

# Introduction to CMOS VLSI Design

## CMOS Basics

Young-Don Bae, Ph.D.

([ceo@donny.co.kr](mailto:ceo@donny.co.kr))

courtesy of David Harris (Harvey Mudd College)

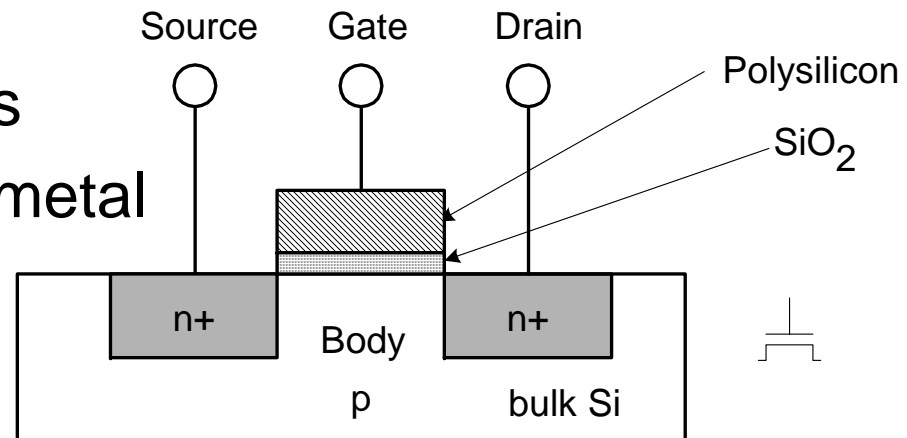


Chungnam National University



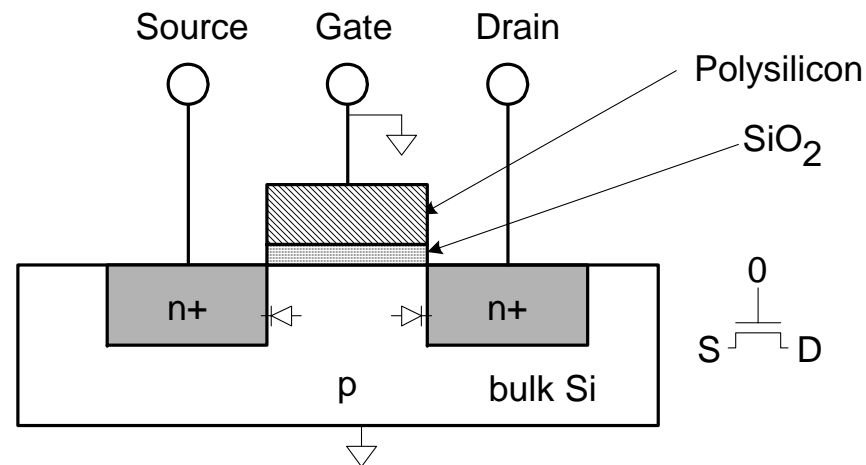
# nMOS Transistor

- ❑ Four terminals: gate, source, drain, body
- ❑ Gate – oxide – body stack looks like a capacitor
  - Gate and body are conductors
  - $\text{SiO}_2$  (oxide) is a very good insulator
  - Called metal – oxide – semiconductor (MOS) capacitor
  - Even though gate is no longer made of metal



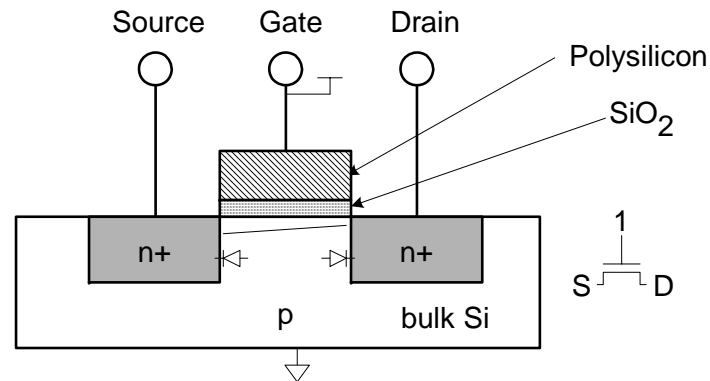
# nMOS Operation

- ❑ Body is usually tied to ground (0 V)
- ❑ When the gate is at a low voltage:
  - P-type body is at low voltage
  - Source-body and drain-body diodes are OFF
  - No current flows, transistor is OFF



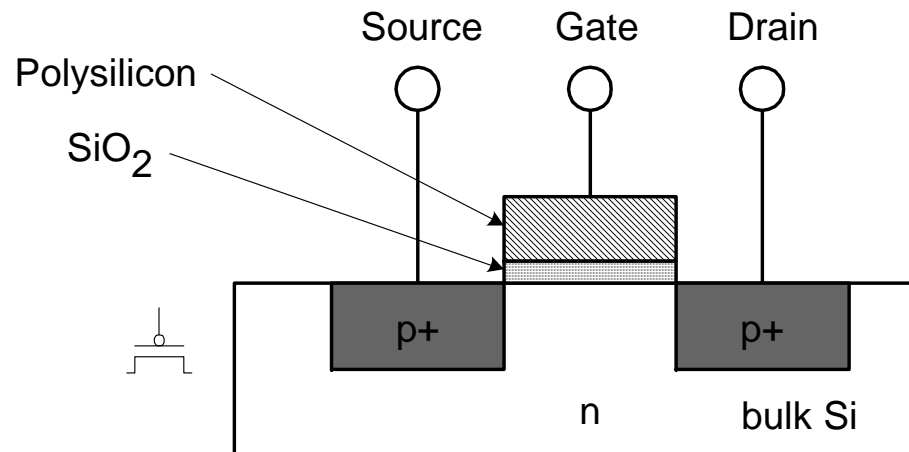
# nMOS Operation Cont.

- When the gate is at a high voltage:
  - Positive charge on gate of MOS capacitor
  - Negative charge attracted to body
  - Inverts a channel under gate to n-type
  - Now current can flow through n-type silicon from source through channel to drain, transistor is ON



# pMOS Transistor

- ❑ Similar, but doping and voltages reversed
  - Body tied to high voltage ( $V_{DD}$ )
  - Gate low: transistor ON
  - Gate high: transistor OFF
  - Bubble indicates inverted behavior

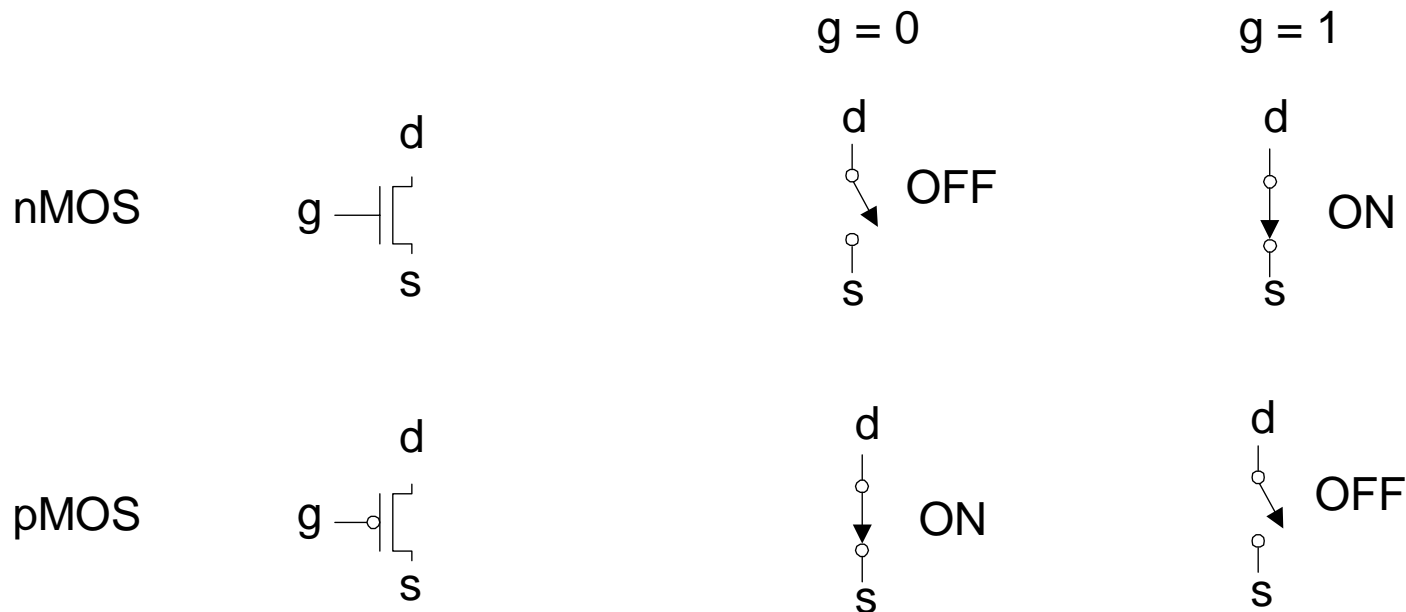


# Power Supply Voltage

- ❑ GND = 0 V
- ❑ In 1980's,  $V_{DD} = 5V$
- ❑  $V_{DD}$  has decreased in modern processes
  - High  $V_{DD}$  would damage modern tiny transistors
  - Lower  $V_{DD}$  saves power
- ❑  $V_{DD} = 3.3, 2.5, 1.8, 1.5, 1.2, 1.0, \dots$

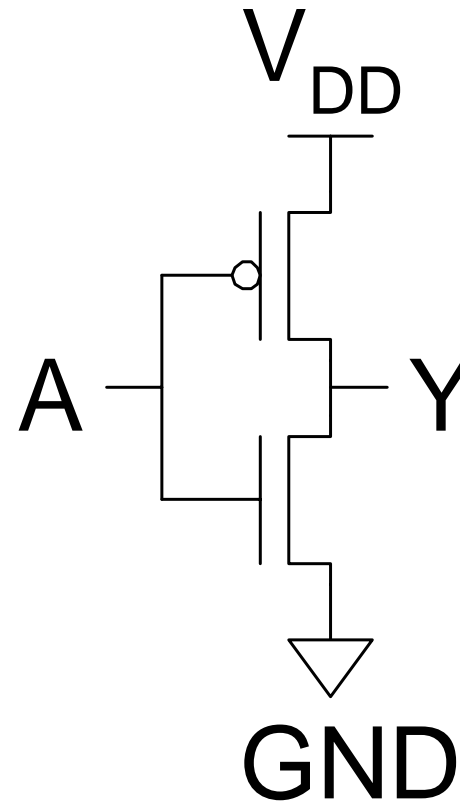
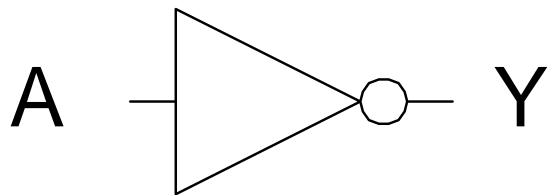
# Transistors as Switches

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



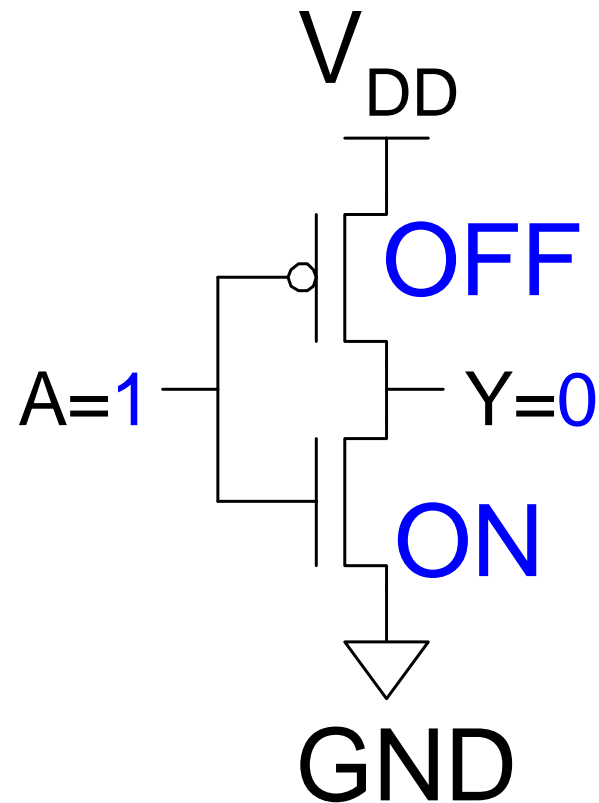
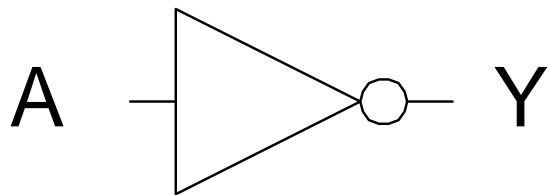
# CMOS Inverter

A	Y
0	
1	



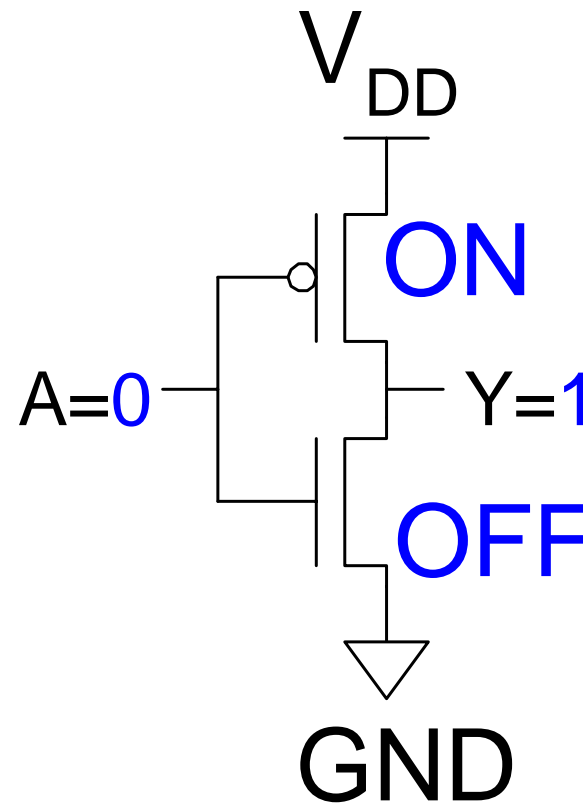
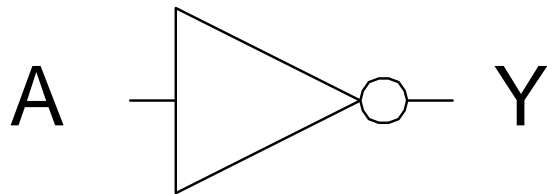
# CMOS Inverter

