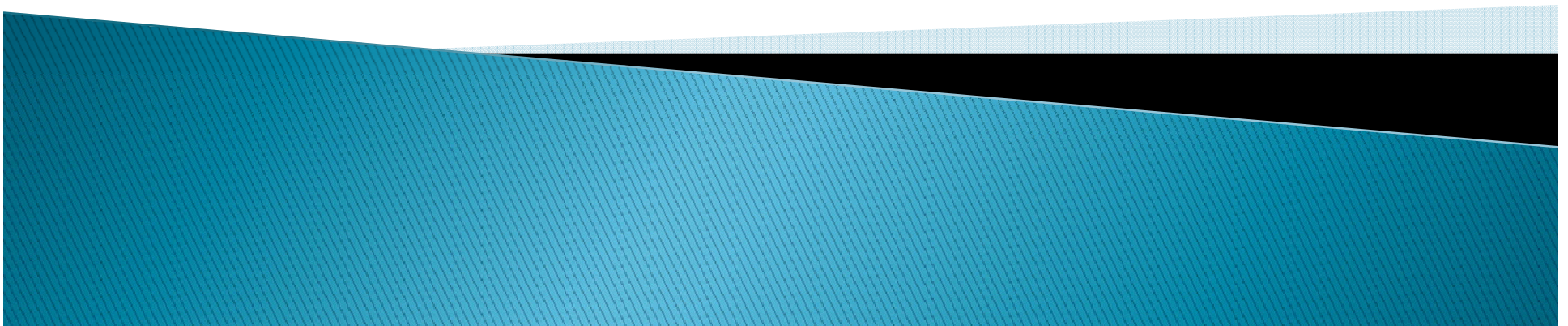


Digital Logic Structures + Assignment helps

CS 210 Tutorial 8



Introductions

- ▶ How do we represent data in a computer?
 - Easy to recognize two conditions:
 - 1. Presence of a voltage – we'll call this state "1"
 - 2. Absence of a voltage – we'll call this state "0"



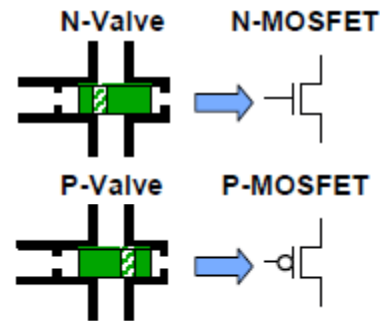
Computer use transistors as switches to manipulate bits

- Before transistors: tubes, electro-mechanical relays (pre 1950s)
- Mechanical adders (punch cards, gears) as far back as mid-1600s

Before describing transistors, we present an analogy...

▶ Transistor: Building Block of Computers

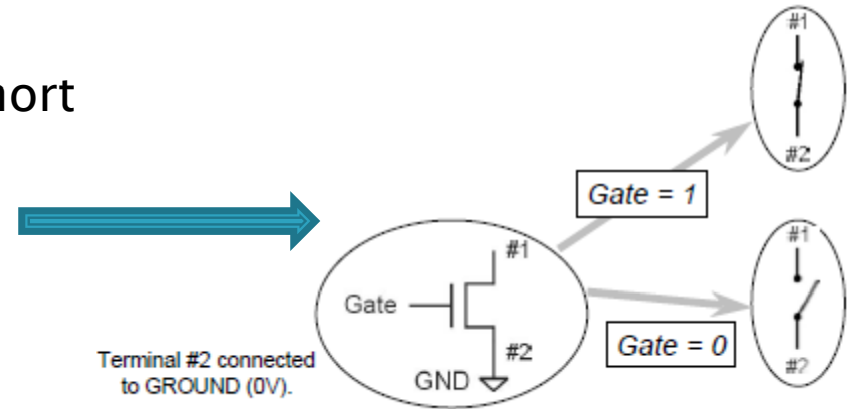
- Two types
 - N-type
 - P-type



Types of transistors

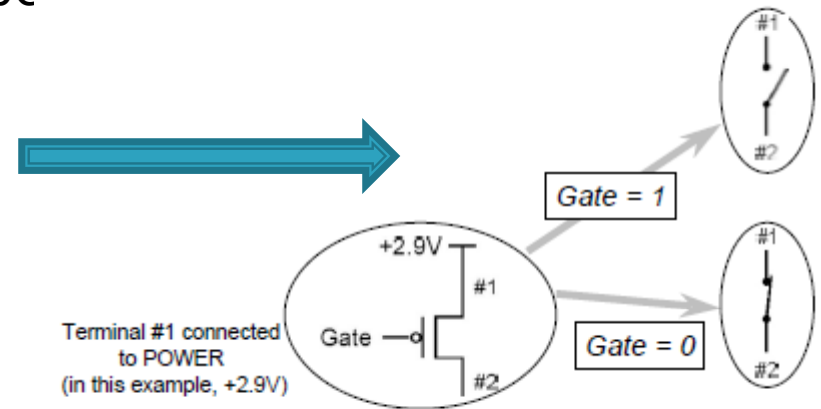
▶ N-type Transistor

- When Gate has positive voltage, short circuit between #1 and #2 (switch closed)
- When Gate has zero voltage, open circuit between #1 and #2 (switch open)



