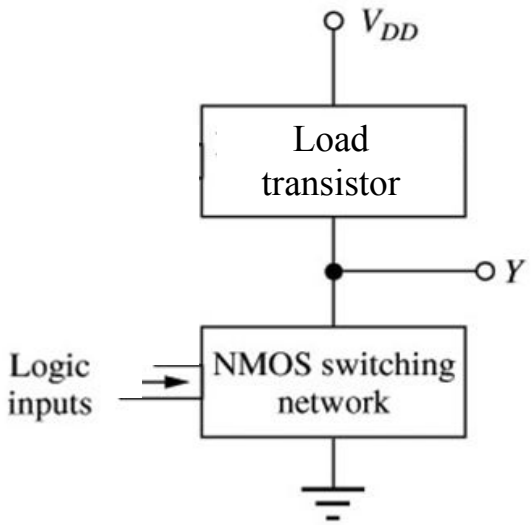
WHY?

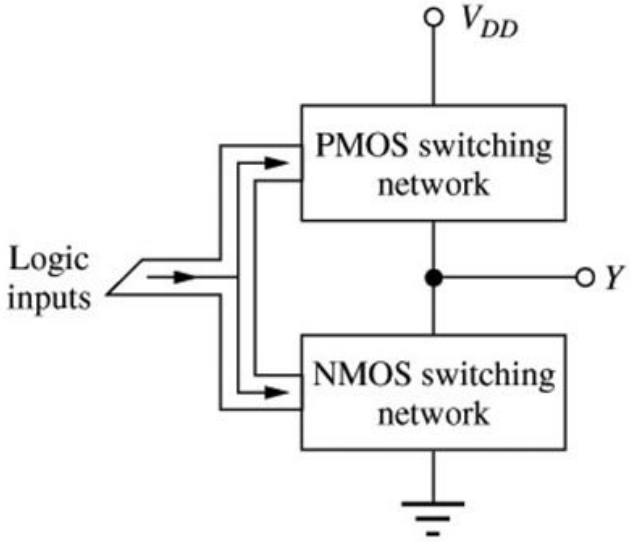
What is the important difference between

NMOS inverter with load transistor

and

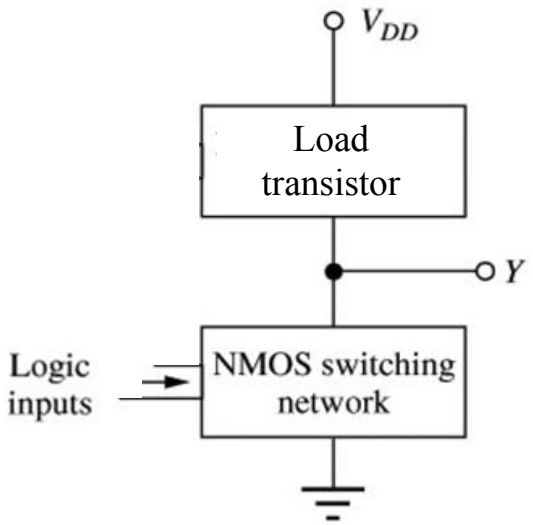
CMOS inverter?





CMOS Logic Design

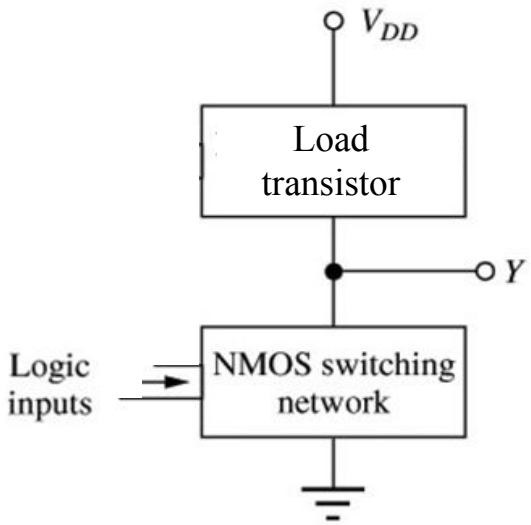
- The design of logic gates for CMOC inverter is different from the similar logic design for NMOS inverters that we considered earlier.
- For NMOS gates, the logic involved only the switching transistor.



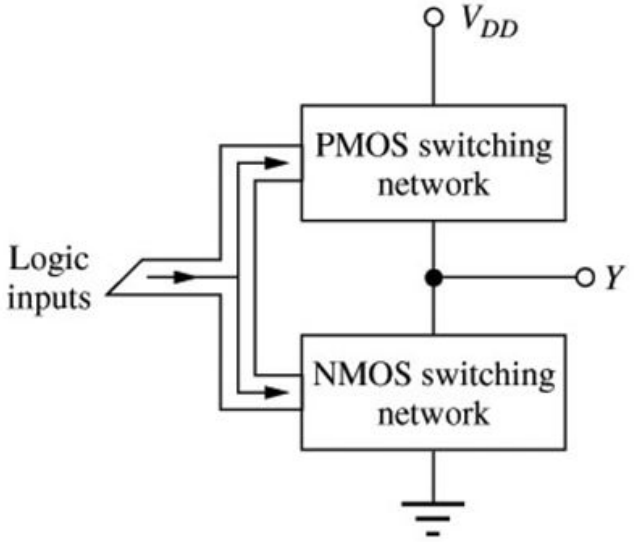
NMOS logic gate structure

CMOS Logic Design

- The design of logic gates for CMOC inverter is different from the similar logic design for NMOS inverters that we considered earlier.
- For NMOS gates, the logic involved only the switching transistor.
- For CMOS, both transistors are involved, since the input affects both in symmetrical way.
- Thus, for each logic input variable in CMOS gate there is *one transistor in NMOS* network and *one transistor in PMOS* network.

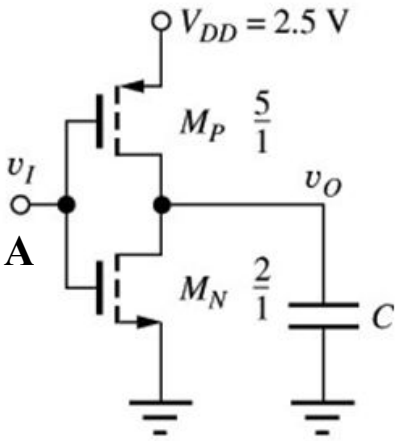


NMOS logic gate structure

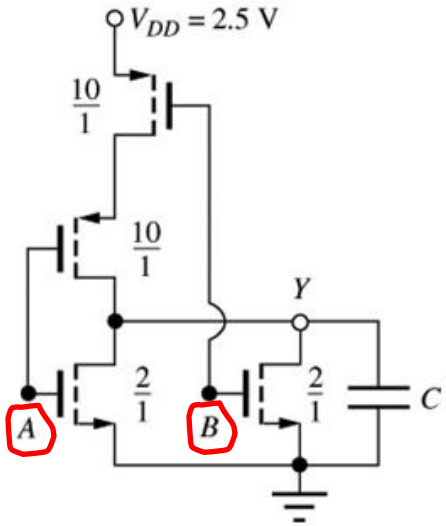


CMOS logic gate structure

CMOS NOR Gate



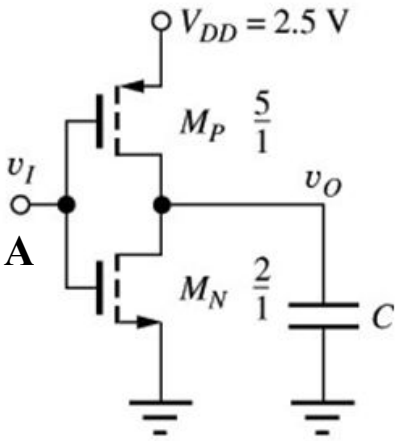
Reference Inverter



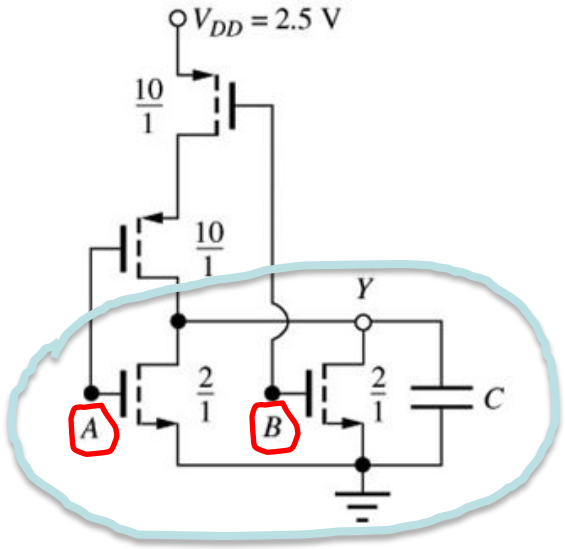
CMOS NOR 2 input gate

CMOS NOR Gate

- For the two input NOR gate, the NMOS portion of the gate is identical to the NMOS gate.



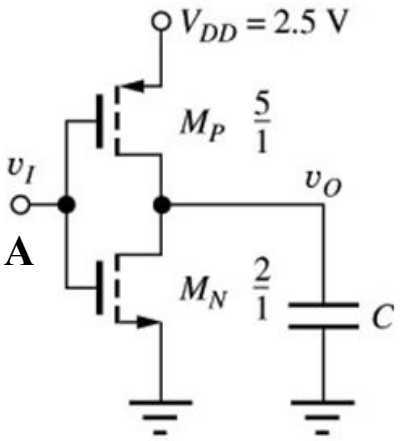
Reference Inverter



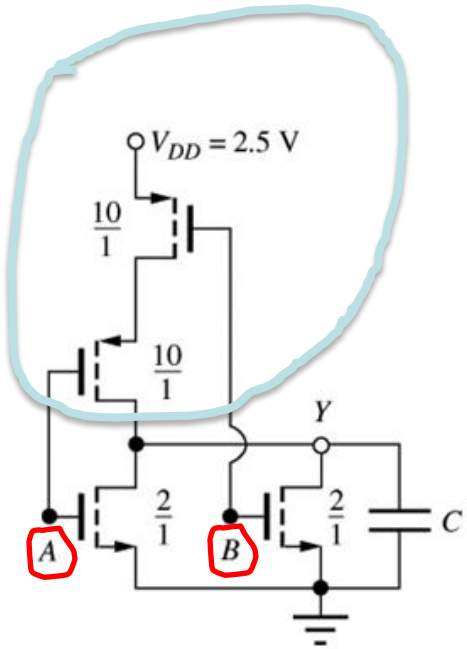
CMOS NOR 2 input gate

CMOS NOR Gate

- For the two input NOR gate, the NMOS portion of the gate is identical to the NMOS gate.
- In the CMOS gate, we must ensure that static current path does not exist through the logic gate, and this requires switching also in the PMOS transistor network 2 PMOS transistors.



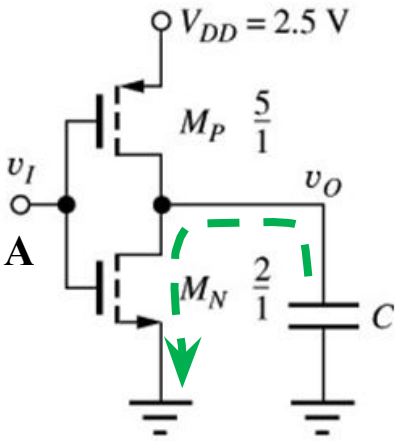
Reference Inverter



CMOS NOR 2 input gate

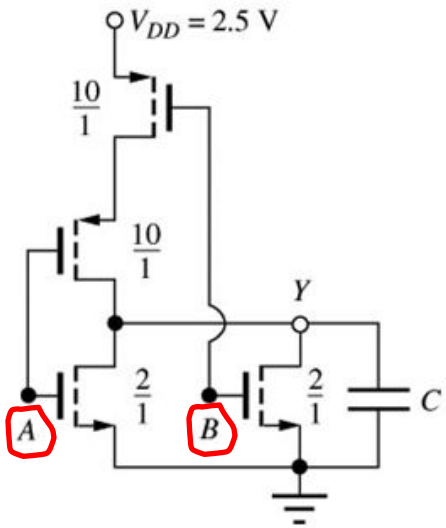
CMOS NOR Gate

- For the two input NOR gate, the NMOS portion of the gate is identical to the NMOS gate.
- In the CMOS gate, we must ensure that static current path does not exist through the logic gate, and this requires switching also in the PMOS transistor network □ 2 PMOS transistors.
- In the NMOS section, the conducting path exists for *A=1* **or** *B=1*.



Reference Inverter

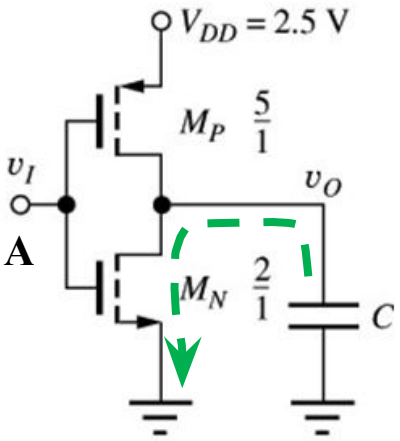
A=1 → Y=0



CMOS NOR 2 input gate

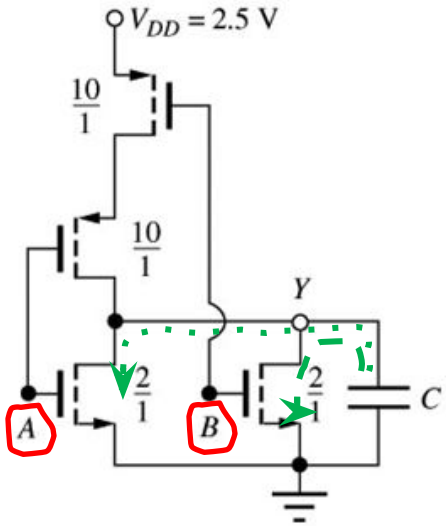
CMOS NOR Gate

- For the two input NOR gate, the NMOS portion of the gate is identical to the NMOS gate.
- In the CMOS gate, we must ensure that static current path does not exist through the logic gate, and this requires switching also in the PMOS transistor network □ 2 PMOS transistors.
- In the NMOS section, the conducting path exists for *A=1* **or** *B=1*.



Reference Inverter

A=1 → Y=0



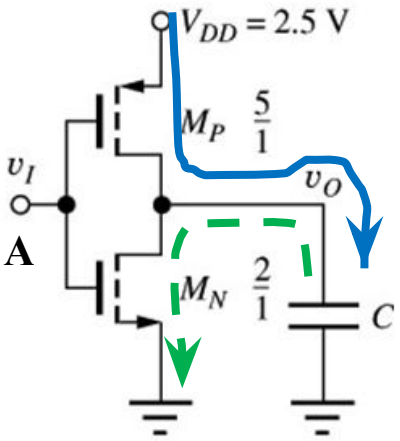
CMOS NOR 2 input gate

A=1 ⋯→ Y=0

B=1 →

CMOS NOR Gate

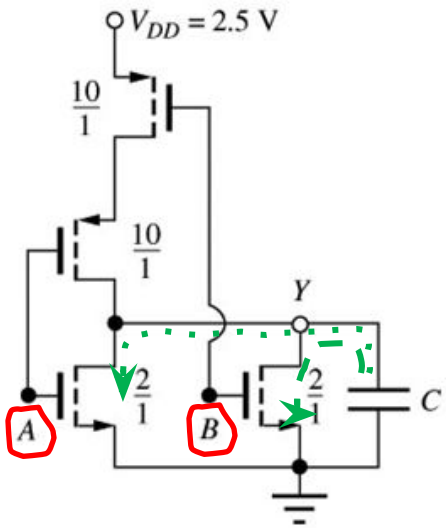
- For the two input NOR gate, the NMOS portion of the gate is identical to the NMOS gate.
- In the CMOS gate, we must ensure that static current path does not exist through the logic gate, and this requires switching also in the PMOS transistor network □ 2 PMOS transistors.
- In the NMOS section, the conducting path exists for *A=1* **or** *B=1*.
- In the PMOS section, the conducting path exists only when *A=0* **and** *B=0*



Reference Inverter

A=1 Y=0

A=0 Y=1



CMOS NOR 2 input gate

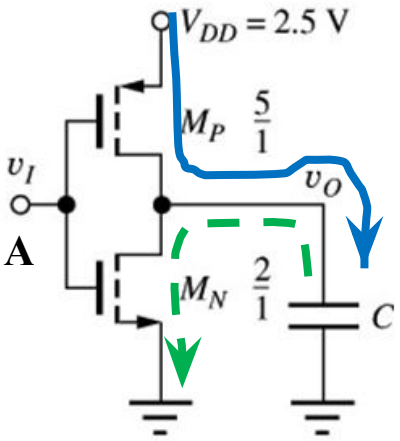
A=1

B=1

Y=0

CMOS NOR Gate

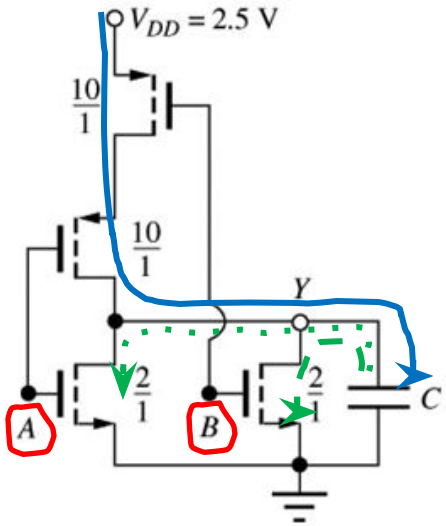
- For the two input NOR gate, the NMOS portion of the gate is identical to the NMOS gate.
- In the CMOS gate, we must ensure that static current path does not exist through the logic gate, and this requires switching also in the PMOS transistor network □ 2 PMOS transistors.
- In the NMOS section, the conducting path exists for *A=1* **or** *B=1*.
- In the PMOS section, the conducting path exists only when *A=0* **and** *B=0*



