

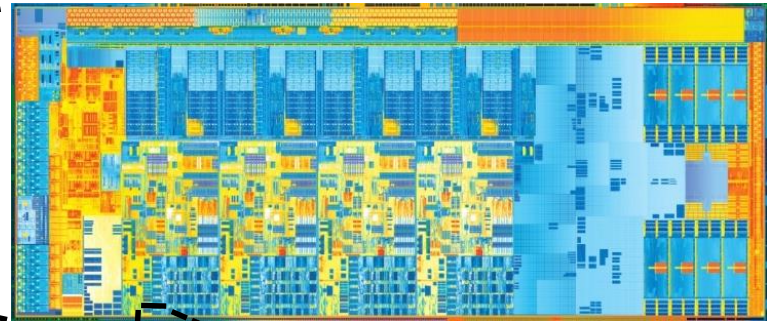
# CMOS Technology

# A Deep Dive Into a Chip

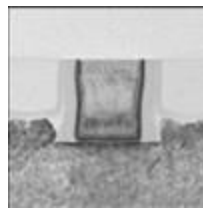
Packaged chip



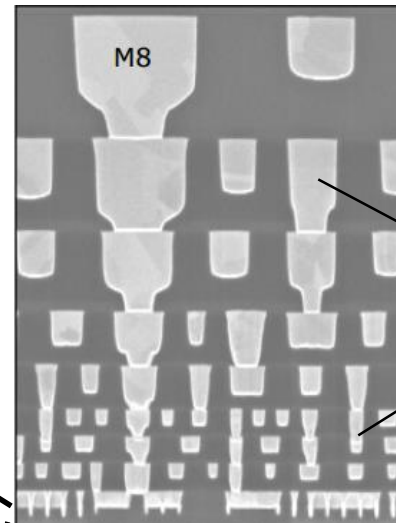
Silicon die (100-400mm<sup>2</sup>)



Transistor (FET)



32 nm



Die cross-section

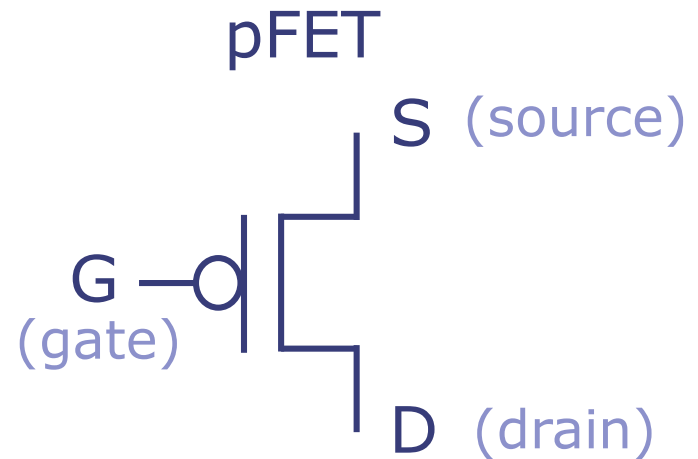
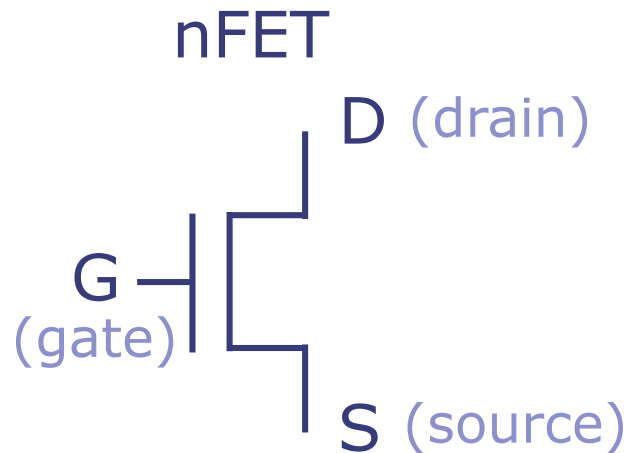
6-15 metal layers (wires)

Source: Intel

# Field-Effect Transistors (FETs)

---

- Nearly all digital systems are built using field-effect transistors, which are **voltage-controlled switches**
- FETs come in two varieties: nFET and pFET



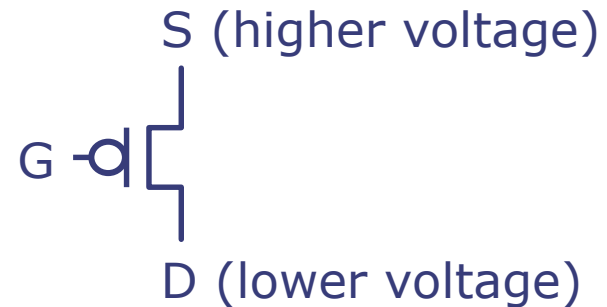
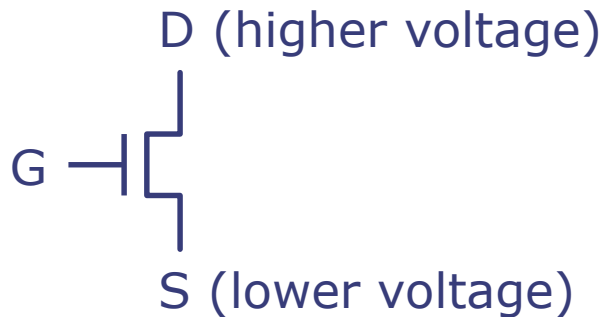
A high voltage at gate ( $G=1$ ) creates a conducting path between source and drain

A low voltage at gate ( $G=0$ ) creates a conducting path between source and drain

