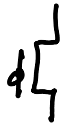
1.12eV

keep ei static

dopant and temperature

is what matters!

CMOS GATES

PMOS = 

James E. Stine, Jr.

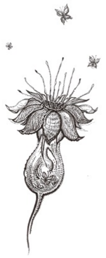
Edward Joullian Endowed Chair in Engineering

Oklahoma State University

Electrical and Computer Engineering Department

Stillwater, OK 74078 USA

james.stine@okstate.edu



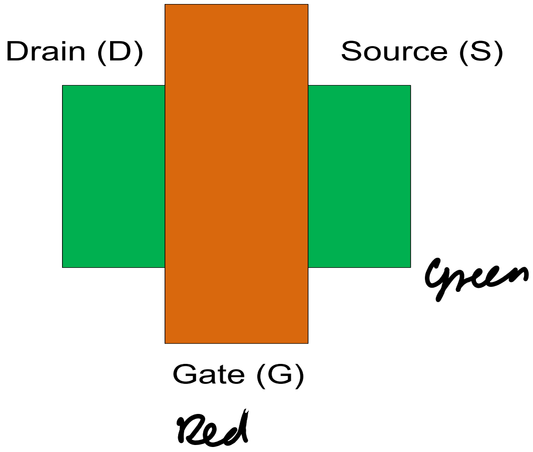
[Wiki]

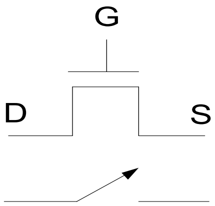
Layers

- To tell equipment where to dope materials, layers were introduced into software tools to give them order.
  - Much of these tools are software categorized into an area called Electronic Design Automation software and can include the Printed Circuit Board area, as well.
- Each color is usually represented by a different layer and possibly number or keyword.
  - Some colors may also have a "stipple" pattern as well (i.e., shading using dots).
  - Historically, some colors have sentimental feelings for people and, thus, are passed down in terms of the representation (e.g, metal1 or 1st layer metal is typically blue) and other colors are random by tool.
- Multiple layers can possibly overlap with each other, but it is the job of the technology file to indicate where there are errors.
  - Sometimes this is called Design Rule Checking (DRC).
  - Most layers have spacing and length rules.
- Certain layers may be grouped together to create a structure, such as a transistor.

Simple Model of a MOSFET

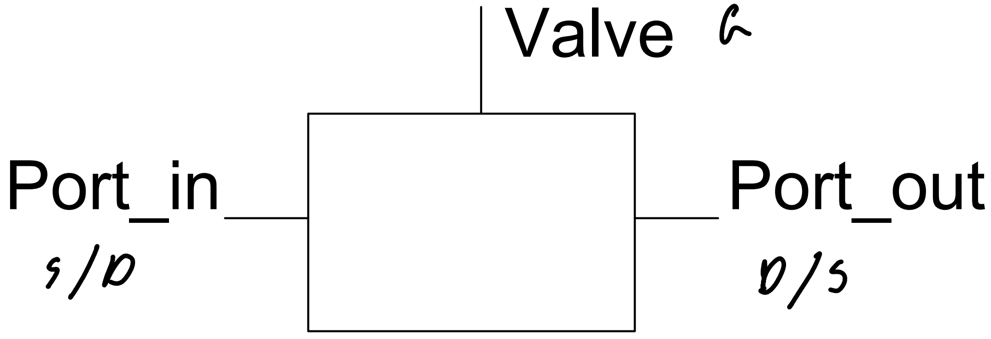
Three Terminal Device





- Source or Drain : depends entirely on the bias.
  - Two ends of the conductive path.
- Gate
  - Controls conductive path.
- Operation
  - Conducts when gate is high
  - Open circuit when gate is low
  - Reality Check: passes zero great, but not ones.

