Disclaimer

The slides presented here are a combination of the CS251 course notes from previous terms, the work of Xiao-Bo Li, and material from the required textbook "Computer Organization and Design, ARM Edition," by David A. Patterson and John L. Hennessy. It is being used here with explicit permission from the authors.

CS251 course policy requires students to delete all course files after the term. Therefore, please do not post these slides to any website or share them.

CS251 - Computer Organization and Design

From Gates to Transistors

Instructor: Zille Huma Kamal

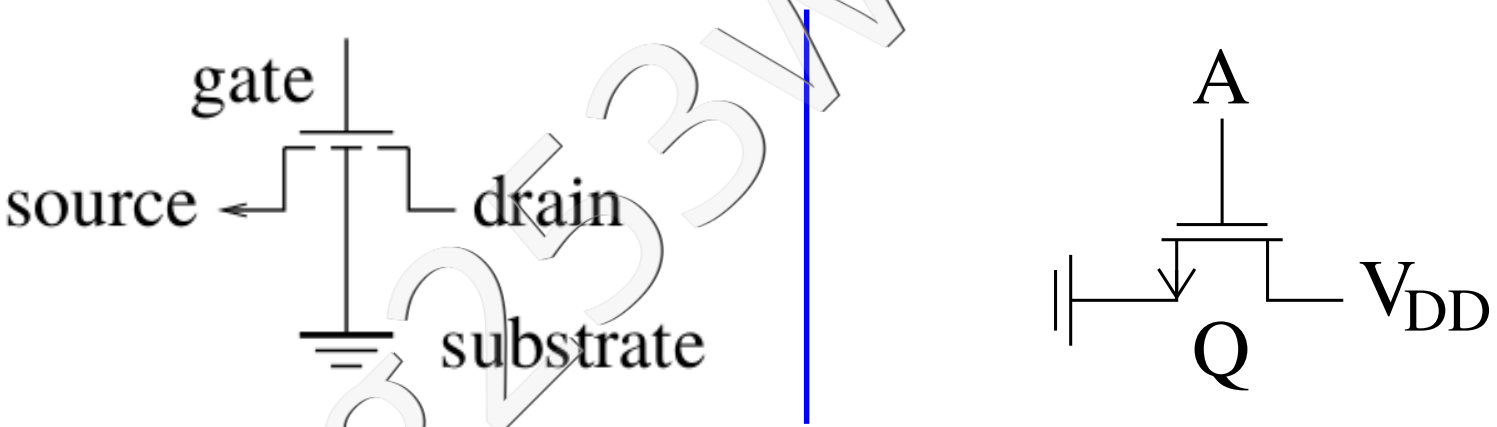
University of Waterloo

Spring 2023

Implementing Gates Using Transistors

- Transistor: an electrically-controlled switch

- An nMOS transistor ("n-transistor") and its symbol

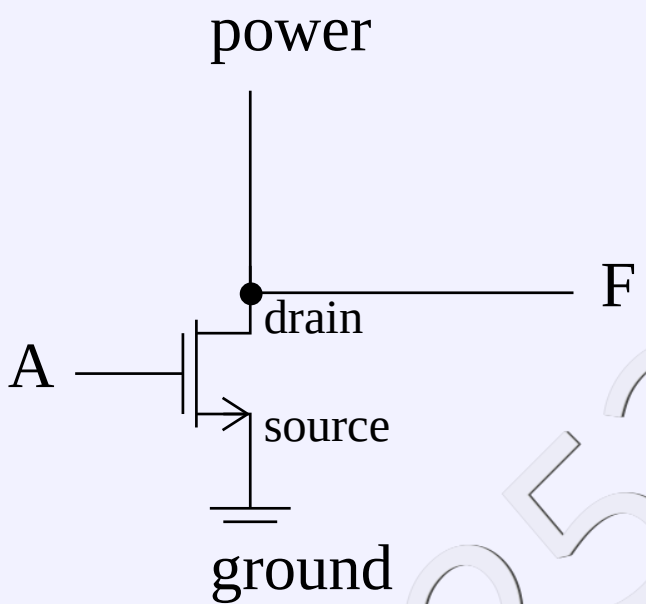


- This behaves like the switch above
- Problem: transmits strong 0 but weak 1

nMOS is naturally inverting circuit (nMOS NOT)

Think About It

Consider the transistor circuit below and complete its truth table:

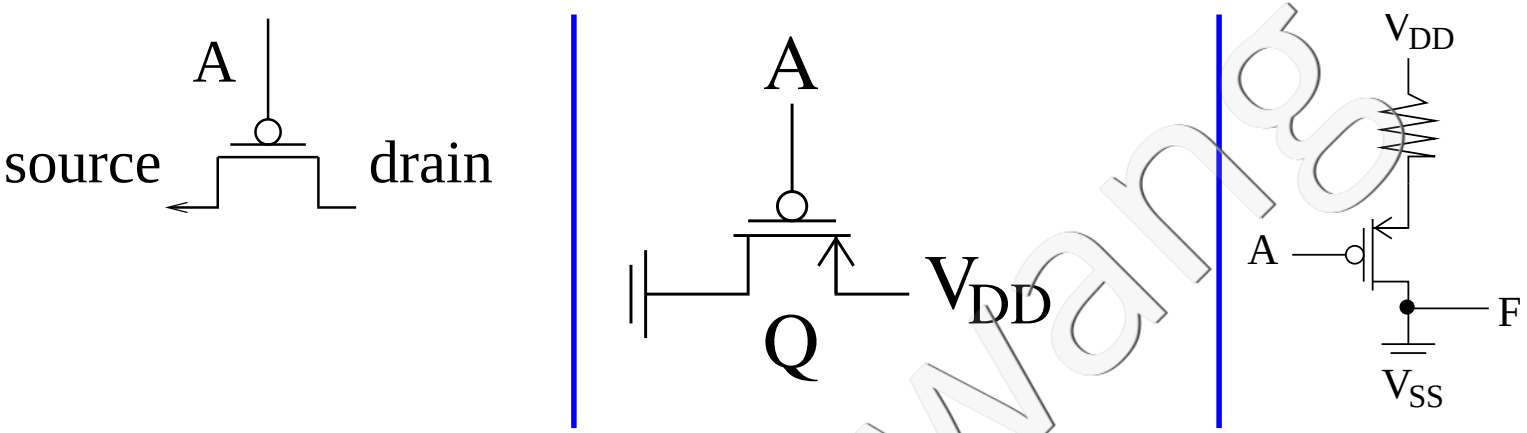


A	F
0	
1	

- If $A = 1$, then low resistance between drain and source ($F = 0$)
- If $A = 0$, then very high resistance between drain and source ($F = 1$)

- **Problem:** When $A = 1$ lots of current flows to the ground (GND) $\Rightarrow$ power is wasted
- **Solution:** Add a resistor between `power` and `drain`

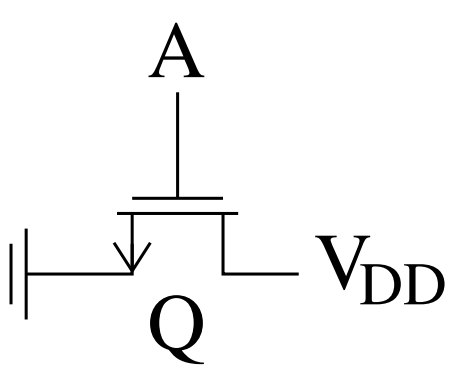
A pMOS transistor

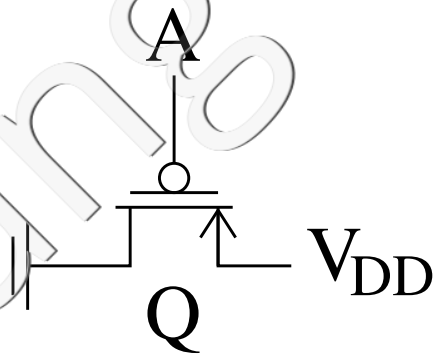


- Opposite behaviour to nMOS:
  - If $A = 1$, high resistance between drain and source
  - If $A = 0$, low resistance between drain and source
  - Transmits strong 1 but weak 0
- "bubble" on gate (input `A`) indicates transistor works in opposite manner to nMOS

Transistor Summary

- Two types of transistors: nMOS, pMOS





nMOS

Input	$A = 0$	$A = 1$
Resistance	High	Low

pMOS

Input	$A = 0$	$A = 1$
Resistance	Low	High

- **nMOS** passes a **strong 0** but when resistance is low, it passes a **weak 1**.
- **pMOS** passes a **strong 1** but when resistance is low, it passes a **weak 0**.

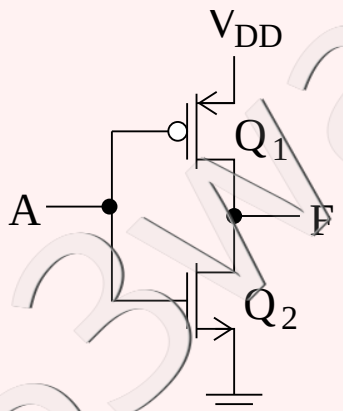
CMOS

- CMOS circuits use both nMOS and pMOS transistors
- Build circuits with "clean" paths to exactly one of power or ground.
- To analyze CMOS circuit:
  - Make truth table with inputs, transistors, and output(s) and
  - evaluate whether transistor's resistance(s) is high (H) or low (L)
  - Output is:
    - ★ **1** if there is a "clean" path to power (1),
    - ★ **0** if there is a "clean" path to ground (0)
  - Make sure there is no clean path to power **AND** ground
    - ★ If there is, the output of the circuit is unknown
- Avoid:
  - Wasteful flow of current from power to ground
  - Weak transmissions

CMOS Circuit Analysis

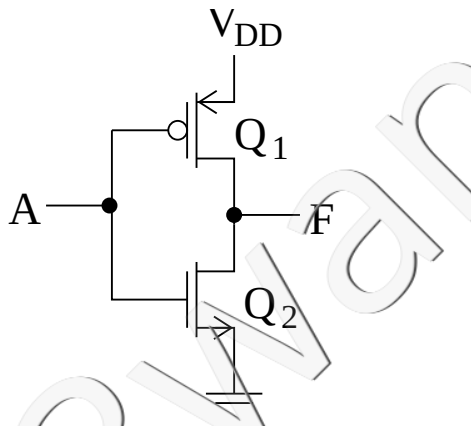
Try this

Complete the truth table for the CMOS circuit below:



A	Q_1	Q_2	F
0			
1			

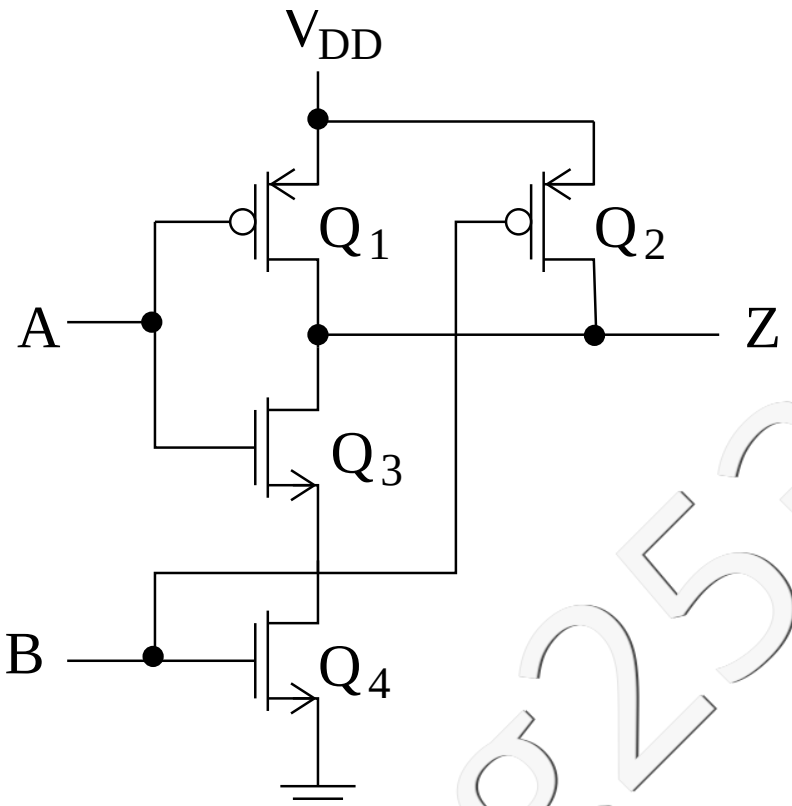
Solution: CMOS NOT



A	Q_1	Q_2	F
0	Low	High	1
1	High	Low	0

The CMOS circuit implements a `NOT` gate.

CMOS NAND

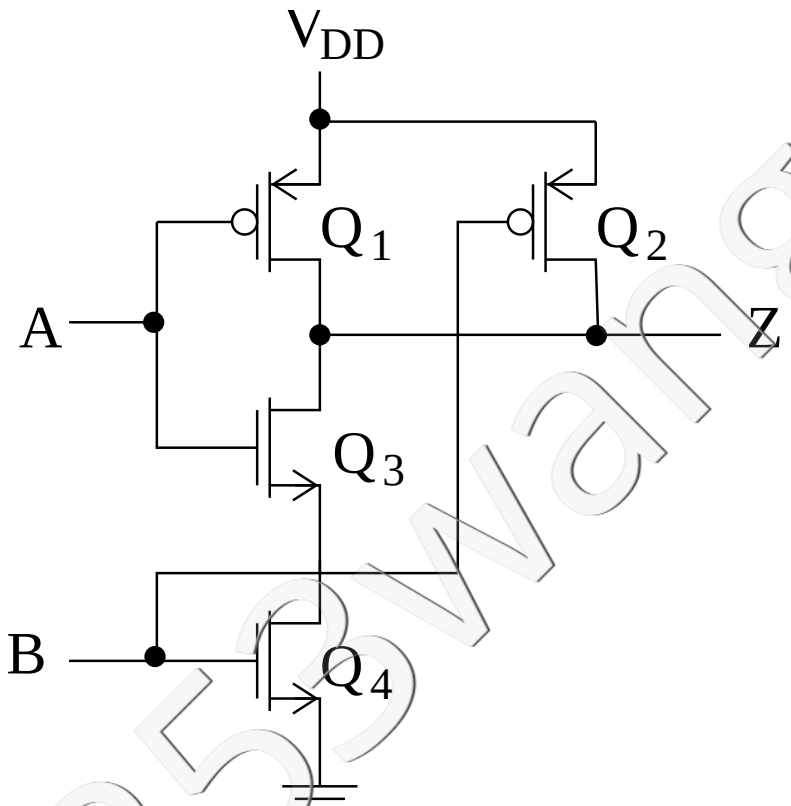


Try this

Complete the truth table for the CMOS circuit shown:

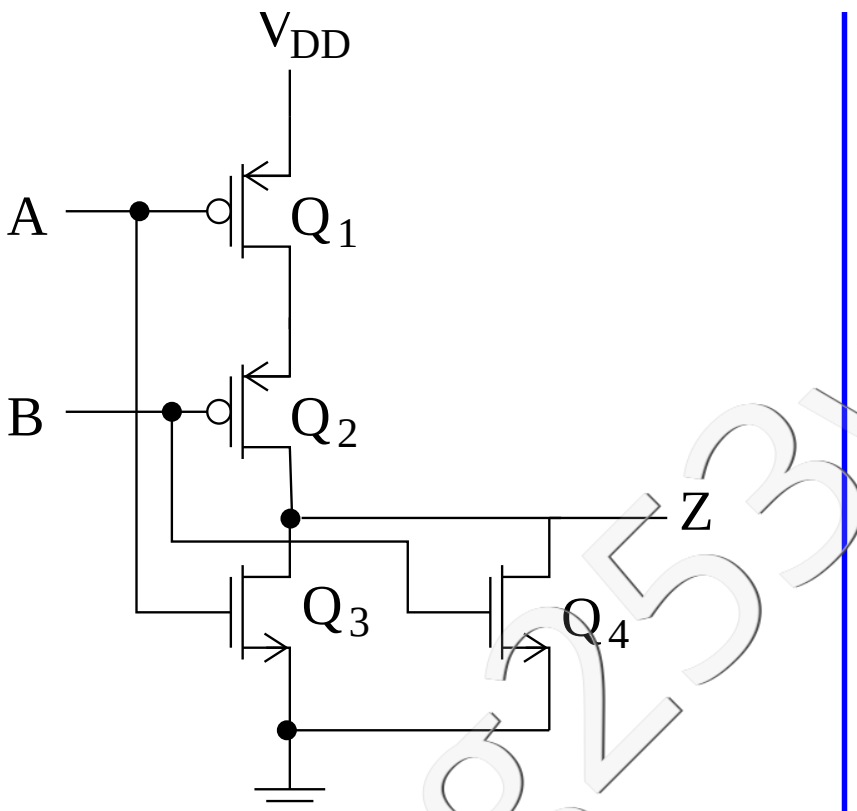
A	B	Q_1	Q_2	Q_3	Q_4	Z
0	0					
0	1					
1	0					
1	1					