# Labeling Source and Drain

---

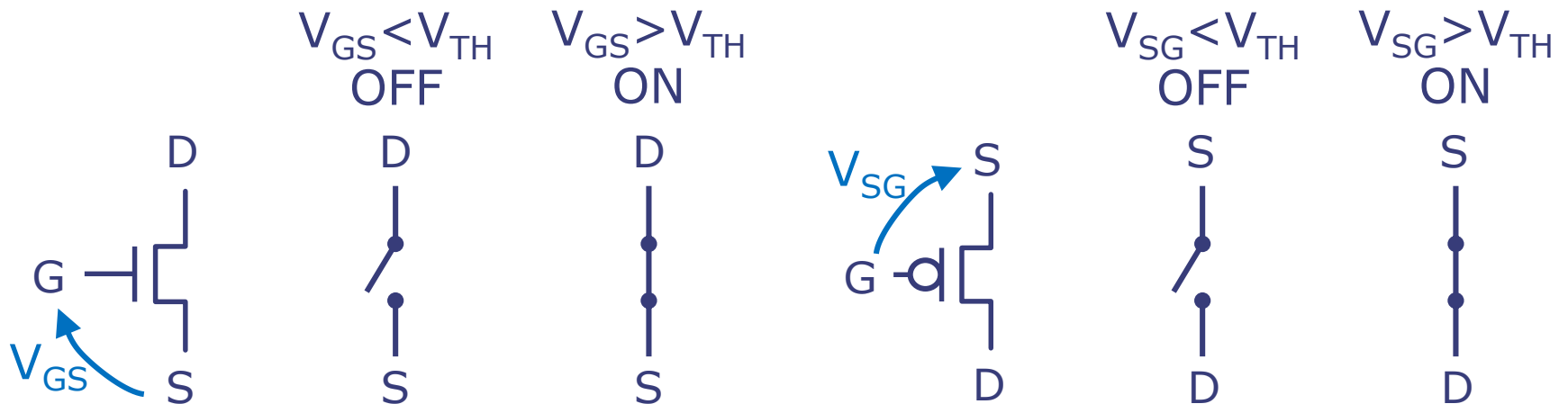
- There is no physical difference between source and drain, called the **diffusion terminals**
- By convention, we label diffusion terminals as source or drain depending on their voltages:
  - On nFETs, source = diffusion terminal at lower voltage
  - On pFETs, source = diffusion terminal at higher voltage



- This convention lets us define the behavior of FETs using the voltage between gate and source

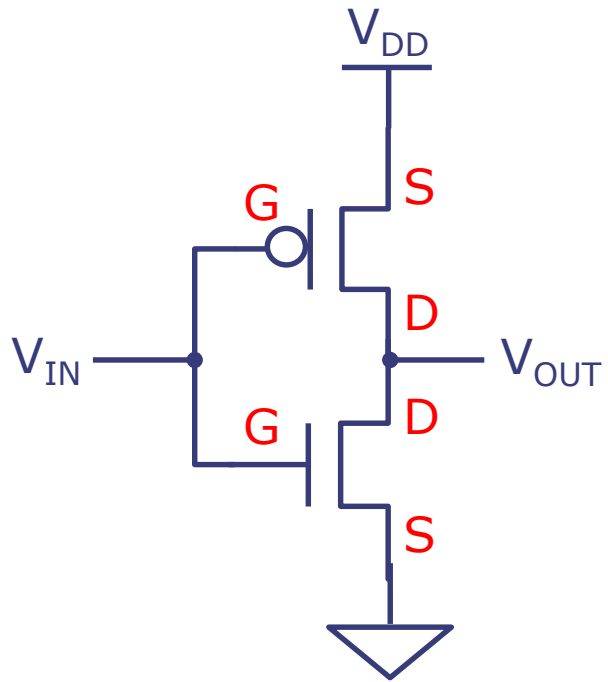
# FET Switching Model

- FETs have a **threshold voltage**  $V_{TH}$
- nFET is ON if the voltage between **gate** and **source**  $V_{GS}$  exceeds  $V_{TH}$ , OFF otherwise
- pFET is ON if the voltage between **source** and **gate**  $V_{SG}$  exceeds  $V_{TH}$ , OFF otherwise

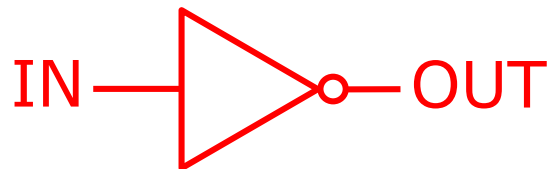


- This is a very simplified model, but it is sufficient to build logic gates

# What Does This Circuit Compute?



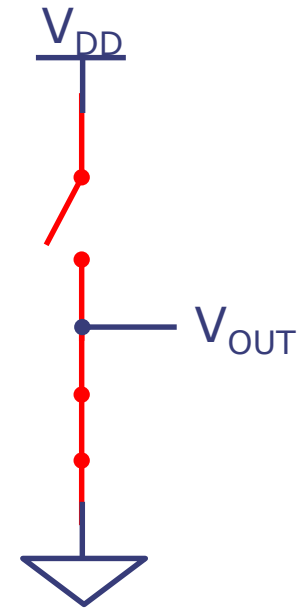
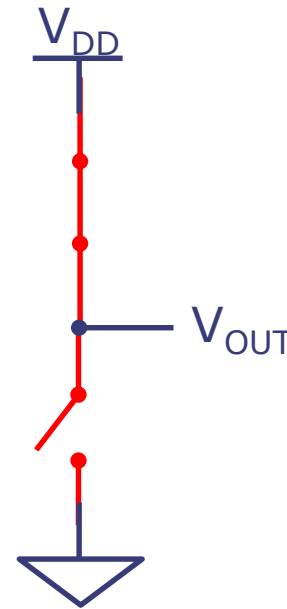
CMOS inverter