Switch Networks



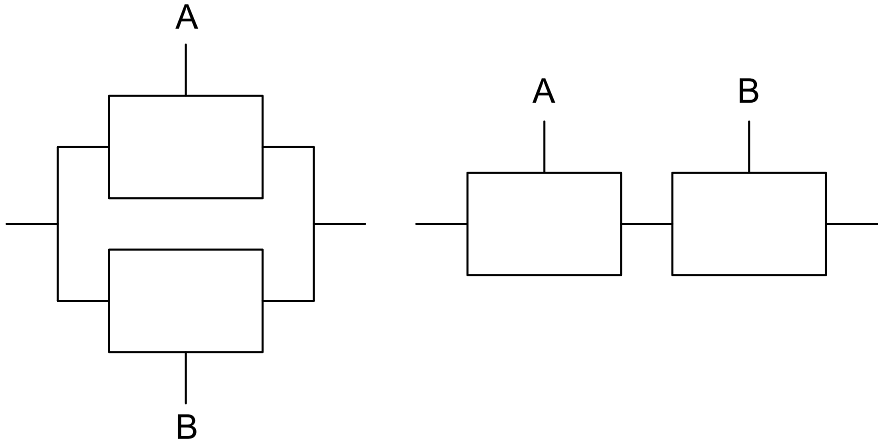
Abstract Switch.

- Since transistors can be modeled as simple switches.
  - Utilize an abstract switch where top is a valve which allows the ports to connect
  - Define a function of a switch network as the input conditions that connect the two terminals of the network *port_out = port_in valve*
- Structure of switch networks defines its logic function:
  - OR functions are constructed by parallel switches.
  - AND functions are constructed by series switches.

Do NMOS first!

The Switch Networks

- The function of a switch is true when the input terminals of the network are connected together.
  - For parallel switches: if either function is on, the function is an OR
  - For series switches: if both functions are on, the function is an AND



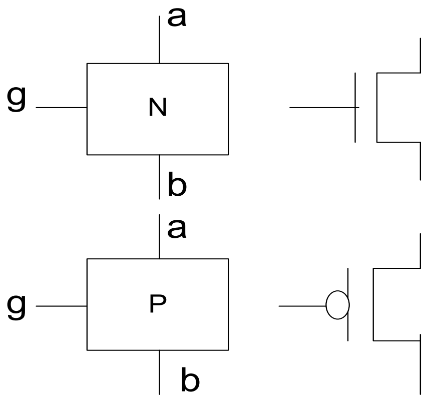
A+B =OR gate

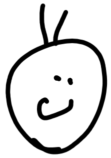
A*B = AND gate

Abstract Switch

The Switch Model

- The simplest model of a MOS transistor is a simple 3-terminal switch!
- n-switch: g=1 closes the switch
- n-switch: passes 1's poorly, good 0's
- p-switch: g=0 closes the switch
- p-switch: passes 0's poorly, good 1's