Solution: CMOS NAND Gate



A	B	Q_1	Q_2	Q_3	Q_4	Z
0	0	Low	Low	High	High	1
0	1	Low	High	High	Low	1
1	0	High	Low	Low	High	1
1	1	High	High	Low	Low	0

CMOS NOR

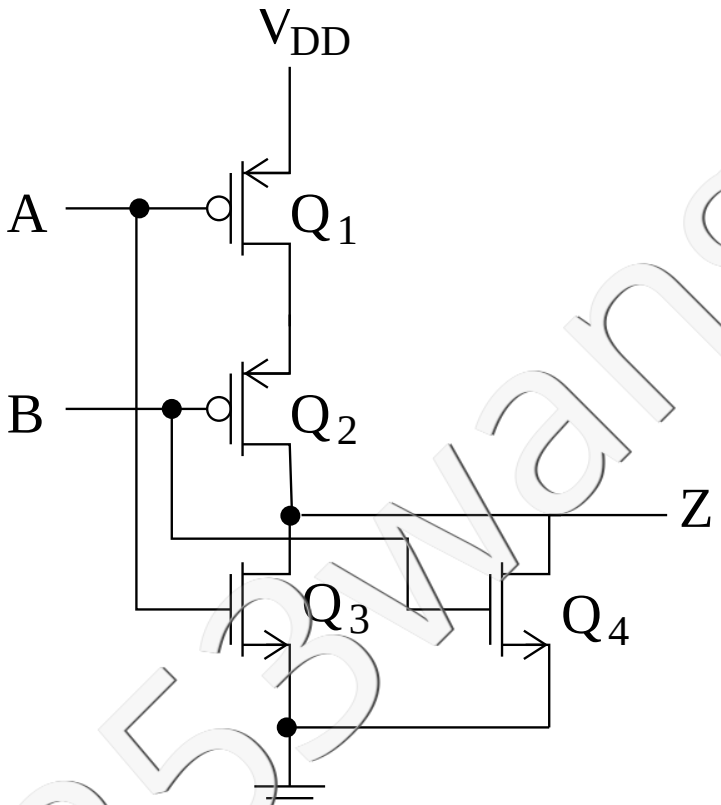


Try this

Complete the truth table for the CMOS circuit shown:

A	B	Q_1	Q_2	Q_3	Q_4	Z
0	0					
0	1					
1	0					
1	1					

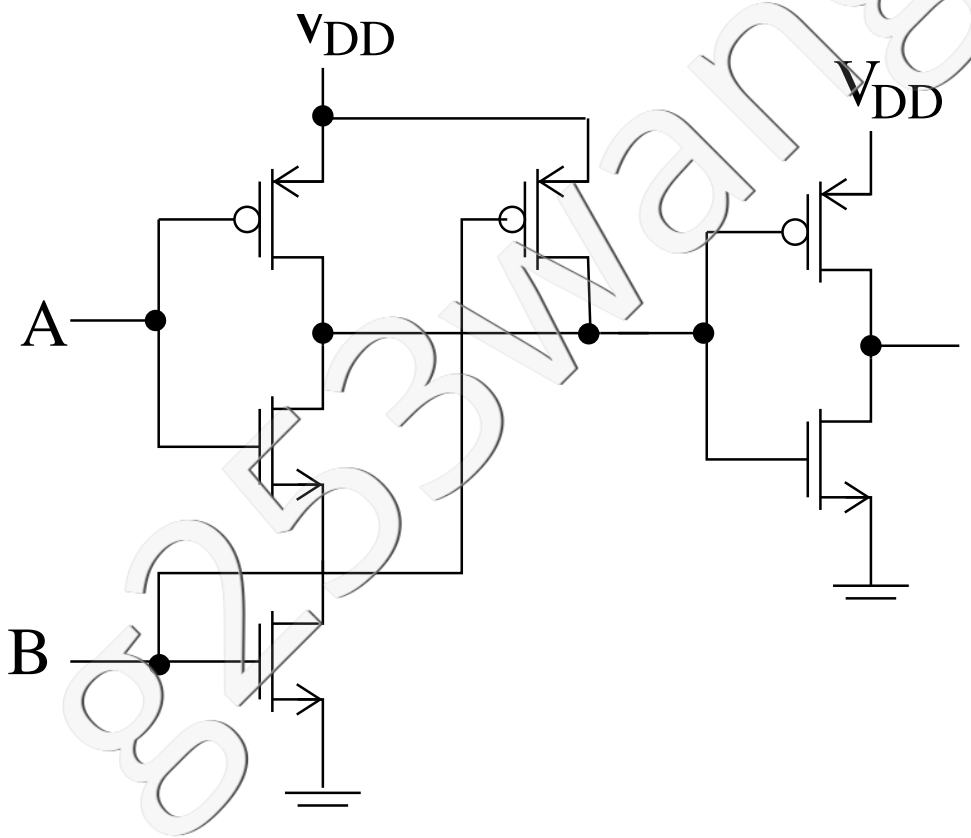
Solution: CMOS NOR



A	B	Q_1	Q_2	Q_3	Q_4	Z
0	0	Low	Low	High	High	1
0	1	Low	High	High	Low	0
1	0	High	Low	Low	High	0
1	1	High	High	Low	Low	0