Assume  $V_{TH} < V_{DD}/2$

$V_{IN} < V_{TH}$

$V_{IN} > V_{DD} - V_{TH}$



IN	OUT
0	1
1	0

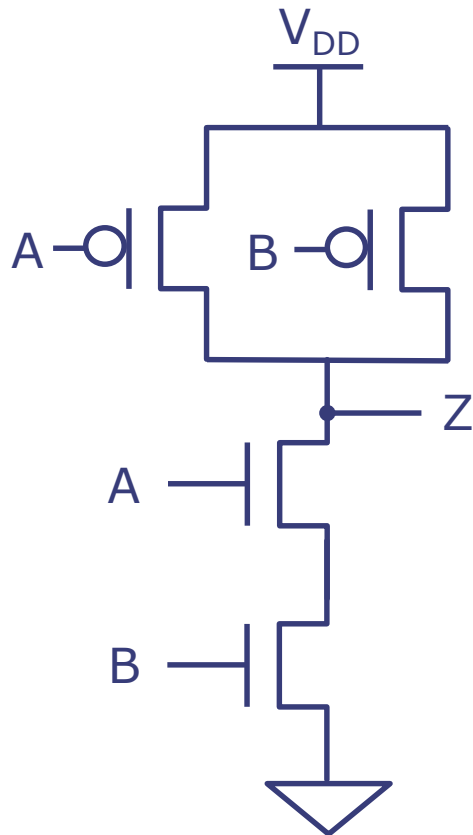
# Note on Terminology

---

- MOSFETs (metal-oxide-semiconductor field-effect transistors) are the most common type of FET
- nFET and pFET are sometimes abbreviated as nMOS and pMOS
- CMOS stands for complementary MOS

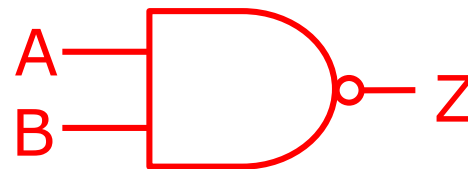
# What Does This Circuit Compute?

---



A	B	Z
0	0	1
0	1	1
1	0	1
1	1	0

CMOS NAND gate

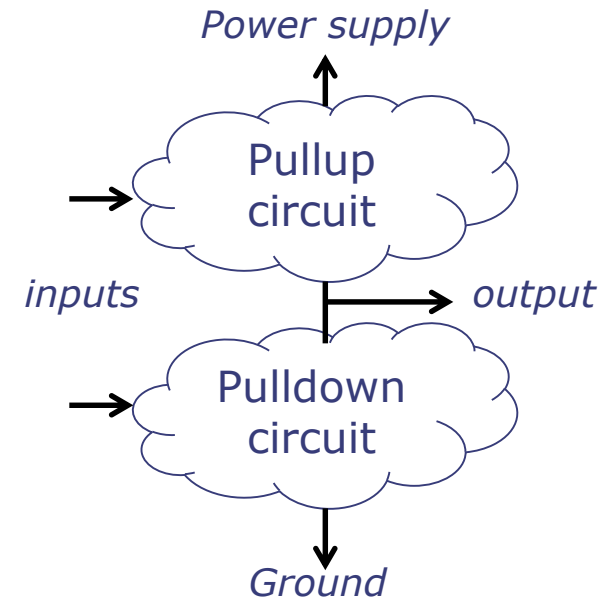




# CMOS Logic

- CMOS gates have **complementary** pullup and pulldown networks, i.e., the pullup is on where the pulldown is off and viceversa

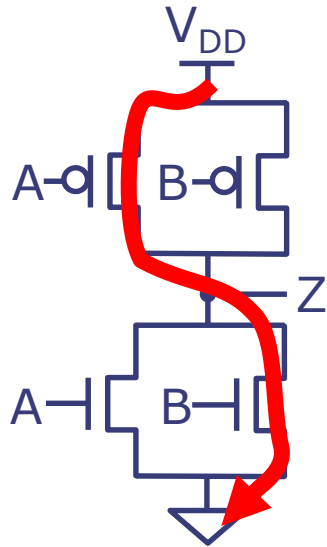
pullup	pulldown	F(inputs)
on	off	driven "1"
off	on	driven "0"
<b>on</b>	<b>on</b>	<b>driven "X"</b>
<b>off</b>	<b>off</b>	<b>no connection</b>



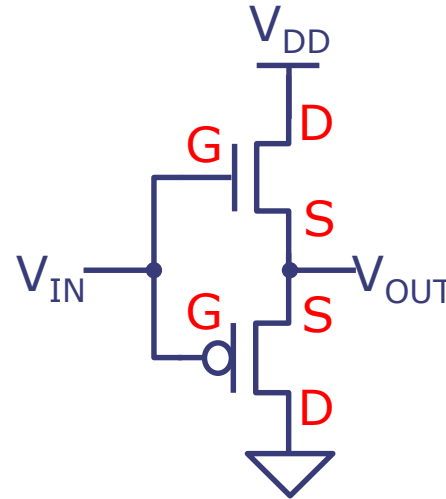
- CMOS uses pFETs to implement the pullup network and nFETs to implement the pulldown network

# Some Questionable Gates

- What can go wrong with the following gates?



$A=0$   $B=1$  or  $A=1$   $B=0$   
connect supply and ground

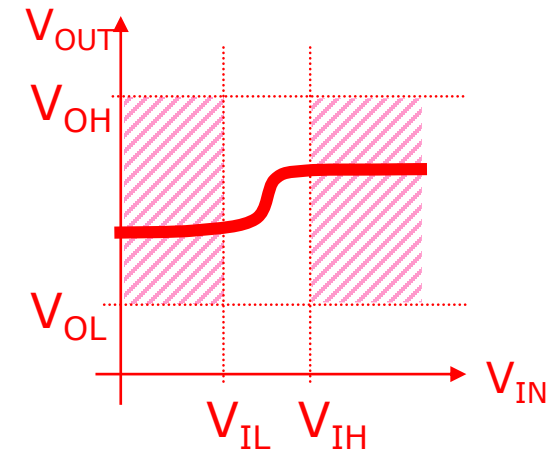


pFET doesn't pull down

$V_{OUT}$  below  $V_{TH}$

nFET doesn't pull up

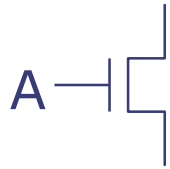
$V_{OUT}$  above  $V_{DD} - V_{TH}$



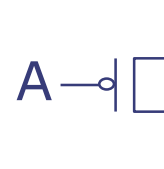
- CMOS Rule #1: Complementary pullup and pulldown networks
- CMOS Rule #2: pFETs in pullup, nFETs in pulldown

