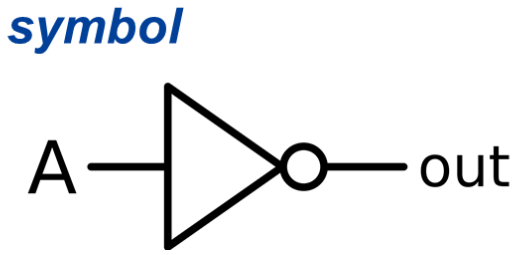
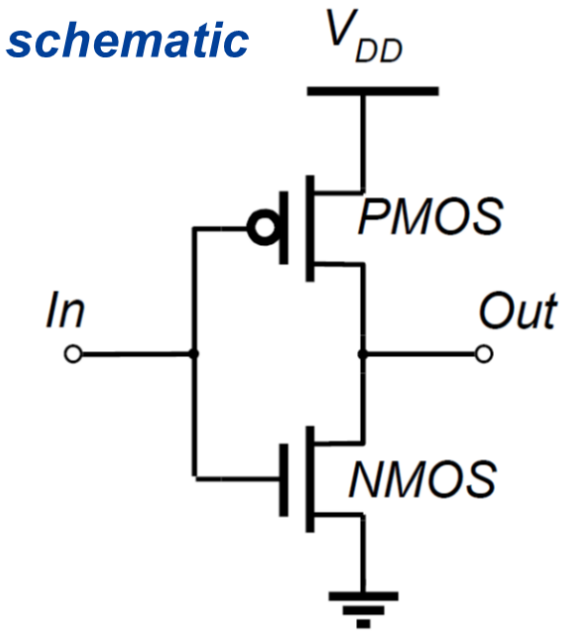
Lecture 5: Static CMOS Logic Circuit

Outline

- **Construction of Static CMOS Logic Gate**
- **Static CMOS Logic Circuit Delay and Power Analysis**
- **Static CMOS Logic Circuit Advanced Delay Analysis**
  - Intermediate Capacitance
  - Delay Optimization

Overview





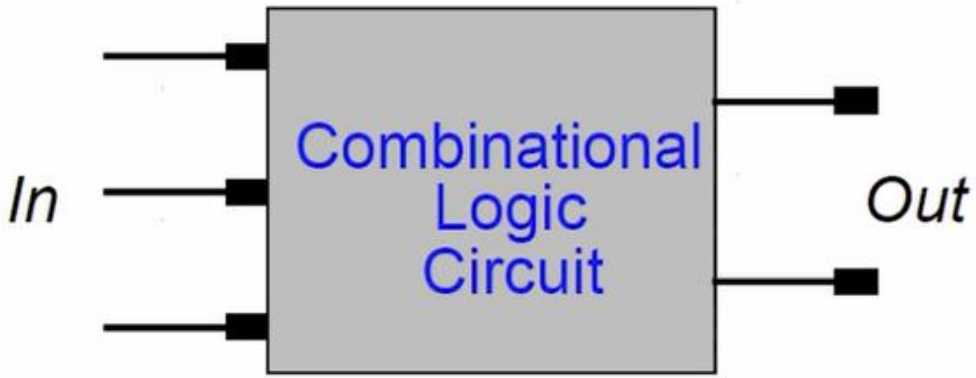
Inverter

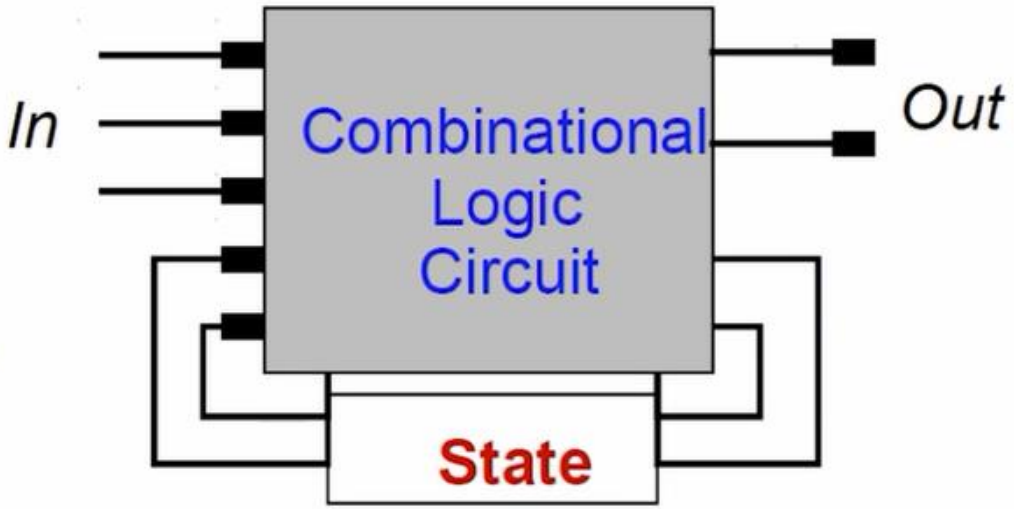
- ✓ DC Characterizations – VTC
- ✓ Propagation Delay
- ✓ Power Consumption
- ✓ Design Optimization

combination

Logic Gate

Combinational vs. Sequential





Combinational

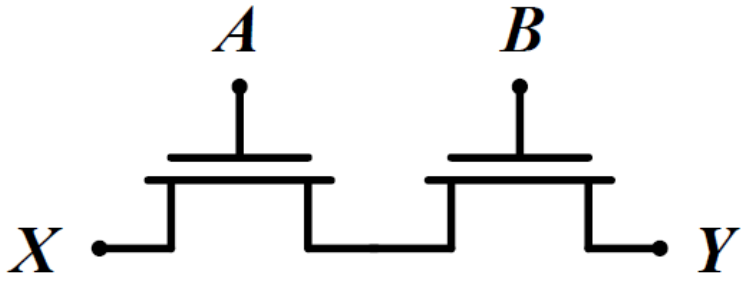
Output = f(In)