A	Y
0	
1	0



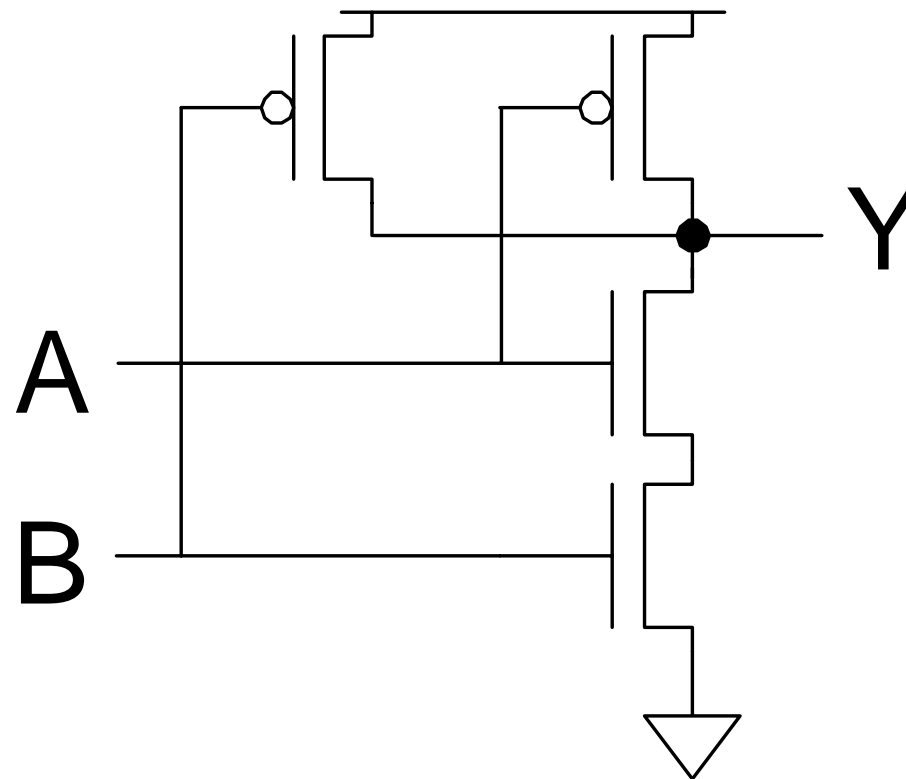
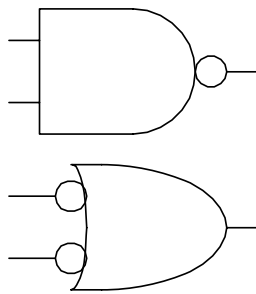
# CMOS Inverter

A	Y
0	1
1	0



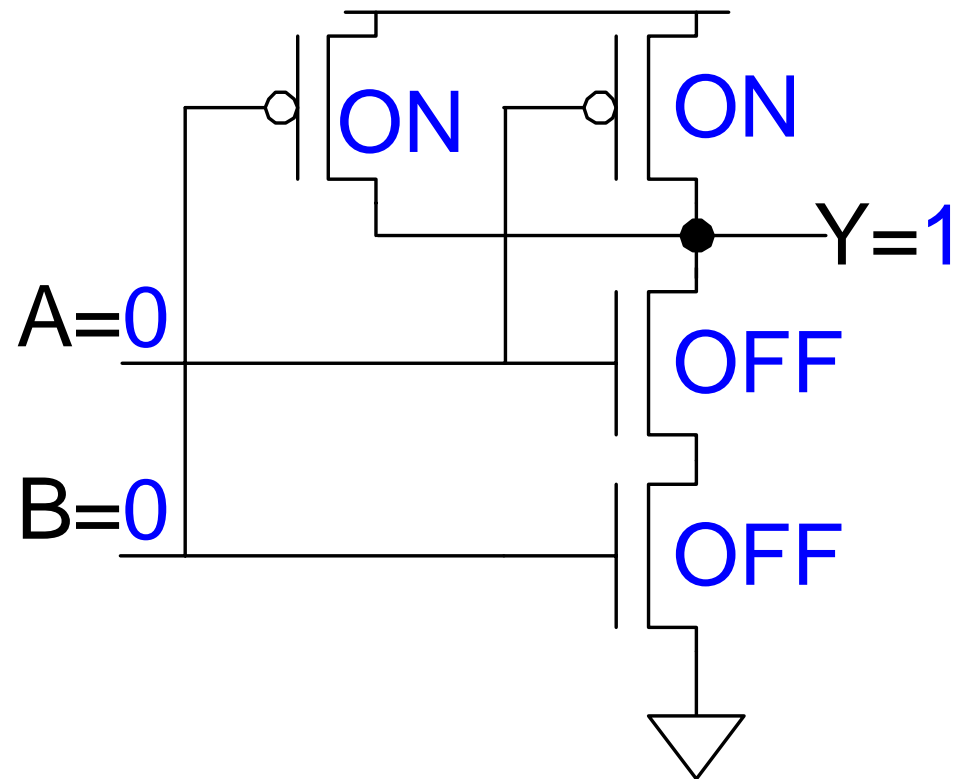
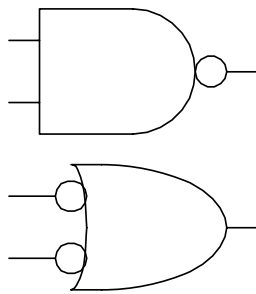
# CMOS NAND Gate

A	B	Y
0	0	
0	1	
1	0	
1	1	



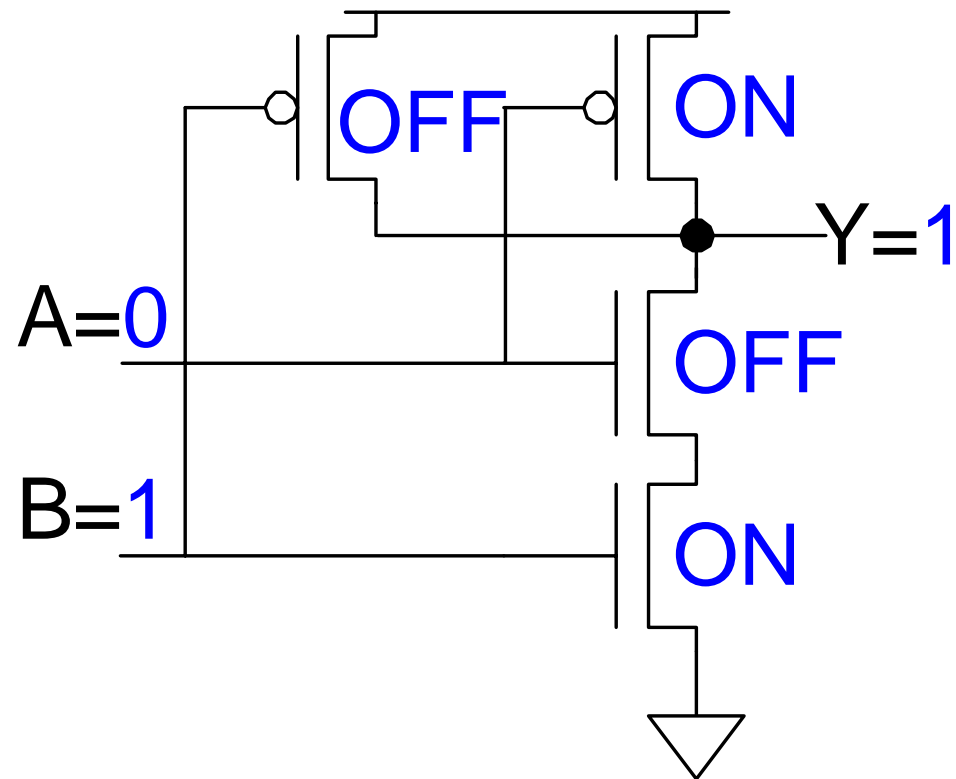
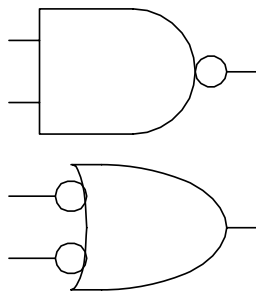
# CMOS NAND Gate

A	B	Y
<b>0</b>	<b>0</b>	<b>1</b>
0	1	
1	0	
1	1	



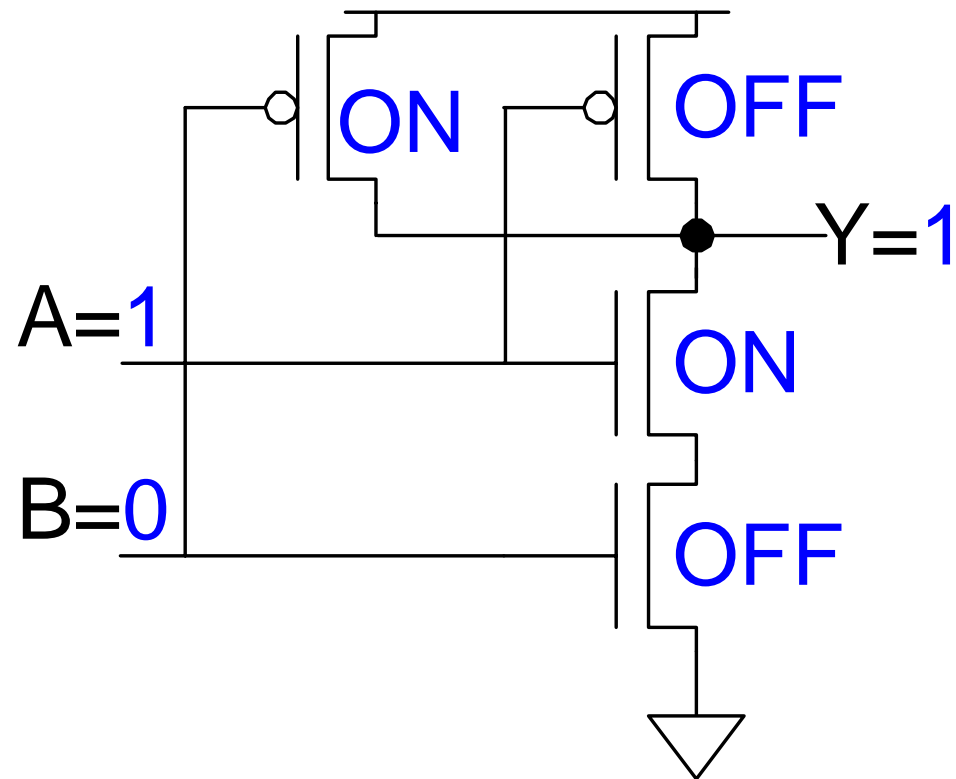
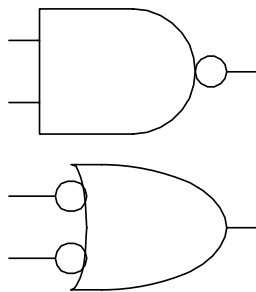
# CMOS NAND Gate

A	B	Y
0	0	1
<b>0</b>	<b>1</b>	<b>1</b>
1	0	
1	1	



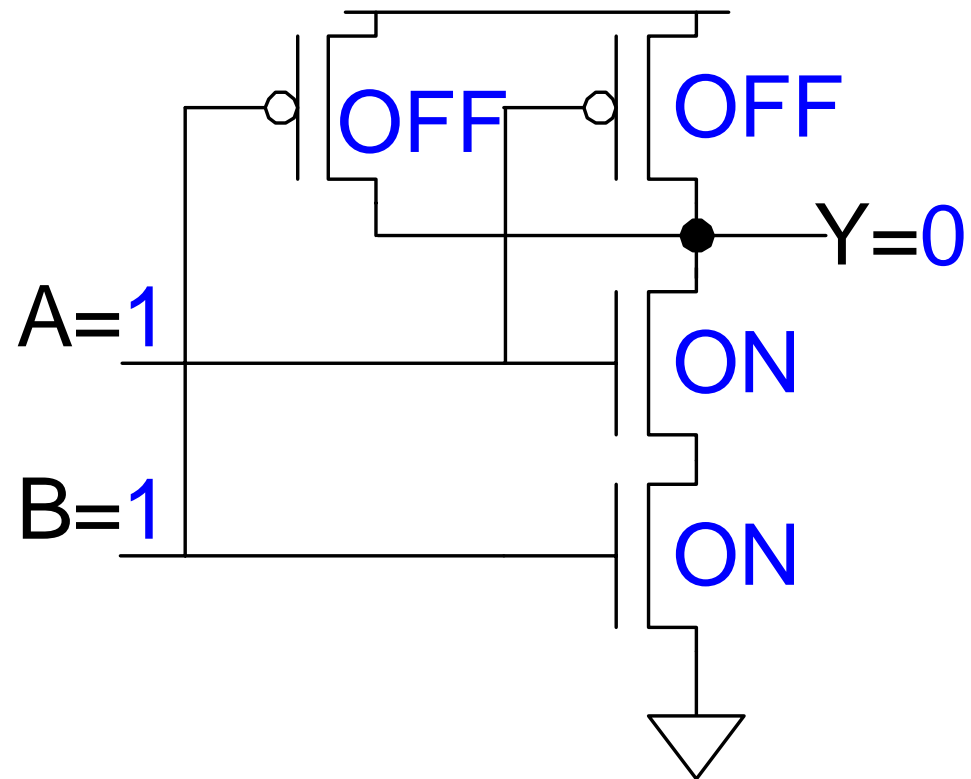
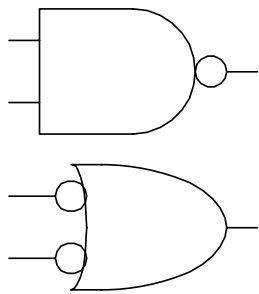
# CMOS NAND Gate