# CMOS Complements

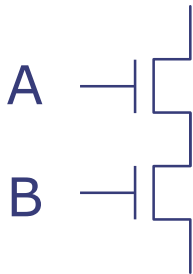
---



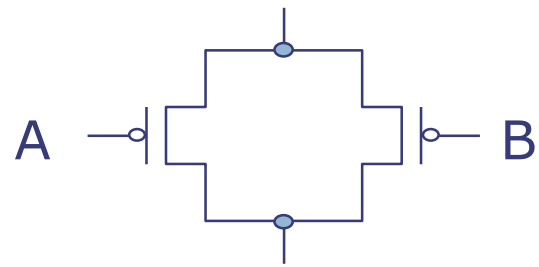
conducts when A is high



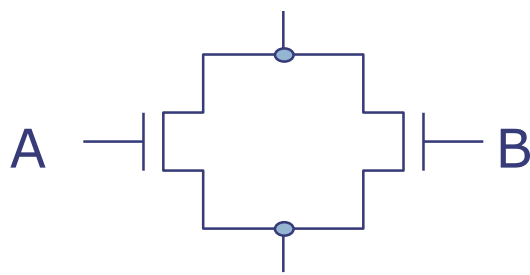
conducts when A is low:  $\bar{A}$



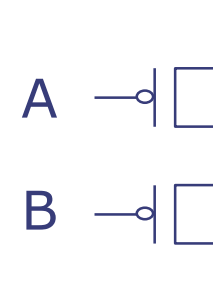
conducts when A is high  
and B is high:  $A \cdot B$



conducts when A is low  
or B is low:  $\bar{A} + \bar{B} = \overline{A \cdot B}$



conducts when A is high  
or B is high:  $A + B$



conducts when A is low  
and B is low:  $\bar{A} \cdot \bar{B} = \overline{A + B}$

# General CMOS Gate Recipe

Step 1. Derive the pullup network that does what you want, e.g.,

$$F = \overline{A} + \overline{B} \times \overline{C}$$

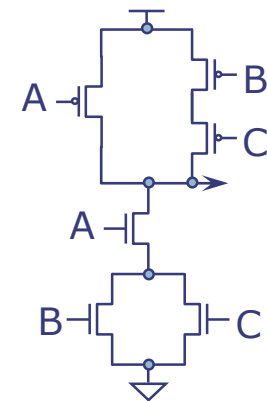
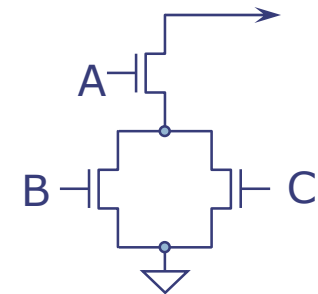
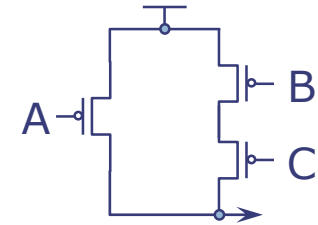
(Determine what combination of inputs generates a high output)

Step 2. Derive complementary pulldown network: replace pFETs with nFETs, series subnets with parallel subnets, and parallel subnets with series subnets

Step 3. Combine pFET pullup network from Step 1 with nFET pulldown network from Step 2 to form the CMOS gate.

*Can CMOS gates implement arbitrary functions?*

**No**



# CMOS Gates are Inverting

---

- In a CMOS gate, rising inputs ( $0 \rightarrow 1$ ) lead to falling outputs ( $1 \rightarrow 0$ ) and viceversa
- On a rising input,
  - nFETs go OFF  $\rightarrow$  ON, so pulldown may connect output to ground
  - pFETs go ON  $\rightarrow$  OFF, so pullup may disconnect output from  $V_{DD}$
  - Output either stays the same or falls
- Corollary: Cannot build non-inverting logic using a single CMOS gate
  - Example: AND