Sequential

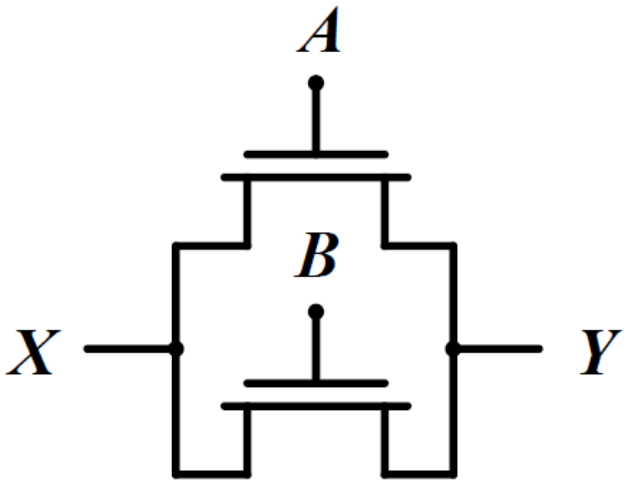
Output = f(In, State)

NMOS Transistors in Series/Parallel

- Transistor ↔ switch controlled by its gate signal
- NMOS switch opens when switch control input is *HIGH*



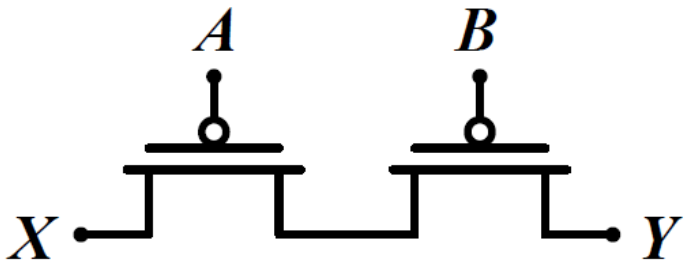
$Y = X$ when A AND B



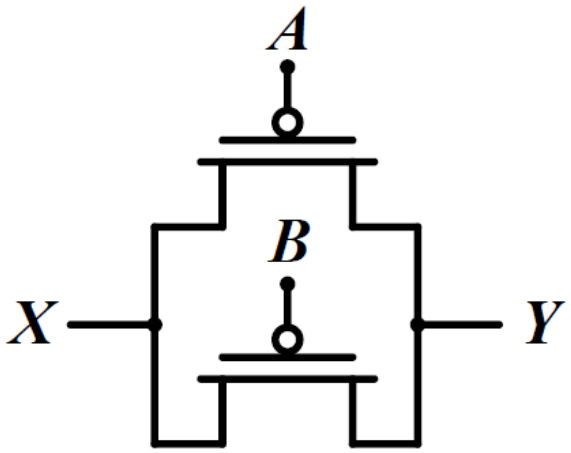
$Y = X$ when A OR B

PMOS Transistors in Series/Parallel

- Transistor ↔ switch controlled by its gate signal
- PMOS switch opens when switch control input is *LOW*