Reference Inverter

A=1 Y=0

A=0 Y=1



CMOS NOR 2 input gate

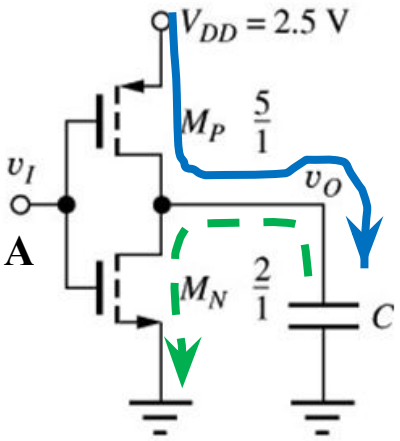
A=1 Y=0

B=1

A=0 & B=0 Y=1

CMOS NOR Gate

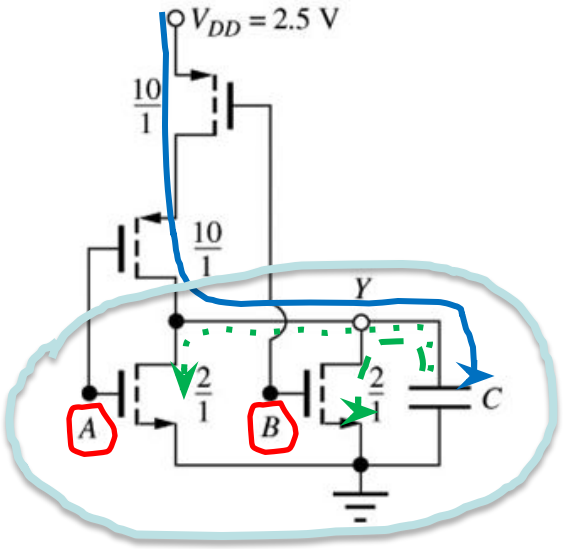
- For the two input NOR gate, the NMOS portion of the gate is identical to the NMOS gate.
- In the CMOS gate, we must ensure that static current path does not exist through the logic gate, and this requires switching also in the PMOS transistor network □ 2 PMOS transistors.
- In the NMOS section, the conducting path exists for *A=1* **or** *B=1*.
- In the PMOS section, the conducting path exists only when *A=0* **and** *B=0*
- Complimentary nature of conducting paths: for **NMOS – parallel**



Reference Inverter

A=1 → Y=0

A=0 → Y=1



CMOS NOR 2 input gate

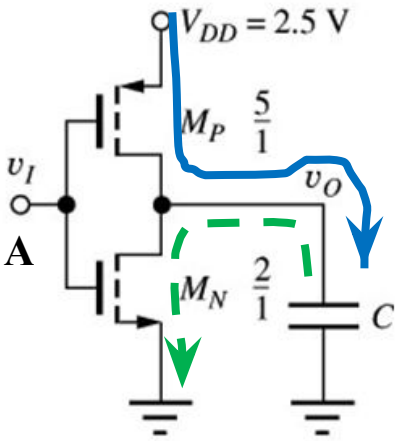
A=1 → Y=0

B=1 →

A=0 & B=0 → Y=1

CMOS NOR Gate

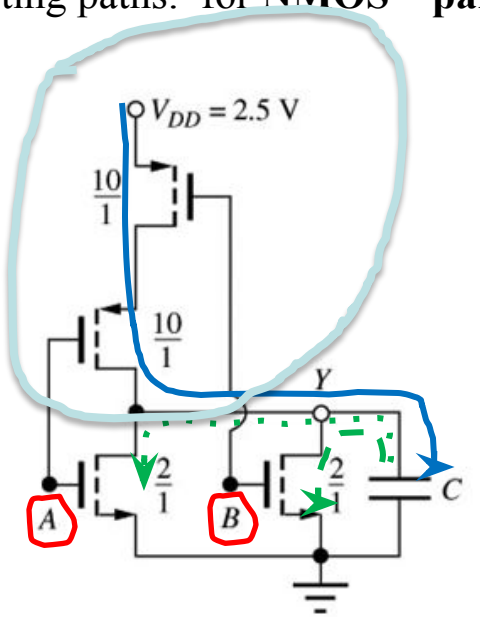
- For the two input NOR gate, the NMOS portion of the gate is identical to the NMOS gate.
- In the CMOS gate, we must ensure that static current path does not exist through the logic gate, and this requires switching also in the PMOS transistor network □ 2 PMOS transistors.
- In the NMOS section, the conducting path exists for *A=1* **or** *B=1*.
- In the PMOS section, the conducting path exists only when *A=0* **and** *B=0*
- Complimentary nature of conducting paths: for **NMOS – parallel,** for **PMOS - series**



Reference Inverter

A=1 → Y=0

A=0 → Y=1



CMOS NOR 2 input gate

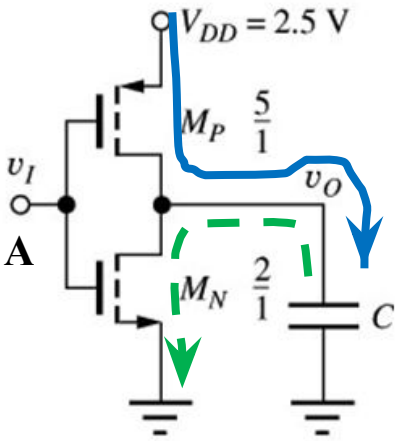
A=1 → Y=0

B=1 →

A=0 & B=0 → Y=1

CMOS NOR Gate

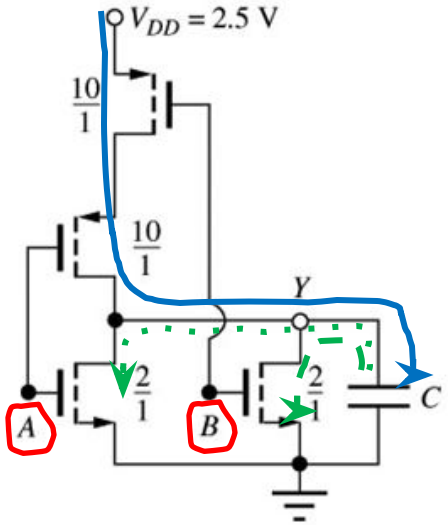
- For the two input NOR gate, the NMOS portion of the gate is identical to the NMOS gate.
- In the CMOS gate, we must ensure that static current path does not exist through the logic gate, and this requires switching also in the PMOS transistor network □ 2 PMOS transistors.
- In the NMOS section, the conducting path exists for *A=1* **or** *B=1*.
- In the PMOS section, the conducting path exists only when *A=0* **and** *B=0*
- Complimentary nature of conducting paths: for **NMOS – parallel,** for **PMOS - series**



Reference Inverter

A=1 → Y=0

A=0 → Y=1

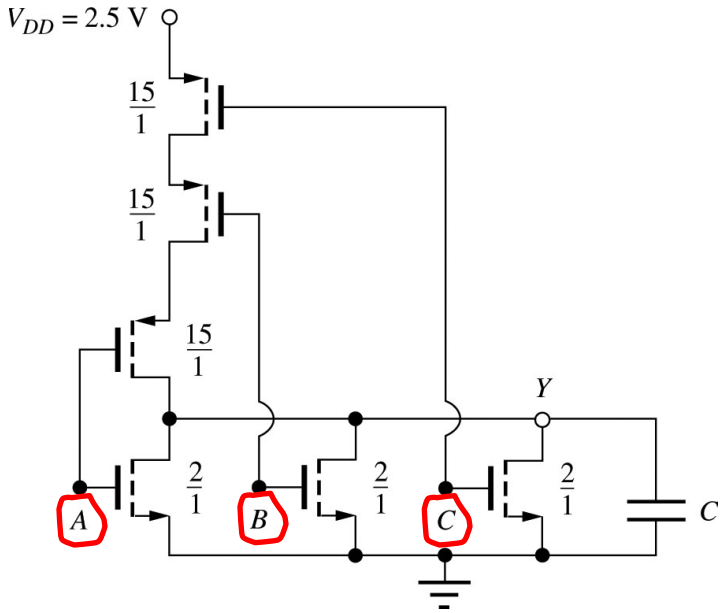


CMOS NOR 2 input gate

A=1 → Y=0

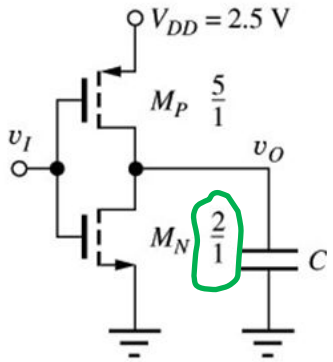
B=1 →

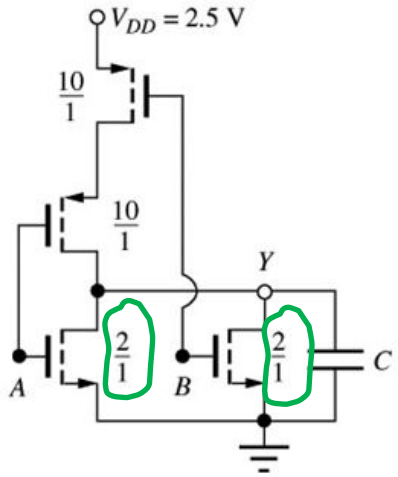
A=0 & B=0 → Y=1



CMOS NOR 3 input gate

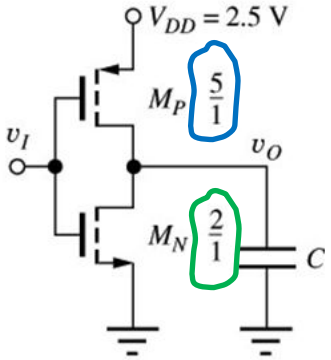
CMOS NOR Gate Transistor Sizing

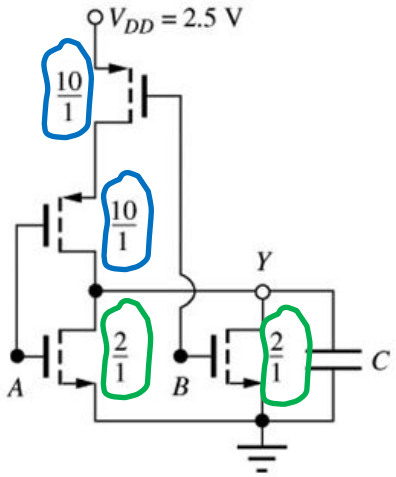




- When sizing the transistors, we attempt to keep the delay times the same as the reference inverter
- To accomplish this, the on-resistance in the PMOS and NMOS branches of the NOR gate must be the same as the reference inverter
- For a two-input NOR gate:
- For the parallel section - keep $(W/L)_N$ the same

CMOS NOR Gate Transistor Sizing

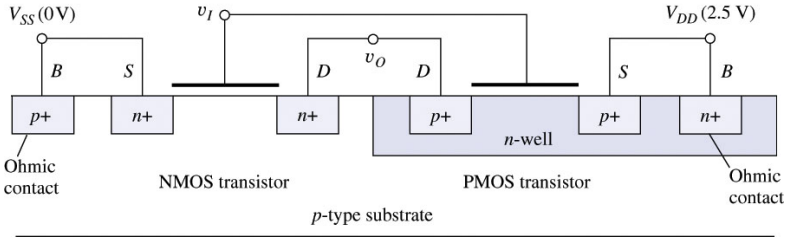




- When sizing the transistors, we attempt to keep the delay times the same as the reference inverter
- To accomplish this, the on-resistance in the PMOS and NMOS branches of the NOR gate must be the same as the reference inverter
- For a two-input NOR gate:
- For the parallel section - keep $(W/L)_N$ the same
- For the series section - $(W/L)_P$ must be made twice as large.

CMOS NOR Gate Body Effect

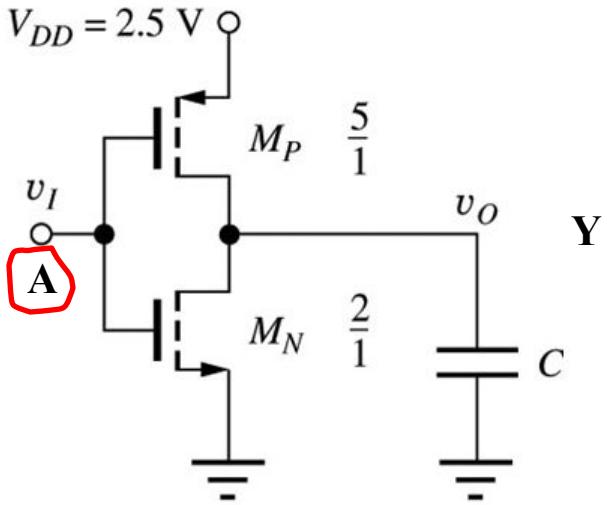
- Due to design, the bottom PMOS body contact is not connected to its source, its threshold voltage changes as V_{SB} changes during switching



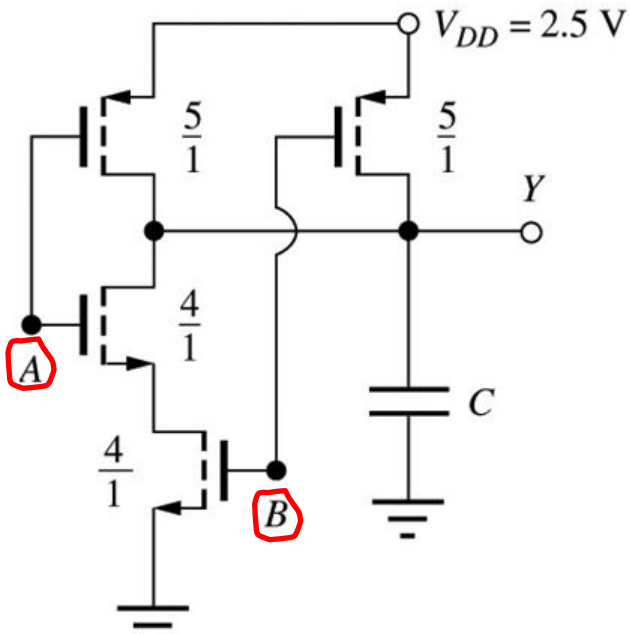
- Once $v_O = V_H$ is reached, the bottom PMOS is not affected by body effect (because all the line from V_{DD} to v_O is at V_H, thus the total on-resistance of the PMOS branch is the same

- However, the rise time is slowed down slightly due to $|V_{TP}|$ being a function of time