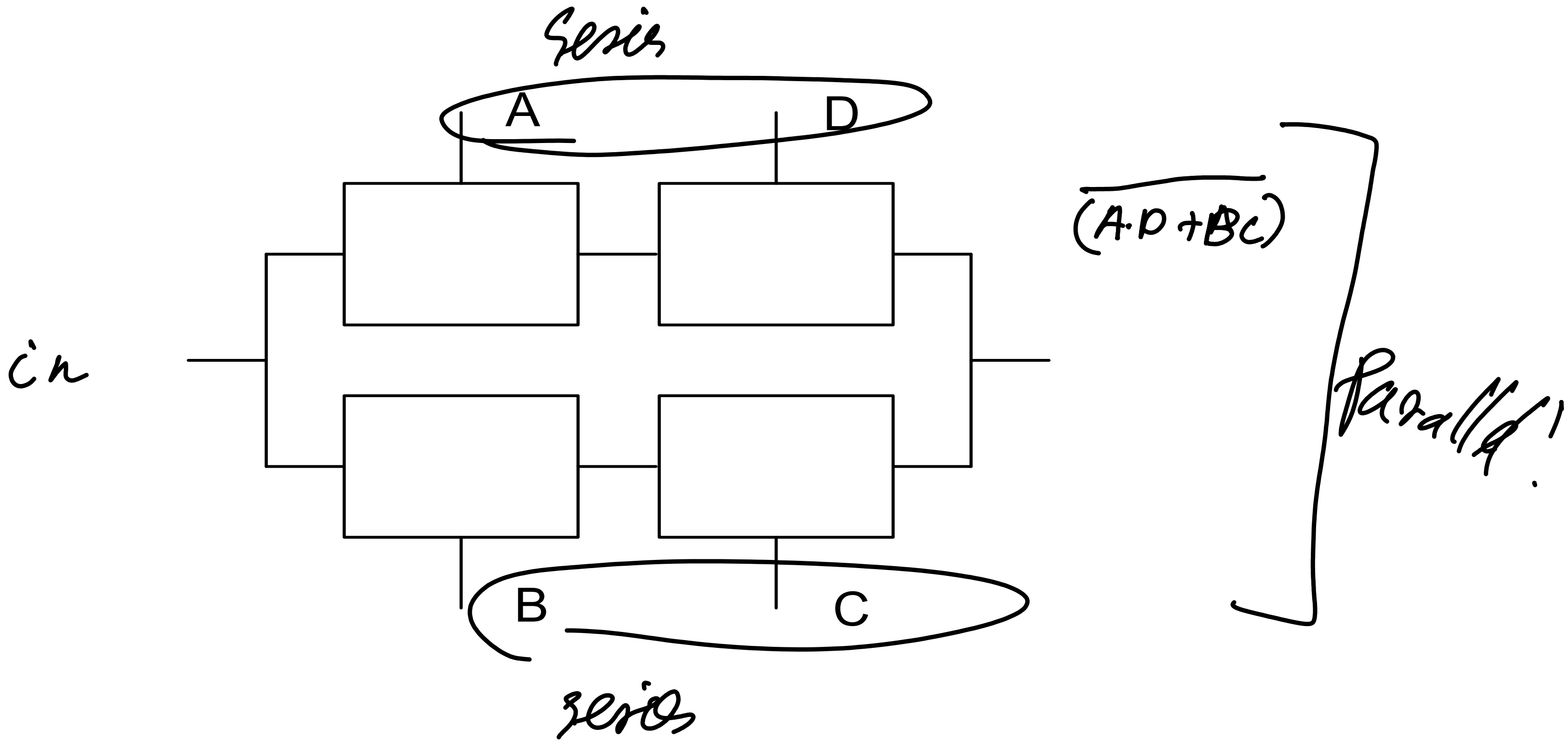


Abstract Switch

Switches and more Switches

- More complex switches are possible.
- Composition rules are simple.
  - Parallel combinations of switches yields an OR of the component switch networks‘ function.
  - Series combinations of switches yields an AND of the component switch networks‘ function.

What function am I?



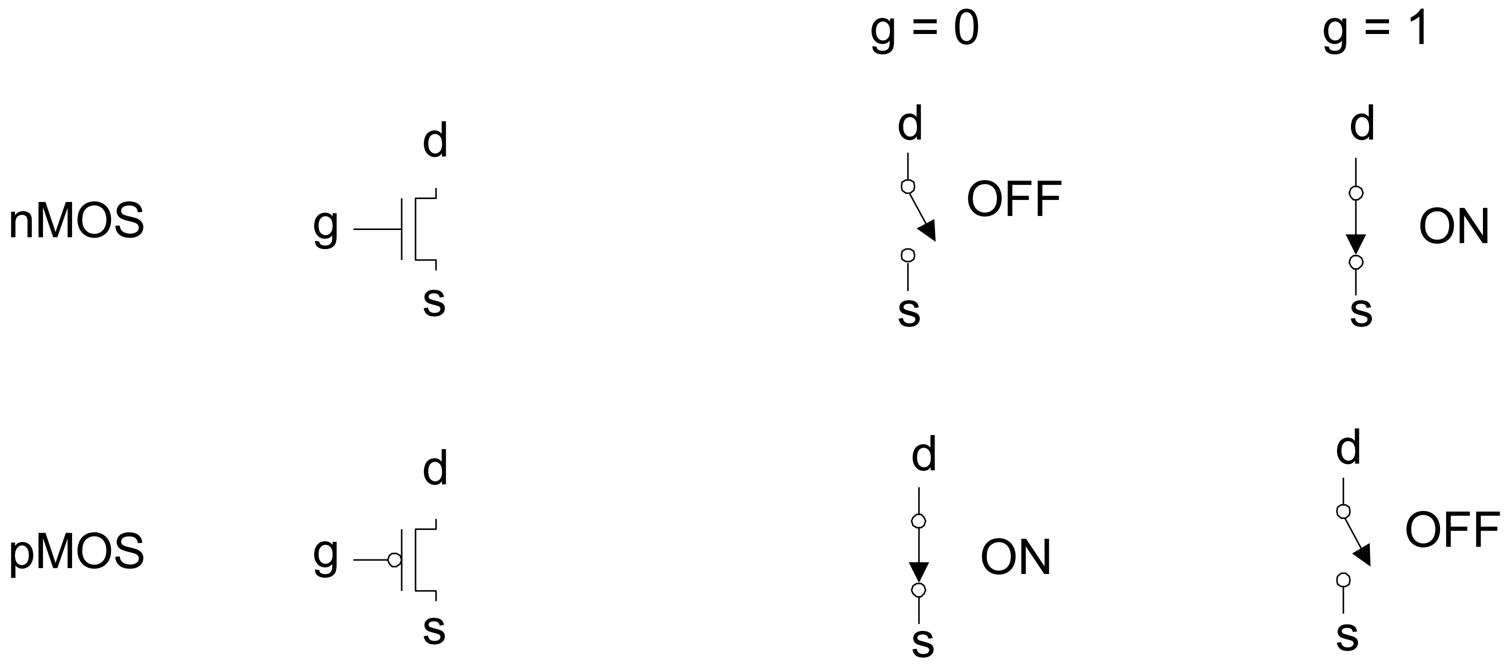
Unknown Function???