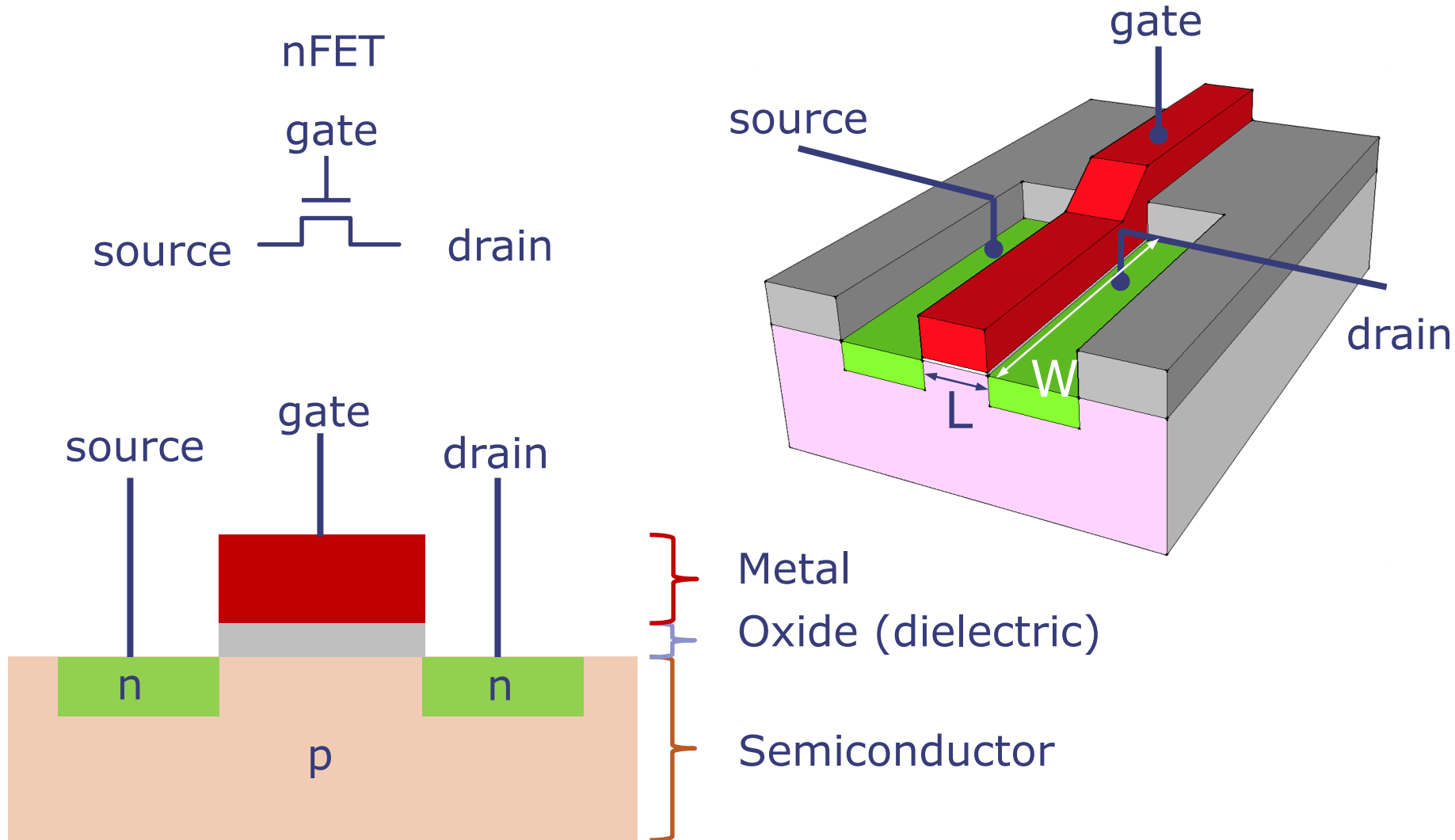
A	B	A·B
0	0	0
0	1	0
1	0	0
1	1	1

rising input rising output

# Analyzing the Delay, Area, and Power of CMOS Gates

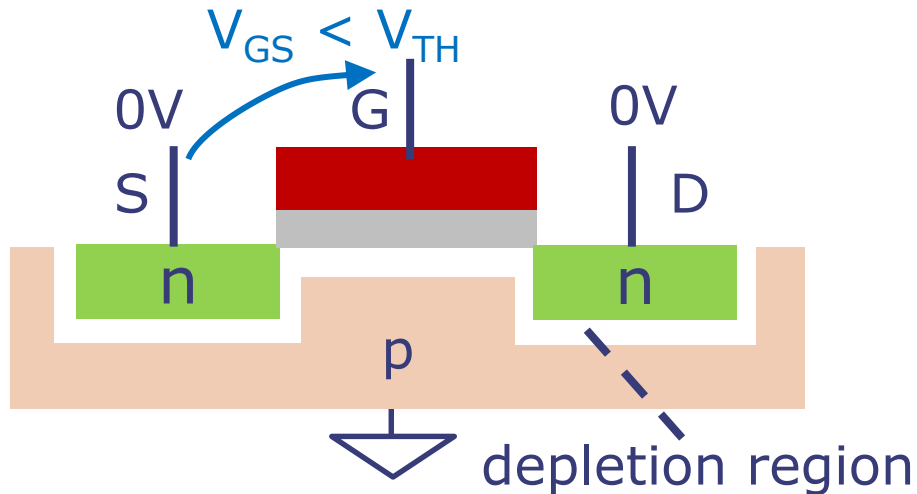
*NOTE: Demystification,  
will not be on the quiz*