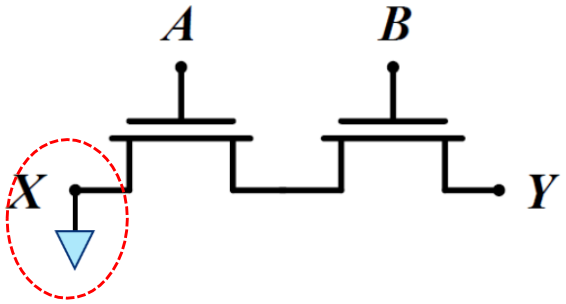


Y = X when $\overline{A}$ AND $\overline{B}$



Y = X when $\overline{A}$ OR $\overline{B}$

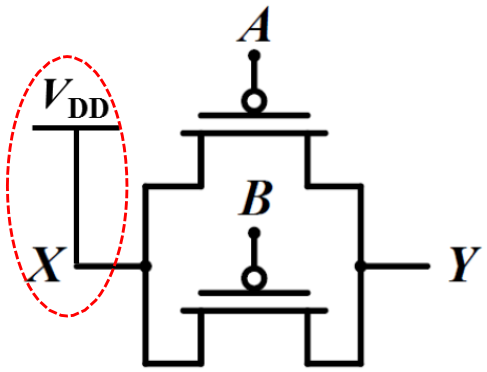
Static CMOS NAND Logic Gates



$Y = 0$ when $A \cdot B$

$\Leftrightarrow$

$Y = \overline{A \cdot B}$



Y = 1 when $\overline{A} + \overline{B}$

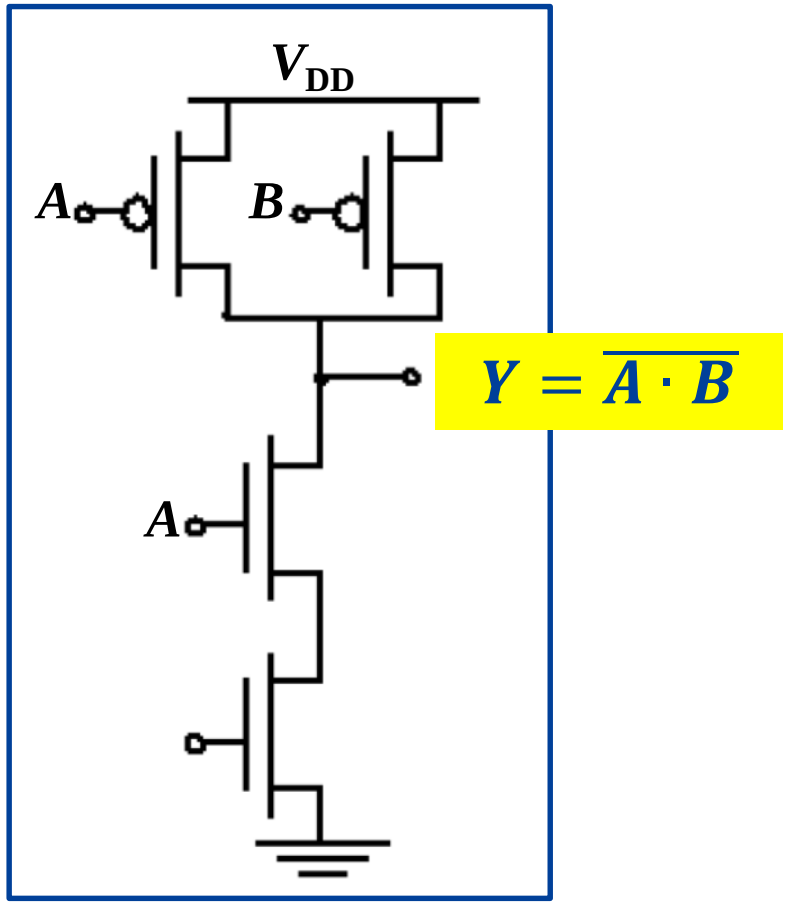
$\Leftrightarrow$

$Y = \overline{A} + \overline{B}$

$\Leftrightarrow$

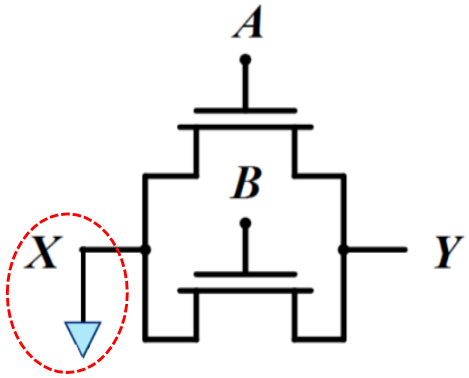
Apply DeMorgan's Theorems

$Y = \overline{A \cdot B}$



2-input NAND

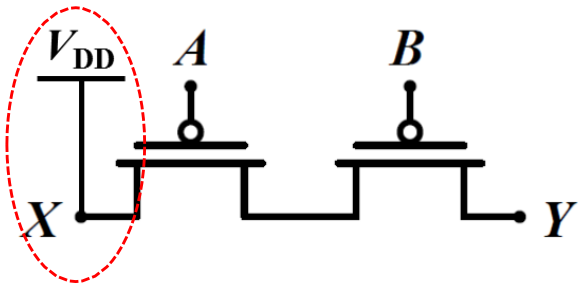
Static CMOS NOR Logic Gates