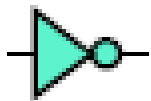
▶ P-type Transistor

- P-type is complementary to n-type
- When Gate has positive voltage, open circuit between #1 and #2 (switch open)
- When Gate has zero voltage, short circuit between #1 and #2 (switch closed)

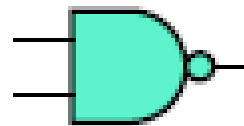


Basic Gates

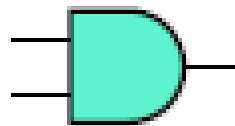
NOT/INV



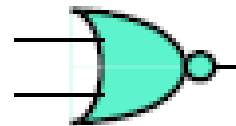
NAND



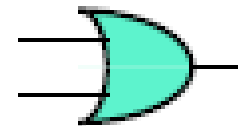
AND



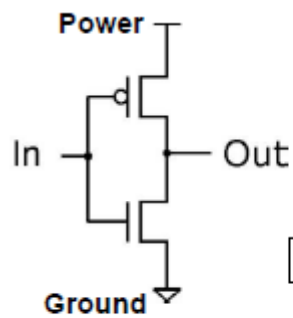
NOR



OR

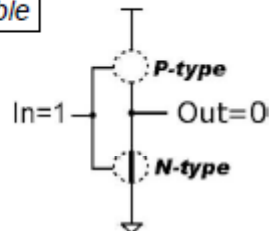
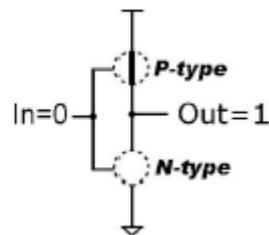


Inverter (NOT Gate)

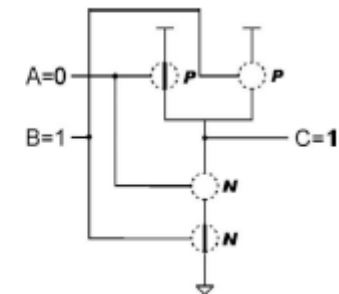
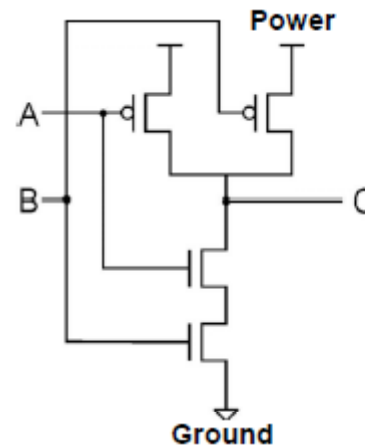


Truth table

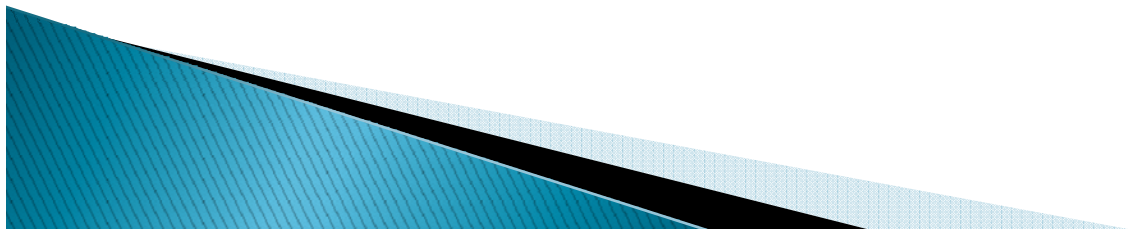
| In | Out |
|----|-----|
| 0 | 1 |
| 1 | 0 |



NAND Gate (NOT-AND)

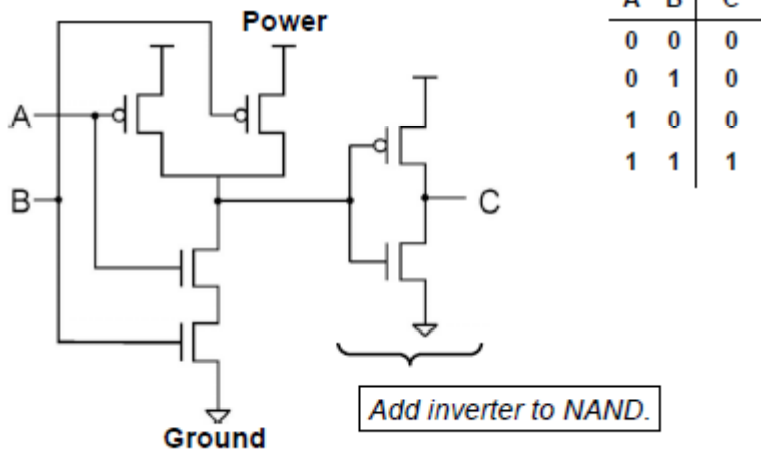


| A | B | C |
|---|---|---|
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