A	B	Y
0	0	1
0	1	1
<b>1</b>	<b>0</b>	<b>1</b>
1	1	



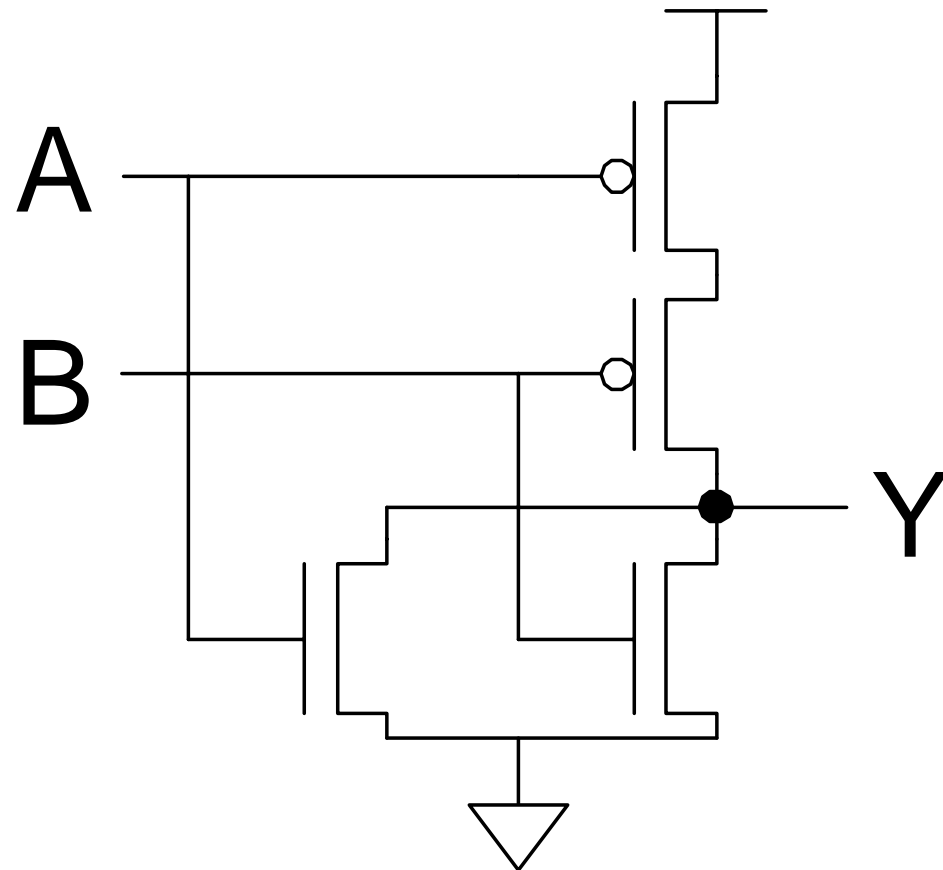
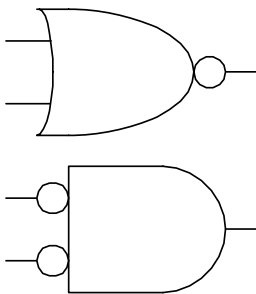
# CMOS NAND Gate

A	B	Y
0	0	1
0	1	1
1	0	1
<b>1</b>	<b>1</b>	<b>0</b>



# CMOS NOR Gate

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



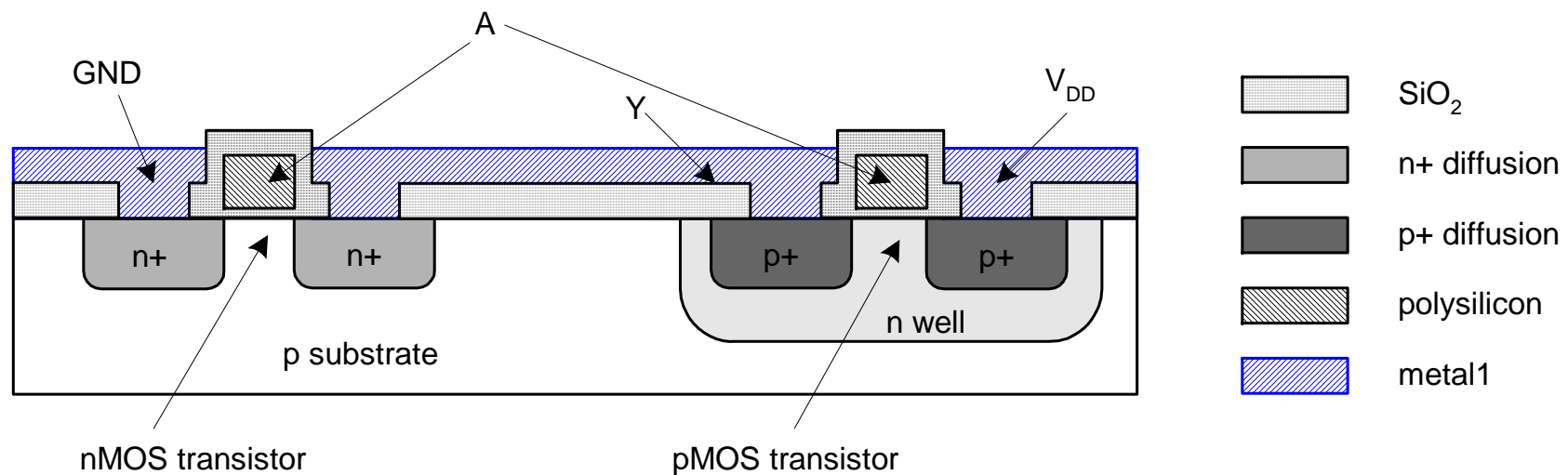
# CMOS Fabrication

---

- ❑ CMOS transistors are fabricated on silicon wafer
- ❑ Lithography process similar to printing press
- ❑ On each step, different materials are deposited or etched
- ❑ Easiest to understand by viewing both top and cross-section of wafer in a simplified manufacturing process

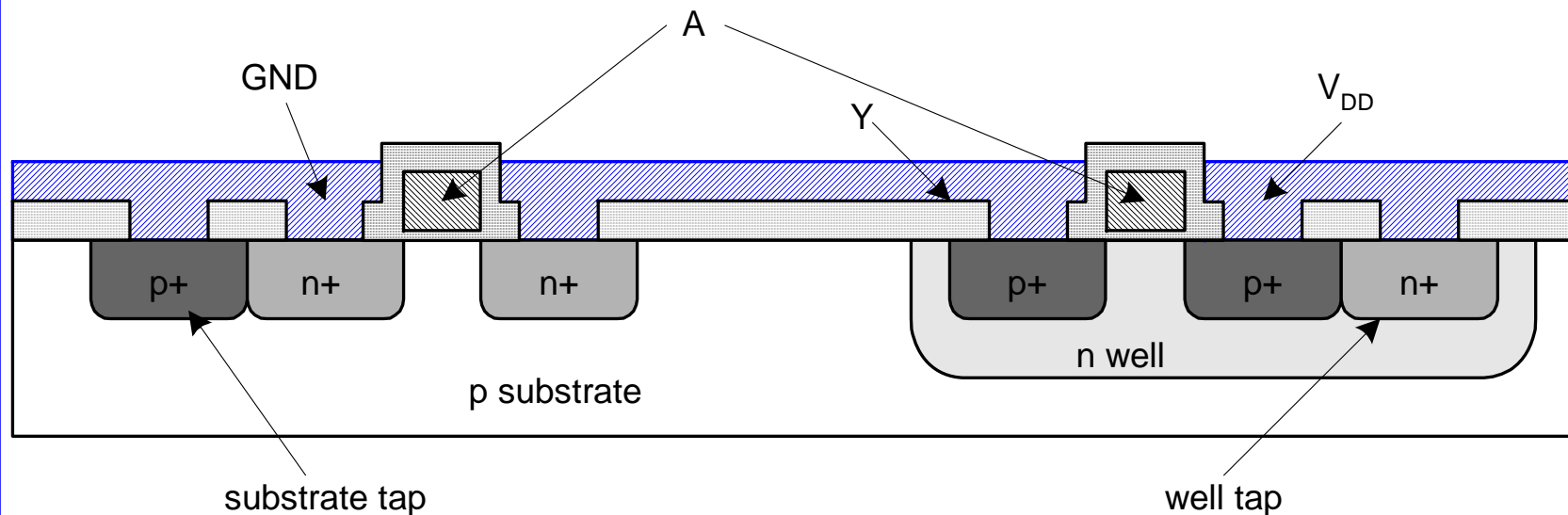
# Inverter Cross-section

- ❑ Typically use p-type substrate for nMOS transistors
- ❑ Requires n-well for body of pMOS transistors



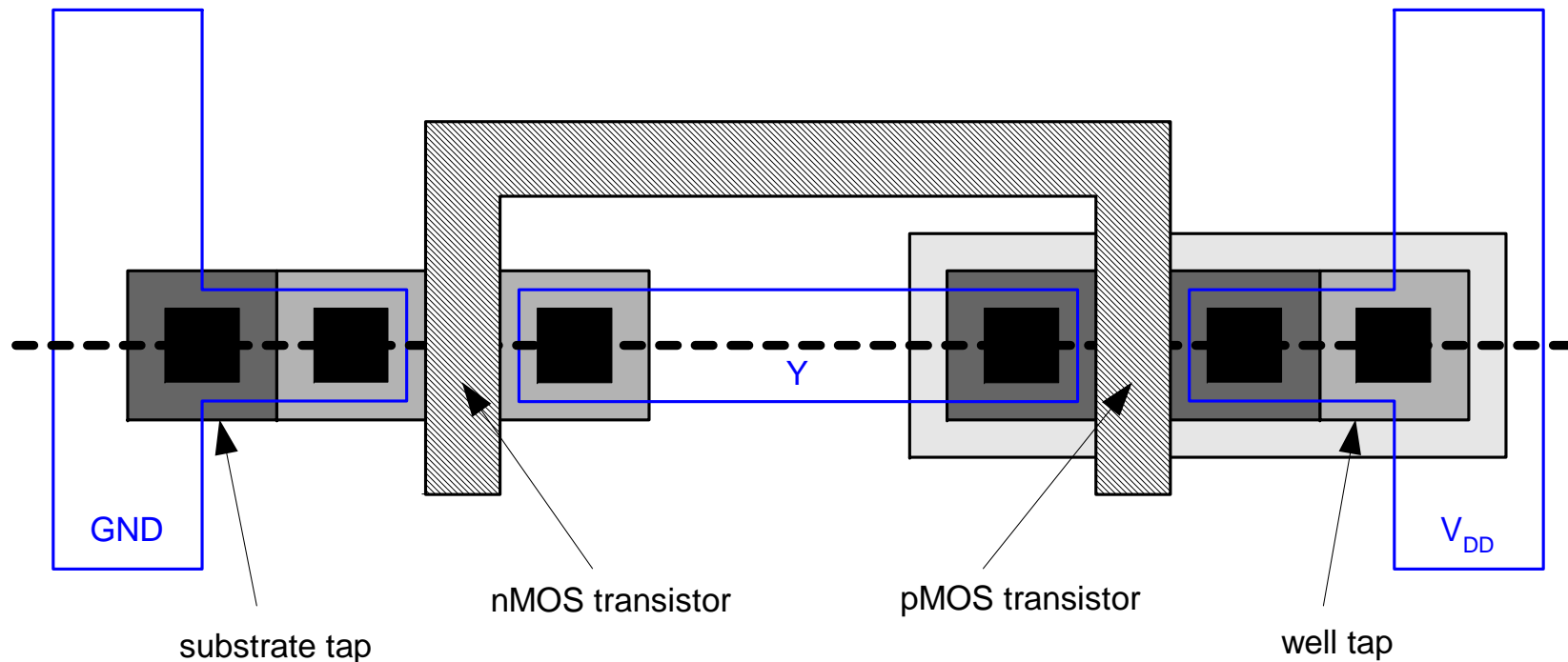
# Well and Substrate Taps

- ❑ Substrate must be tied to GND and n-well to  $V_{DD}$
- ❑ Metal to lightly-doped semiconductor forms poor connection called Schottky Diode
- ❑ Use heavily doped well and substrate contacts / taps



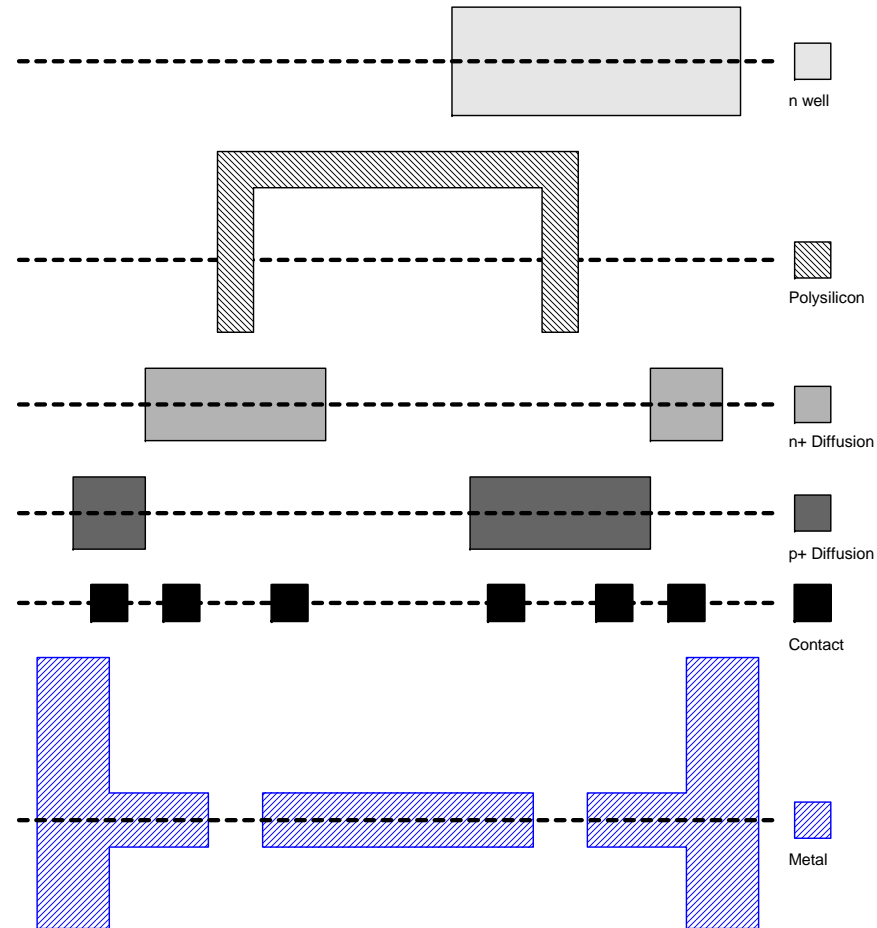
# Inverter Mask Set

- ❑ Transistors and wires are defined by *masks*
- ❑ Cross-section taken along dashed line