$Y = 0$ when $A + B$

$\Leftrightarrow$

$Y = \overline{A + B}$



Y = 1 when $\overline{A} \cdot \overline{B}$

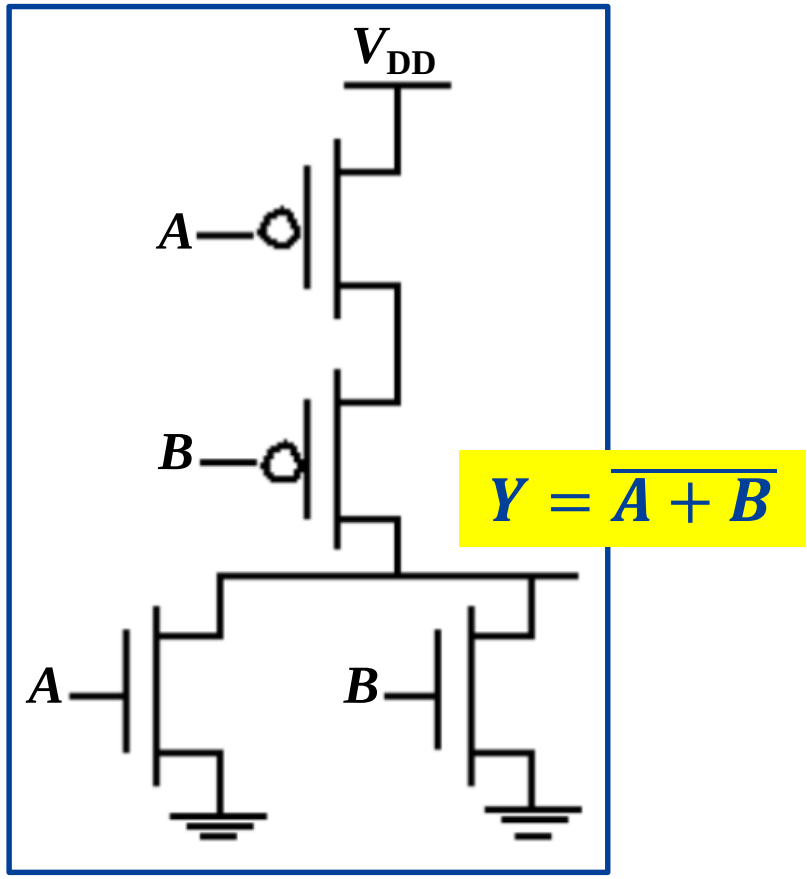
$\Leftrightarrow$

$Y = \overline{A} \cdot \overline{B}$

$\Leftrightarrow$

Apply DeMorgan's Theorems

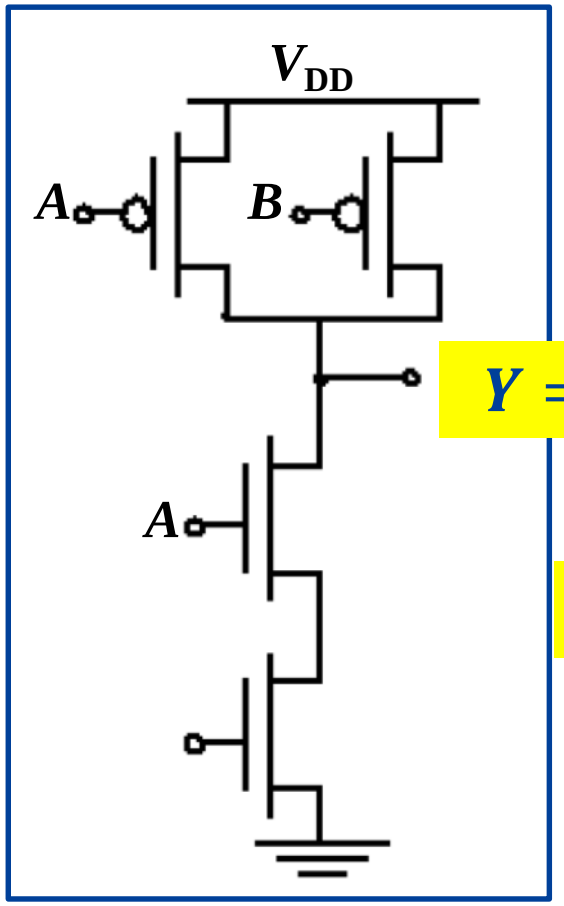
$Y = \overline{A + B}$



2-input NOR

Example: NAND & NOR Logic Gates

2-input NAND

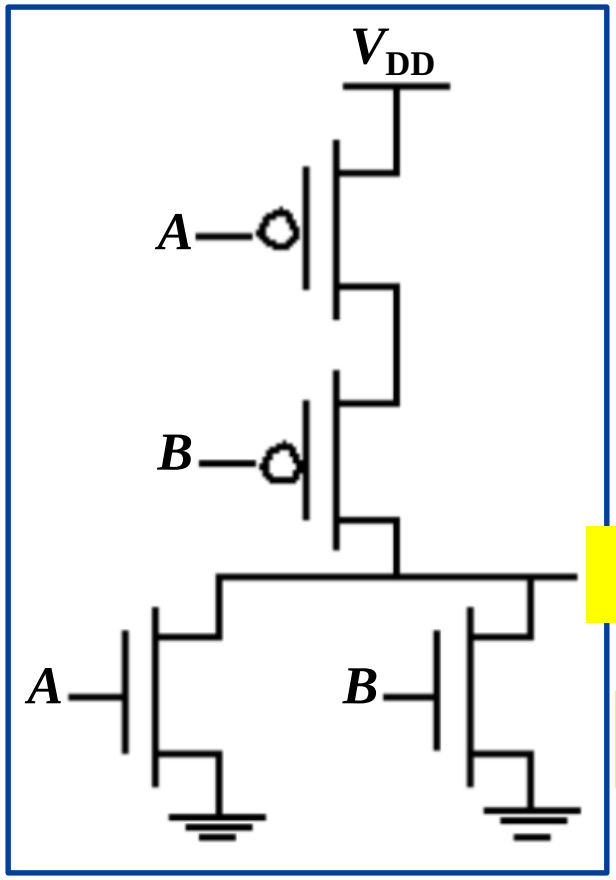


$$Y = \overline{A \cdot B}$$

$$Y = \overline{A_1 \cdot A_2 \cdots A_n}$$

n-input NAND

2-input NOR



$$Y = \overline{A + B}$$

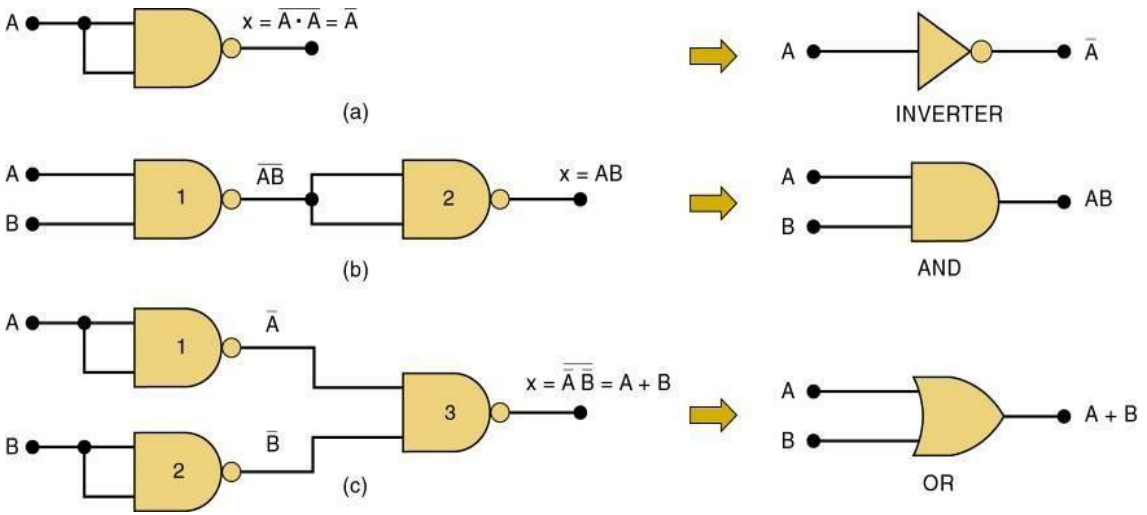
$$Y = \overline{A_1 + A_2 \cdots + A_n}$$