# MOSFET Physical Structure

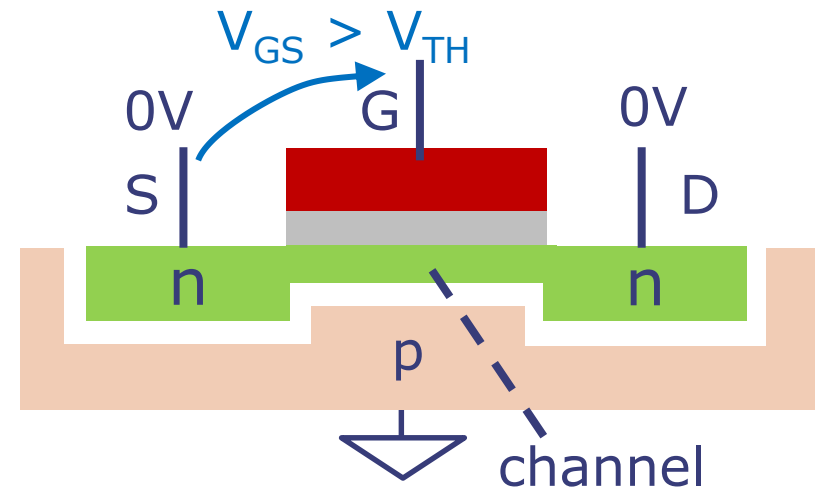


# MOSFET Electrical View

With  $V_{GS} < V_{TH}$ , almost no current flows between source and drain



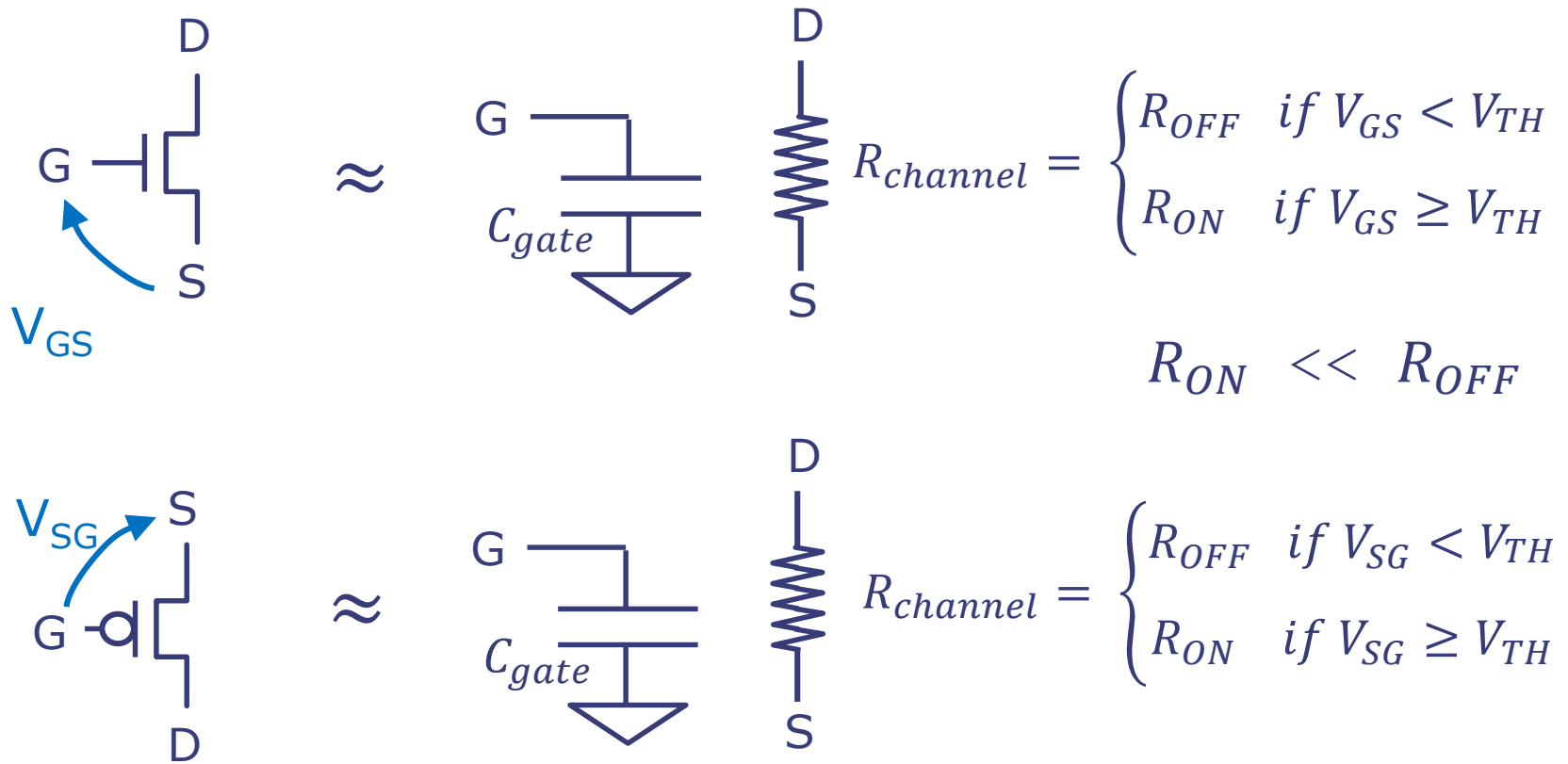
As  $V_{GS}$  reaches  $V_{TH}$ , a channel forms between source and drain



The shape of the channel (and its resistance) also depends on the voltage at the drain. But a low-resistance channel will exist while  $V_{GS} > V_{TH}$



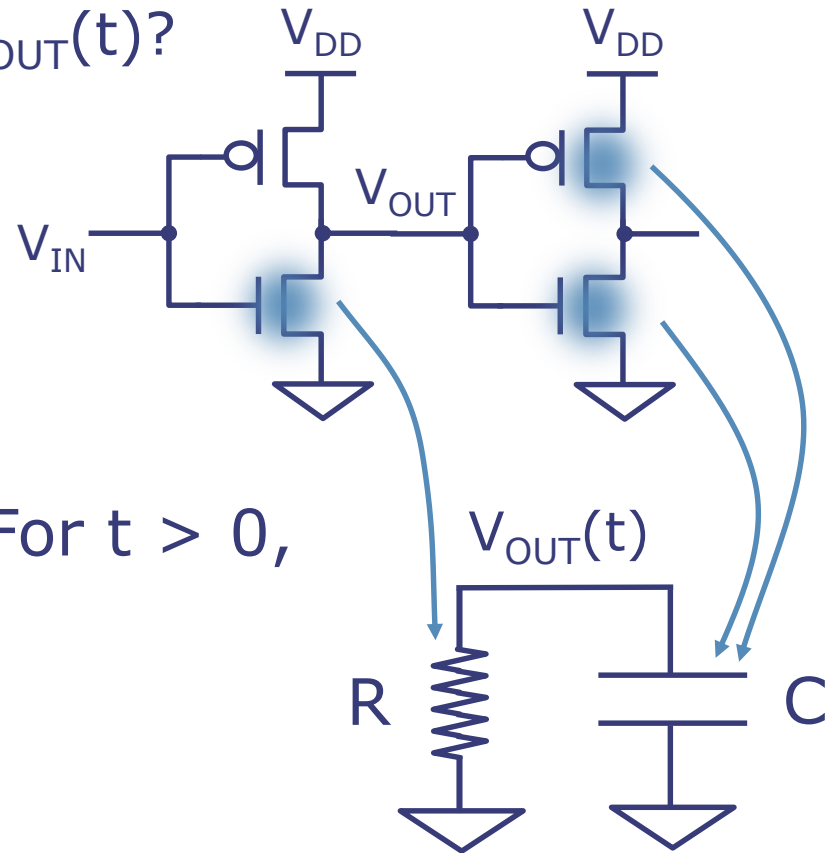
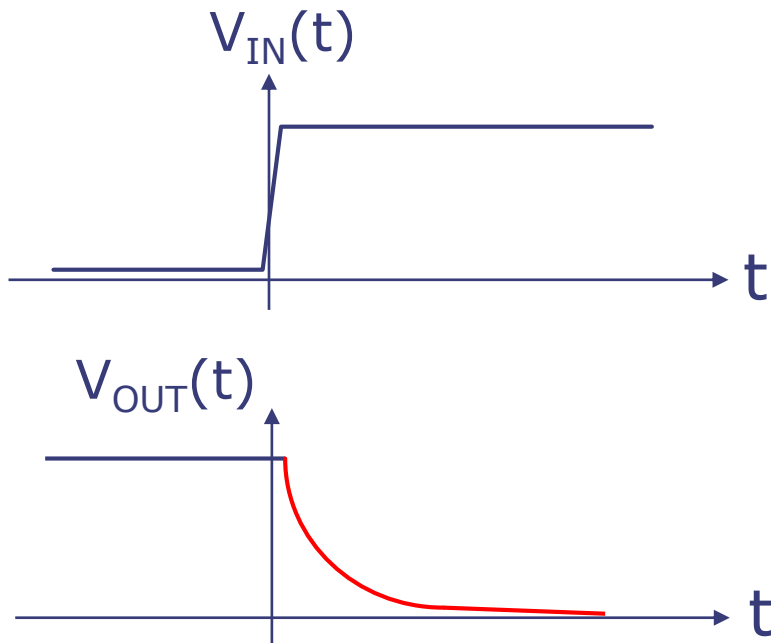
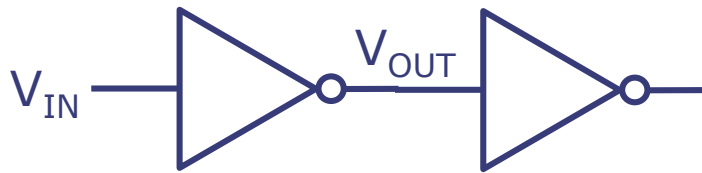
# FET *First-Order* Electrical Model