CMOS NAND Gates

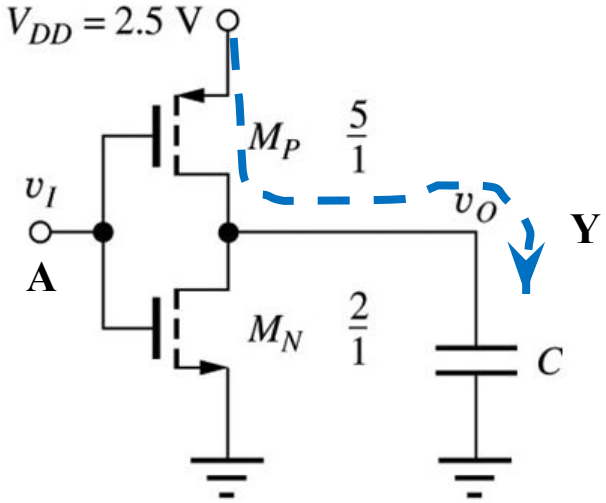


Reference Inverter



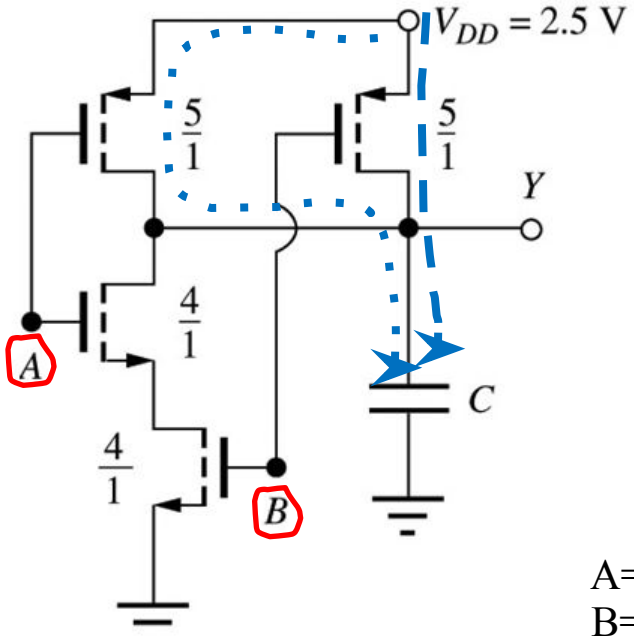
CMOS NAND gate

CMOS NAND Gates



Reference Inverter

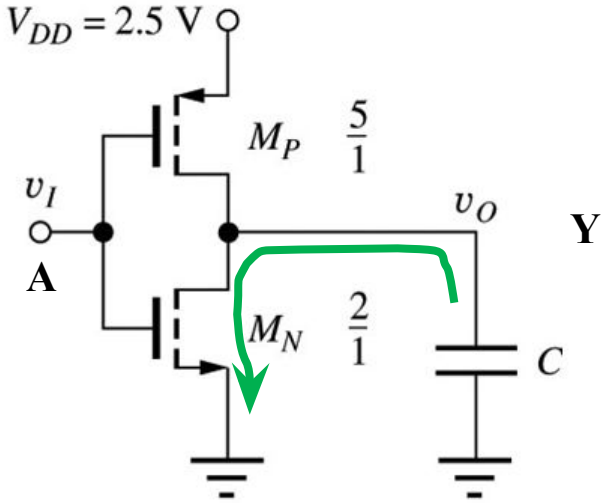
A=0 → Y=1



A=0 ⋯→ Y=1
B=0 —→ Y=1

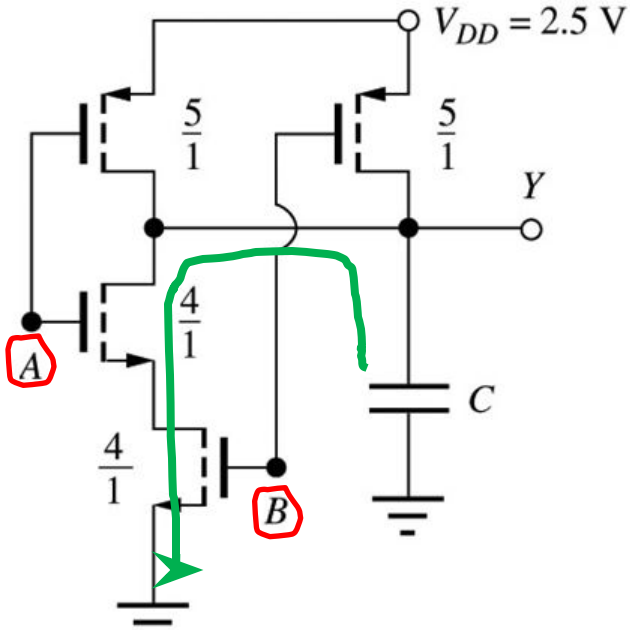
CMOS NAND gate

CMOS NAND Gates



Reference Inverter

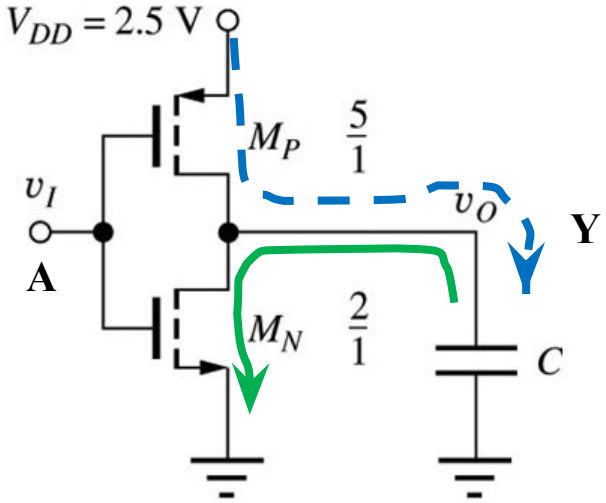
A=1 → Y=0



CMOS NAND gate

A=1 & B=1 → Y=0

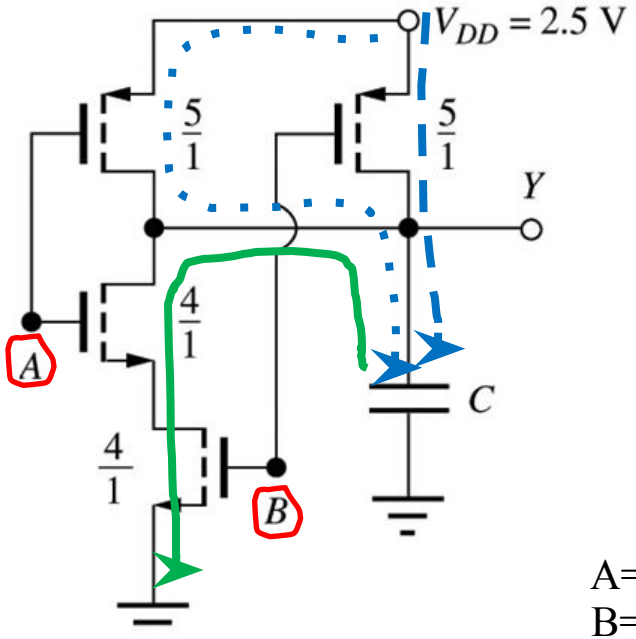
CMOS NAND Gates



Reference Inverter

A=0 → Y=1

A=1 → Y=0



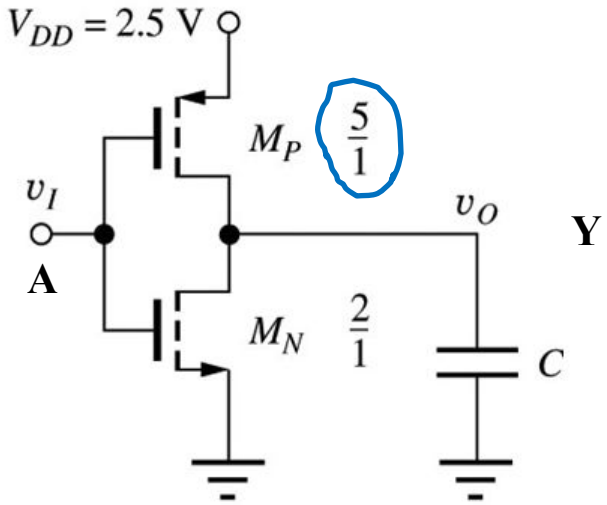
CMOS NAND gate

A=0 → Y=1

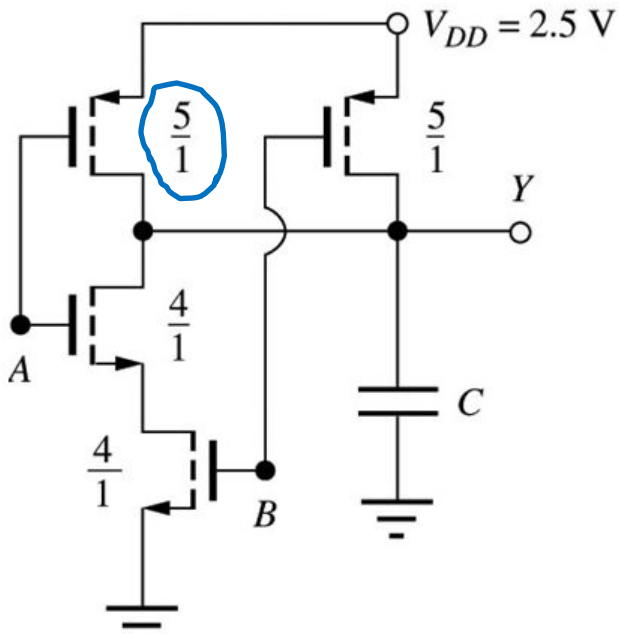
B=0 → Y=1

A=1 & B=1 → Y=0

CMOS NAND Gates



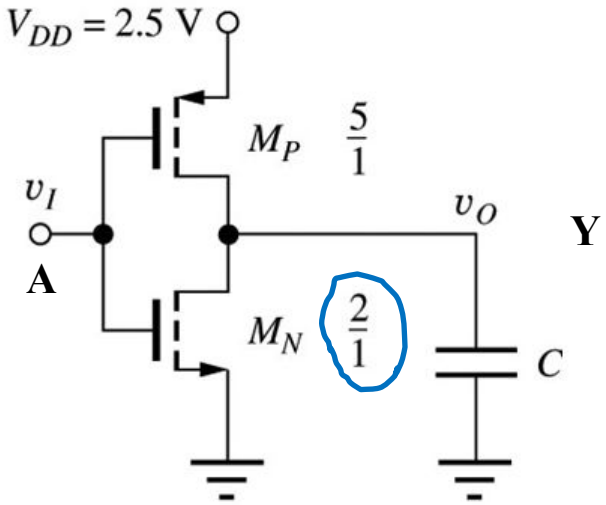
Reference Inverter



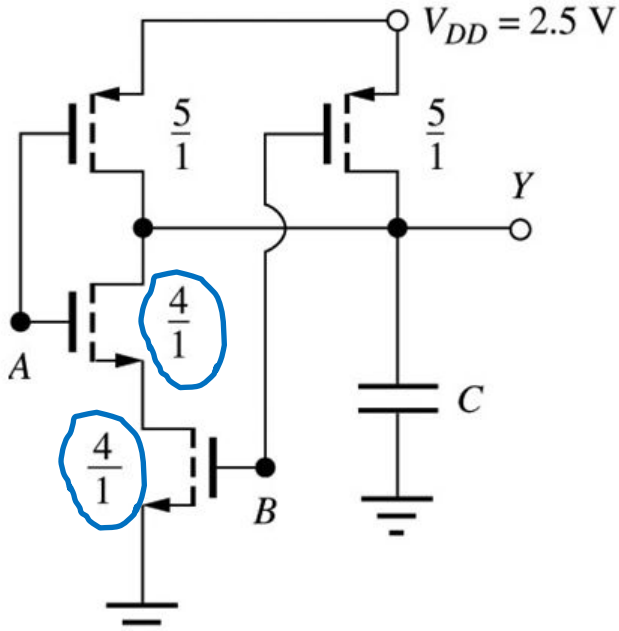
CMOS NAND gate

- The same rules apply for sizing the NAND gate devices as for the NOR gate, except now the NMOS transistors are in series
- $(W/L)_p$ will be the same size of that of the reference inverter

CMOS NAND Gates



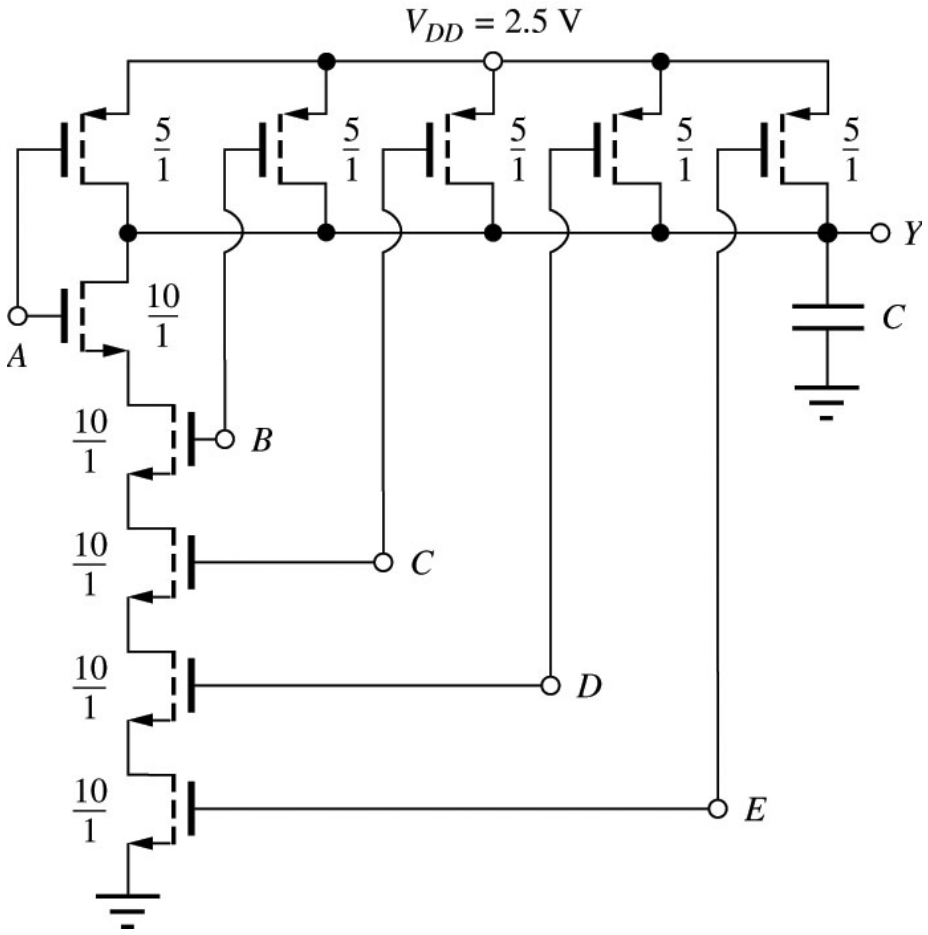
Reference Inverter



CMOS NAND gate

- The same rules apply for sizing the NAND gate devices as for the NOR gate, except now the NMOS transistors are in series
- $(W/L)_P$ will be the same size of that of the reference inverter
- $(W/L)_N$ will be twice the size of that of the reference inverter

Multi-Input CMOS NAND Gates



Complex CMOS Logic Gate Design

- The design of the complex CMOS logic gates is **more complicated** process then NMOS design because it involves designing **two logic circuits:** NMOS and PMOS.

Complex CMOS Logic Gate Design

- The design of the complex CMOS logic gates is **more complicated** process then NMOS design because it involves designing **two logic circuits:** NMOS and PMOS.
- The **NMOS** logic circuits **repeats the original logic function** to be implemented.

Complex CMOS Logic Gate Design

- The design of the complex CMOS logic gates is **more complicated** process then NMOS design because it involves designing **two logic circuits:** NMOS and PMOS.
- The **NMOS** logic circuits **repeats the original logic function** to be implemented.
- However the **PMOS** logic circuit should be designed as **complementary to the original logic function**, which is another complication. This difference, as compared to the pure NMOS logic design, is due to the fact that in NMOS logic the input goes only to the switching part – the load is providing inversion of the resulting voltage. In the CMOS gate design, the input is applied also to the load portion (PMOS), and has to be inverted, thus by DeMorgan's law, the logic has to be complementary.

Complex CMOS Logic Gate Design

- The design of the complex CMOS logic gates is **more complicated** process then NMOS design because it involves designing **two logic circuits:** NMOS and PMOS.
- The **NMOS** logic circuits **repeats the original logic function** to be implemented.
- However the **PMOS** logic circuit should be designed as **complementary to the original logic function**, which is another complication. This difference, as compared to the pure NMOS logic design, is due to the fact that in NMOS logic the input goes only to the switching part – the load is providing inversion of the resulting voltage. In the CMOS gate design, the input is applied also to the load portion (PMOS), and has to be inverted, thus by DeMorgan's law, the logic has to be complementary.
- The process consists of **two** steps:

  - design the **NMOS circuit** structure **and** the corresponding **graph** of the logic function

Complex CMOS Logic Gate Design

- The design of the complex CMOS logic gates is **more complicated** process then NMOS design because it involves designing **two logic circuits:** NMOS and PMOS.
- The **NMOS** logic circuits **repeats the original logic function** to be implemented.
- However the **PMOS** logic circuit should be designed as **complementary to the original logic function**, which is another complication. This difference, as compared to the pure NMOS logic design, is due to the fact that in NMOS logic the input goes only to the switching part – the load is providing inversion of the resulting voltage. In the CMOS gate design, the input is applied also to the load portion (PMOS), and has to be inverted, thus by DeMorgan's law, the logic has to be complementary.
- The process consists of **two** steps:

  - design the **NMOS circuit** structure **and** the corresponding **graph** of the logic function

  - build the **graph** for the complementary logic function and then design the complementary logic PMOS circuit **OR invert** the original function using the **DeMorgan's law** and design the corresponding network.

Complex CMOS Logic Gate Design Example

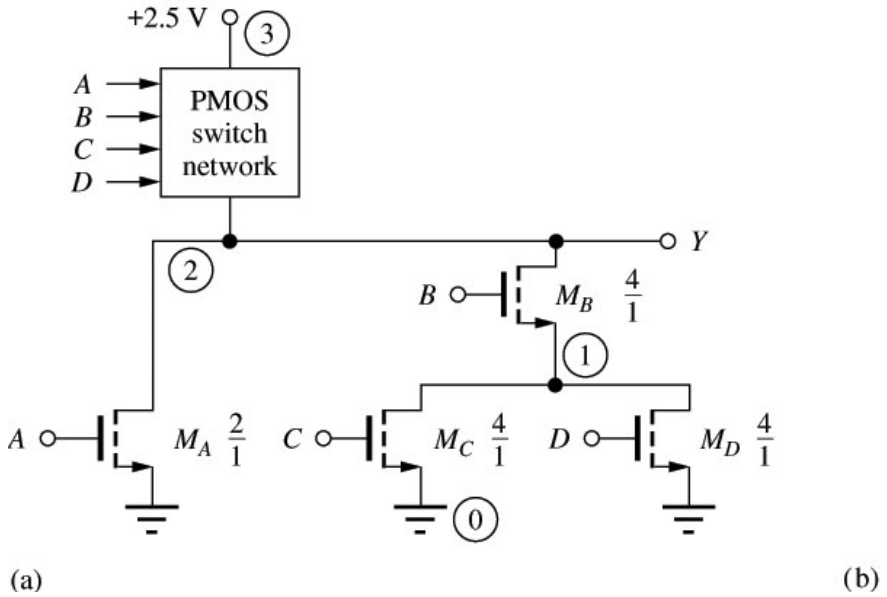
- Design a CMOS logic gate that yields the function: $Y = \overline{A + B(C + D)}$ from the reference inverter with $(W/L)_{p,ref} = 5/1$ and $(W/L)_{n,ref} = 2/1$

Complex CMOS Logic Gate Design Example

- Design a CMOS logic gate that yields the function: $Y = \overline{A + B(C + D)}$ from the reference inverter with $(W/L)_{p,ref} = 5/1$ and $(W/L)_{n,ref} = 2/1$
- By inspection (knowing $\overline{Y}$), the NMOS section of the gate can be designed as the following.