n-input NOR

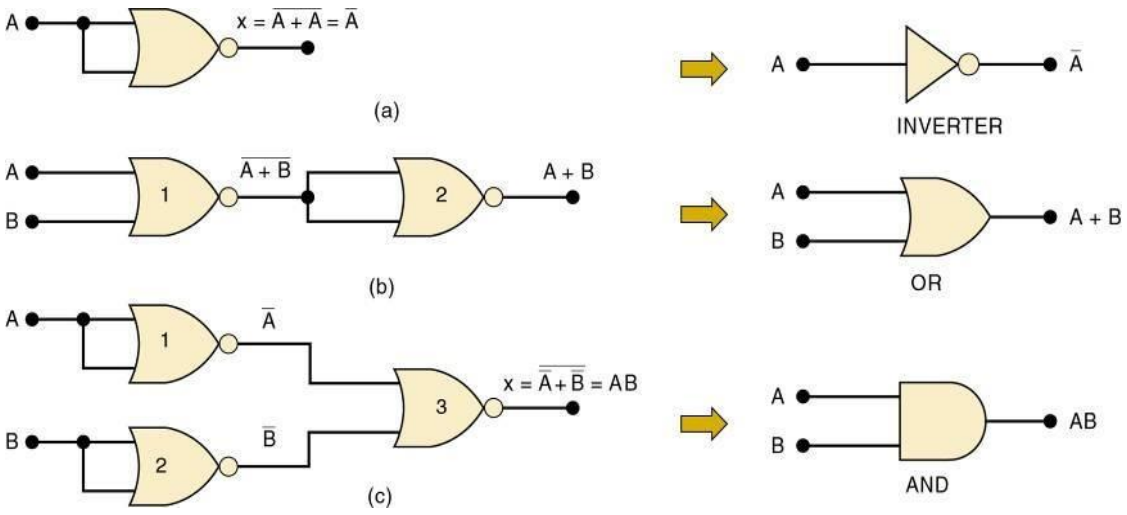
- ***n*-input NAND can be constructed by the combination of #n PMOS in parallel to V_{DD} and #n NMOS in series to GND.**
- ***n*-input NOR can be constructed by the combination of #n NMOS in parallel to GND and #n PMOS in series to V_{DD}.**

Other Static CMOS Logic Gates

- All combinational logic circuits can be implemented by the combinations of static CMOS NAND and/or NOR gates!



Combination of NAND



Combination of NOR

Example: Constructing a Complex Gate

$$Y = \overline{D + A(B + C)}$$

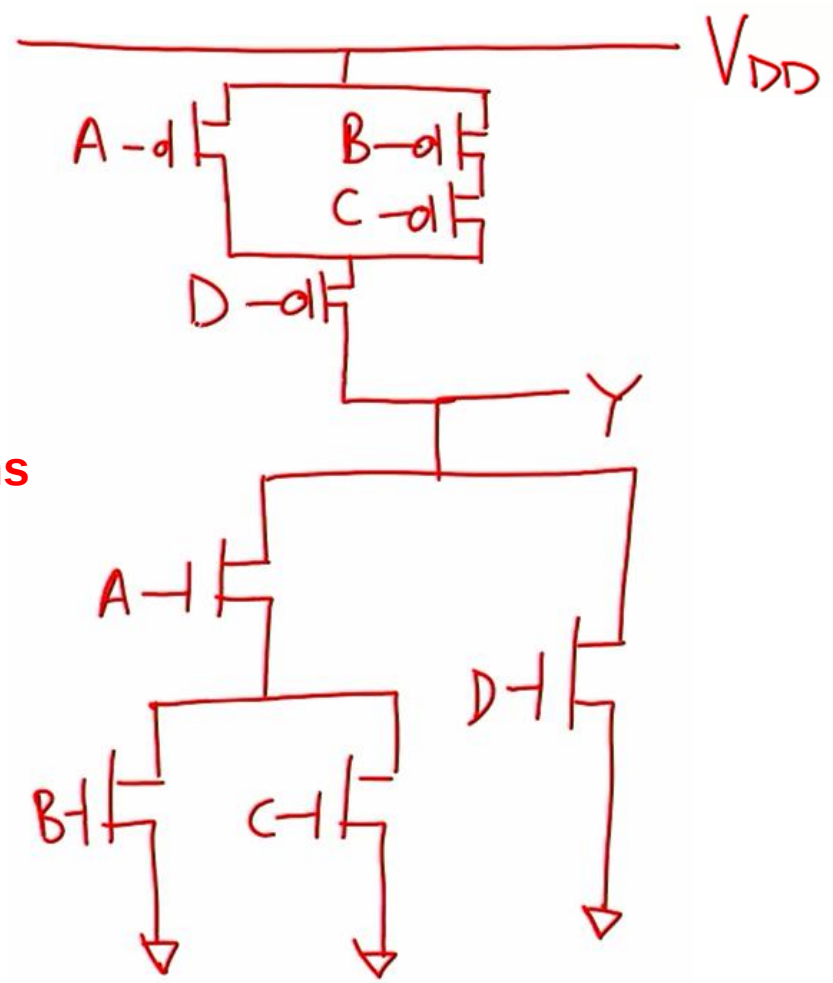
$$Y = \overline{D}\,(\overline{A} + \overline{B}\,\overline{C})$$

PMOSs:

Apply DeMorgan's Theorems

$$Y = \overline{D + A(B + C)}$$

NMOSs:



Example: Constructing XOR Gate

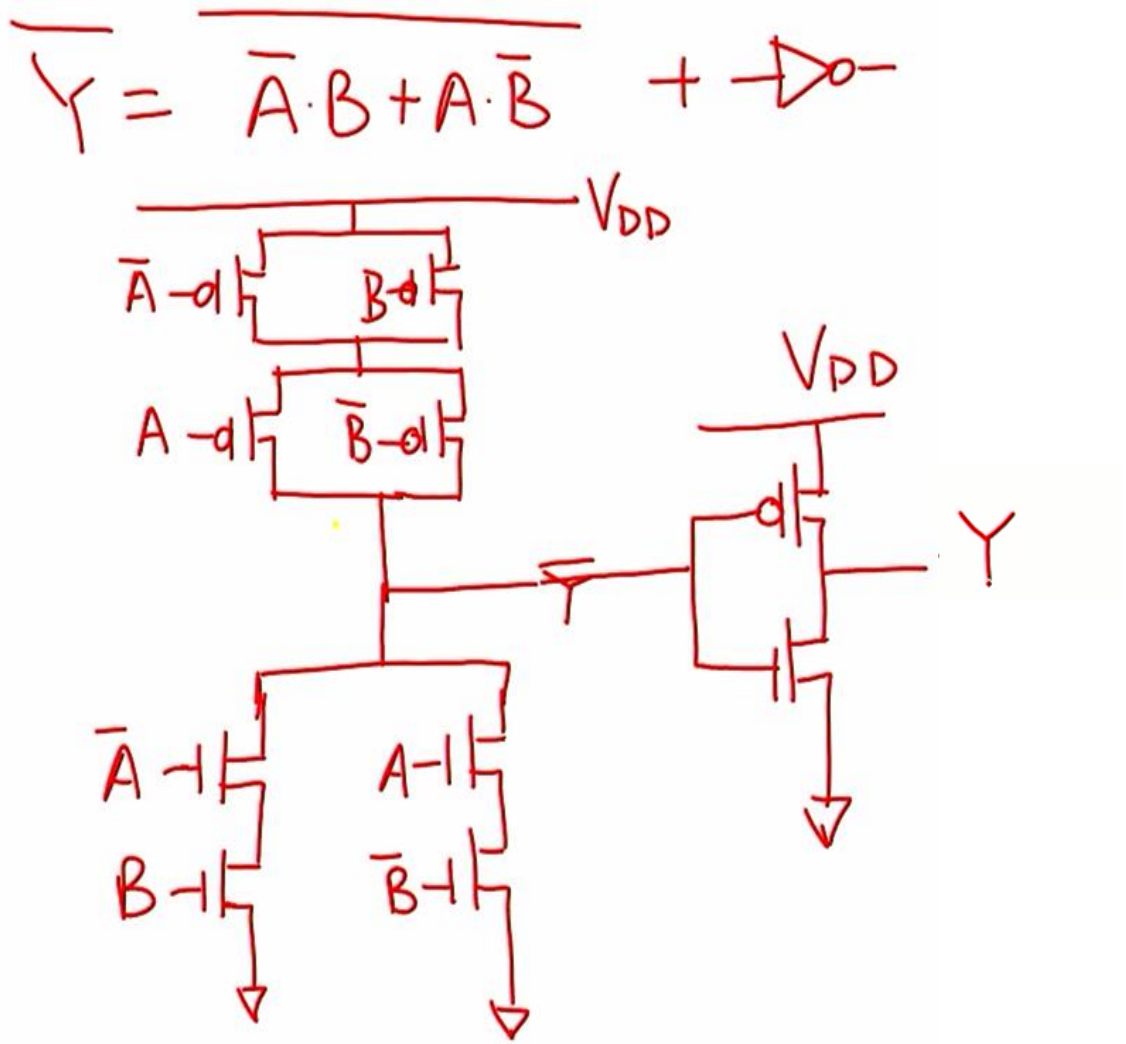
***KEY*: Apply Standard SOP(Sum Of Product)**
Process for *Y*