- Simplest possible model that lets us reason about delay, area, and power. Not very accurate!

# CMOS Gate Delay

Consider the following circuit.  
Given  $V_{IN}(t)$ , can you derive  $V_{OUT}(t)$ ?

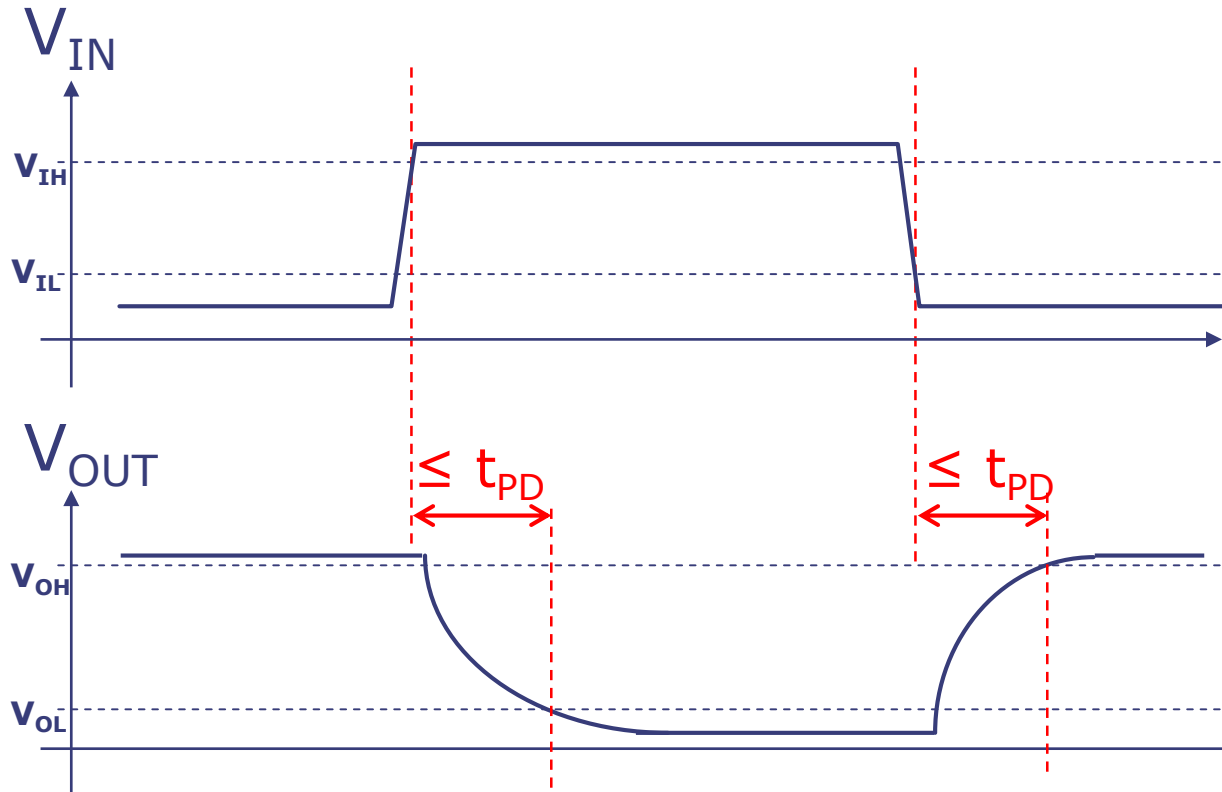


For  $t > 0$ ,

$$V_{OUT}(t) = V_{OUT}(0)e^{-t/RC}$$

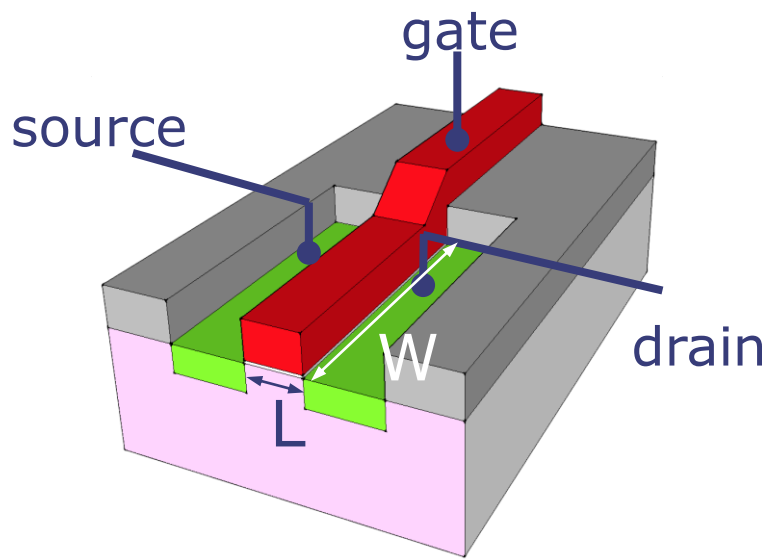
# Propagation Delay

Propagation delay ( $t_{pD}$ ): Upper bound on the delay from **valid inputs** to **valid outputs**.