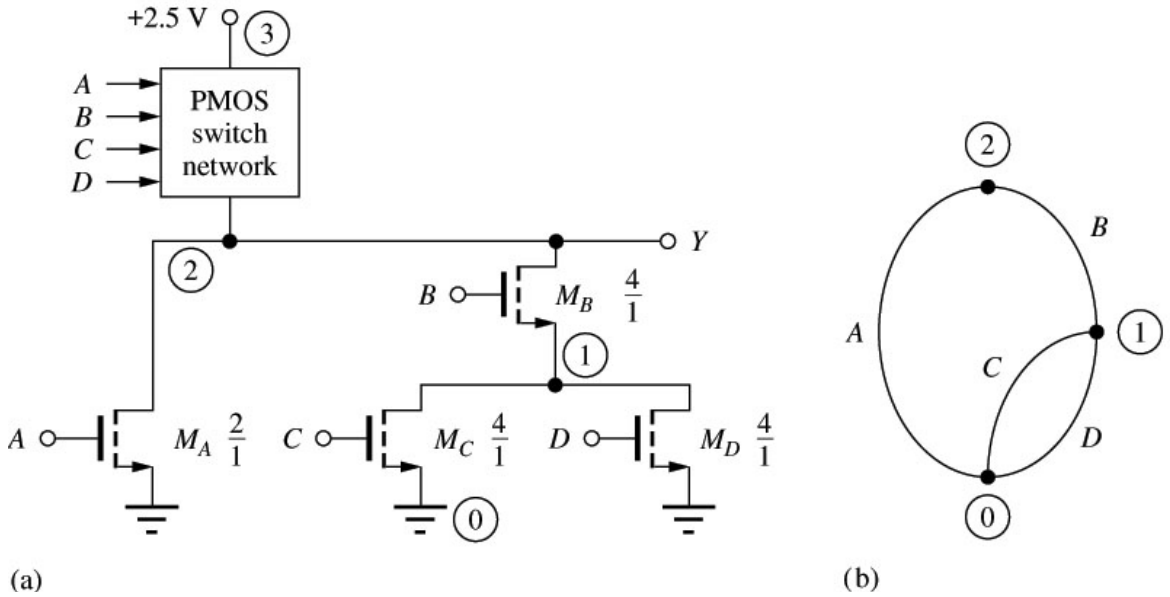
Complex CMOS Logic Gate Design Example

- Design a CMOS logic gate that yields the function: $Y = \overline{A + B(C + D)}$ from the reference inverter with $(W/L)_{p,ref} = 5/1$ and $(W/L)_{n,ref} = 2/1$
- By inspection (knowing $\overline{Y}$), the NMOS section of the gate can be designed as the following.



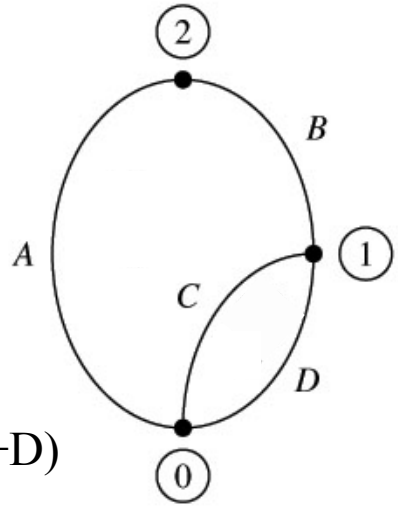
Complex CMOS Logic Gate Design Example

- Design a CMOS logic gate that yields the function: $Y = \overline{A + B(C + D)}$ from the reference inverter with $(W/L)_{p,ref} = 5/1$ and $(W/L)_{n,ref} = 2/1$
- By inspection (knowing $\overline{Y}$), the NMOS section of the gate can be designed as the following.
- Then the corresponding graph can be drawn: each arc represent an NMOS transistor and each node correspond to the circuit node.



Complex CMOS Logic Gate Design Example

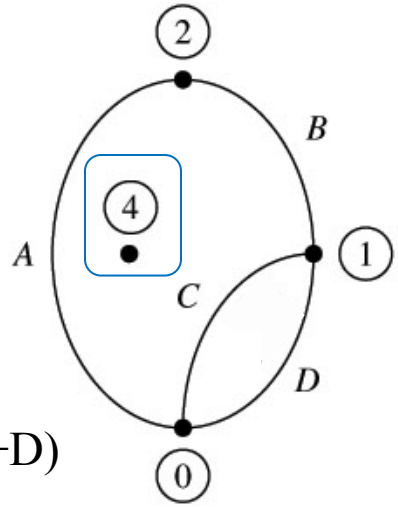
- The PMOS logic should invert Y. Since each PMOS transistor provide its own inversion, we use DeMorgan's law to create complementary logic: $\overline{Y} = \overline{A + B(C + D)} = \overline{A}(\overline{B} + \overline{C}\overline{D})$
- To draw the graph for the complementary PMOS logic:



Y = A + B(C +D)

Complex CMOS Logic Gate Design Example

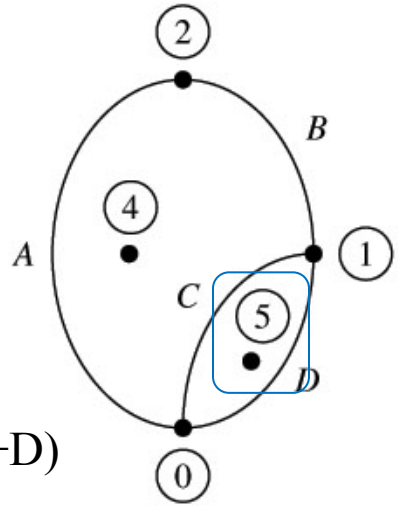
- The PMOS logic should invert Y. Since each PMOS transistor provide its own inversion, we use DeMorgan's law to create complementary logic: $\overline{Y} = \overline{A + B(C + D)} = \overline{A}(\overline{B} + \overline{C}\overline{D})$
- To draw the graph for the complementary PMOS logic:
  - place nodes in the interior of each enclosed path: **(4)**



Y = A + B(C +D)

Complex CMOS Logic Gate Design Example

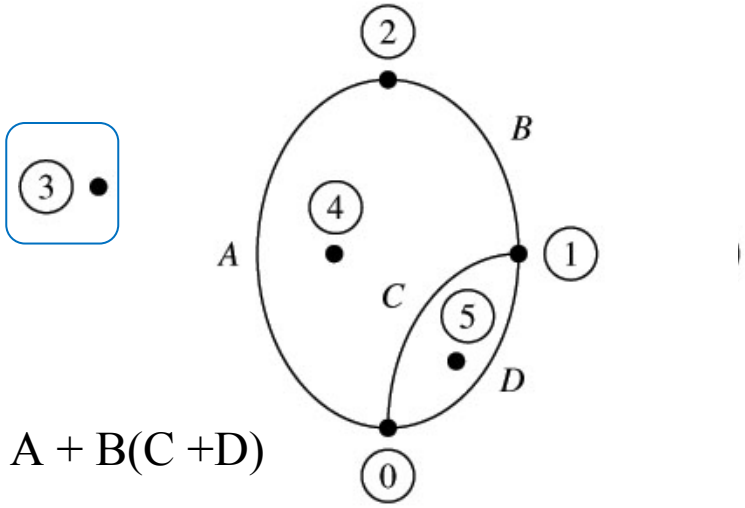
- The PMOS logic should invert Y. Since each PMOS transistor provide its own inversion, we use DeMorgan's law to create complementary logic: $\overline{Y} = \overline{A + B(C + D)} = \overline{A}(\overline{B} + \overline{C}\overline{D})$
- To draw the graph for the complementary PMOS logic:
  - place nodes in the interior of each enclosed path: (4) and **(5)**



Y = A + B(C +D)

Complex CMOS Logic Gate Design Example

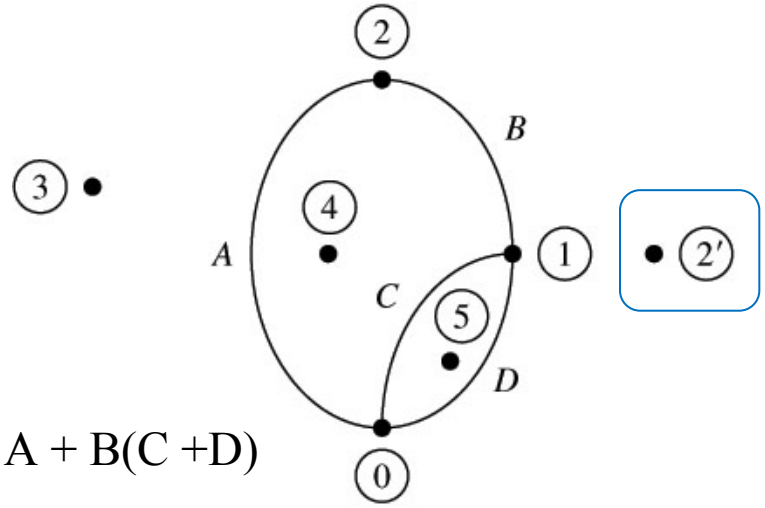
- The PMOS logic should invert Y. Since each PMOS transistor provide its own inversion, we use DeMorgan's law to create complementary logic: $\overline{Y} = \overline{A + B(C + D)} = \overline{A}(\overline{B} + \overline{C}\overline{D})$
- To draw the graph for the complementary PMOS logic:
  - place nodes in the interior of each enclosed path: (4) and (5)
  - place two more outside the graph for V_{DD} **(3)**



Y = A + B(C +D)

Complex CMOS Logic Gate Design Example

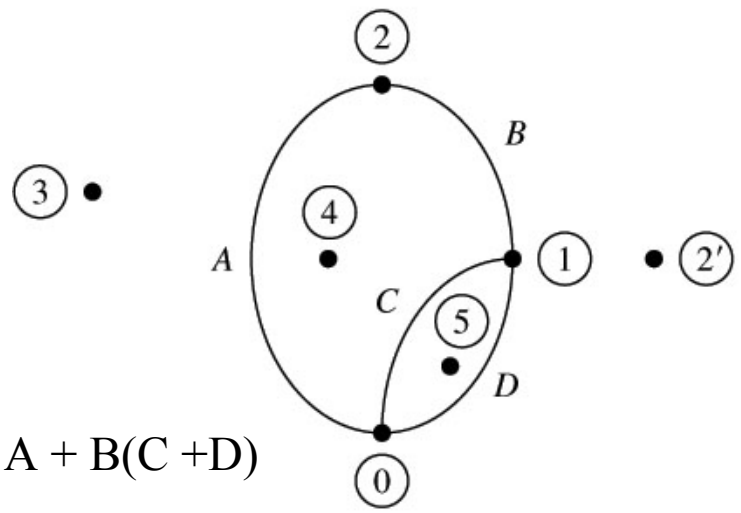
- The PMOS logic should invert Y. Since each PMOS transistor provide its own inversion, we use DeMorgan's law to create complementary logic: $\overline{Y} = \overline{A + B(C + D)} = \overline{A}(\overline{B} + \overline{C}\overline{D})$
- To draw the graph for the complementary PMOS logic:
  - place nodes in the interior of each enclosed path: (4) and (5)
  - place two more outside the graph for V_{DD} (3) and the complementary output **(2')**

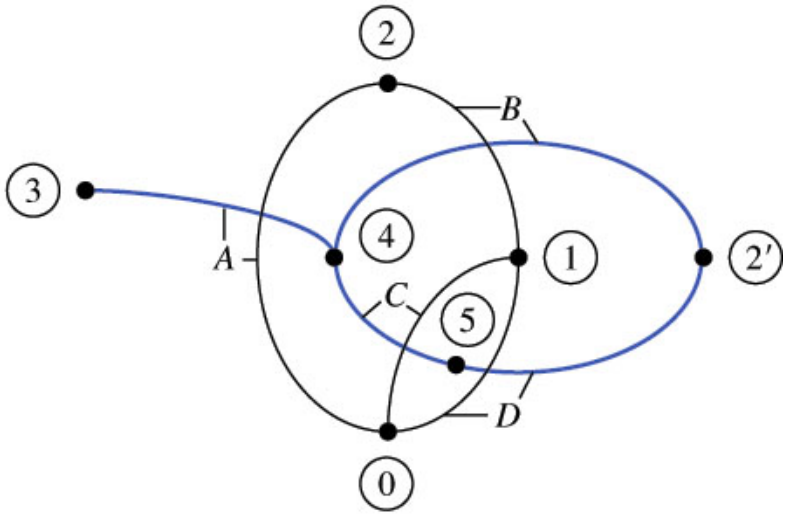


Y = A + B(C +D)

Complex CMOS Logic Gate Design Example

- The PMOS logic should invert Y. Since each PMOS transistor provide its own inversion, we use DeMorgan's law to create complementary logic: $\overline{Y} = \overline{A + B(C + D)} = \overline{A}(\overline{B} + \overline{C}\overline{D})$
- To draw the graph for the complementary PMOS logic:
  - place nodes in the interior of each enclosed path: (4) and (5)
  - place two more outside the graph for V_{DD} (3) and the complementary output (2')
- Connect all of the nodes by drawing the arcs (PMOS transistors) that cut the arcs of the original NMOS graph

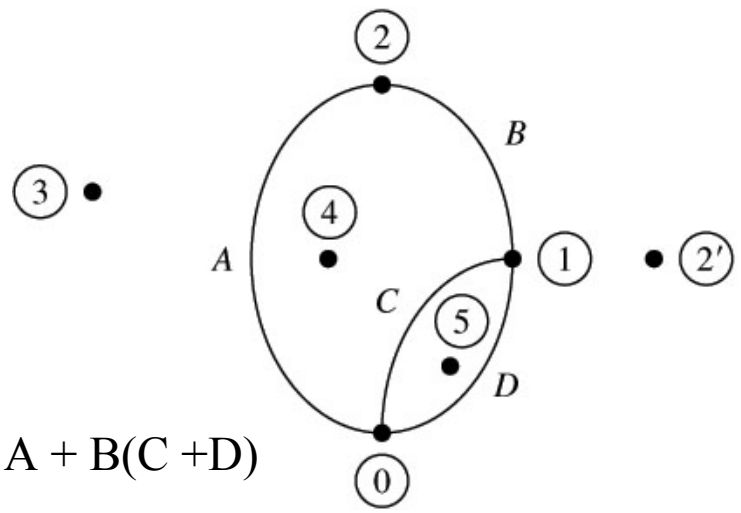




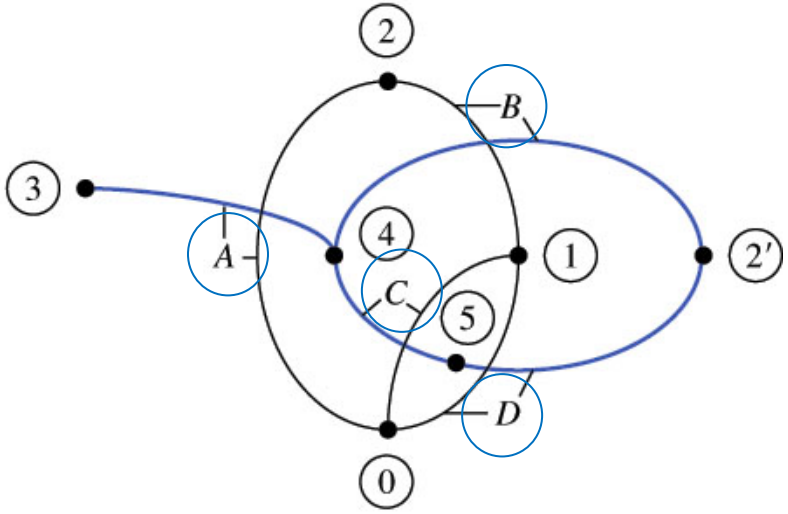
Y = A + B(C +D)

Complex CMOS Logic Gate Design Example

- The PMOS logic should invert Y. Since each PMOS transistor provide its own inversion, we use DeMorgan's law to create complementary logic: $\overline{Y} = \overline{A + B(C + D)} = \overline{A}(\overline{B} + \overline{C}\overline{D})$
- To draw the graph for the complementary PMOS logic:
  - place nodes in the interior of each enclosed path: (4) and (5)
  - place two more outside the graph for V_{DD} (3) and the complementary output (2')
- Connect all of the nodes by drawing the arcs (PMOS transistors) that cut the arcs of the original NMOS graph and label them as the arcs they intersect.

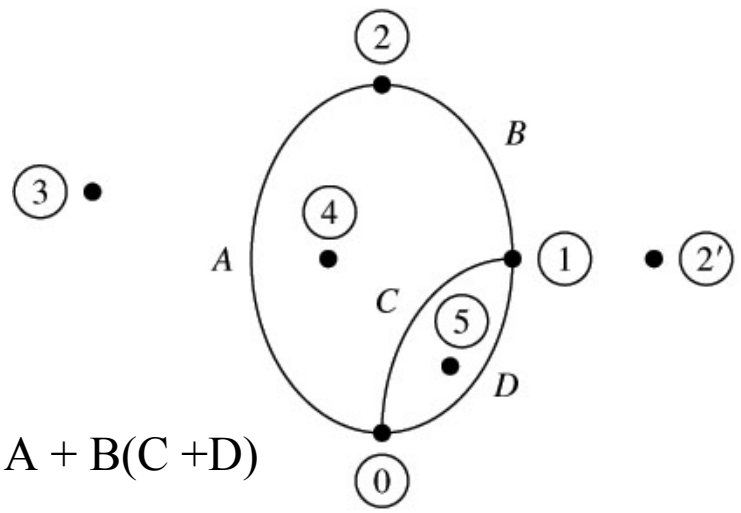


Y = A + B(C +D)

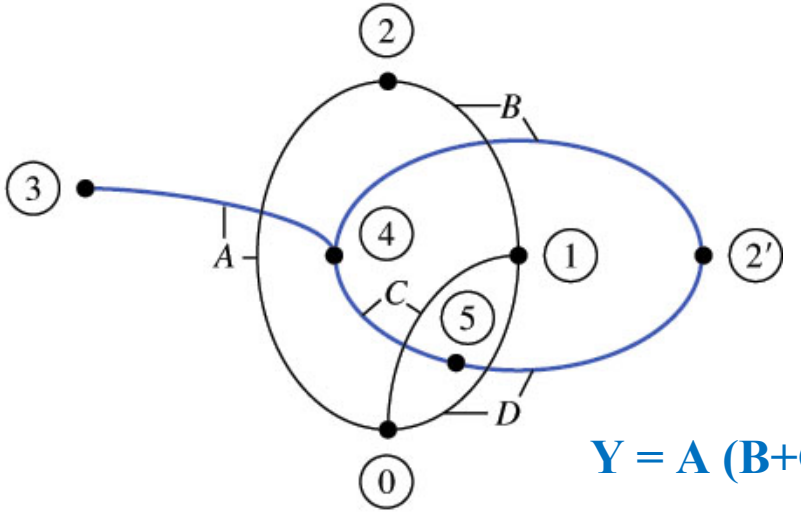


Complex CMOS Logic Gate Design Example

- The PMOS logic should invert Y. Since each PMOS transistor provide its own inversion, we use DeMorgan's law to create complementary logic: $\overline{Y} = \overline{A + B(C + D)} = \overline{A}(\overline{B} + \overline{C}\overline{D})$
- To draw the graph for the complementary PMOS logic:
  - place nodes in the interior of each enclosed path: (4) and (5)
  - place two more outside the graph for V_{DD} (3) and the complementary output (2')
- Connect all of the nodes by drawing the arcs (PMOS transistors) that cut the arcs of the original NMOS graph and label them as the arcs they intersect.
- This construction results in the minimum PMOS logic network that has one transistor per logical input.