Rules for Switches

- Using switch networks, we can build up a simple kind of logic. The basic idea is to use switches to route one or more inputs to the output.
- There are two rules you must follow for switch logic to work.
  - The primary output must always be connected to one of the inputs
    - That is, the OR of all the switch-networks to the output must be 1.
  - Two or more inputs must not be connected together
    - That is, the AND of any two of the switch-networks to the output must be 0 unless they are both constants and have the same value.

Transistors as Switches

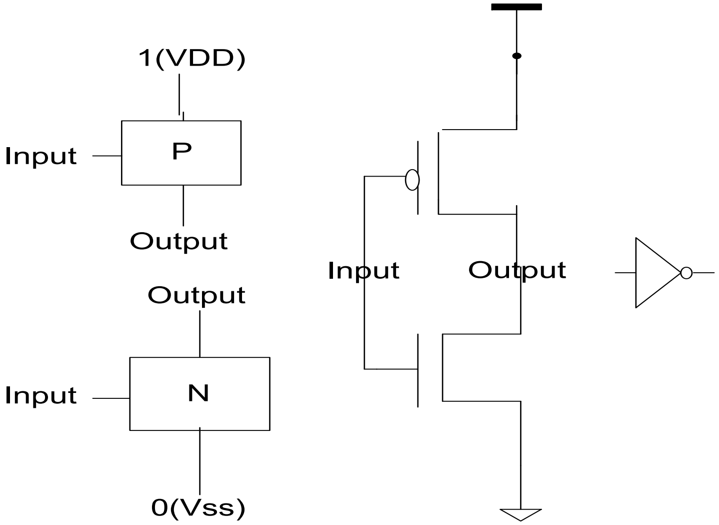
- We can view MOS transistors as electrically controlled switches
- Voltage at gate controls path from source to drain



Logic Inversion

- To achieve any function easily, it would be advantageous to create an inverted signal.
  - That is, the ability to change a signals digital level from a 1 to a 0 and vice-versa.
    - An inverter can be built using two (n and p) MOS transistors.

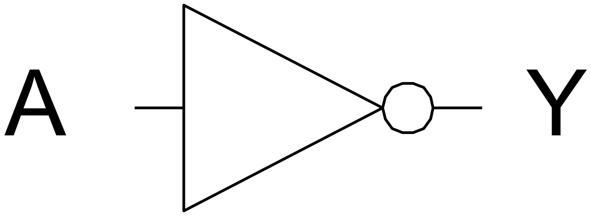
OSU mascot Pistol Pete
photo by Jim Beckel
© 2002, The Daily Oklahoman