CMOS AND

- To get AND and OR, add inverter at end
- Example:



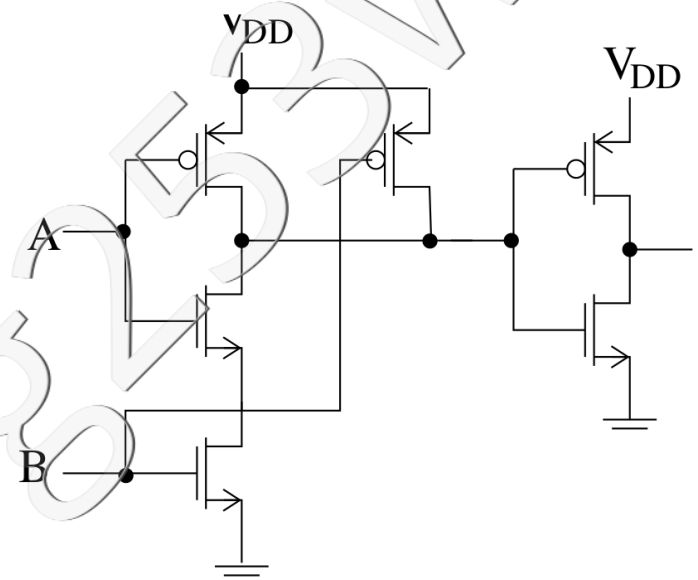
- Thus, NAND is preferred to AND in actual circuits

CMOS Strong AND Gate

Place a CMOS NOT gate after the CMOS NAND gate to get the AND gate.

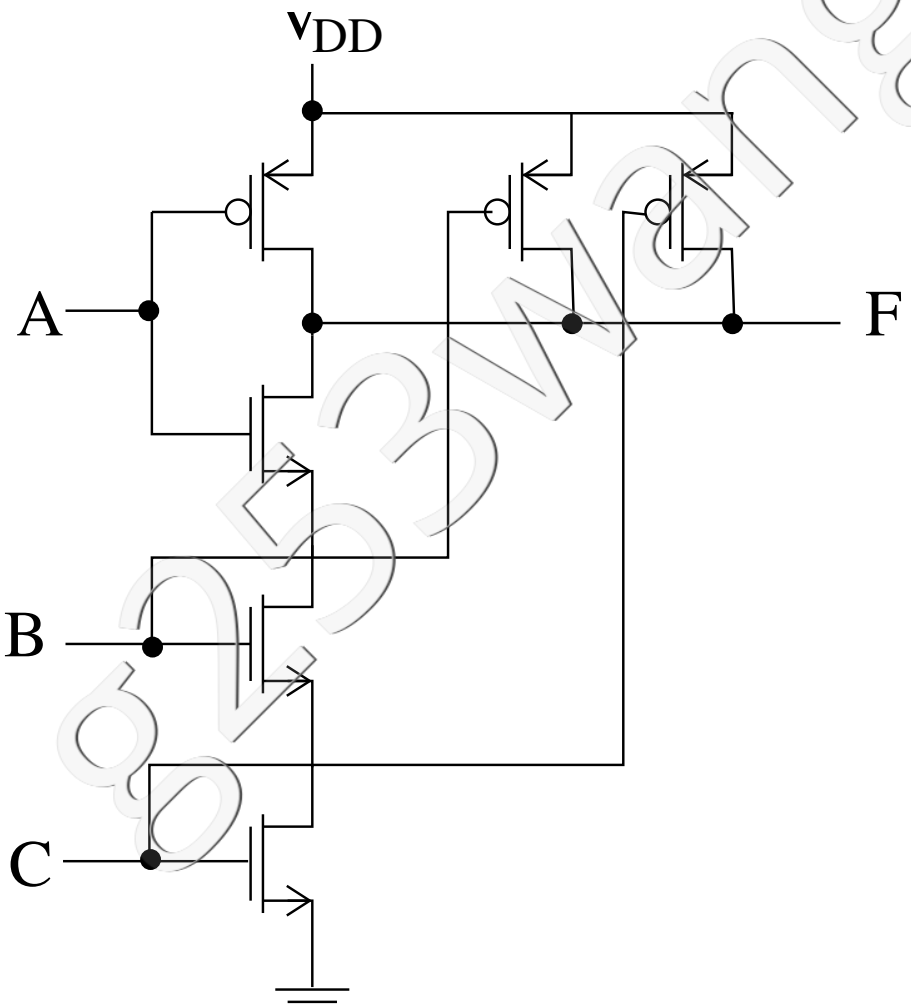
- The NAND gate needs 4 transistors.
- The NOT gate needs 2 transistors.
- The AND gate needs 6 transistors.

The NAND gate design is better than AND gate.