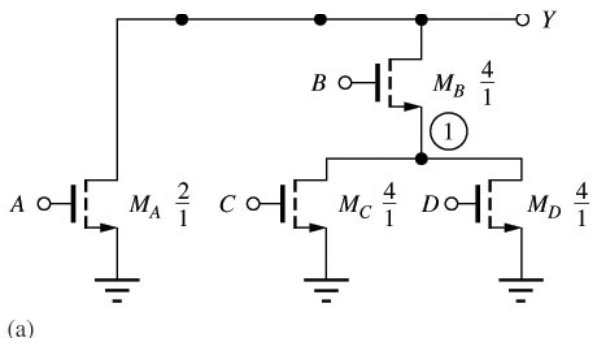
Y = A + B(C +D)



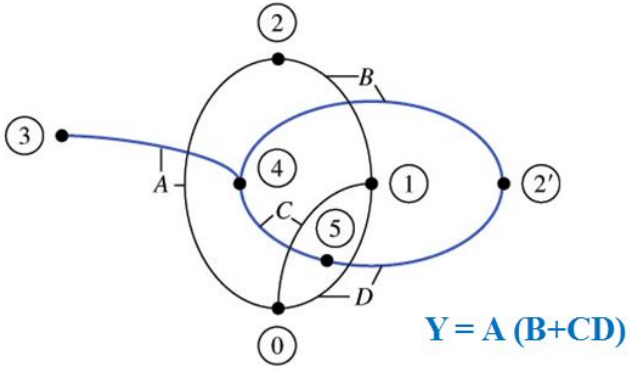
Y = A (B+CD)

Complex CMOS Logic Gate Design Example

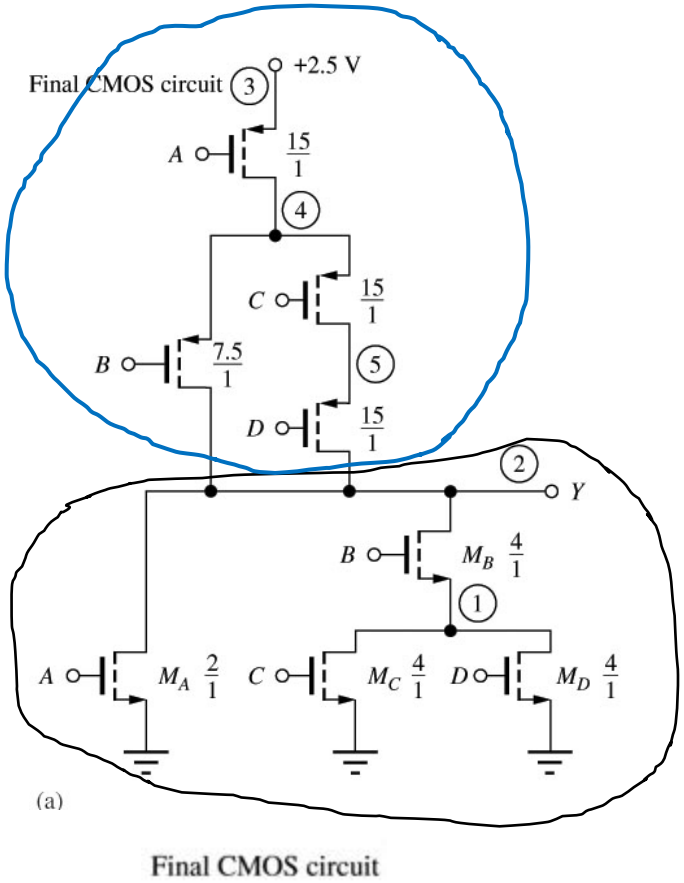
- From the PMOS graph, the PMOS network can now be drawn for the final CMOS logic gate



Final CMOS circuit

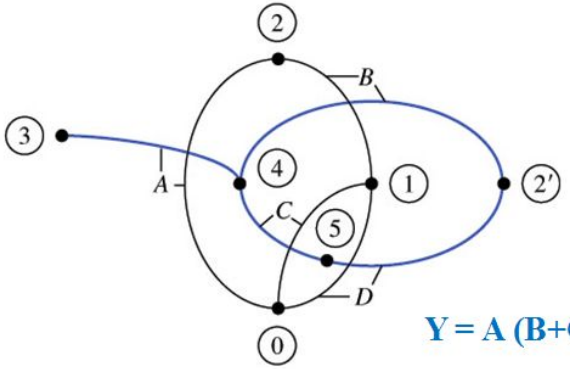


Complex CMOS Logic Gate Design Example

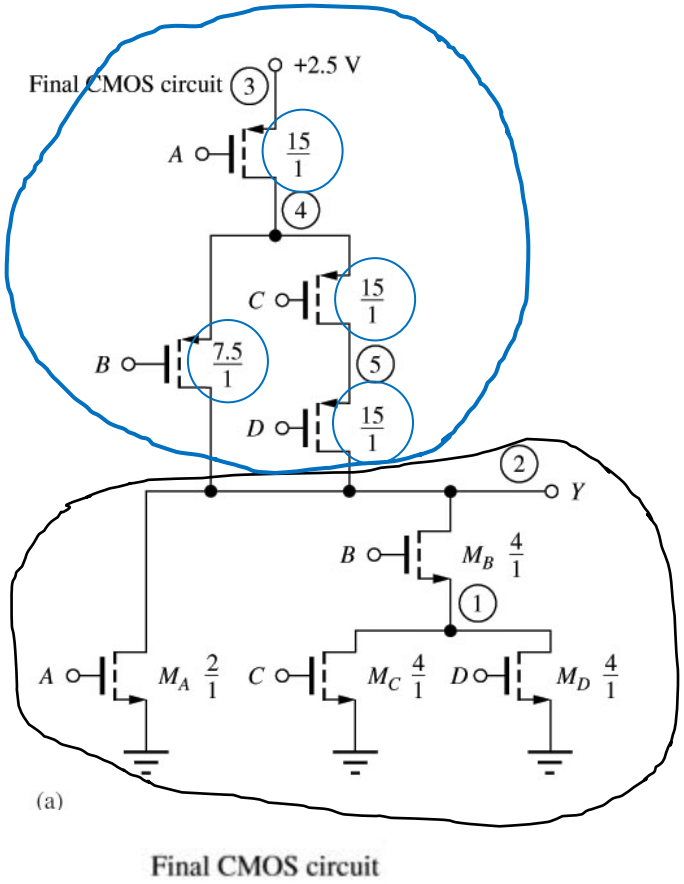


Final CMOS circuit

- From the PMOS graph, the PMOS network can now be drawn for the final CMOS logic gate

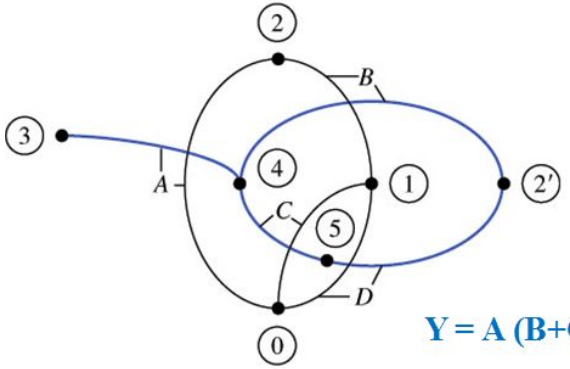


Complex CMOS Logic Gate Design Example



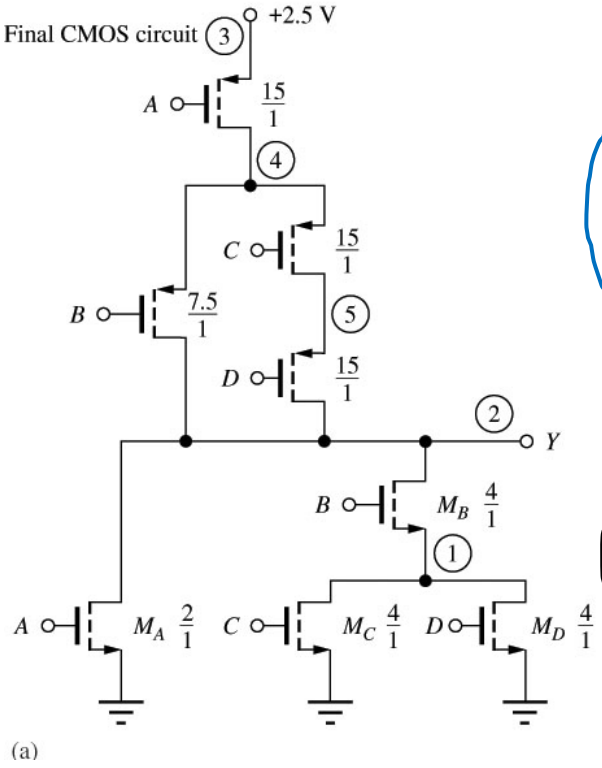
Final CMOS circuit

- From the PMOS graph, the PMOS network can now be drawn for the final CMOS logic gate

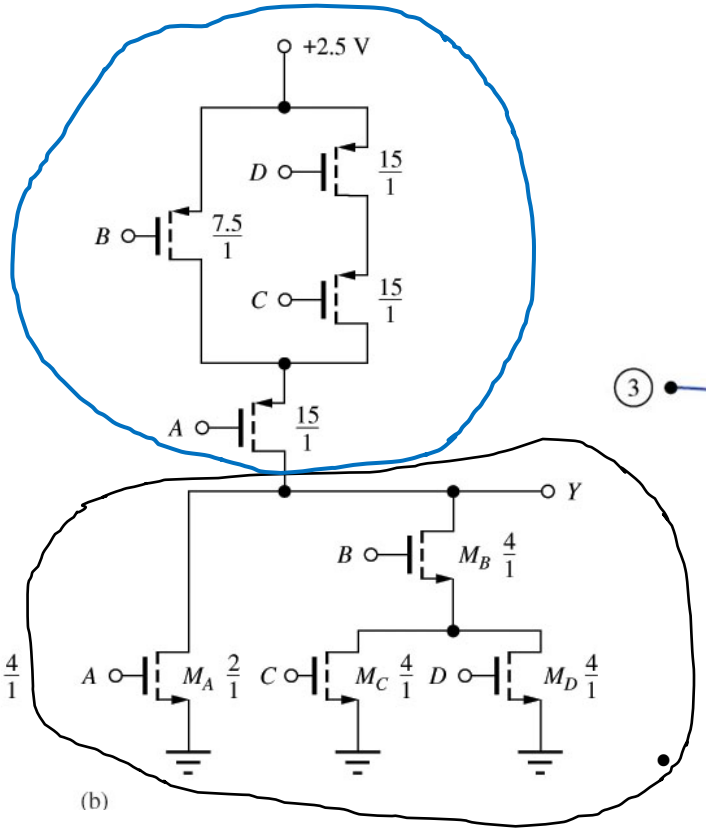


- For sizing we once again consider the longest PMOS path, where **$(W/L)_{p,ref} = 5/1$.**

Complex CMOS Logic Gate Design Example

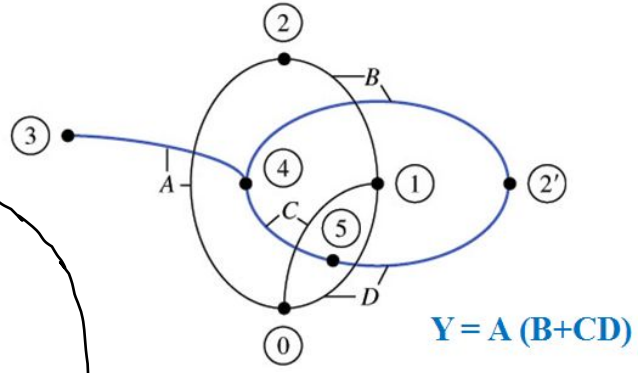


Final CMOS circuit



The alternative designs for PMOS circuit

- From the PMOS graph, the PMOS network can now be drawn for the final CMOS logic gate



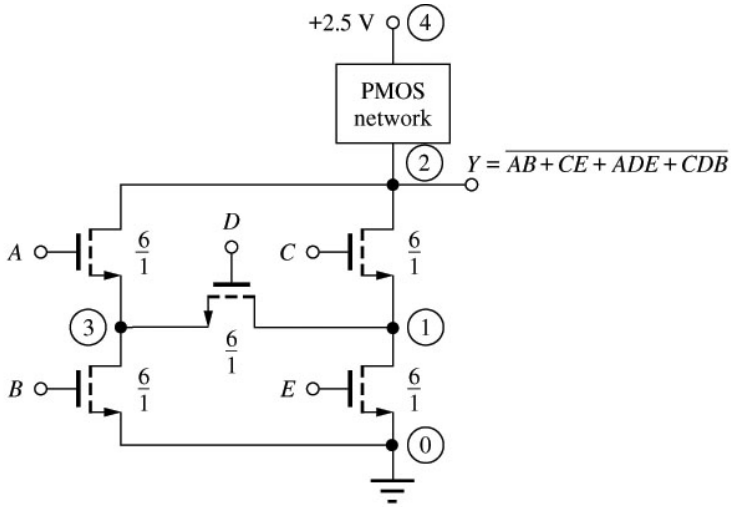
- For sizing we once again consider the longest PMOS path, where $(W/L)_{p,ref} = 5/1$.

Complex CMOS Gate with a Bridging Transistor - Design Example

- Design a CMOS gate that implements the following logic function using the same reference inverter sizes as the previous example:
  $Y = \overline{AB + CE + ADE + CDB}$

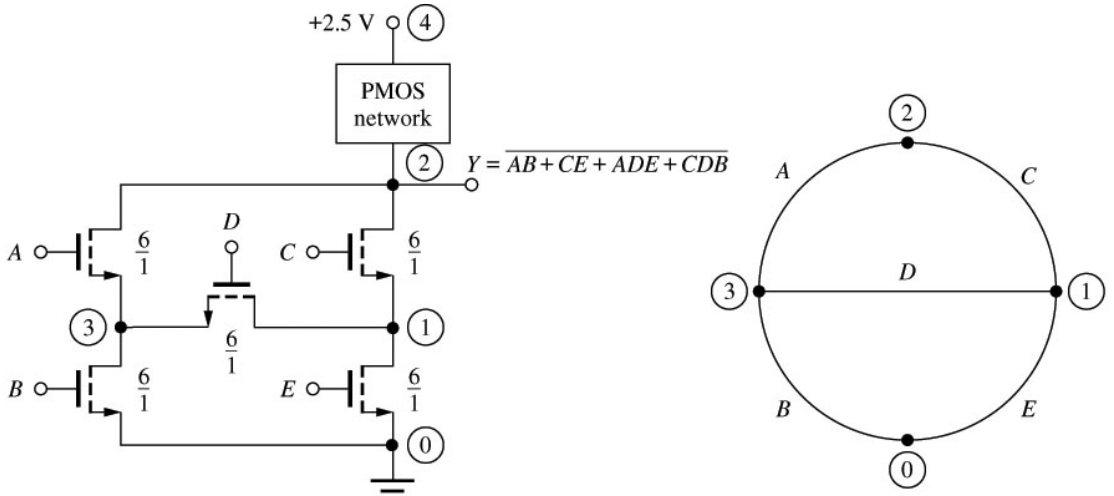
Complex CMOS Gate with a Bridging Transistor - Design Example

- Design a CMOS gate that implements the following logic function using the same reference inverter sizes as the previous example:
  $Y = \overline{AB + CE + ADE + CDB}$
- The NMOS branch can be realized in the following manner using bridging NMOS D to implement $\overline{Y}$.