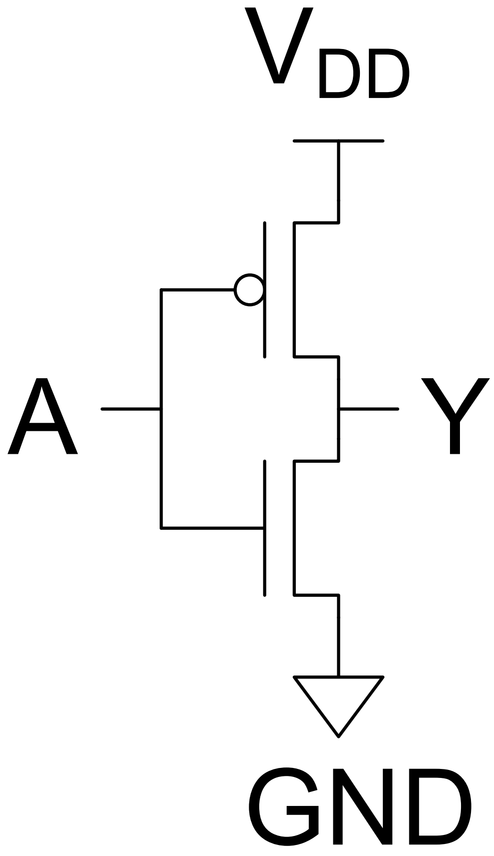


Inverter

CMOS Inverter

A	Y
0	
1	0



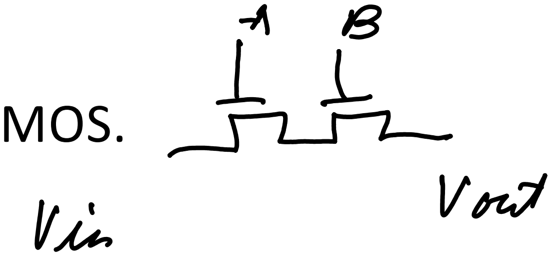


CMOS Logic

Example 1

	Input A	Input B	Gate		
Case	(NMOS)	(NMOS)	1	2	Output
1	0	0	Off	Off	0
2	0	1	Off	On	0
3	1	0	On	Off	0
4	1	1	On	On	1

Series NMOS.

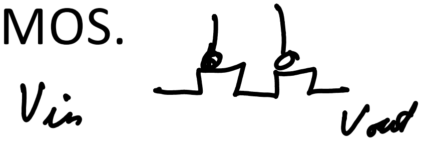


CMOS Logic

Example 2

	Input A	Input B	Gate		
Case	(PMOS)	(PMOS)	1	2	Output
1	0	0	On	On	1
2	0	1	On	Off	0
3	1	0	Off	Off	0
4	1	1	Off	Off	0

Series PMOS.

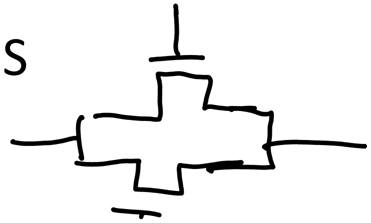


CMOS Logic

Example 3

	Input A	Input B	Gate		
Case	(NMOS)	(NMOS)	1	2	Output
1	0	0	Off	Off	0
2	0	1	Off	On	1
3	1	0	On	Off	1
4	1	1	On	On	1

Parallel NMOS



CMOS Logic

Example 4

	Input A	Input B	Gate		
Case	(PMOS)	(PMOS)	1	2	Output
1	0	0	On	On	1
2	0	1	On	Off	1
3	1	0	Off	On	1
4	1	1	Off	Off	0

Parallel PMOS.

Logic Gates

The general structure of a complex logic gate can be created by the following steps:

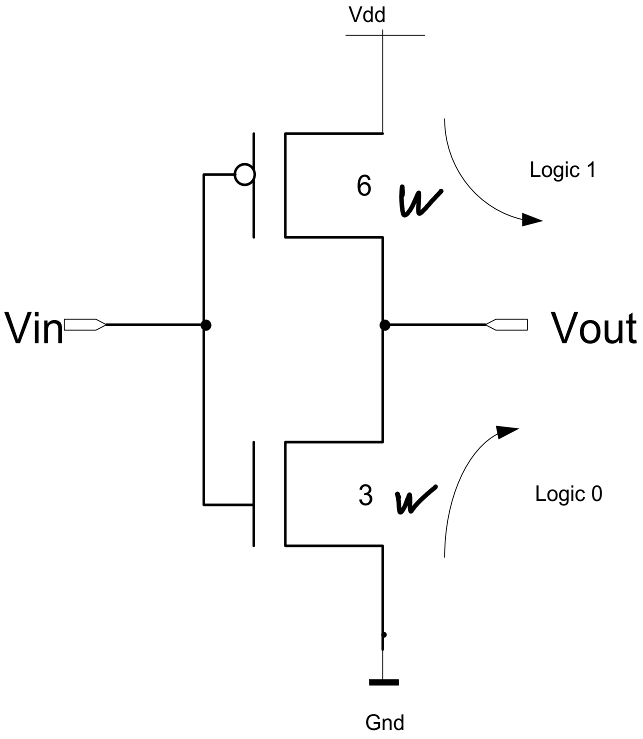
- Provide a complementary pair (nMOS and pMOS with a common gate) for each input.
- Replace the nMOS with an array of nMOSs that connect the output to ground (VSS).
- Replace the pMOS with an array of pMOS gates that connect the output to VDD.
  - That is, the ability to change a signals digital level from a 1 to a 0 and vice-versa.
    - An inverter can be built using two (n and p) MOS transistors.