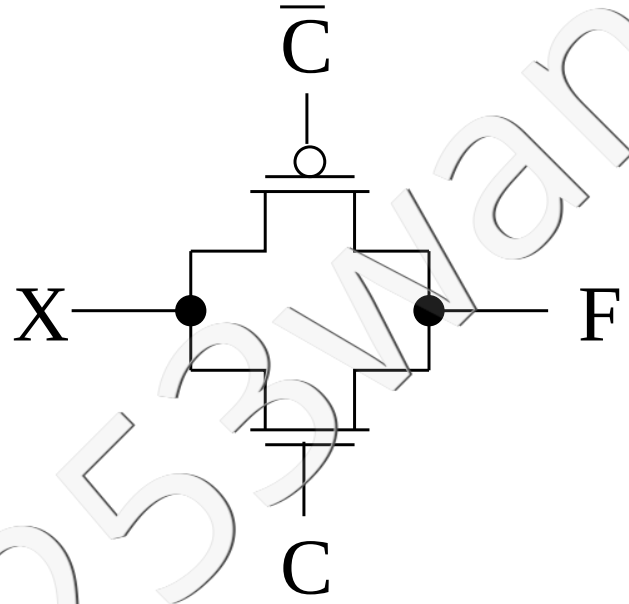
CMOS 3 Input NAND

- *n*-input NAND: 2 transistors per input
- Example: 3 Input NAND



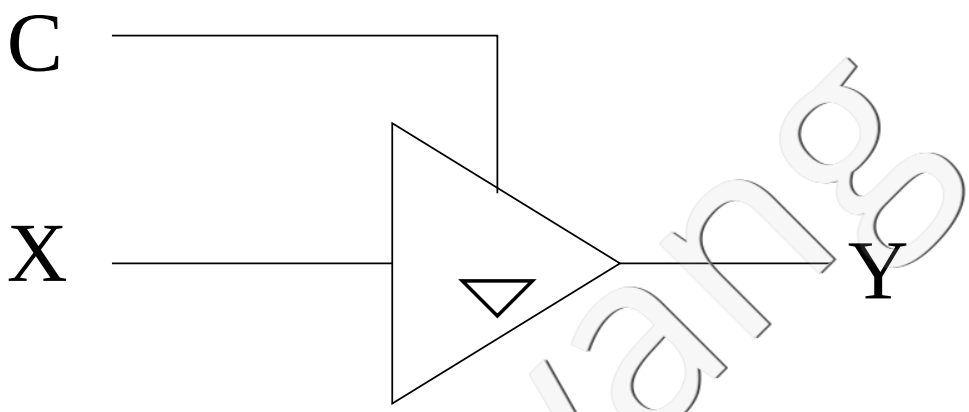
Three-state buffer Gate

- Has three outputs 0, 1, and *floating* (connected to neither power or ground)

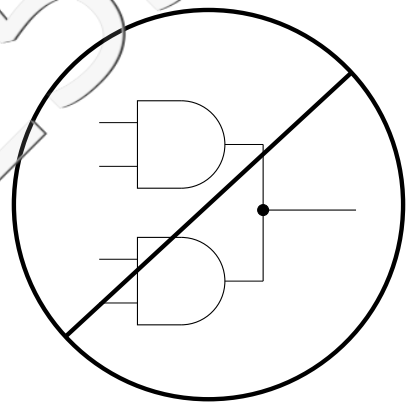


- $C = 1$, then
  - NMOS gate passes 0 well
  - $\bar{C} = 0$ and PMOS gate passes 1 well
- $C = 0$, then $\bar{C} = 1$ and both transistors are off (output is floating).

Using Three-State Buffers

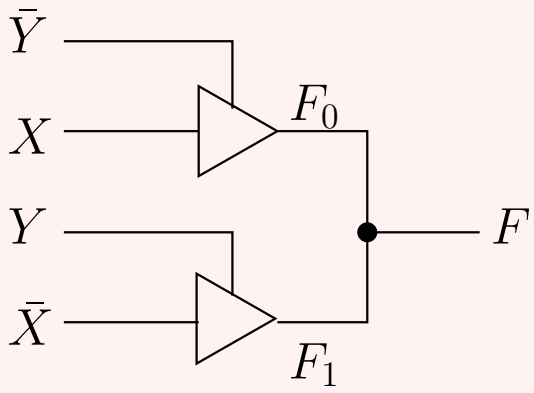


- High-impedance outputs can be "tied together" without problems
- Normally, do not tie output lines together



XOR from Three-State Buffers

Try this



Complete the truth table for the circuit illustrated.

X	Y	$\bar{X}$	$\bar{Y}$	F_0	F_1	F
0	0					
0	1					
1	0					
1	1					

Circuit analysis with tri-stage gates:

- Label floating output as '—' or Z
- Tied lines better have exactly one non-floating!

Solution: XOR from Three-state Buffers

In Summary:

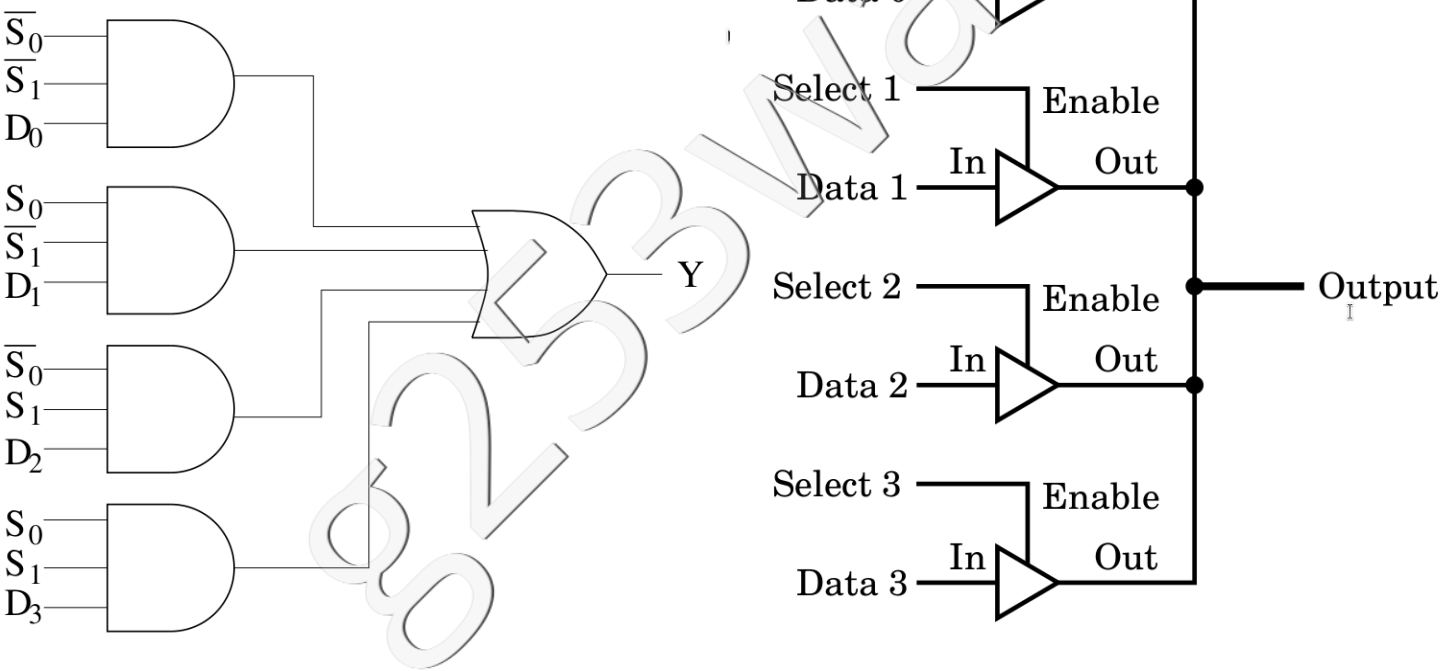
X	Y	$\overline{X}$	$\overline{Y}$	F_0	F_1	F
0	0	1	1	0	Z	0
0	1	1	0	Z	1	1
1	0	0	1	1	Z	1
1	1	0	0	Z	0	0

Transistor Count for XOR

- A 2-input XOR using CMOS NAND
  - needs 4 NAND gates
  - Each NAND is a 2 input NAND, which requires 4 transistors
  - TOTAL: $4 \times 4 = 16$ transistors.
- Three-state buffer XOR:
  - Three-state buffer: 4 transistors
  - TOTAL: 4 transistors

Tri-State Buffer and Mux

We can use the tri-state buffer to make a mux. Such a mux does not need a big OR gate. Exactly one select line is 1.



(a) 4-to-1 mux with OR gate. (b) Tri-state mux without OR gate.

CMOS Summary

- Terminology
  - Metal-Oxide-Semiconductor (MOS)
  - N-type Metal-Oxide-Semiconductor (NMOS)
  - P-type Metal-Oxide-Semiconductor (PMOS)
  - Complementary Metal Oxide Semiconductor (CMOS)
- Motivation
  - Assembly instruction $->$ Instruction Set
  - CMOS $->$ Logic Gates
  - Logic Gates $->$ Logic Circuits
- Key Ideas
  - A **short circuit** is a low resistance path between power and ground.
  - A **float state** is being disconnected from both power and ground.
  - For strong transmission, always connect nMOS to ground and pMOS to power.
  - Note that if we have a two-level circuit (OR of ANDs, or sum of products), we can replace each gate by NAND, and the resulting circuit computes the same function.
  - Therefore, we can continue to use AND-OR design.