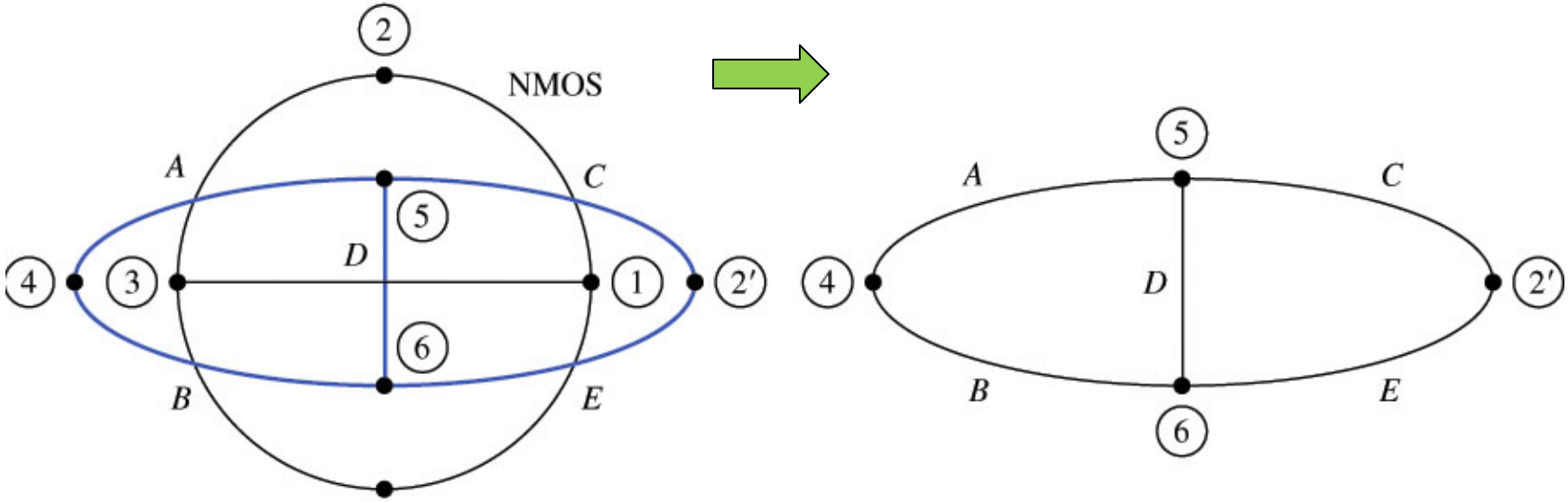
Complex CMOS Gate with a Bridging Transistor - Design Example

- Design a CMOS gate that implements the following logic function using the same reference inverter sizes as the previous example:
  $Y = \overline{AB + CE + ADE + CDB}$
- The NMOS branch can be realized in the following manner using bridging NMOS D to implement $\overline{Y}$. The corresponding NMOS graph is shown to the right.

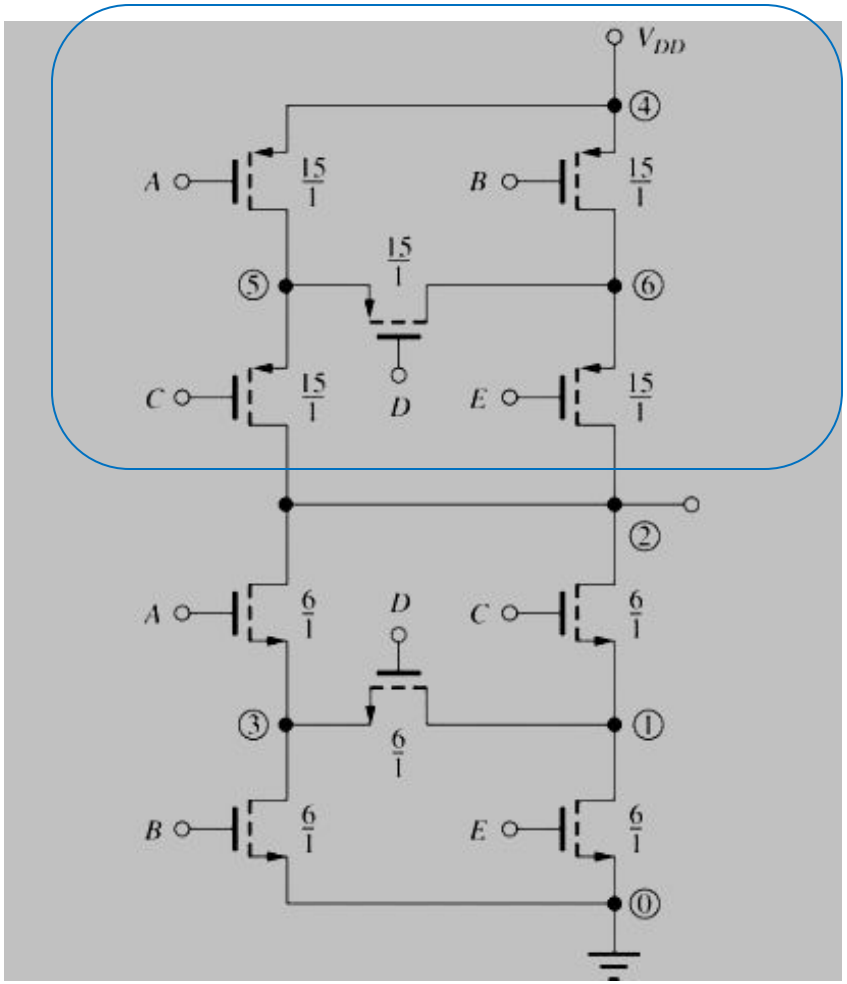


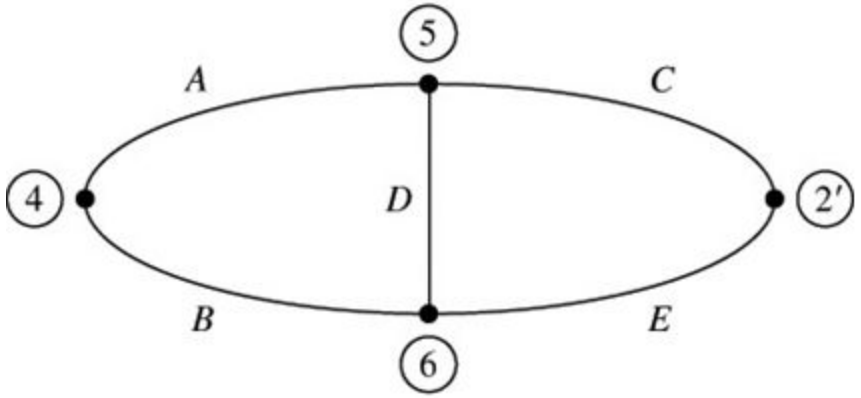
Complex CMOS Gate with a Bridging Transistor - Design Example

- By using the same technique as before, the PMOS graph can now be drawn



Complex CMOS Gate with a Bridging Transistor - Design Example

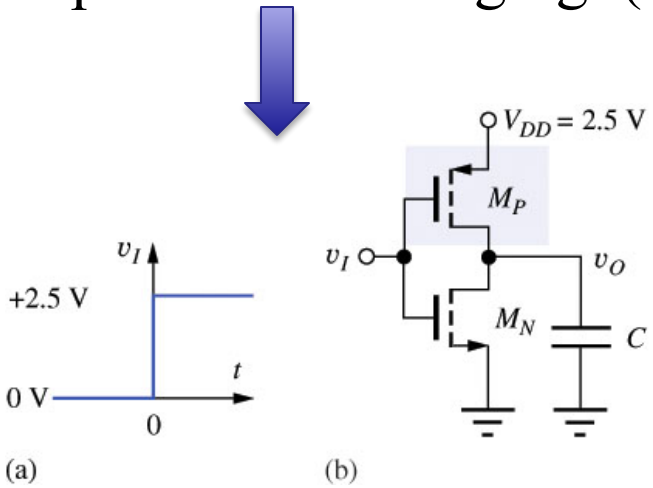




- By using the PMOS graph, the PMOS network can now be realized as shown (considering the longest path for sizing)

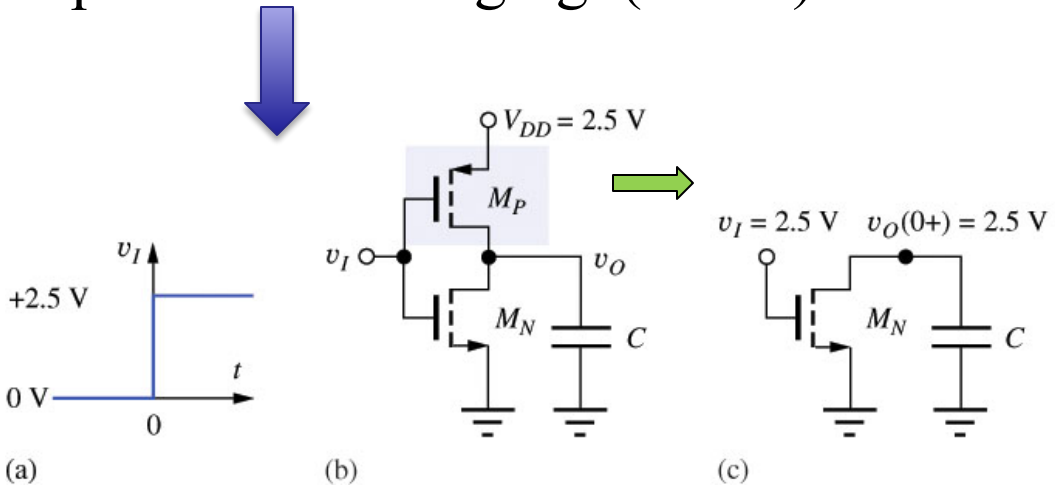
Dynamic Behavior of the CMOS Inverter Propagation Delay Estimate

- The propagation delays are created by the process of capacitive discharging (L□ H)



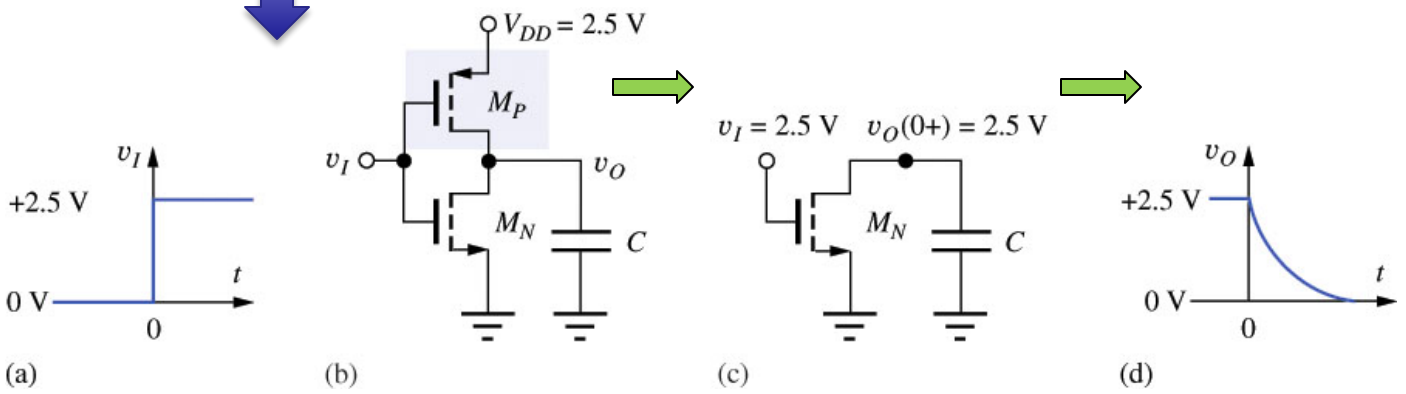
Dynamic Behavior of the CMOS Inverter Propagation Delay Estimate

- The propagation delays are created by the process of capacitive discharging (L□ H)



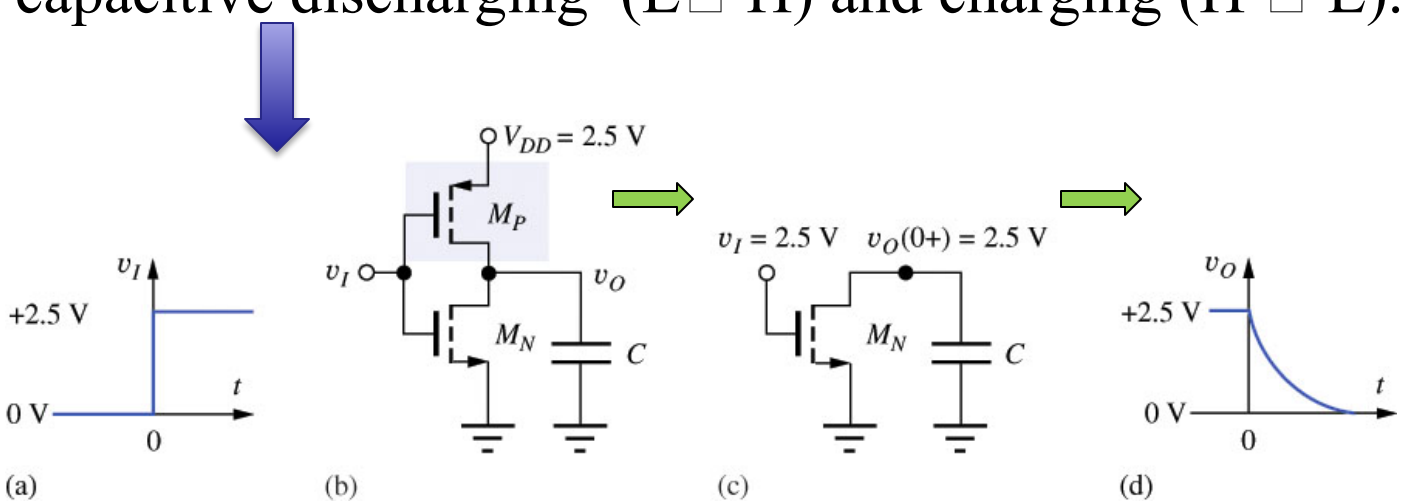
Dynamic Behavior of the CMOS Inverter Propagation Delay Estimate

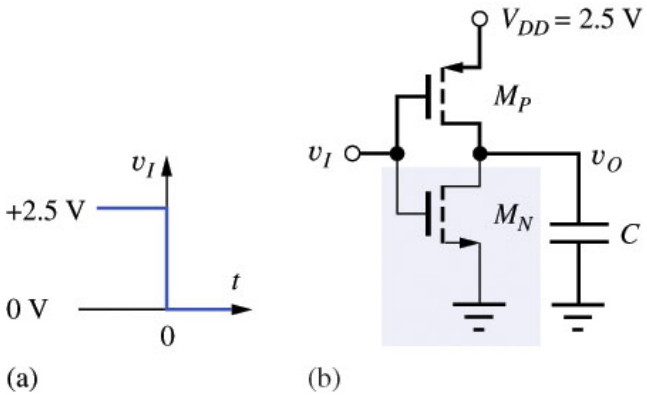
- The propagation delays are created by the process of capacitive discharging (L□ H)



Dynamic Behavior of the CMOS Inverter Propagation Delay Estimate

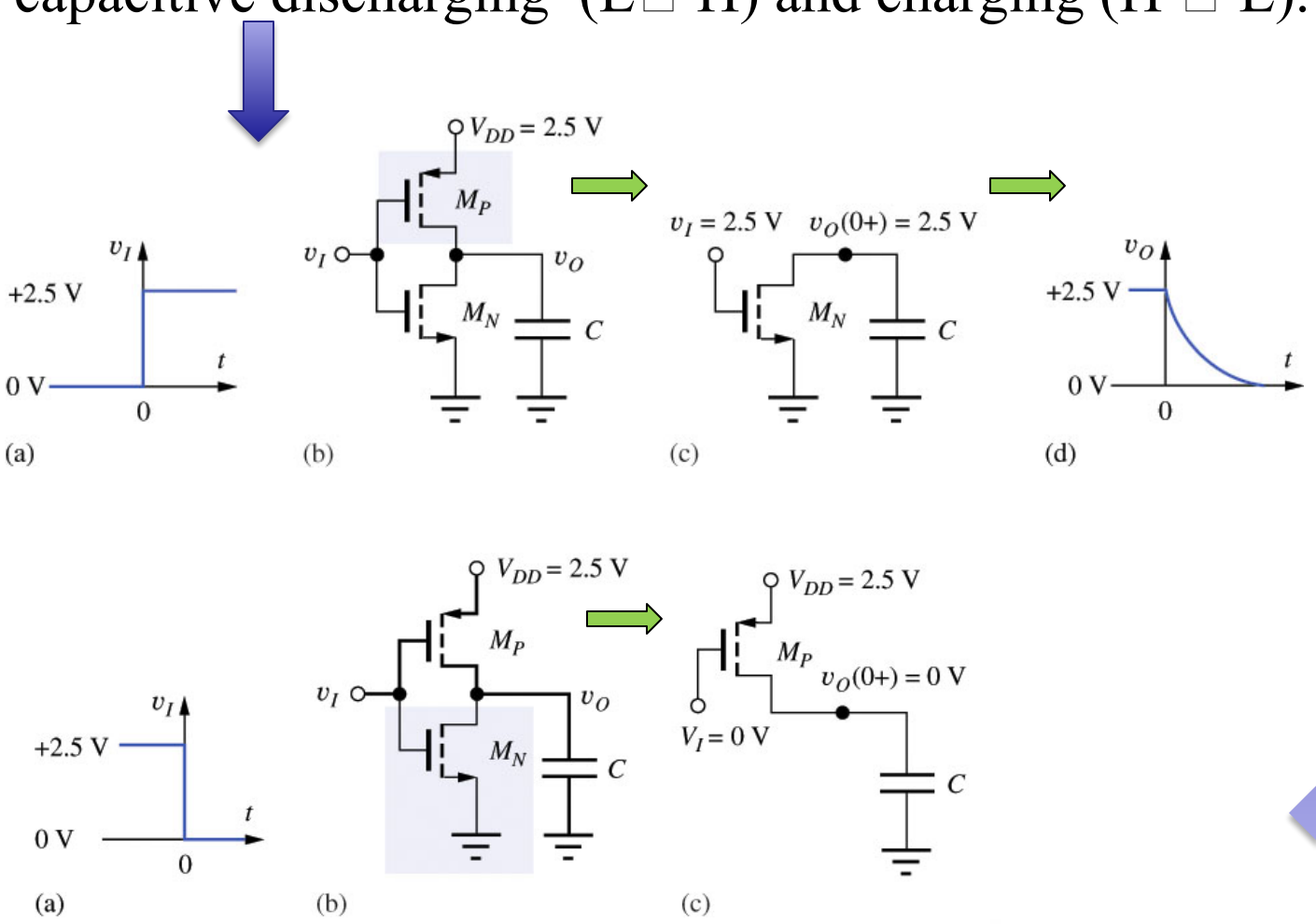
- The propagation delays are created by the process of capacitive discharging (L□ H) and charging (H □ L).