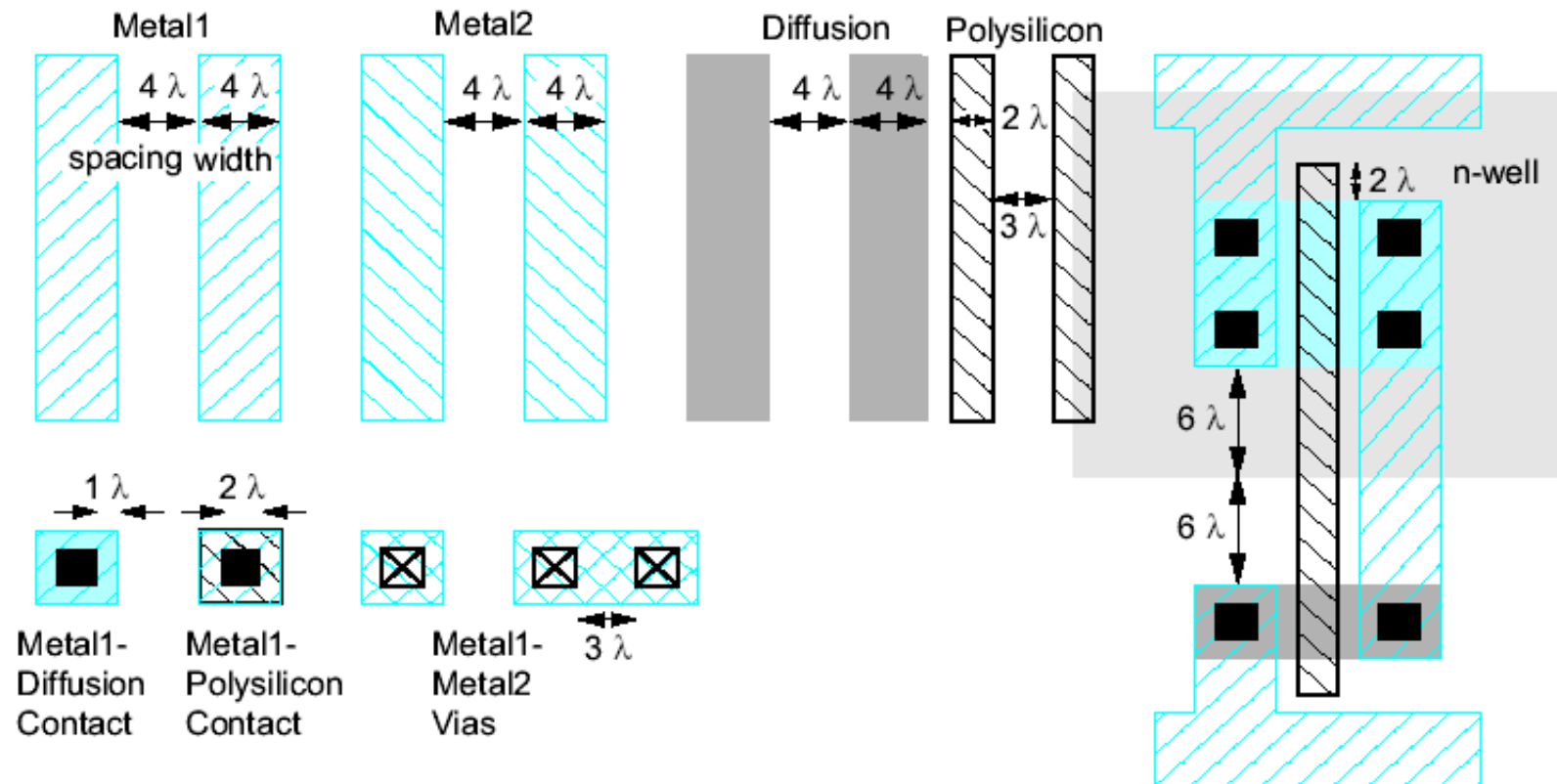
# Detailed Mask Views

- Six masks
  - n-well
  - Polysilicon
  - n+ diffusion
  - p+ diffusion
  - Contact
  - Metal



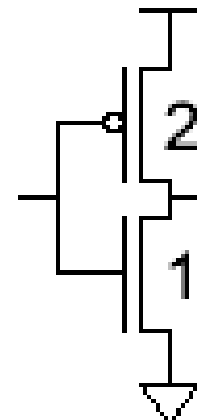
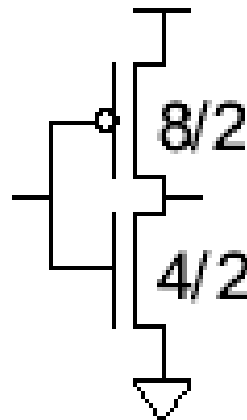
# Simplified Design Rules

- Conservative rules to get you started



# Inverter Layout

- Transistor dimensions specified as Width / Length
  - Minimum size is  $4\lambda / 2\lambda$ , sometimes called 1 unit
  - In  $f = 0.6 \mu\text{m}$  process, this is  $1.2 \mu\text{m}$  wide,  $0.6 \mu\text{m}$  long



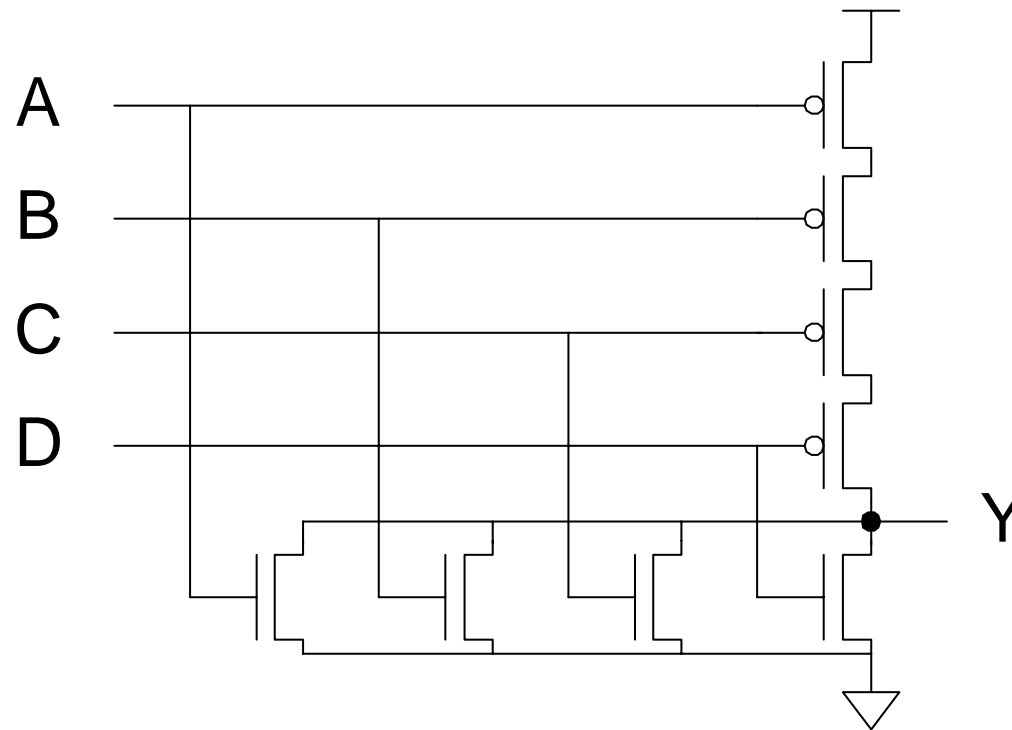
# CMOS Gate Design

---

- Activity:
  - Sketch a 4-input CMOS NOR gate

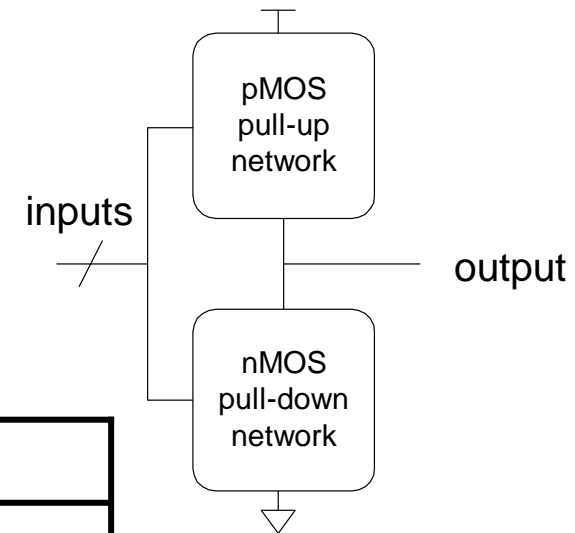
# CMOS Gate Design

- Activity:
  - Sketch a 4-input CMOS NOR gate



# Complementary CMOS

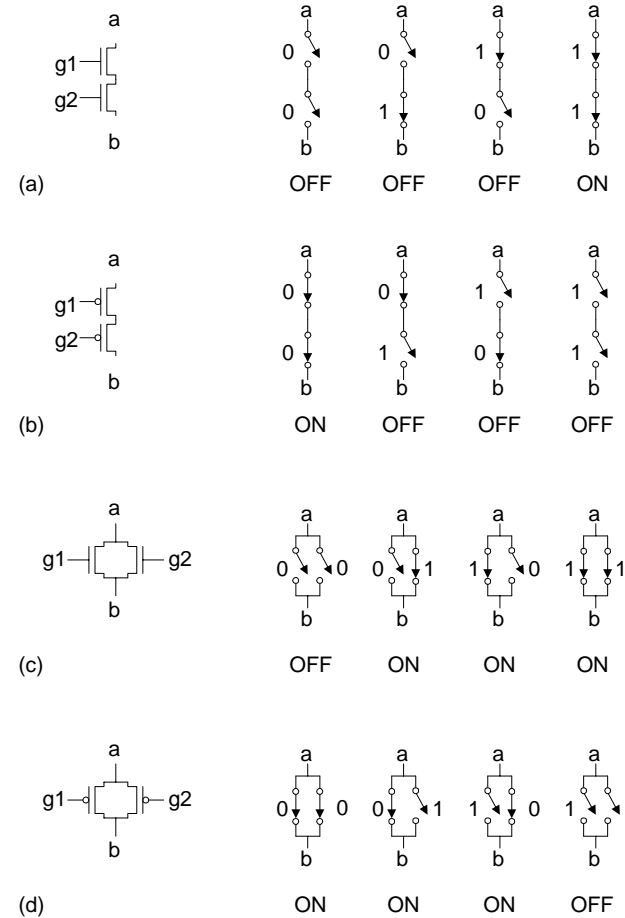
- Complementary CMOS logic gates
  - nMOS *pull-down network*
  - pMOS *pull-up network*
  - a.k.a. static CMOS



	Pull-up OFF	Pull-up ON
Pull-down OFF	Z (float)	1
Pull-down ON	0	X (crowbar)

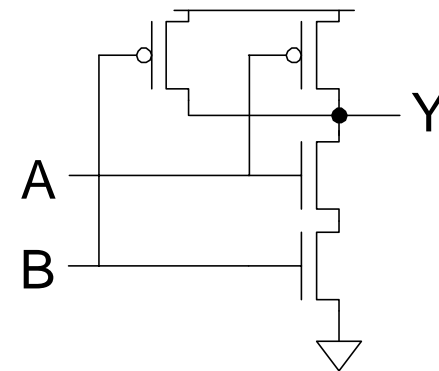
# Series and Parallel

- ❑ nMOS: 1 = ON
- ❑ pMOS: 0 = ON
- ❑ *Series*: both must be ON
- ❑ *Parallel*: either can be ON



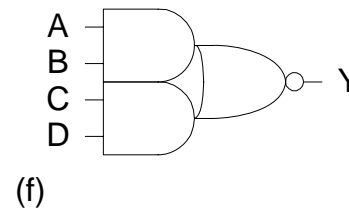
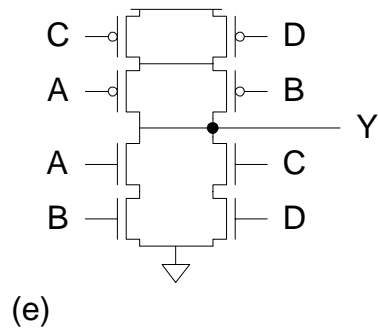
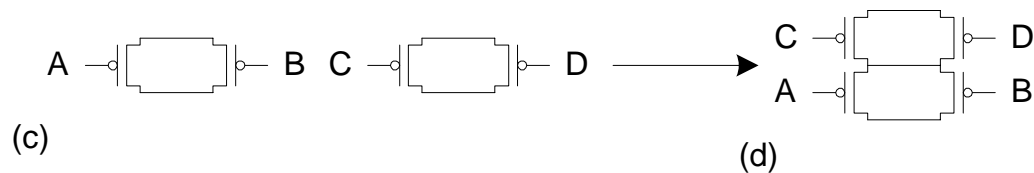
# Conduction Complement

- ❑ Complementary CMOS gates always produce 0 or 1
- ❑ Ex: NAND gate
  - Series nMOS:  $Y=0$  when both inputs are 1
  - Thus  $Y=1$  when either input is 0
  - Requires parallel pMOS
- ❑ Rule of *Conduction Complements*
  - Pull-up network is complement of pull-down
  - Parallel  $\rightarrow$  series, series  $\rightarrow$  parallel