Operation of the CMOS Inverter Circuit

Perfect load

- Voltage Vin controls the conduction modes of both transistors.
- When Vin = VDD, the nMOS is ON and transmits the ground (Logic 0).
- When Vin = VSS, the pMOS is ON and transmits the power supply voltage (VDD) to the output (Logic 1).
- The search for the perfect load is over!!!

Operation of the CMOS Inverter Circuit

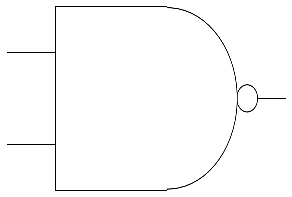


$$\sum_{k=1}^{n} V_k = 0$$

The CMOS NAND Gate

Input A	Input B	Output
0	0	1
0	1	1
1	0	1
1	1	0

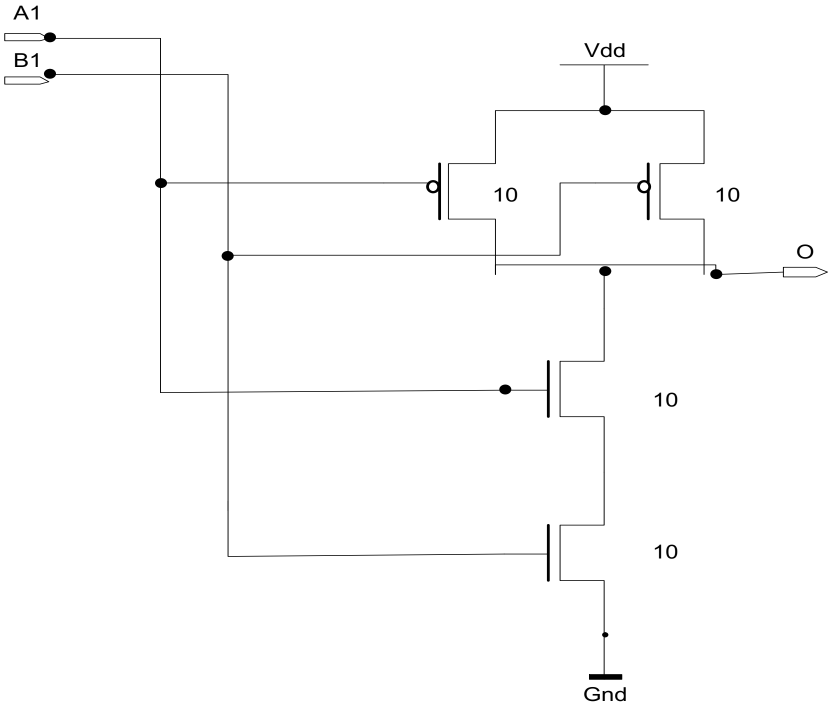
Table 5: Truth Table for Logic NAND.



The CMOS NAND Gate

- Want to have complementary pair that act in tandem.
- When one is on, you want the other off
- Connect two N-transistors in series from output to VSS
- Connect two P-transistors in parallel from output to VDD

The CMOS NAND Gate



NAND Gate

The CMOS NAND Gate

Input A1	Input B1	Output
0 V	0 V	VDD
0 V	5 V	VDD
5 V	0 V	VDD
5 V	5 V	VSS

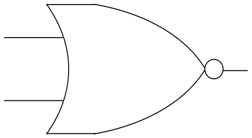
Parallel PMOS

$$f = (\overline{A1} + \overline{B1}) \cdot 1 + (A1 \cdot B1) \cdot 0$$
$$f = (\overline{A1} + \overline{B1})$$
$$f = \overline{(A1 \cdot B1)}$$