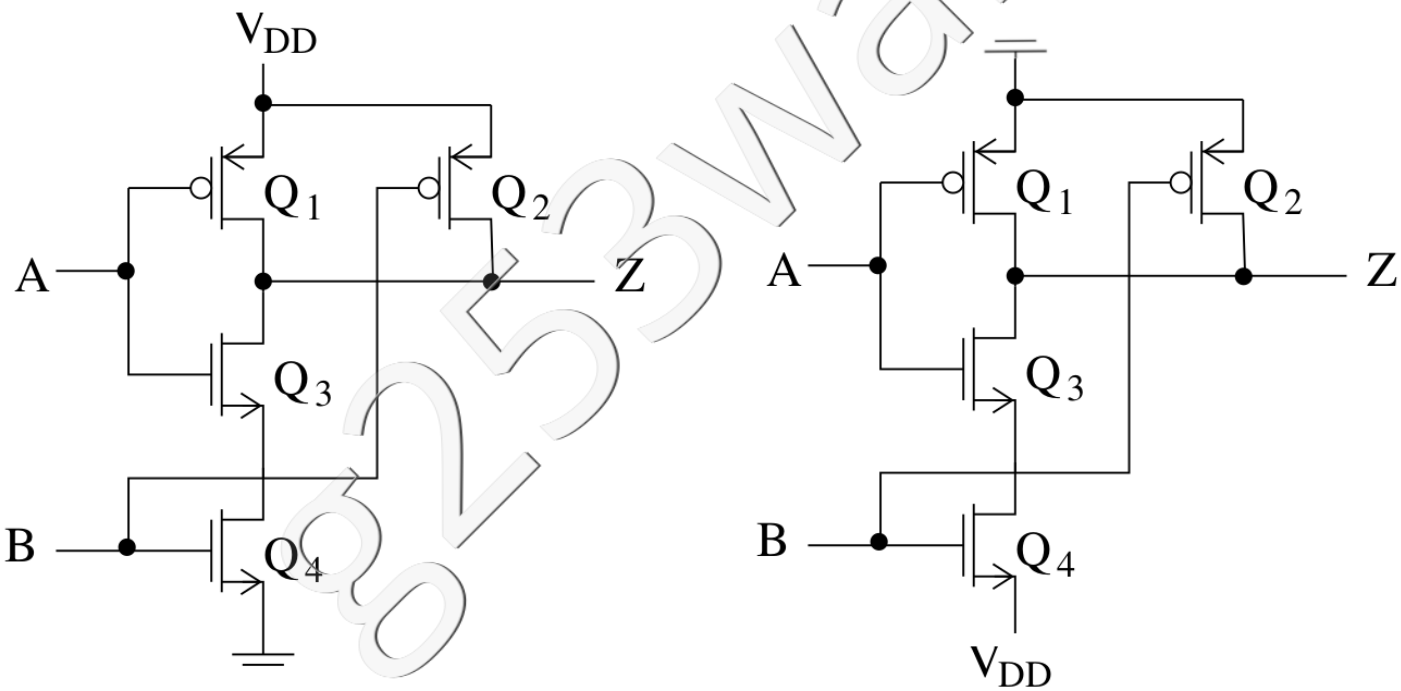
Additional Slides

Additional slides for curious minds

CMOS Weak AND Gate

In the NAND gate, if we swap power with ground we get the AND gate. We do **not** use this design. Example:

- When $A = 0$ and $B = 0$, Q_1 and Q_2 have low resistance. But since Q_1 and Q_2 are PMOS, they transmit weak 0s between Z and ground.

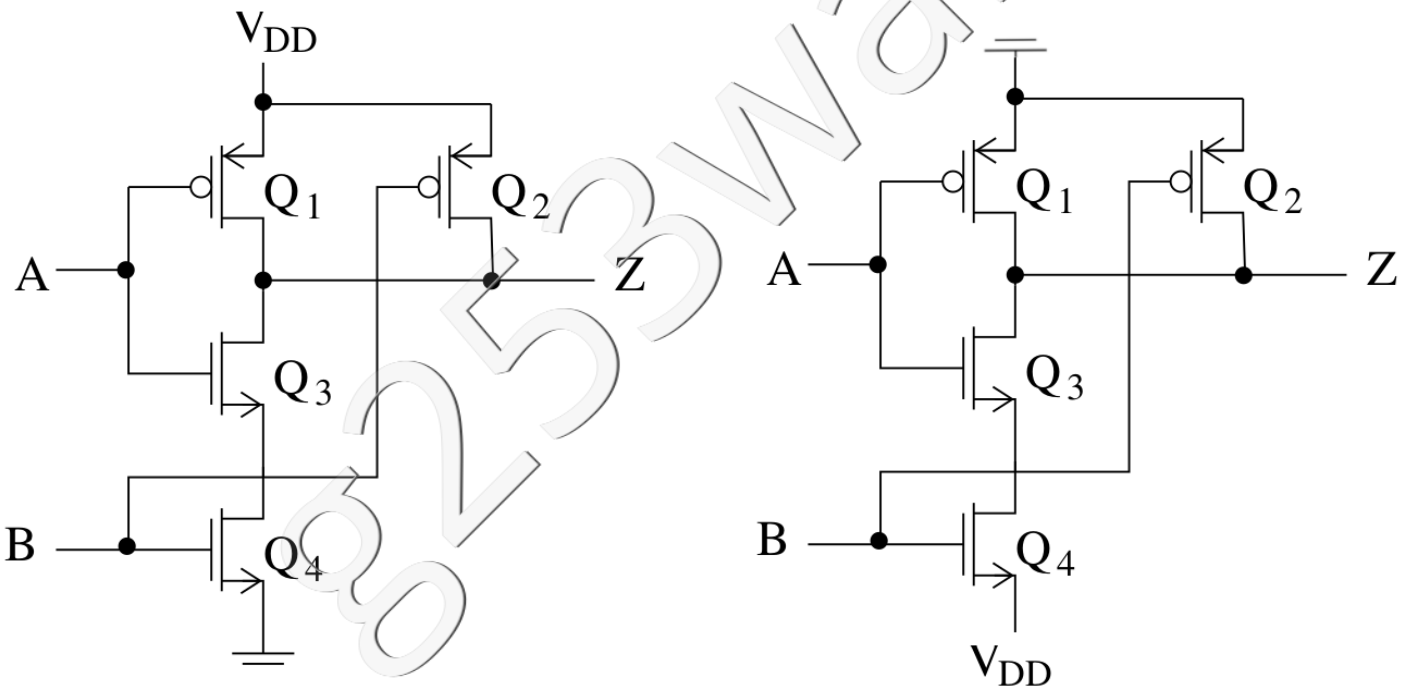


(c) NAND gate (d) Weak AND gate

CMOS Weak AND Gate

In the NAND gate, if we swap power with ground we get the AND gate. We do **not** use this design. Example:

- When $A = 1$ and $B = 1$, Q_3 and Q_4 have low resistance. But since Q_3 and Q_4 are NMOS they transmit weak 1s between power and Z.



(e) NAND gate

(f) Weak AND gate