# Compound Gates

- *Compound gates* can do any inverting function
- Ex:  $Y = \overline{A \square B + C \square D}$  (AND-AND-OR-INVERT, AOI22)

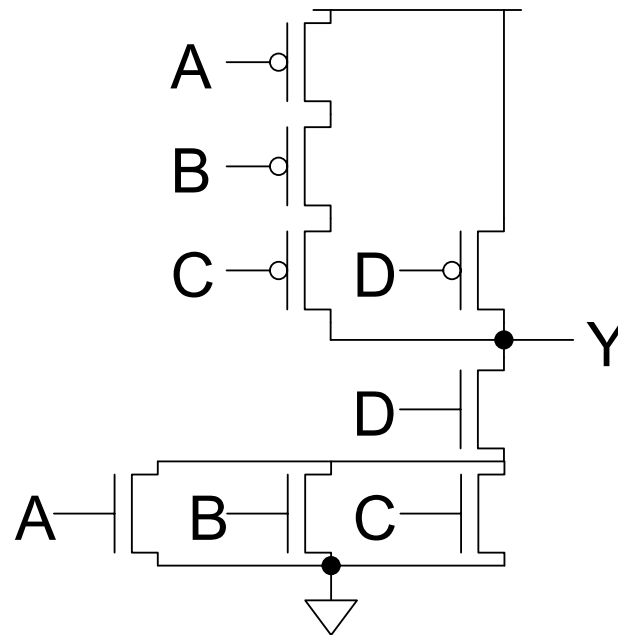


# Example: O3A1

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

# Example: O3AI

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

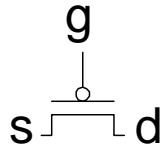
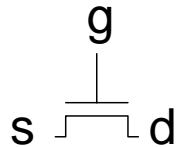


# Signal Strength

- ❑ *Strength* of signal
  - How close it approximates ideal voltage source
- ❑  $V_{DD}$  and GND rails are strongest 1 and 0
- ❑ nMOS pass strong 0
  - But degraded or weak 1
- ❑ pMOS pass strong 1
  - But degraded or weak 0
- ❑ Thus nMOS are best for pull-down network

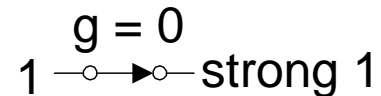
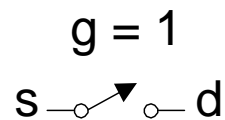
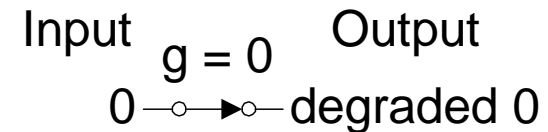
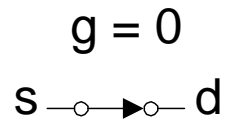
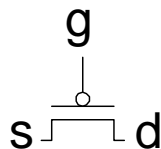
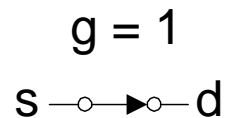
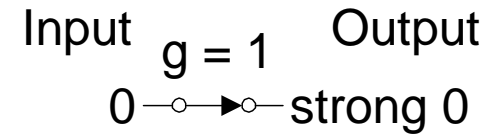
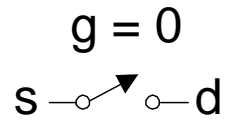
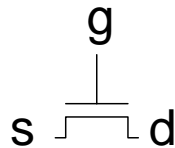
# Pass Transistors

- Transistors can be used as switches



# Pass Transistors

- Transistors can be used as switches



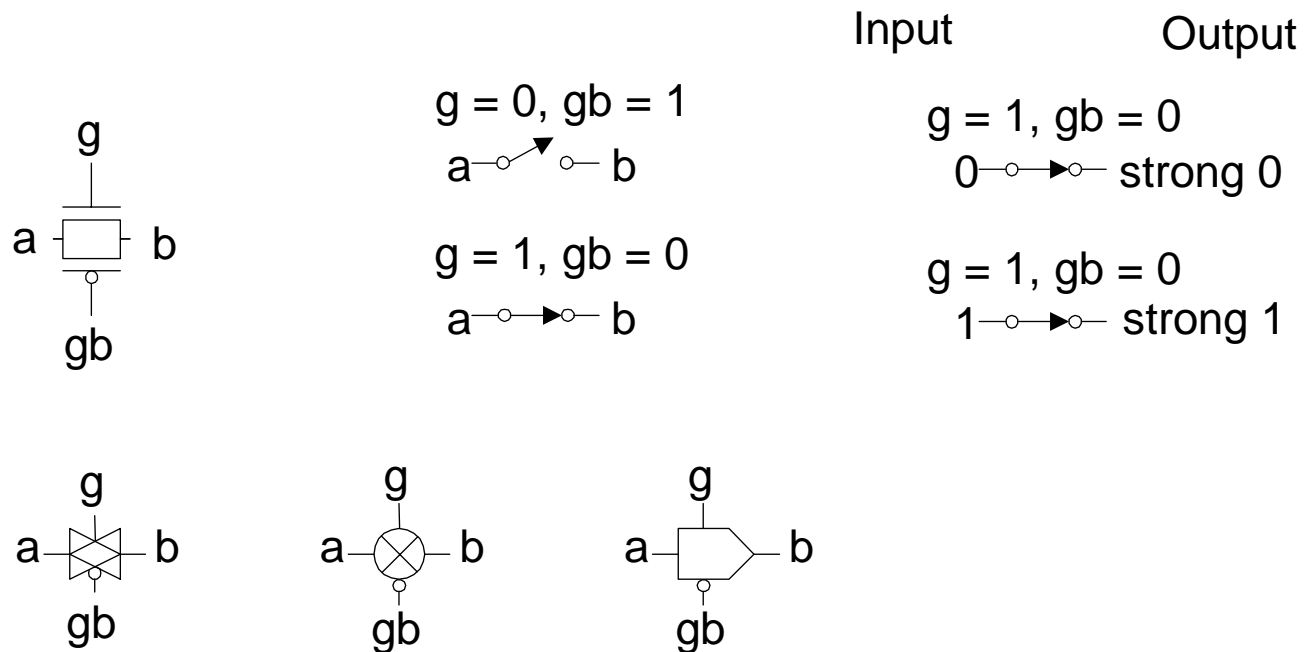
# Transmission Gates

---

- ❑ Pass transistors produce degraded outputs
- ❑ *Transmission gates* pass both 0 and 1 well

# Transmission Gates

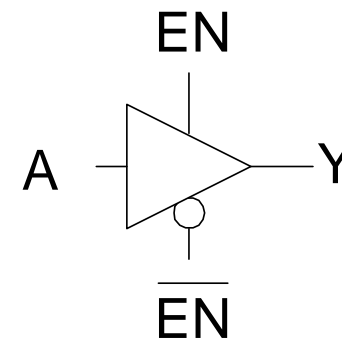
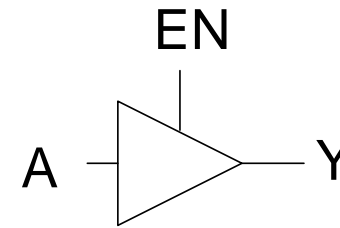
- ❑ Pass transistors produce degraded outputs
- ❑ *Transmission gates* pass both 0 and 1 well



# Tristates

- ❑ *Tristate buffer* produces Z when not enabled

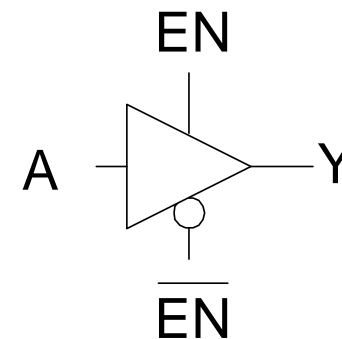
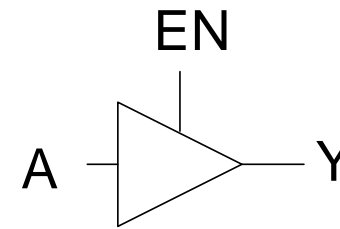
EN	A	Y
0	0	
0	1	
1	0	
1	1	



# Tristates

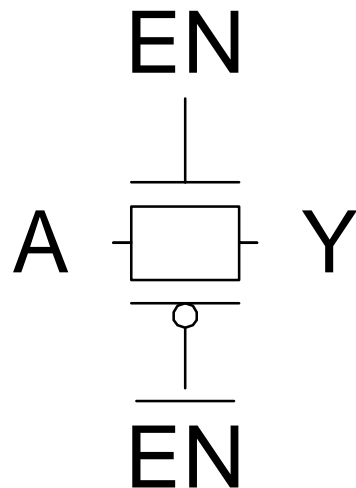
- ❑ *Tristate buffer* produces Z when not enabled

EN	A	Y
0	0	Z
0	1	Z
1	0	0
1	1	1



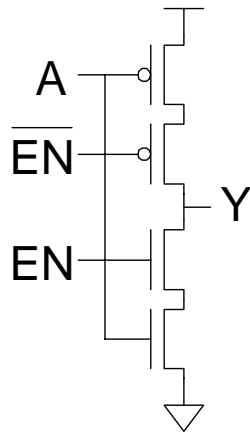
# Nonrestoring Tristate

- ❑ Transmission gate acts as tristate buffer
  - Only two transistors
  - But *nonrestoring*
    - Noise on A is passed on to Y



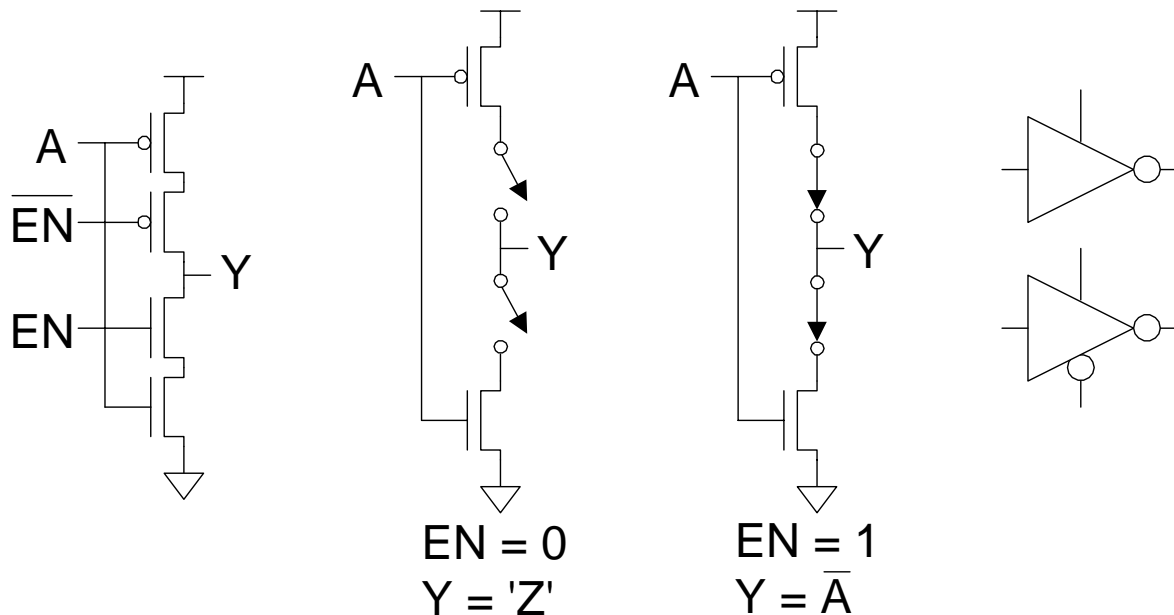
# Tristate Inverter

- ❑ Tristate inverter produces restored output
  - Violates conduction complement rule
  - Because we want a Z output



# Tristate Inverter

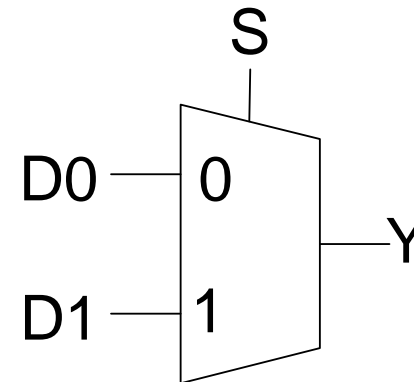
- Tristate inverter produces restored output
  - Violates conduction complement rule
  - Because we want a Z output



# Multiplexers

- 2:1 *multiplexer* chooses between two inputs

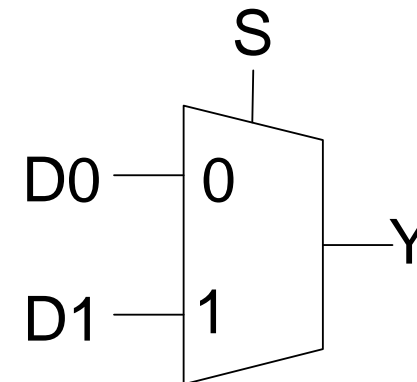
S	D1	D0	Y
0	X	0	
0	X	1	
1	0	X	
1	1	X	



# Multiplexers

- 2:1 multiplexer chooses between two inputs

S	D1	D0	Y
0	X	0	0
0	X	1	1
1	0	X	0
1	1	X	1

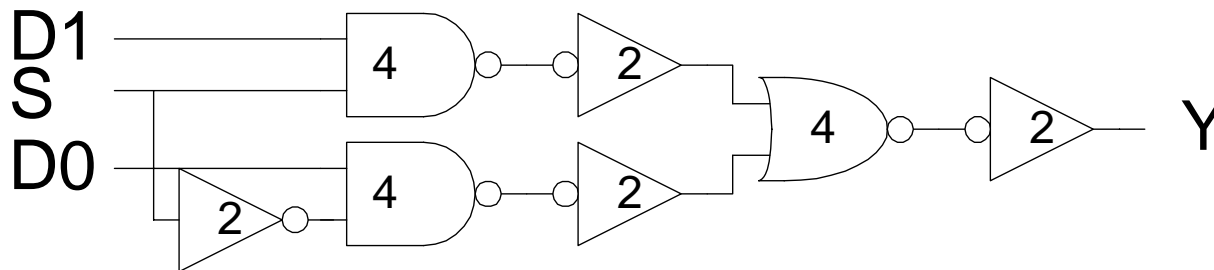
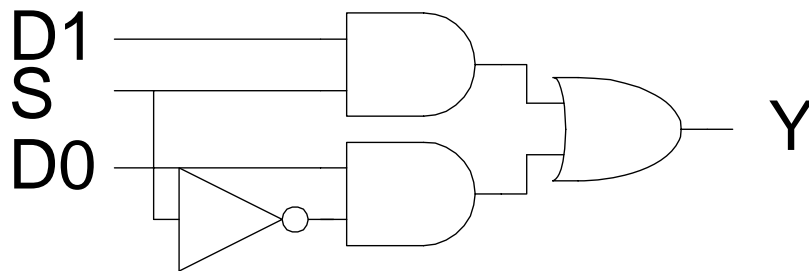


# Gate-Level Mux Design

- ❑  $Y = SD_1 + \bar{S}D_0$  (too many transistors)
- ❑ How many transistors are needed?