Static 2-input XOR Gate

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0

$$Y = \overline{A} \cdot B + A \cdot \overline{B}$$

$$\overline{Y} = \overline{\overline{A} \cdot B + A \cdot \overline{B}}$$



Example: Constructing XOR Gate

***KEY*: Apply Standard SOP(Sum Of Product)**

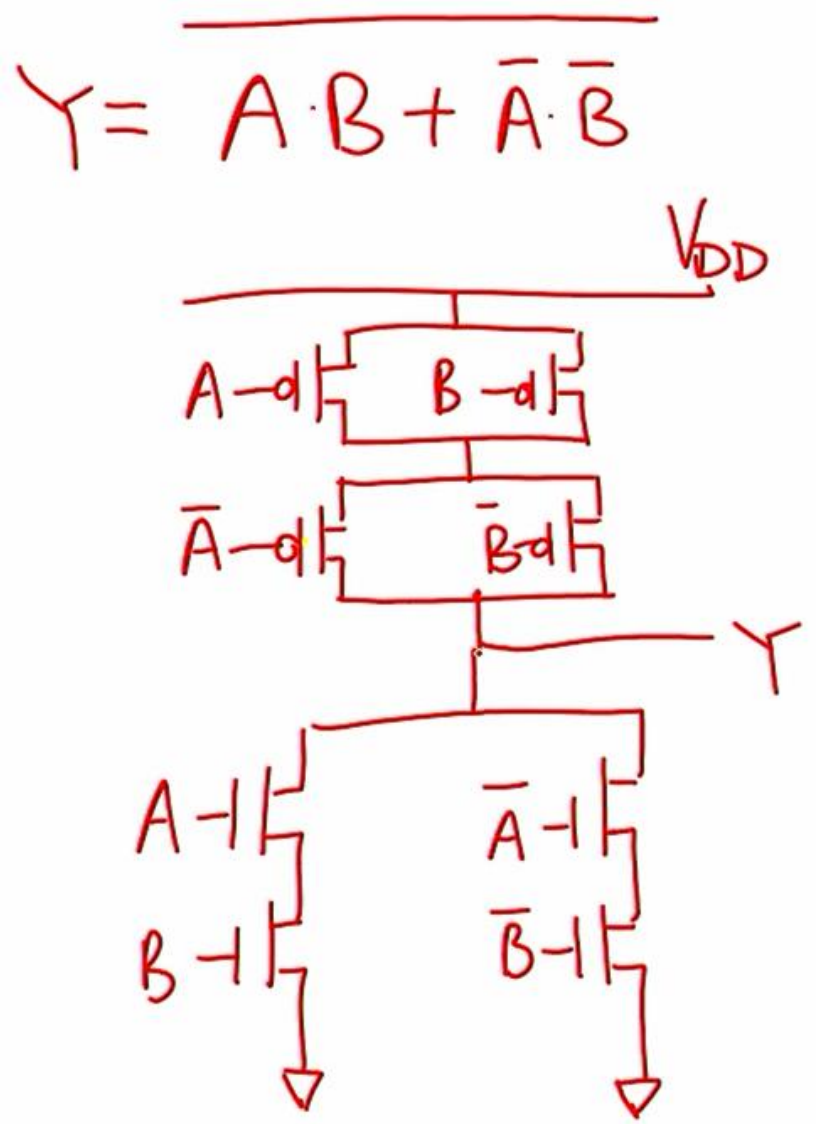
Process for $\overline{Y}$

Static 2-input XOR Gate

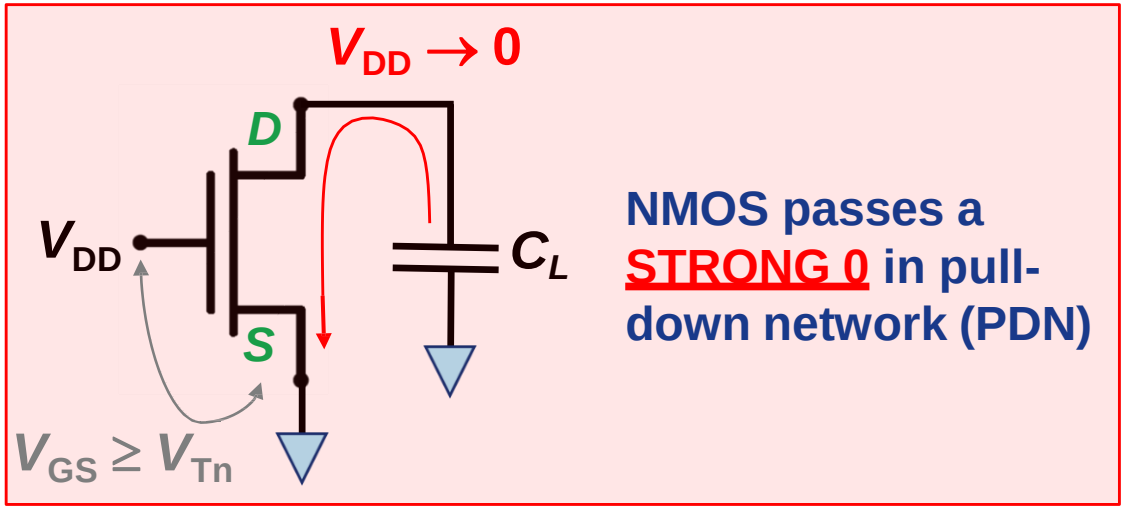
A	B	Y	$\overline{Y}$
0	0	0	1
0	1	1	0
1	0	1	0
1	1	0	1

$$\overline{Y} = A \cdot B + \overline{A} \cdot \overline{B}$$

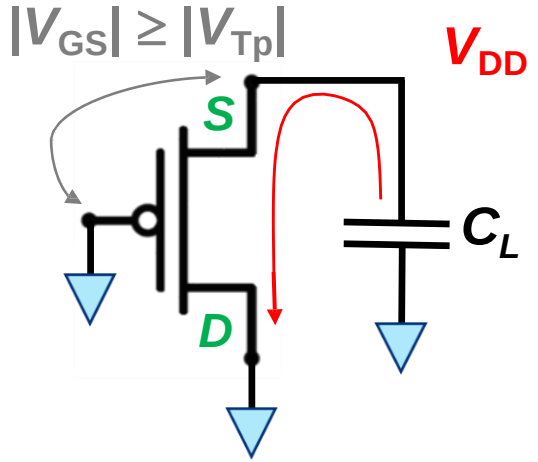
$$Y = \overline{A \cdot B + \overline{A} \cdot \overline{B}}$$



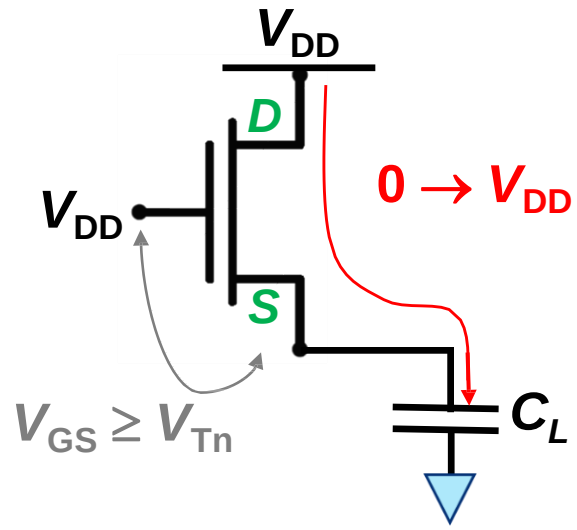
Weak and Strong PUN & PDN



NMOS passes a **STRONG 0** in pull-down network (PDN)