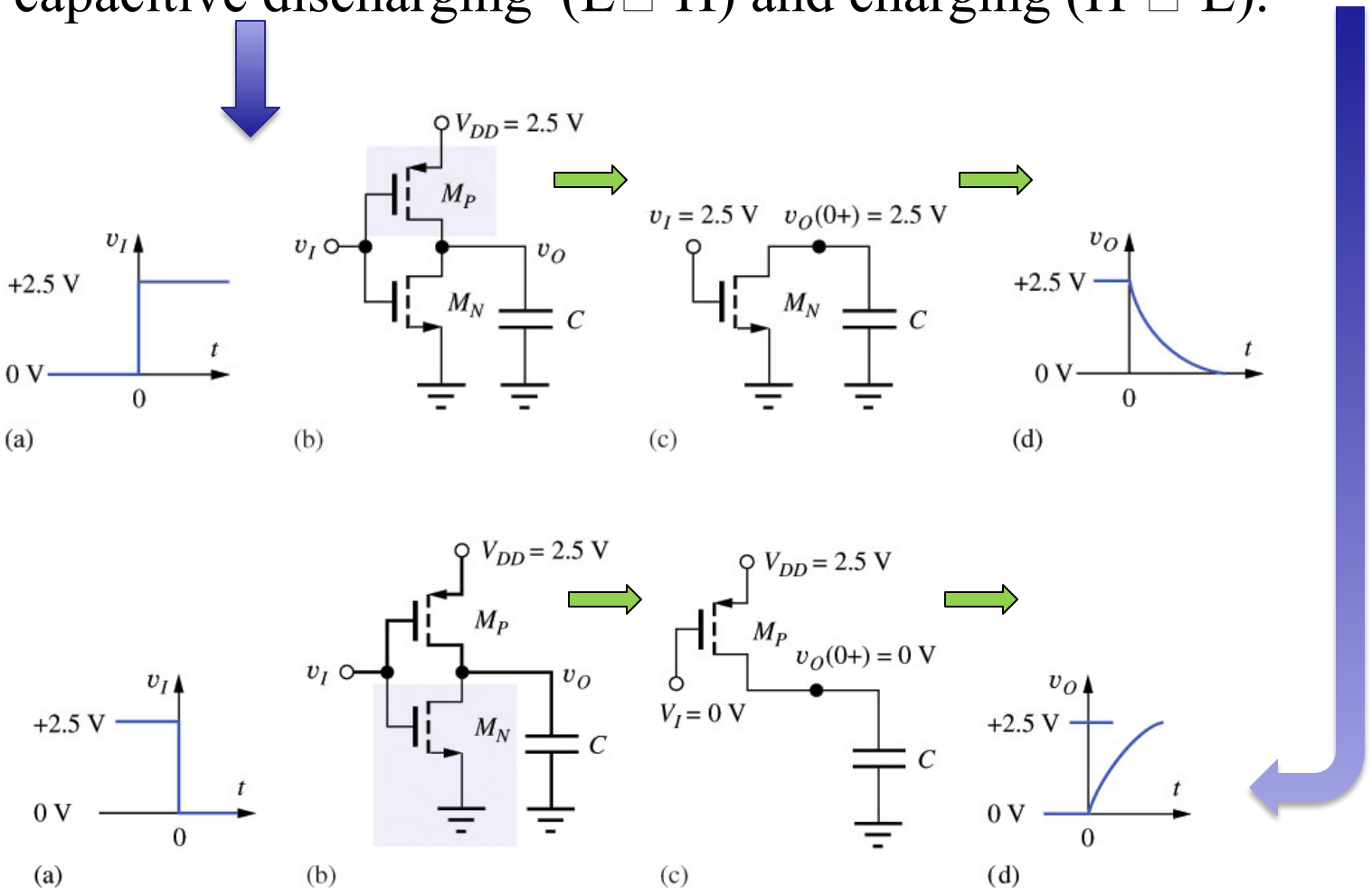
Dynamic Behavior of the CMOS Inverter Propagation Delay Estimate

- The propagation delays are created by the process of capacitive discharging (L□ H) and charging (H □ L).

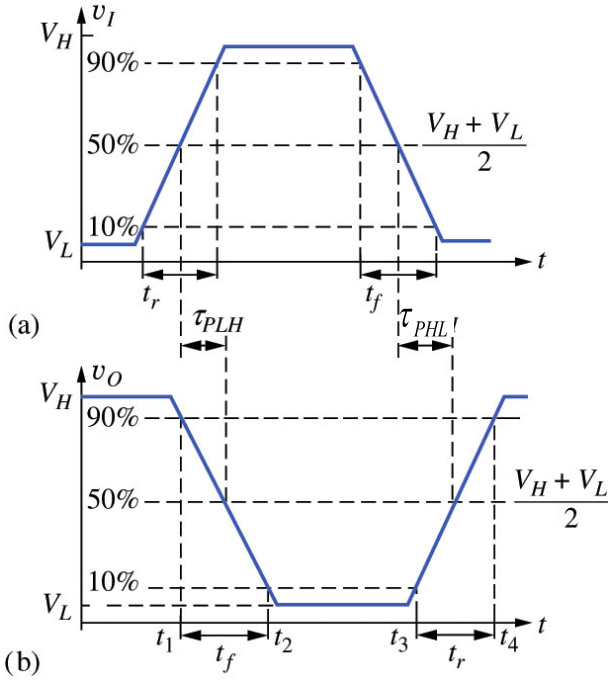


Dynamic Behavior of the CMOS Inverter Propagation Delay Estimate

- The propagation delays are created by the process of capacitive discharging (L□ H) and charging (H □ L).



Propagation Delay & Rise and Fall Times

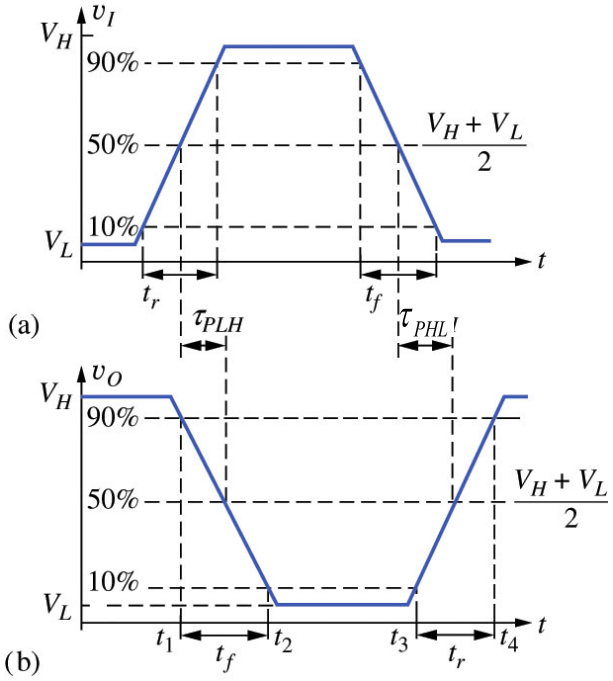


Propagation Delay & Rise and Fall Times

- To estimate the propagation delay, we use the approximate expression, which employs equivalent resistance of a transistor in the ON state, that was developed for the NMOS.

$$\tau_{PHL} = 1.2 R_{onN} C \qquad \tau_{LH} = 1.2 R_{onP} C$$

$$R_{onN} = \frac{1}{K_n \left(V_H - V_{TN} \right)} \qquad R_{onP} = \frac{1}{K_p \left(V_H + V_{TP} \right)}$$

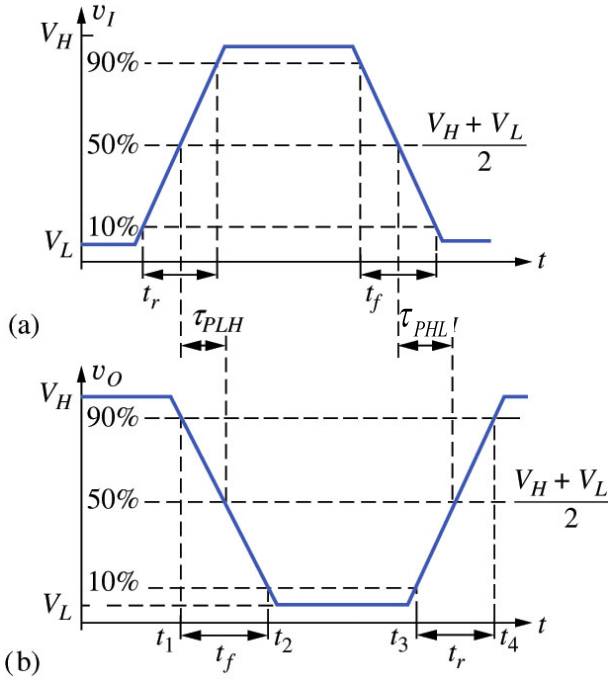


Propagation Delay & Rise and Fall Times

- To estimate the propagation delay, we use the approximate expression, which employs equivalent resistance of a transistor in the ON state, that was developed for the NMOS.

$$\tau_{PHL} = 1.2 R_{onN} C \qquad \tau_{LH} = 1.2 R_{onP} C$$

$$R_{onN} = \frac{1}{K_n \left(V_H - V_{TN} \right)} \qquad R_{onP} = \frac{1}{K_p \left(V_H + V_{TP} \right)}$$



- If it is assumed the inverter is "symmetrical" with $(W/L)_P = 2.5(W/L)_N$, then $\tau_{PLH} = \tau_{PHL}$ and

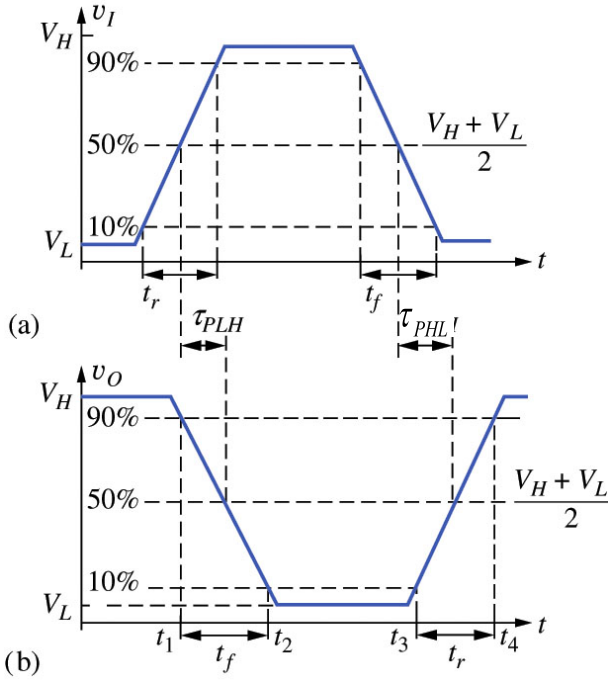
$$\tau_p = \frac{\tau_{PHL} + \tau_{PLH}}{2} = \tau_{PHL} = 1.2 R_{onN} C$$

Propagation Delay & Rise and Fall Times

- To estimate the propagation delay, we use the approximate expression, which employs equivalent resistance of a transistor in the ON state, that was developed for the NMOS.

$$\tau_{PHL} = 1.2 R_{onN} C \qquad \tau_{LH} = 1.2 R_{onP} C$$

$$R_{onN} = \frac{1}{K_n \left(V_H - V_{TN} \right)} \qquad R_{onP} = \frac{1}{K_p \left(V_H + V_{TP} \right)}$$



- If it is assumed the inverter is "symmetrical" (with $(W/L)_P = 2.5(W/L)_N$, since typically $K'_N = 2.5\, K'_P$) then $\tau_{PLH} = \tau_{PHL}$ and

$$\tau_p = \frac{\tau_{PHL} + \tau_{PLH}}{2} = \tau_{PHL} = 1.2 R_{onN} C$$

- The rise and fall times are given by the following approximate expressions:

$$t_f = 3\tau_{PHL}, \quad t_r = 3\tau_{PLH}$$

Reference Inverter Design Example

- Design a reference inverter to achieve a delay of 250ps with a 0.2pF load given the following information:

$$V_{DD} = 3.3V,\ C = 0.2\,pF$$
$$\tau_p = 250\,ps,\ V_{TN} = -V_{TP} = 0.75V$$

Reference Inverter Design Example

- Design a reference inverter to achieve a delay of 250ps with a 0.2pF load given the following information:

$$V_{DD} = 3.3V,\ C = 0.2pF$$
$$\tau_p = 250ps,\ V_{TN} = -V_{TP} = 0.75V$$

- Assuming the inverter is symmetrical and using the typical values (Table 7.1):

$$K_n^{'} = 100\frac{\mu A}{V^2},\ K_p^{'} = 40\frac{\mu A}{V^2}$$
$$\tau_p = \tau_{PHL} = \tau_{PLH} = 250ps$$

Reference Inverter Design Example

- Design a reference inverter to achieve a delay of 250ps with a 0.2pF load given the following information:

$$V_{DD} = 3.3V,\ C = 0.2pF$$
$$\tau_p = 250ps,\ V_{TN} = -V_{TP} = 0.75V$$

- Assuming the inverter is symmetrical and using the typical values (Table 7.1):

$$K_n^{'} = 100\frac{\mu A}{V^2},\ K_p^{'} = 40\frac{\mu A}{V^2}$$
$$\tau_p = \tau_{PHL} = \tau_{PLH} = 250ps$$

- Solving for R_{onN}:

$$R_{onN} = \frac{\tau_{PHL}}{1.2C} = 1040\,\Omega$$

Reference Inverter Design Example

- Design a reference inverter to achieve a delay of 250ps with a 0.2pF load given the following information:

$$V_{DD} = 3.3V,\ C = 0.2\,pF$$
$$\tau_p = 250\,ps,\ V_{TN} = -V_{TP} = 0.75V$$

- Assuming the inverter is symmetrical and using the typical values (Table 7.1):

$$K_n^{'} = 100\frac{\mu A}{V^2},\ K_p^{'} = 40\frac{\mu A}{V^2}$$
$$\tau_p = \tau_{PHL} = \tau_{PLH} = 250\,ps$$

- Solving for R_{onN}:

$$R_{onN} = \frac{\tau_{PHL}}{1.2C} = 1040\,\Omega$$

- Then solving R_{ON} for the transistor ratios:

$$R_{onN} = \frac{1}{K_n\left(\frac{W}{L}\right)_N\left(V_H - V_{TN}\right)}$$

Reference Inverter Design Example

- Design a reference inverter to achieve a delay of 250ps with a 0.2pF load given the following information:

$$V_{DD} = 3.3V,\ C = 0.2\,pF$$
$$\tau_p = 250\,ps,\ V_{TN} = -V_{TP} = 0.75V$$

- Assuming the inverter is symmetrical and using the typical values (Table 7.1):

$$K_n^{'} = 100\frac{\mu A}{V^2},\ K_p^{'} = 40\frac{\mu A}{V^2}$$
$$\tau_p = \tau_{PHL} = \tau_{PLH} = 250\,ps$$

- Solving for R_{onN}:

$$R_{onN} = \frac{\tau_{PHL}}{1.2C} = 1040\,\Omega$$

- Then using equation for R_{ON} :

$$R_{onN} = \frac{1}{K_n\left(\frac{W}{L}\right)_N\left(V_H - V_{TN}\right)}$$

we get

$$\left(\frac{W}{L}\right)_n = \frac{1}{K_n^{'}R_{onN}\left(V_{DD} - V_{TN}\right)} = \frac{3.77}{1}$$

$$\left(\frac{W}{L}\right)_p = \frac{K_n^{'}}{K_p^{'}}\left(\frac{W}{L}\right)_n = 2.5\left(\frac{W}{L}\right)_n = \frac{9.43}{1}$$

Using similarity btw current expressions for N and P

Performance Scaling

- State-of-the-art short length technologies are hard to analyze – the first order *i-v* models may not be accurate for such short scales.

Performance Scaling

- State-of-the-art short length technologies are hard to analyze – the first order *i-v* models may not be accurate for such short scales.
- However, two relationships continue to be true:
  - delay remains proportional to the total load capacitance

Performance Scaling

- State-of-the-art short length technologies are hard to analyze – the first order *i-v* models may not be accurate for such short scales.
- However, two relationships continue to be true:
  - delay remains proportional to the total load capacitance
  - delay is inverse proportional to the (*W/L*)

$$\tau_{PHL} = 1.2 R_{onN} C$$

$$R_{onN} = \frac{1}{K'_n \frac{W}{L}(V_H - V_{TN})}$$

Performance Scaling

- State-of-the-art short length technologies are hard to analyze – the first order *i-v* models may not be accurate for such short scales.
- However, two relationships continue to be true:
  - delay remains proportional to the total load capacitance
  - delay is inverse proportional to the (*W/L*)
- Thus, scaling can be used to set new *W/L* for a new load capacitance relative to reference gate simulation with a reference load capacitance:

$$\tau_P' = \frac{(W/L)_{ref}}{(W/L)'} \times \left(\frac{C_L'}{C_{Lref}}\right) \times \tau_{Pref} \Rightarrow \left(\frac{W}{L}\right)' = \left(\frac{W}{L}\right)_{ref} \times \left(\frac{\tau_{Pref}}{\tau_P'}\right) \times \left(\frac{C_L'}{C_{Lref}}\right)$$

Performance Scaling

- State-of-the-art short length technologies are hard to analyze – the first order *i-v* models may not be accurate for such short scales.
- However, two relationships continue to be true:
  - delay remains proportional to the total load capacitance
  - delay is inverse proportional to the (*W/L*)
- Thus, scaling can be used to set new *W/L* for a new load capacitance relative to reference gate simulation with a reference load capacitance:

$$\tau'_P = \frac{(W/L)_{ref}}{(W/L)'} \times \left(\frac{C_L'}{C_{Lref}}\right) \times \tau_{Pref} \Rightarrow \left(\frac{W}{L}\right)' = \left(\frac{W}{L}\right)_{ref} \times \left(\frac{\tau_{Pref}}{\tau'_P}\right) \times \left(\frac{C_L'}{C_{Lref}}\right)$$

Scaling allows us to calculate a new geometry (W/L)' in terms of a target load and delay.