# Gate-Level Mux Design

- ❑  $Y = SD_1 + \bar{S}D_0$  (too many transistors)
- ❑ How many transistors are needed? 20



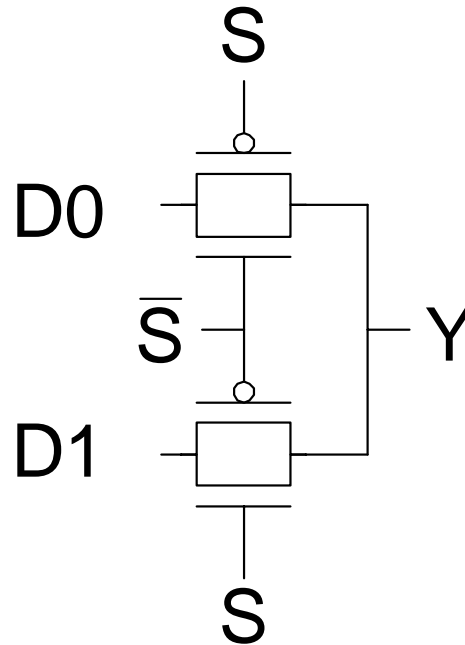
# Transmission Gate Mux

---

- ❑ Nonrestoring mux uses two transmission gates

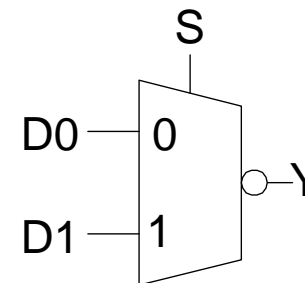
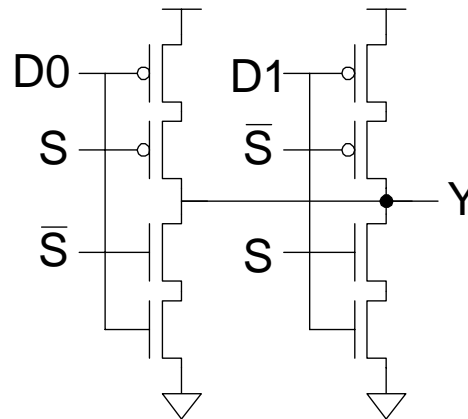
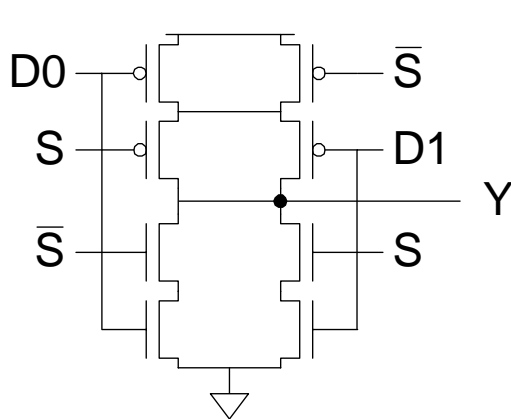
# Transmission Gate Mux

- Nonrestoring mux uses two transmission gates
  - Only 4 transistors



# Inverting Mux

- ❑ Inverting multiplexer
  - Use compound AOI22
  - Or pair of tristate inverters
  - Essentially the same thing
- ❑ Noninverting multiplexer adds an inverter

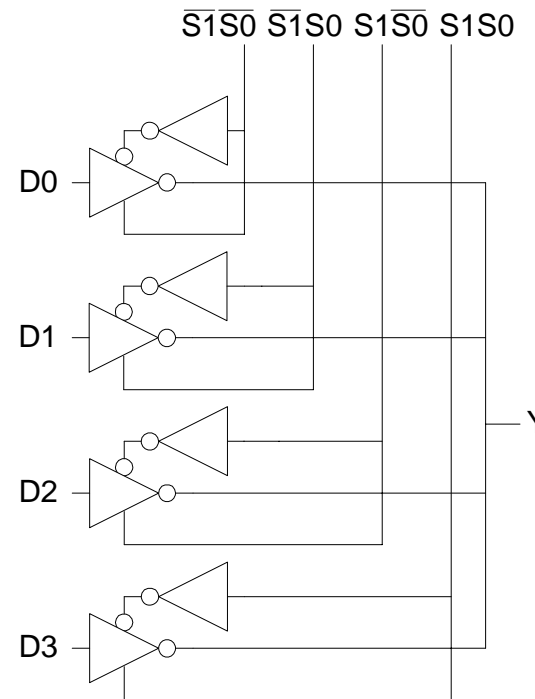
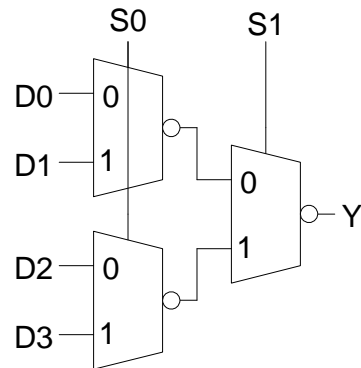


# 4:1 Multiplexer

- 4:1 mux chooses one of 4 inputs using two selects

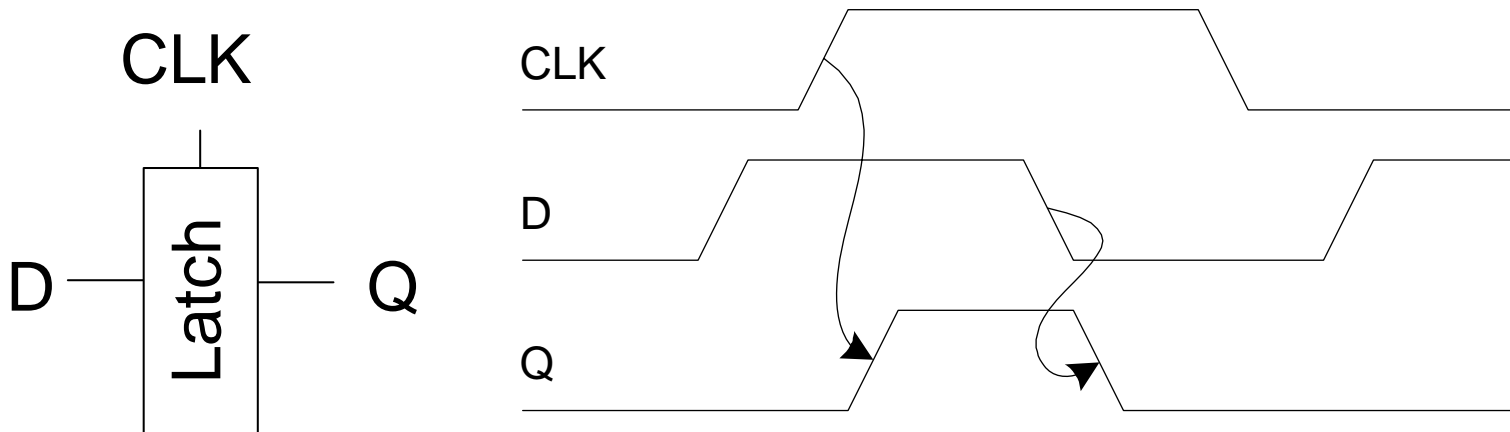
# 4:1 Multiplexer

- 4:1 mux chooses one of 4 inputs using two selects
  - Two levels of 2:1 muxes
  - Or four tristates



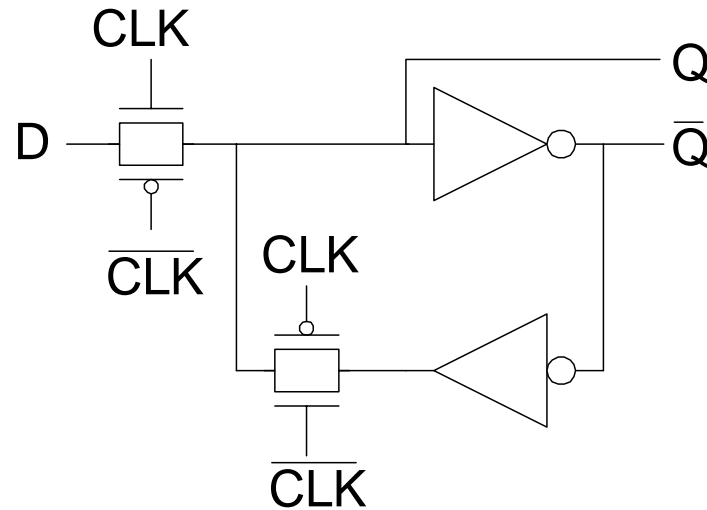
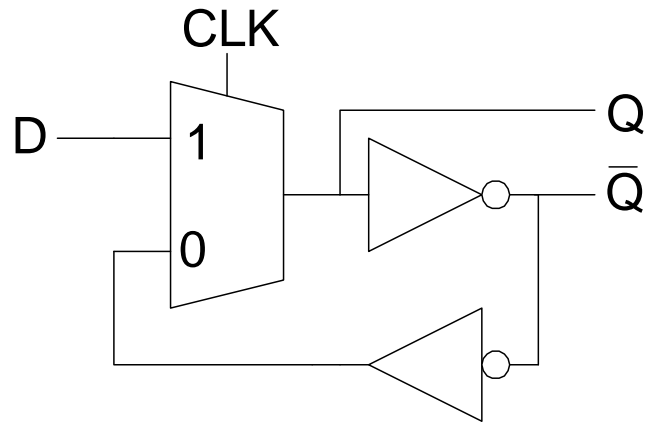
# D Latch

- ❑ When  $CLK = 1$ , latch is *transparent*
  - D flows through to Q like a buffer
- ❑ When  $CLK = 0$ , the latch is *opaque*
  - Q holds its old value independent of D
- ❑ a.k.a. *transparent latch* or *level-sensitive latch*



# D Latch Design

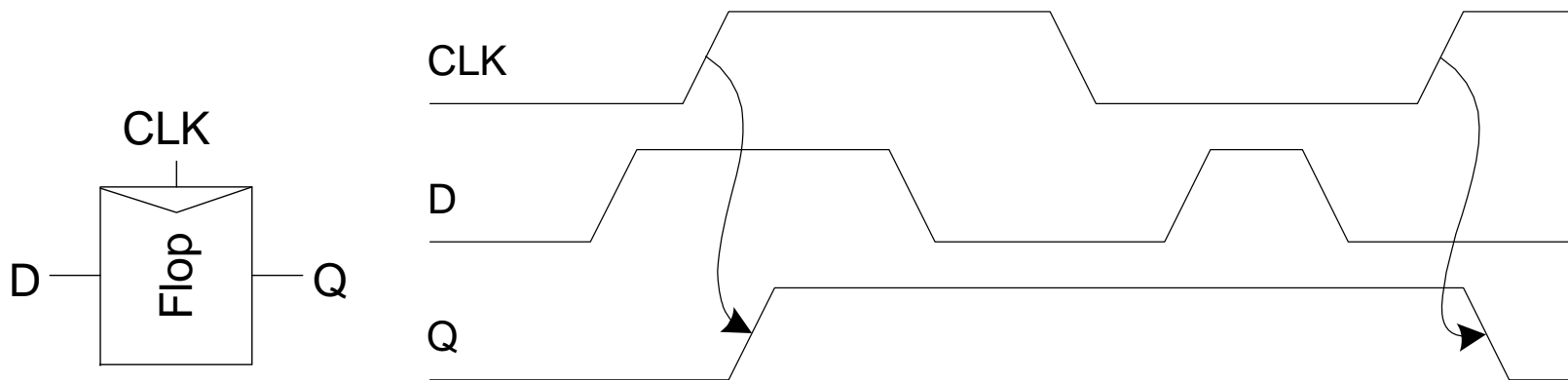
- ❑ Multiplexer chooses D or old Q





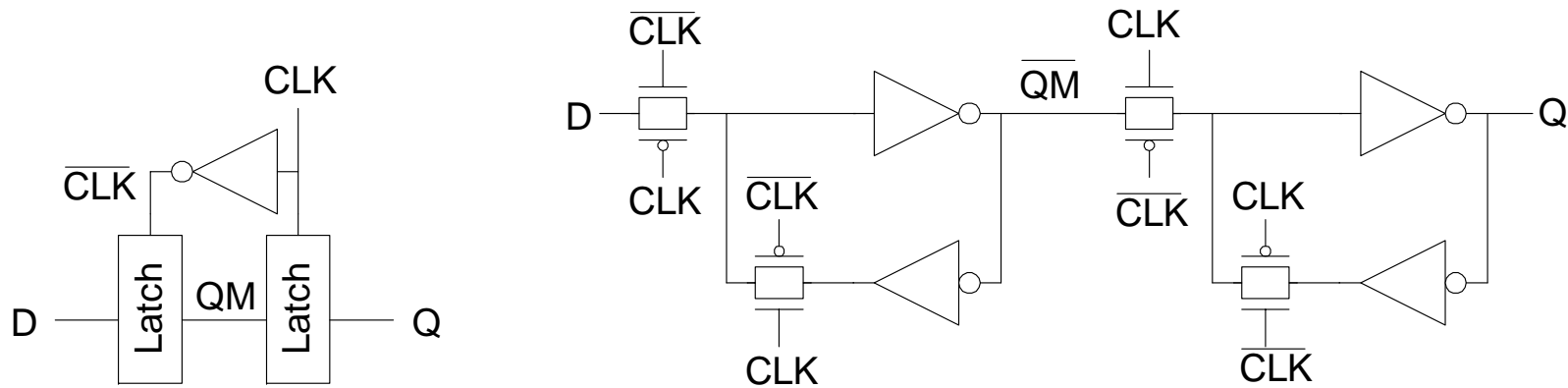
# D Flip-flop

- ❑ When CLK rises, D is copied to Q
- ❑ At all other times, Q holds its value
- ❑ a.k.a. *positive edge-triggered flip-flop, master-slave flip-flop*