The CMOS NOR Gate

Input A	Input B	Output
0	0	1
0	1	1
1	0	1
1	1	0

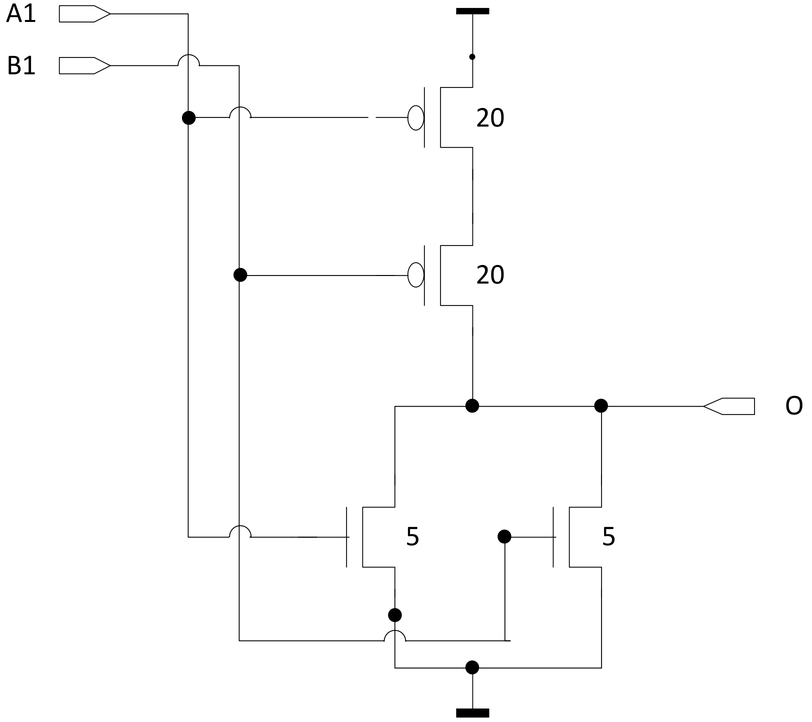
Truth Table for Logic Nor.



The CMOS NOR Gate

- Want to have complementary pair that act in tandem.
- When one is on, you want the other off
- Connect two N-transistors in parallel from output to VSS
- Connect two P-transistors in series from output to VDD

The CMOS NAND Gate



NOR Gate

The CMOS NAND Gate

Input A1	Input B1	Output
0 V	0 V	VDD
0 V	5 V	VSS
5 V	0 V	VSS
5 V	5 V	VSS

Parallel NMOS

$$f = (\overline{A1} \cdot \overline{B1}) \cdot 1 + (A1 \cdot B1) \cdot 0$$
$$f = (\overline{A1} \cdot \overline{B1})$$
$$f = \overline{(A1 + B1)}$$

The Static CMOS Gate

- As you can see each gate will have complementary pair.
- In addition, you can see they always act in tandem.
- Since the top part of the circuit (the PMOS) act by pulling the logic level up they are called *Pullup devices* which is abbreviated as **PUN**

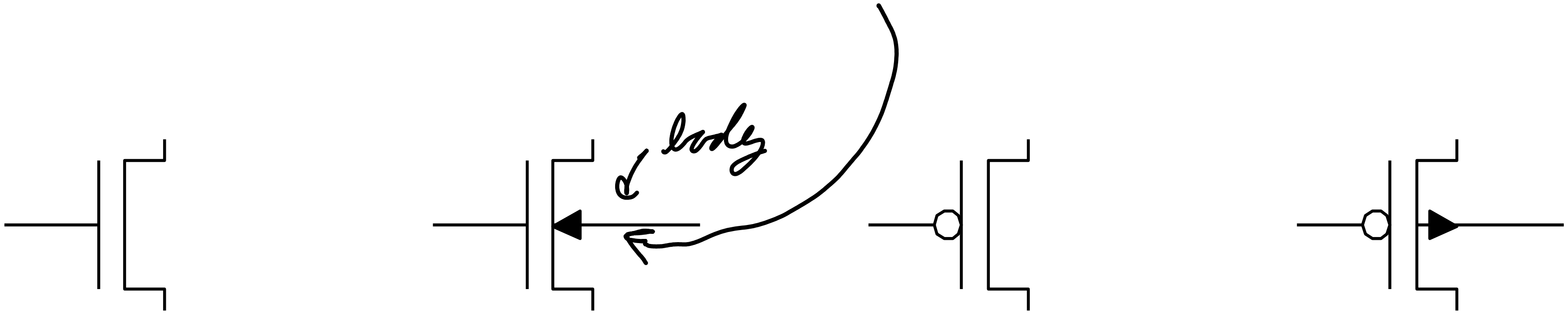
Pull up network

- Since the bottom part of the circuit (the NMOS) act by pulling the logic level down they are called *Pulldown devices* which is abbreviated as **PDN**