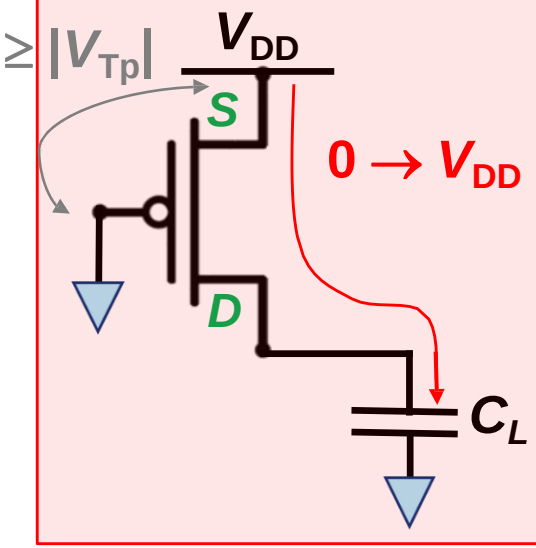


PMOS passes a **WEAK 0** in pull-down network (PDN)



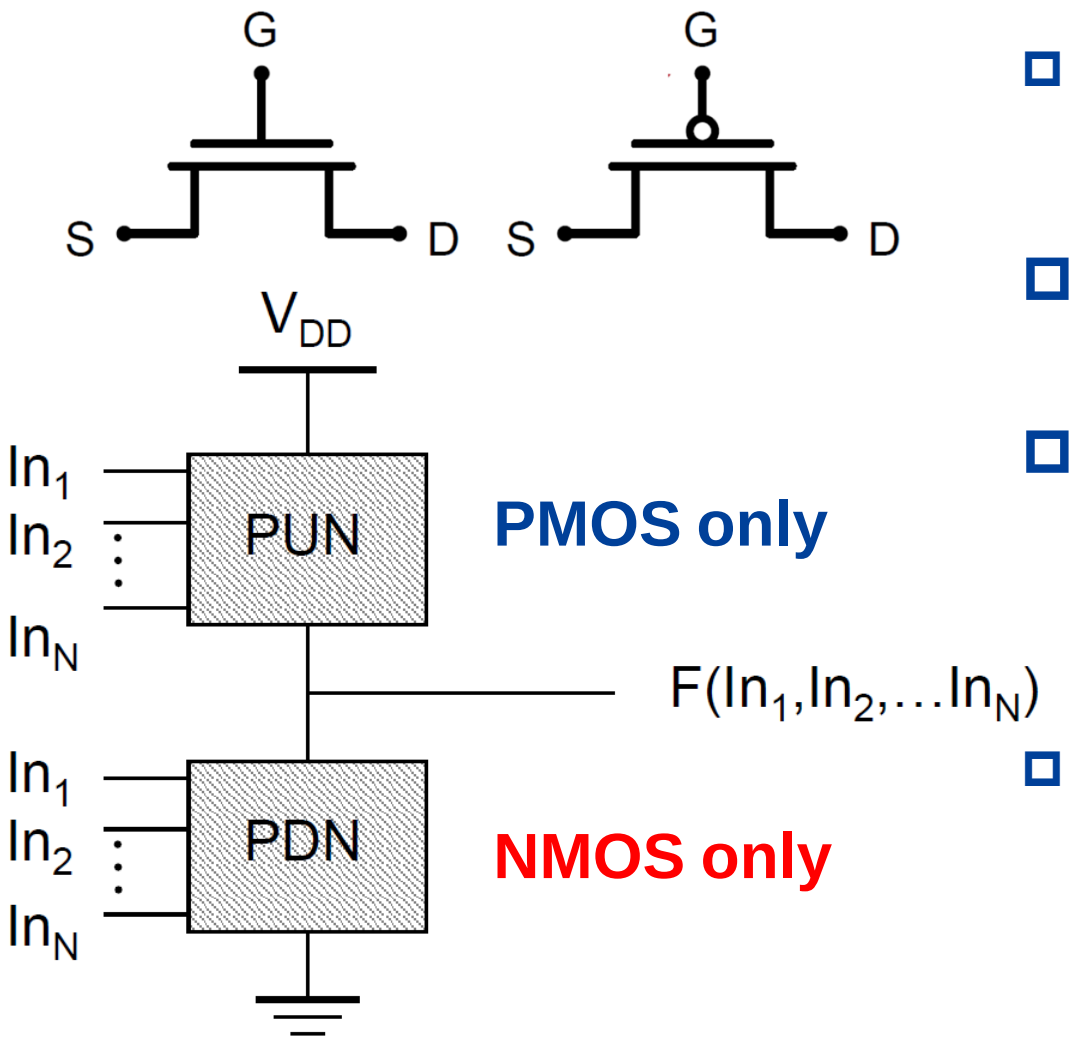
NMOS passes a **WEAK 1** in pull-up network (PUN)

$|V_{GS}| \geq |V_{Tp}|$



PMOS passes a **STRONG 1** in pull-up network (PUN)

Summary



- The pull-up network (PUN) is made of PMOS transistors ONLY while the pull-down network (PDN) is made of NMOS transistors ONLY.
- Logic inputs connect to the *Gate* terminal of MOSFETs.
- The output of Static CMOS logic gate is connected to V_{DD} or GND via a *low-resistive path* at every point in time (except during the switching transients).
- The outputs of the static CMOS logic gates at all times are the value of the Boolean function, implemented by the circuit at steady state.