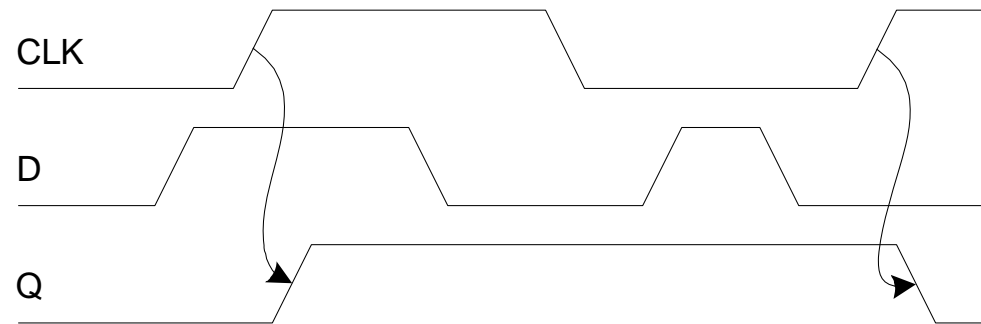
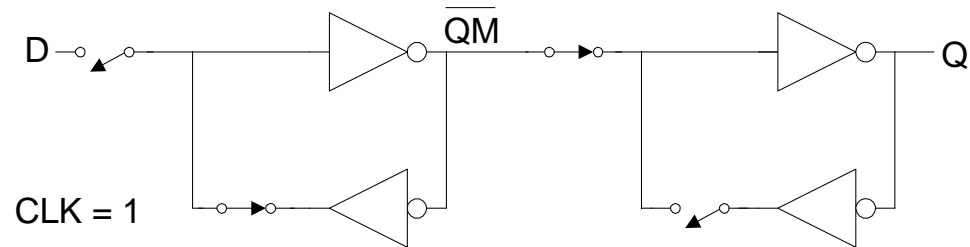
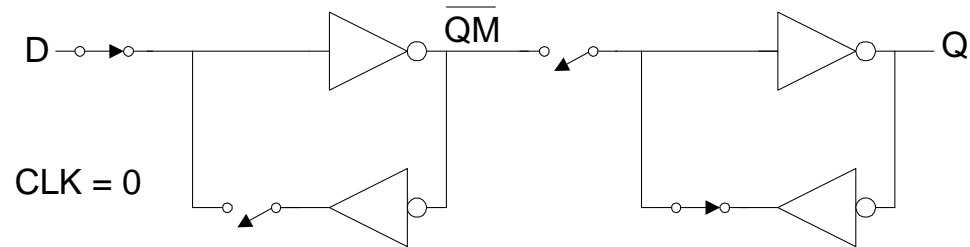


# D Flip-flop Design

- Built from master and slave D latches

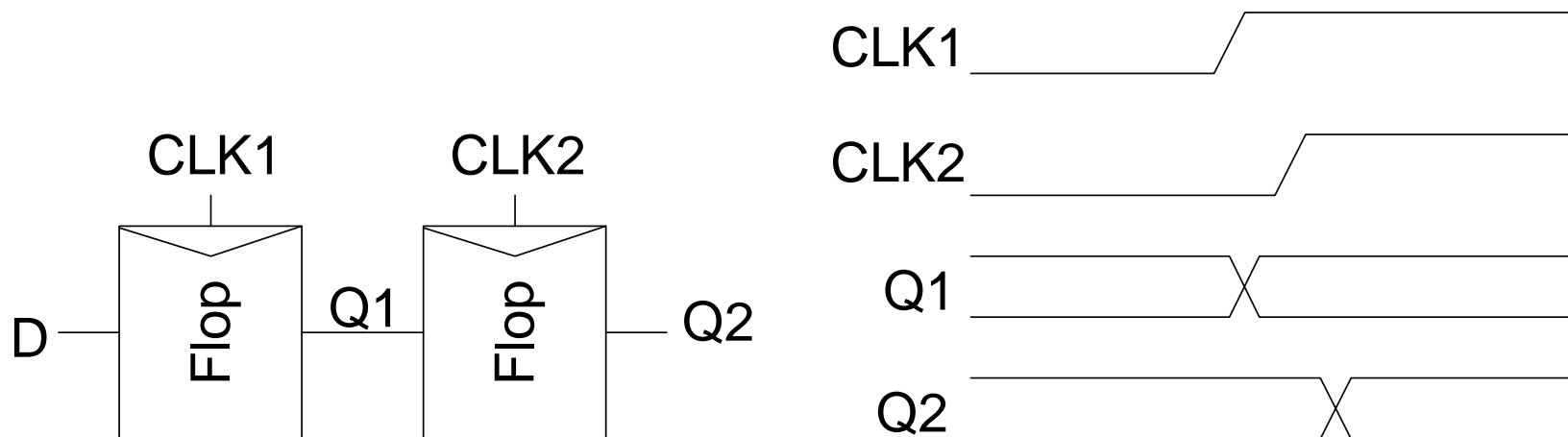


# D Flip-flop Operation



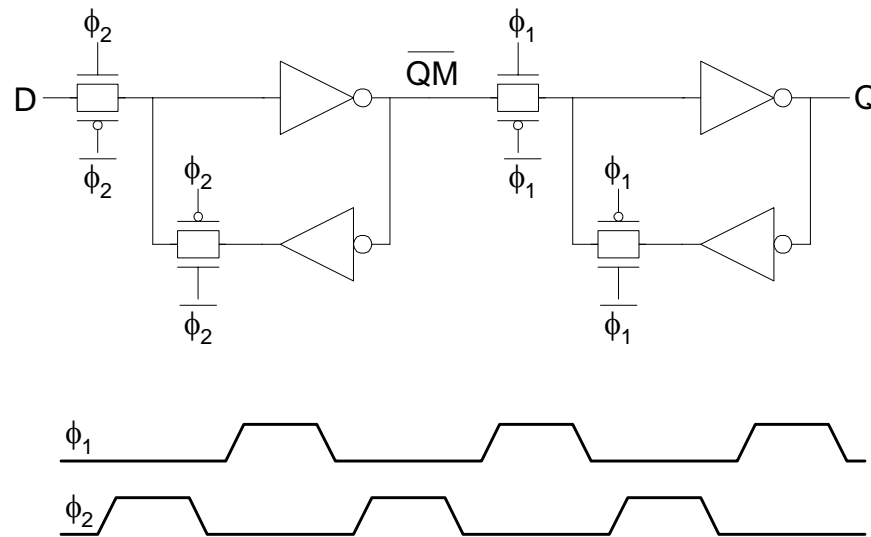
# Race Condition

- ❑ Back-to-back flops can malfunction from clock skew
  - Second flip-flop fires late
  - Sees first flip-flop change and captures its result
  - Called *hold-time failure* or *race condition*



# Nonoverlapping Clocks

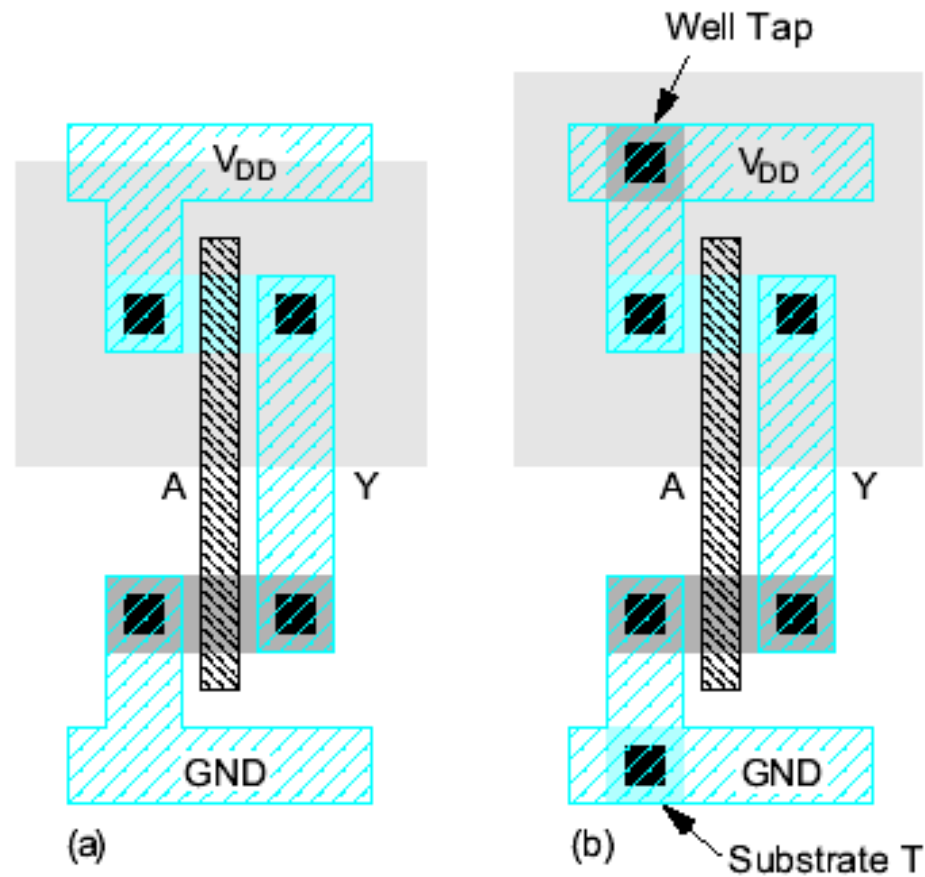
- ❑ Nonoverlapping clocks can prevent races
  - As long as nonoverlap exceeds clock skew
- ❑ We will use them in this class for safe design
  - Industry manages skew more carefully instead



# Gate Layout

- ❑ Layout can be very time consuming
  - Design gates to fit together nicely
  - Build a library of standard cells
- ❑ Standard cell design methodology
  - $V_{DD}$  and GND should abut (standard height)
  - Adjacent gates should satisfy design rules
  - nMOS at bottom and pMOS at top
  - All gates include well and substrate contacts

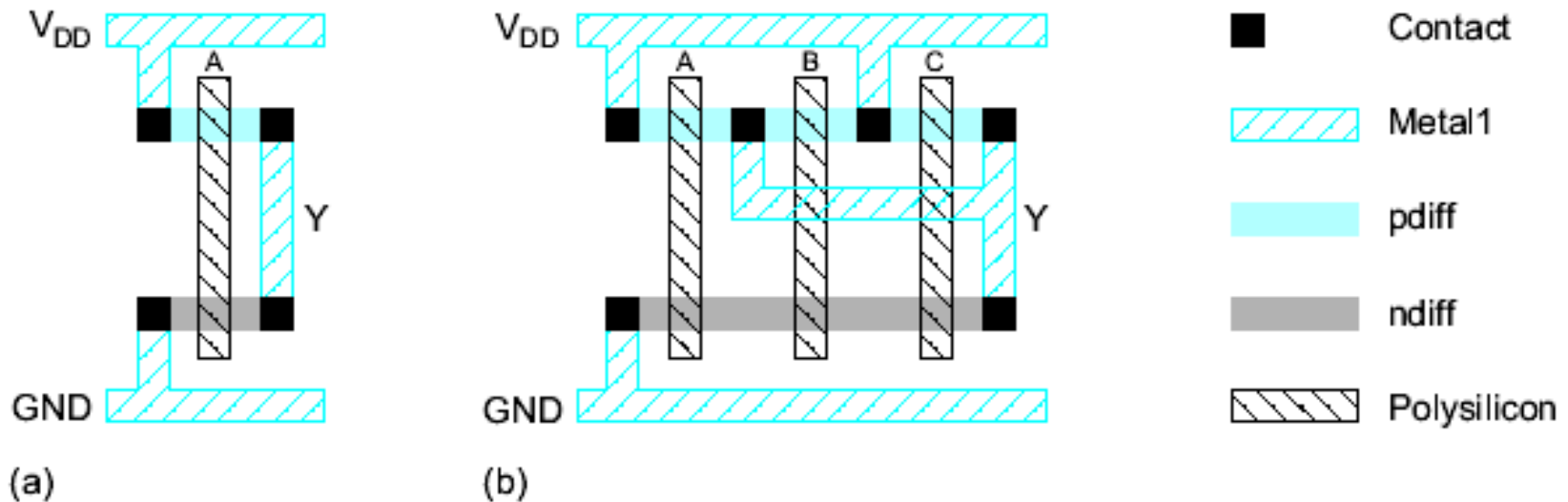
# Example: Inverter





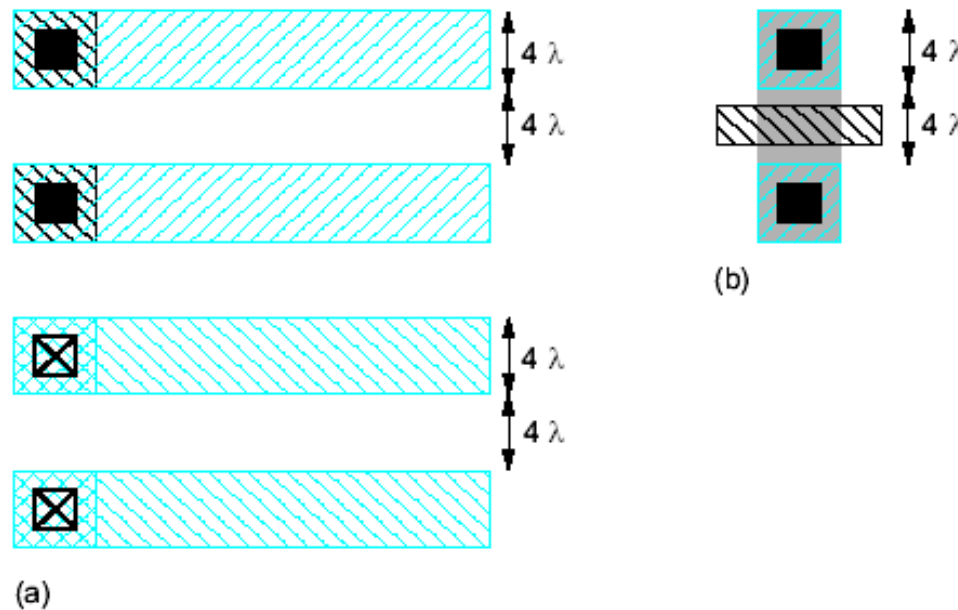
# Stick Diagrams

- *Stick diagrams* help plan layout quickly
  - Need not be to scale
  - Draw with color pencils or dry-erase markers