Pull down network

Symbols

- So far, we have treated transistors as ideal switches
- An ON transistor passes a finite amount of current
  - Depends on terminal voltages
  - Derive current-voltage (I-V) relationships
- Transistor gate, source, drain all have capacitance
  - Capacitance and current determine speed

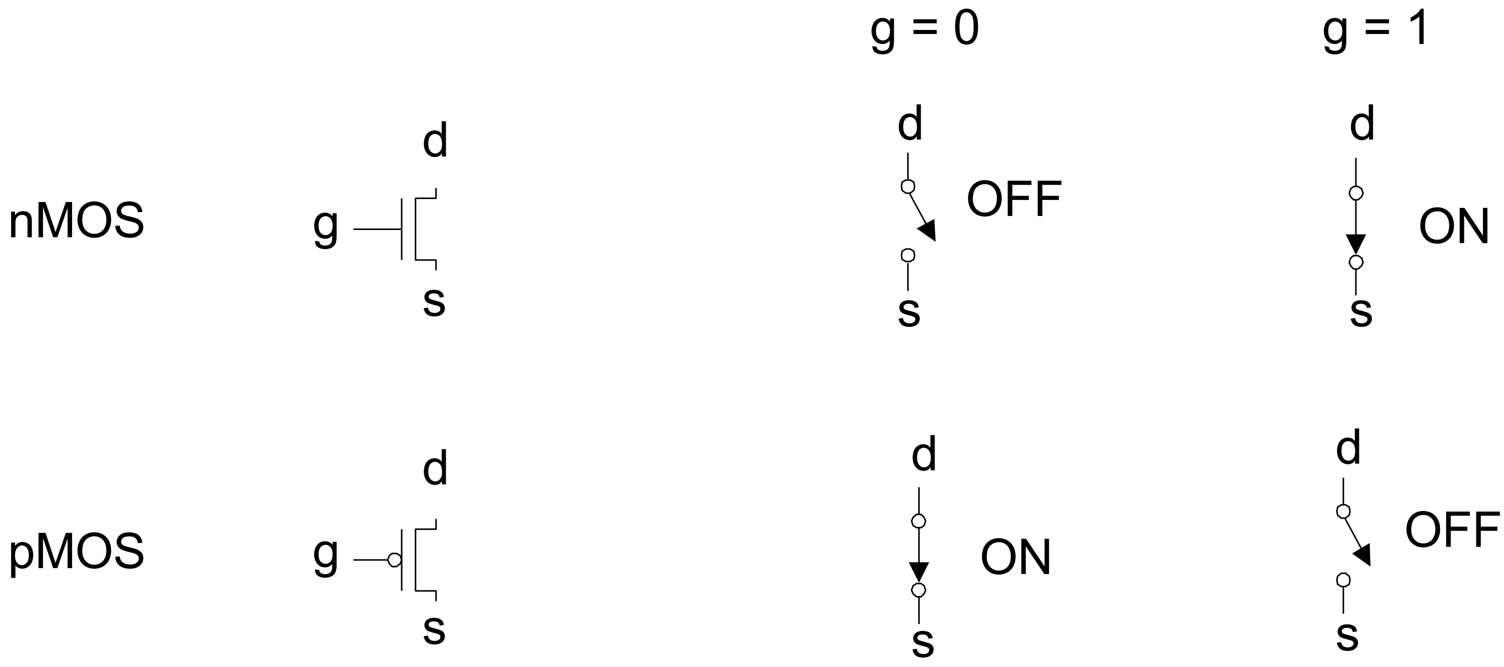


point towards n-type semiconductor

body

Transistors as Switches

- We can view MOS transistors as electrically controlled switches
- Voltage at gate controls path from source to drain



Summary

- CMOS gates are easy to create for most inverting-type logic.
- They also produce a very ”clean” VTC which is great for digital systems.
- Resiliency is also key for CMOS as they are great at switching for systems.
- There is an equal number of pMOS and nMOS transistors in a system for most gates.
- Most CMOS gates tend to use around 4 as the maximum number of inputs to a digital gate.
  - This is changing for different technologies and lower feature sizes.