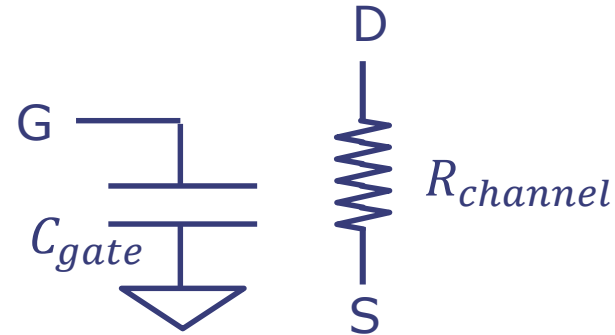


To minimize  $t_{pD}$ , must keep resistances and capacitances low

# MOSFET Sizing



$\approx$



*How do  $C_{gate}$  and  $R_{channel}$  change with  $L$  and  $W$ ?*

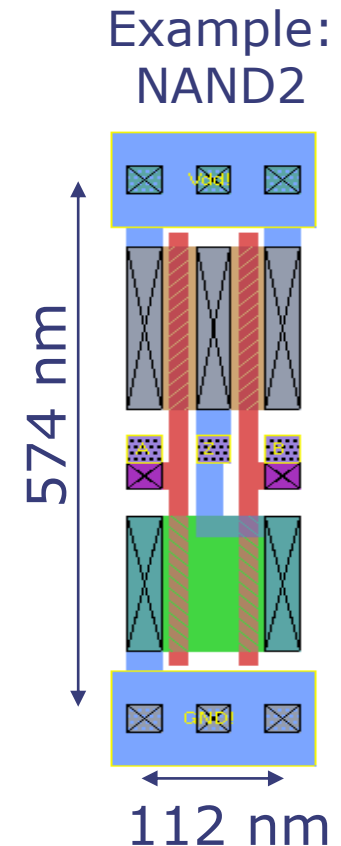
$$C_{gate} \propto L \cdot W$$

$$R_{channel} \propto L/W$$

- CMOS gates use MOSFETs with smallest possible  $L$  and choose  $W$  to set performance
  - Wider FETs drive more current (lower  $R$ ), but their gates are harder to drive (higher  $C$ ) and they take more area

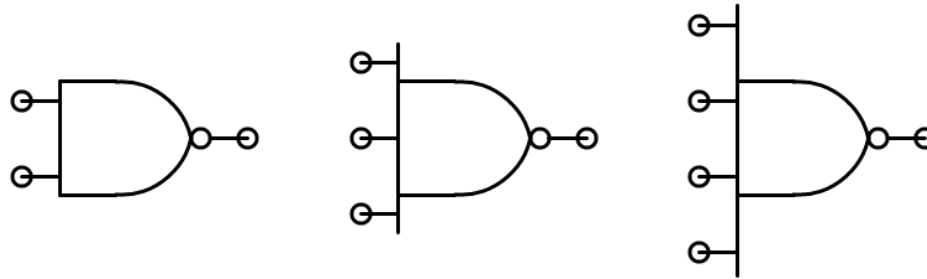
# Standard Cell Libraries

- A standard cell library provides implementations of common gates (NAND, NOR, XOR, etc.) for a specific implementation technology
- Each gate includes
  - Electrical parameters (e.g.,  $R_s$  and  $C_s$ )
  - Physical layout
- Synthesis tools use gates from the standard library instead of sizing and placing individual transistors



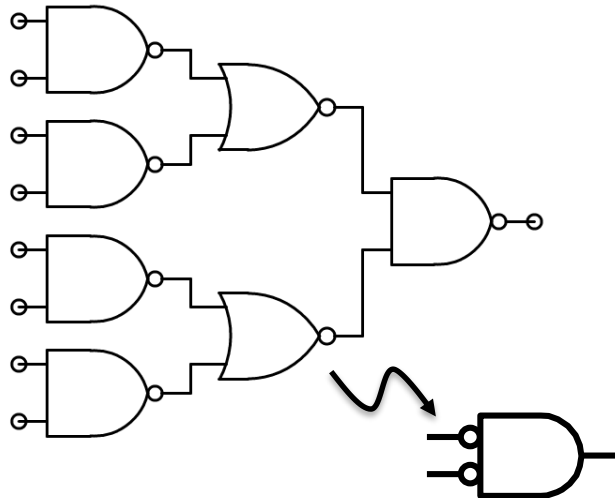
# Wide (High-Fanin) Gates

Most standard cell libraries include 2-, 3- and 4-input devices:



But for a large number of inputs, the series connections of too many MOSFETs can lead to very large effective  $R_{\text{pulldown}}$  or  $R_{\text{pullup}}$ . Instead, use trees of smaller devices...

Example:  
8-input  
NAND



*How does  $t_{pD}$  grow with the number of inputs  $N$ ?*

If we use a single CMOS gate,  $t_{pD} \propto N$

If we use a tree of gates,  $t_{pD} \propto \log(N)$

# CMOS Power Dissipation

---

- Total power dissipation:  $P = P_{dynamic} + P_{static}$
- Dynamic power: Caused by  $0 \leftrightarrow 1$  transitions of nodes in the circuit
  - Charging/discharging each capacitor consumes  $\frac{1}{2}CV_{DD}^2$  energy
  - If on average  $C_S$  capacitance across the chip switches each cycle, and there are  $f_{CLK}$  cycles per second

$$P_{dynamic} = \frac{1}{2} C_S V_{DD}^2 f_{CLK}$$

- Static power: Caused by
  - Subthreshold leakage: Even when the FET is off, a very small current flows from source to drain ( $R_{OFF} < \infty$ )
  - Tunneling current: Gate and channel are separated by a very thin ( $< 1\text{nm}$ ) dielectric, so some electrons tunnel through

$$P_{static} = I_{static} V_{DD}$$

- Static power is typically 10-30% of total power

# Summary

---

- FETs behave as voltage-controlled switches
- CMOS gates:
  - Use complementary pullup and pulldown networks
  - Use pFETs in pullup, nFETs in pulldown network
- CMOS gates are inverting (rising inputs can only cause falling outputs, and viceversa)



# Thank you!

Next lecture:  
Combinational logic and  
introduction to Minispec