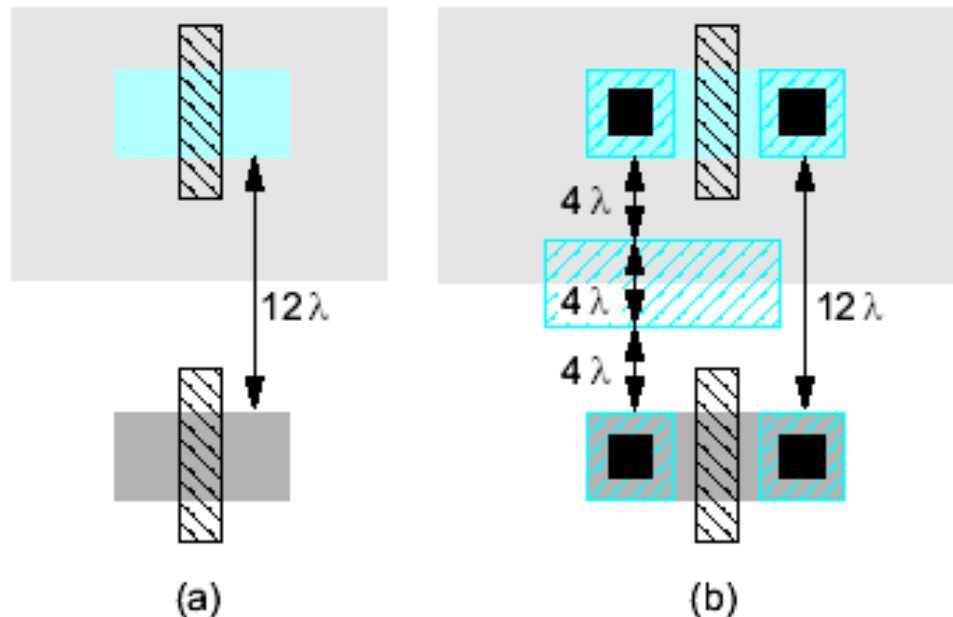
# Wiring Tracks

- A *wiring track* is the space required for a wire
  - $4 \lambda$  width,  $4 \lambda$  spacing from neighbor =  $8 \lambda$  pitch
- Transistors also consume one wiring track



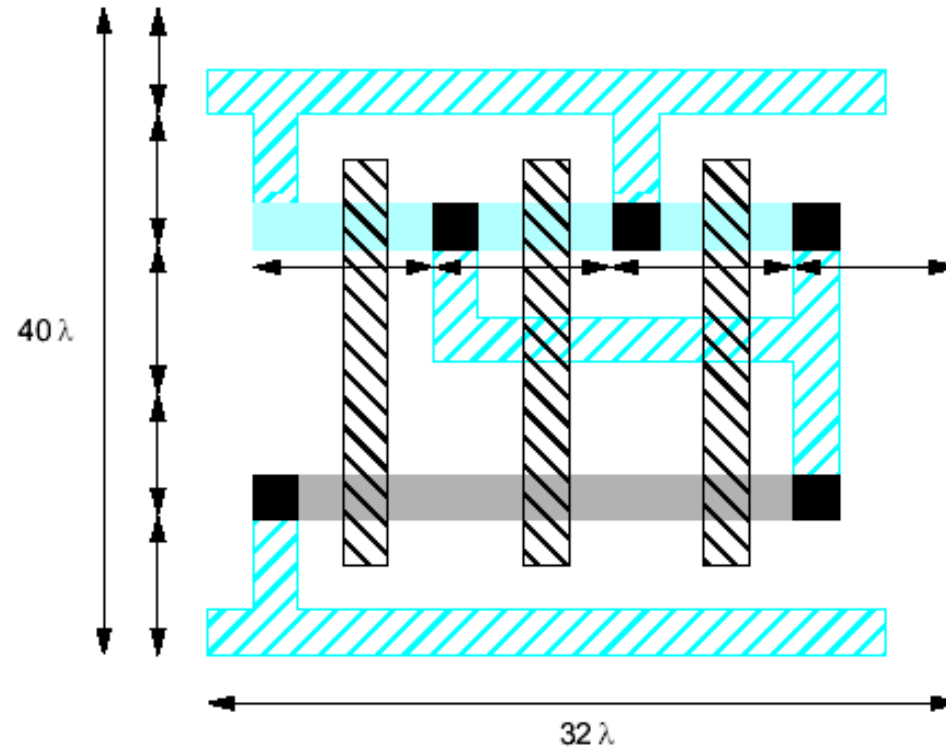
# Well spacing

- ❑ Wells must surround transistors by  $6\lambda$ 
  - Implies  $12\lambda$  between opposite transistor flavors
  - Leaves room for one wire track



# Area Estimation

- Estimate area by counting wiring tracks
  - Multiply by 8 to express in  $\lambda$



# Example: O3AI

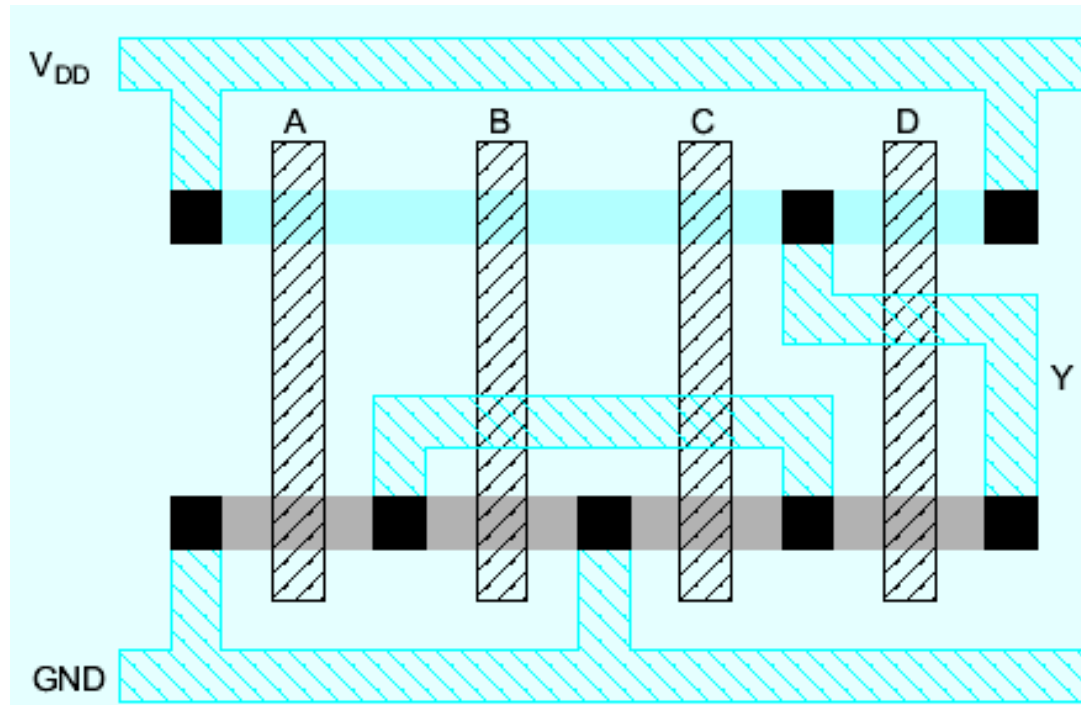
- Sketch a stick diagram for O3AI and estimate area

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

# Example: O3AI

- Sketch a stick diagram for O3AI and estimate area

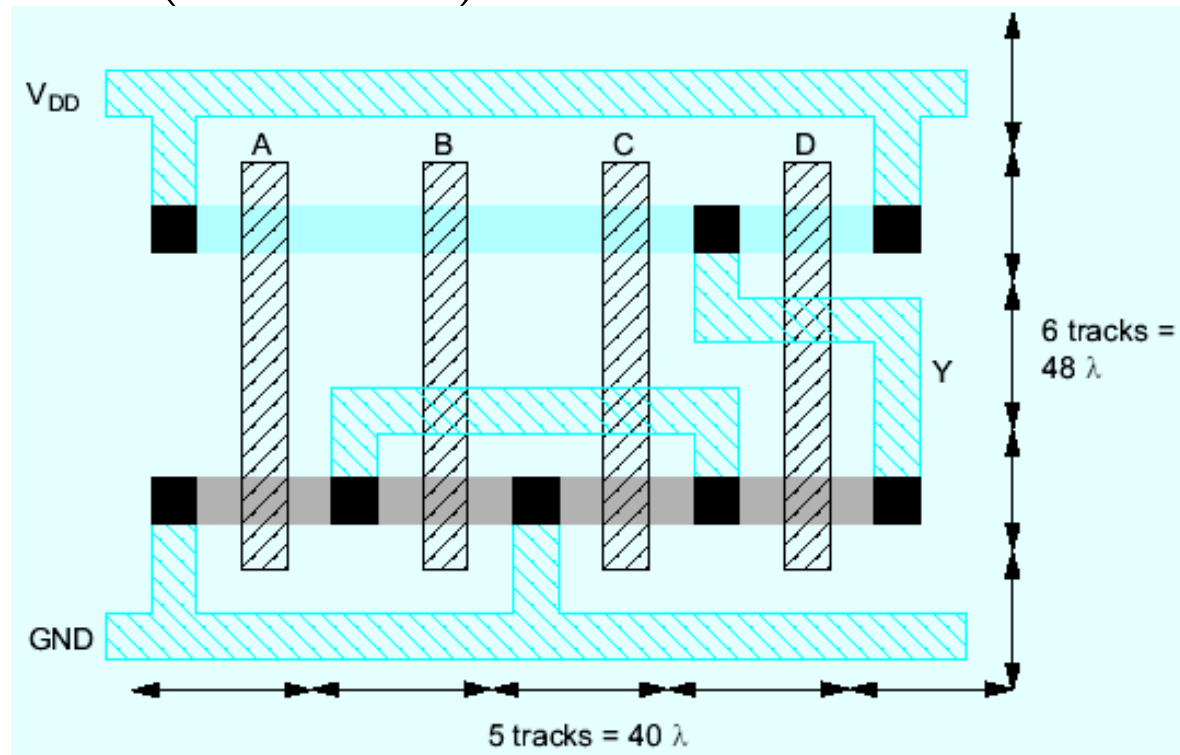
$$- Y = \overline{(A + B + C)} \square D$$



# Example: O3AI

- Sketch a stick diagram for O3AI and estimate area

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

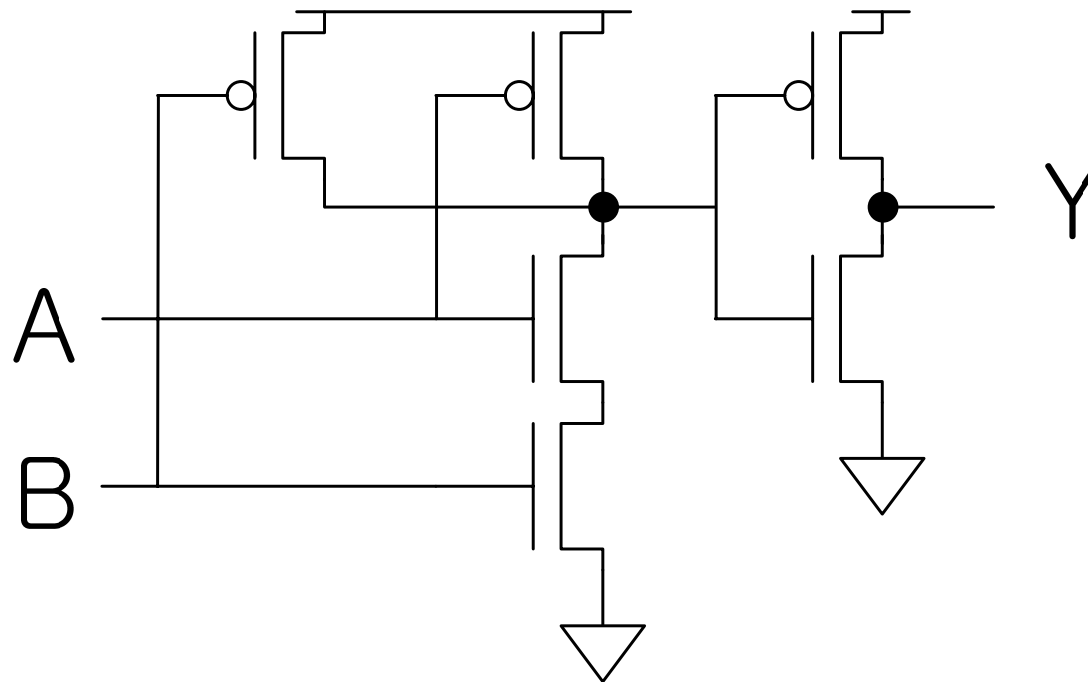


# Notice

---

- ❑ Course Materials
  - <http://www.donny.co.kr>
  - e-mail: [ceo@donny.co.kr](mailto:ceo@donny.co.kr)
  - Tel: 019-690-2054
  
- ❑ Homework Deadline
  - 10/15

# Q1 Solution: 2-input AND



# Q2 Solution: Complex Gate

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