Performance Scaling Example

- Consider a reference inverter with a delay of 3.16 ns and W/L = 2/1.
- What is the delay if an inverter has a W/L 4 time larger than the transistors of the reference inverter and twice the load capacitance?

Performance Scaling Example

- Consider a reference inverter with a delay of 3.16 ns and *W/L* = 2/1.
- What is the delay if an inverter has a *W/L* 4 time larger than the transistors of the reference inverter and twice the load capacitance?

$$\tau_P = \frac{(2/1)}{(8/1)'} \times \left(\frac{2pF'}{1pF} \right) \times \ 3.16\ ns = 1.58\ ns$$

Scaling allows us to calculate *new geometry (W/L)' or delay* relative to a reference design.

Static Power Dissipation

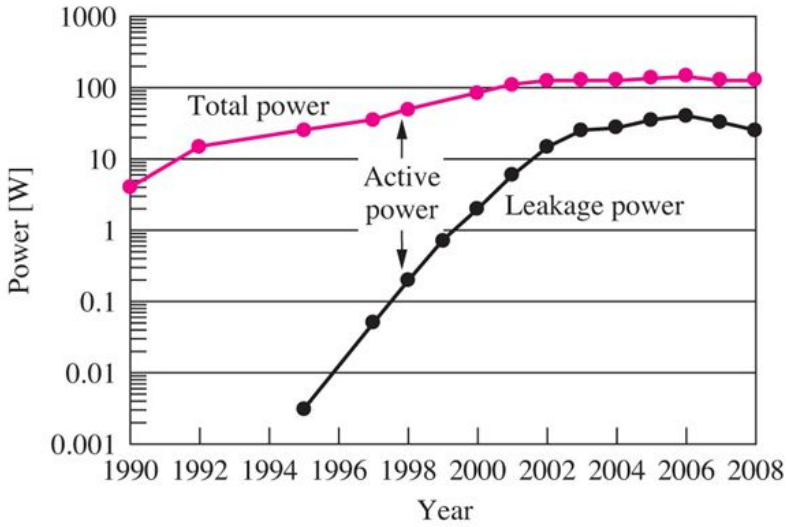
- **CMOS logic is considered to have no static power dissipation.**

Static Power Dissipation

- **CMOS logic is considered to have no static power dissipation.**
- This is not completely accurate since MOS transistors have leakage currents associated with the reverse-biased drain-to-substrate connections as well as sub-threshold leakage current between the drain and source.

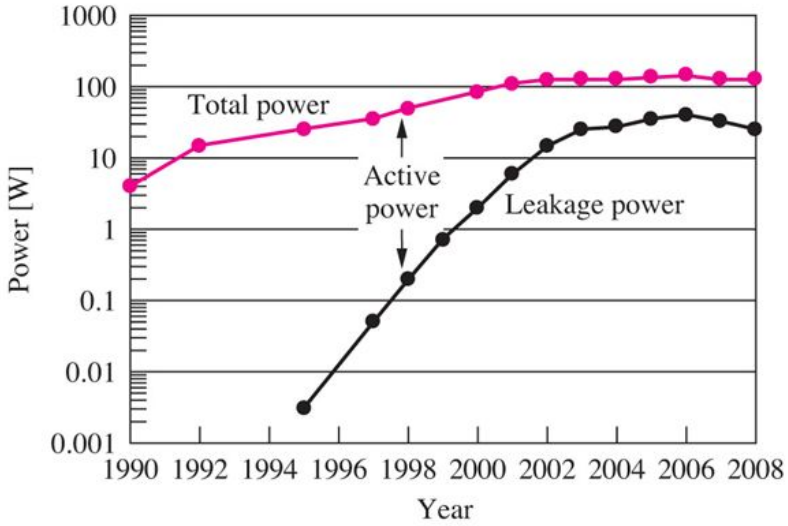
Static Power Dissipation

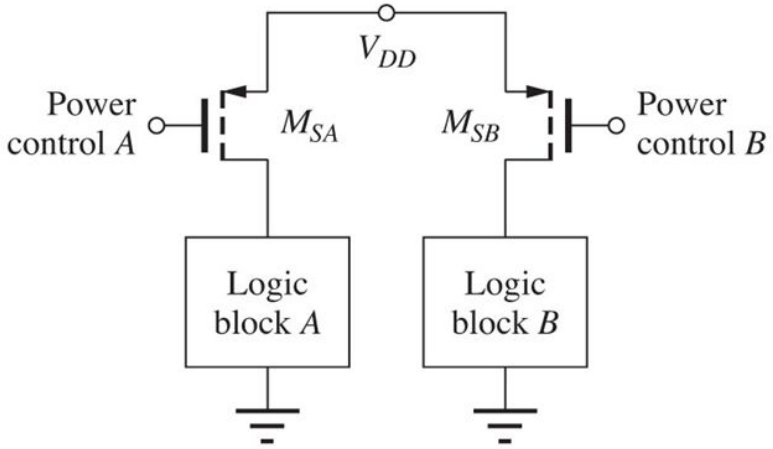
- **CMOS logic is considered to have no static power dissipation.**
- This is not completely accurate since MOS transistors have **leakage currents** associated with the reverse-biased drain-to-substrate connections as well as sub-threshold leakage current between the drain and source.
- For the sub 0.1 μm logic technologies, the leakage power approaches 30% of total chip power.



Static Power Dissipation

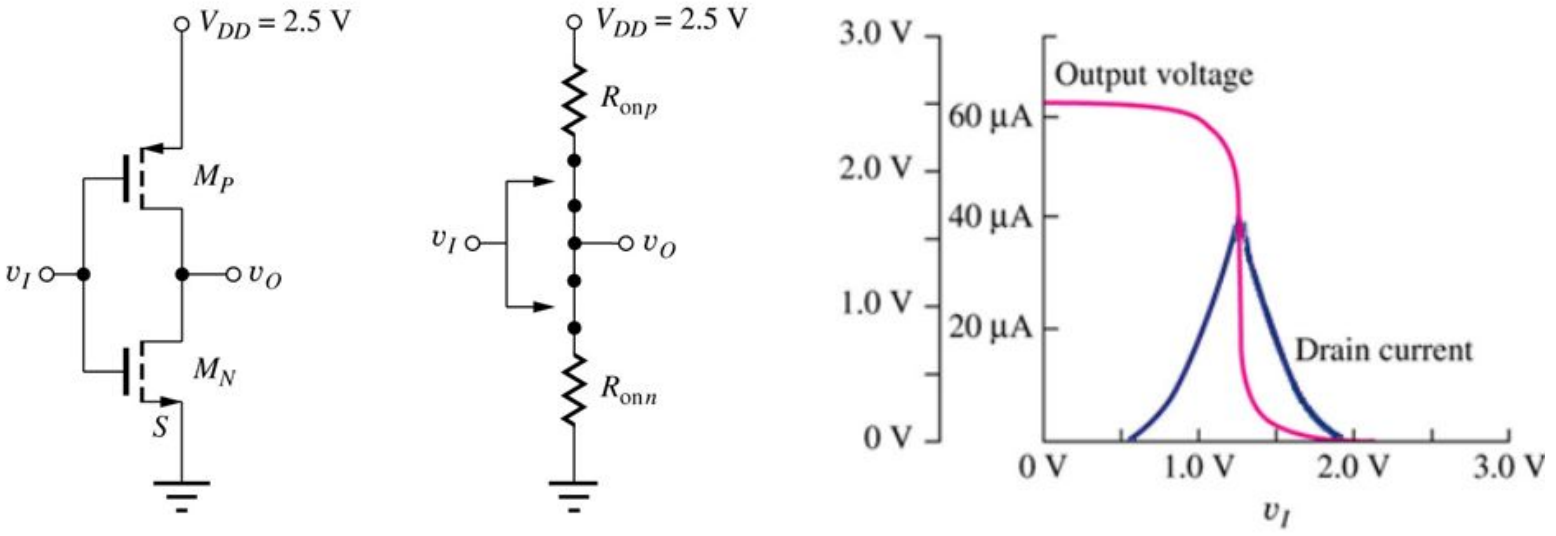
- **CMOS logic is considered to have no static power dissipation.**
- This is not completely accurate since MOS transistors have **leakage currents** associated with the reverse-biased drain-to-substrate connections as well as sub-threshold leakage current between the drain and source.
- For the sub 0.1 μm logic technologies, the leakage power approaches 30% of total chip power.
- Special methods are developed to reduce the static power, like adding large PMOS to control the power to the certain logic blocks when they are not needed, which can be done either using hardware or software control.





Dynamic Power Dissipation

- There are two components that contributes to dynamic power dissipation:
- As we found earlier, the capacitive load charging/discharging at a frequency f is responsible for power dissipation $P_D = CV^2_{DD} f$
- The second mechanism is created by drain current through both transistors during the short period of switching when both transistors are ON and in saturation region.
- That current exists when $V_{TN} < v_I < (V_{DD} - |V_{TP}|)$ and reaches max when $v_I = v_O = V_{DD}/2$.



Power-Delay Product

- The power-delay product is defined as $PDP = P_{av}\tau_P$

Power-Delay Product

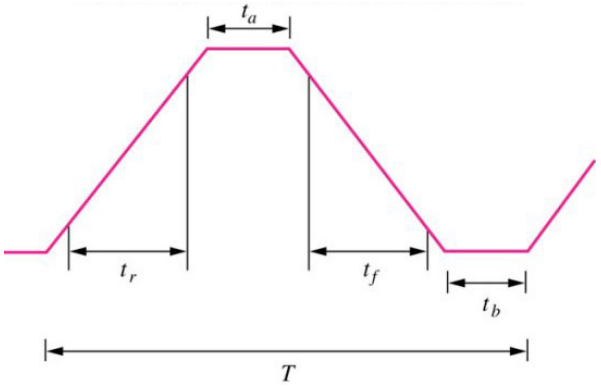
- The power-delay product is defined as $PDP = P_{av}\tau_P$
- This is the important characteristic that tells how much energy is needed to change the state of the circuit.
- Early logic had PDP as 10 to 100 pJ, the current has PDP in 10 to 100 fJ.

Power-Delay Product

- The power-delay product is defined as $PDP = P_{av}\tau_P$
- This is the important characteristic that tells how much energy is needed to change the state of the circuit.
- Early logic had PDP as 10 to 100 pJ, the current has PDP in 10 to 100 fJ.
- As we know, for the high frequency logic, the dominant component is charging/discharging power $P_{av} = CV_{DD}^2 f$, where f=1/T.

Power-Delay Product

- The power-delay product is defined as $PDP = P_{av}\tau_P$
- This is the important characteristic that tells how much energy is needed to change the state of the circuit.
- Early logic had PDP as 10 to 100 pJ, the current has PDP in 10 to 100 fJ.
- As we know, for the high frequency logic, the dominant component is charging/discharging power $P_{av} = CV_{DD}^2 f$, where f=1/T.
- For a symmetrical inverter waveform, the switching period is minimized by making t_a and t_b close to zero and t_r and t_f about 80% of T.

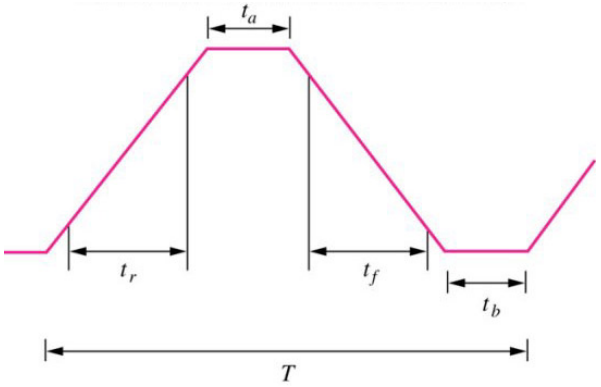


Power-Delay Product

- The power-delay product is defined as $PDP = P_{av}\tau_P$
- This is the important characteristic that tells how much energy is needed to change the state of the circuit.
- Early logic had PDP as 10 to 100 pJ, the current has PDP in 10 to 100 fJ.
- As we know, for the high frequency logic, the dominant component is charging/discharging power $P_{av} = CV_{DD}^2 f$, where f=1/T.
- For a symmetrical inverter waveform, the switching period is minimized by making t_a and t_b close to zero and t_r and t_f about 80% of T.
- Thus

$$T = t_r + t_a + t_f + t_b \approx \frac{2t_r}{0.8} = \frac{2(2\tau_P)}{0.8} = 5\tau_P$$

$$PDP \approx \frac{CV_{DD}^2}{5\tau_P}\tau_P = \frac{CV_{DD}^2}{5}$$

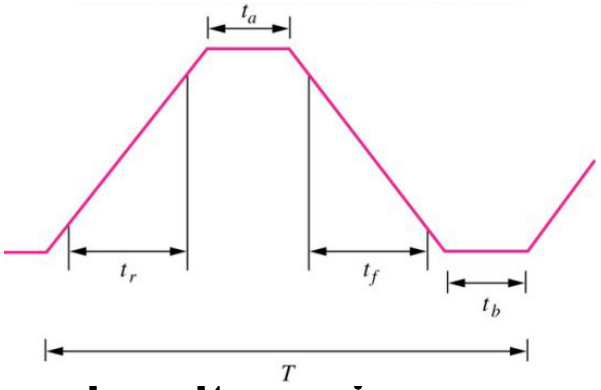


Power-Delay Product

- The power-delay product is defined as $PDP = P_{av}\tau_P$
- This is the important characteristic that tells how much energy is needed to change the state of the circuit.
- Early logic had PDP as 10 to 100 pJ, the current has PDP in 10 to 100 fJ.
- As we know, for the high frequency logic, the dominant component is charging/discharging power $P_{av} = CV_{DD}^2 f$, where f=1/T.
- For a symmetrical inverter waveform, the switching period is minimized by making t_a and t_b close to zero and t_r and t_f about 80% of T.
- Thus

$$T \approx t_r + t_a + t_f + t_b = \frac{2t_r}{0.8} = \frac{2(2\tau_P)}{0.8} = 5\tau_P$$

$$PDP \approx \frac{CV_{DD}^2}{5\tau_P}\tau_P = \frac{CV_{DD}^2}{5}$$



- **This shows importance of reducing the power supply voltage, since PDP is reduced proportionally to square of V_{DD}.**

Minimum Size Gate Design and Performance

- With CMOS technology, there is an design area/propagation delay tradeoff that needs to be considered.

Minimum Size Gate Design and Performance

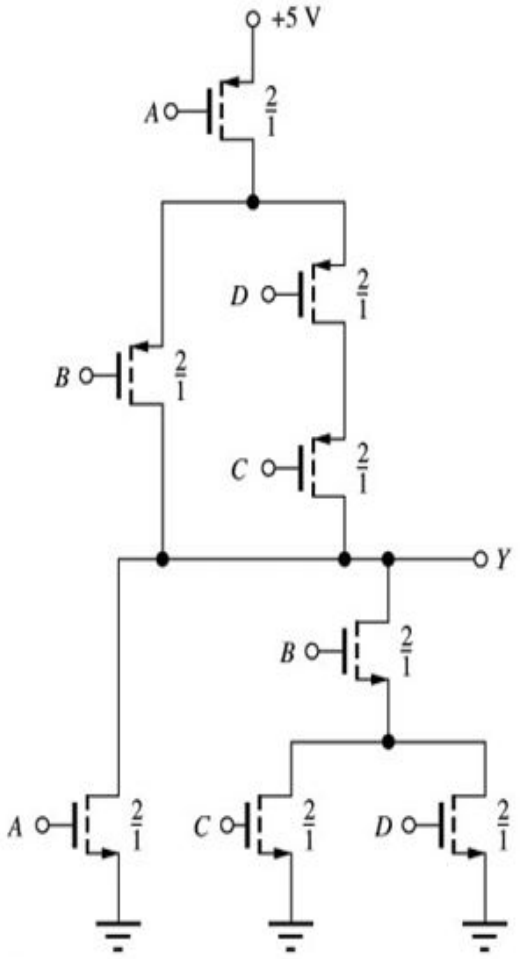
- With CMOS technology, there is an design area/propagation delay tradeoff that needs to be considered.
- In the AND sequences, in order to minimize delay we want to provide the max current and has to increase the width of design.

Minimum Size Gate Design and Performance

- With CMOS technology, there is an design area/propagation delay tradeoff that needs to be considered.
- In the AND sequences, in order to minimize delay we want to provide the max current and has to increase the width of design.
- However this increases the total design area and the density of transistors.

Minimum Size Gate Design and Performance

- With CMOS technology, there is an design area/propagation delay tradeoff that needs to be considered.
- In the AND sequences, in order to minimize delay we want to provide the max current and has to increase the width of design.
- However this increases the total design area and the density of transistors.
- Thus if delay is of lesser importance, we can use the minimum geometry design for ALL transistors



Minimum Size Gate Design and Performance

- With CMOS technology, there is an design area/propagation delay tradeoff that needs to be considered.
- In the AND sequences, in order to minimize delay we want to provide the max current and has to increase the width of design.
- However this increases the total design area and th density of transistors.
- Thus if delay is of lesser importance, we can use the minimum geometry design for ALL transistors

- **If minimum feature sized are used for both devices, then the τ_{PLH} will be increased compared to the symmetrical reference inverter**