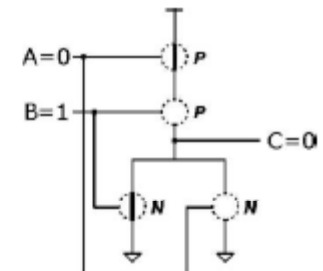
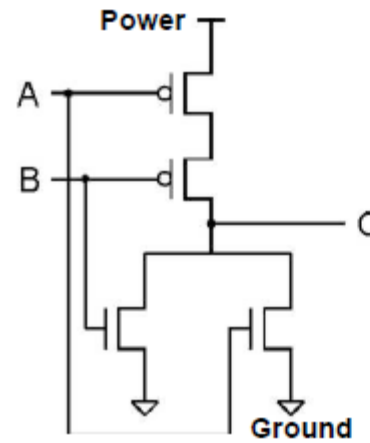


Basic Gates

AND Gate

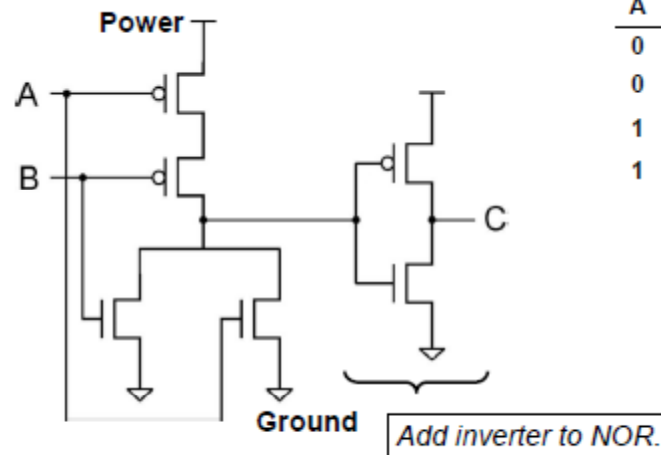


NOR Gate (NOT-OR)



| A | B | C |
|---|---|---|
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 0 |

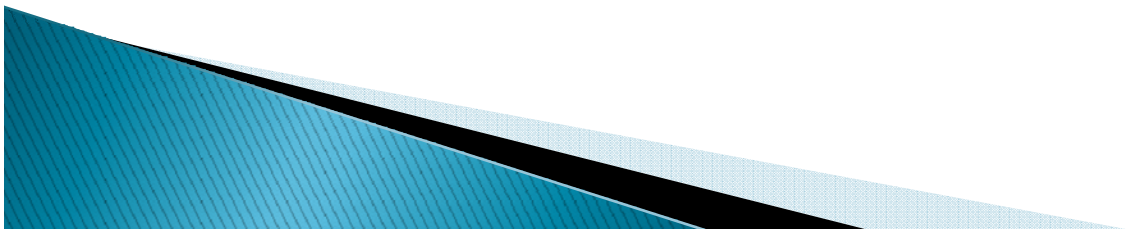
OR Gate



| A | B | C |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

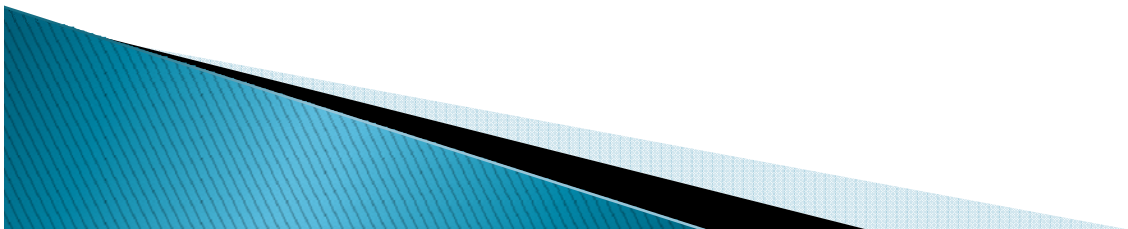
Exercise

- ▶ 1. Warm-up:
 - Write out the truth tables for AND, OR having 2-bit inputs
 - and XOR using 3-bit input
 - Hint for XOR: use the property of associativity; $A \oplus B \oplus C = A \oplus (B \oplus C)$



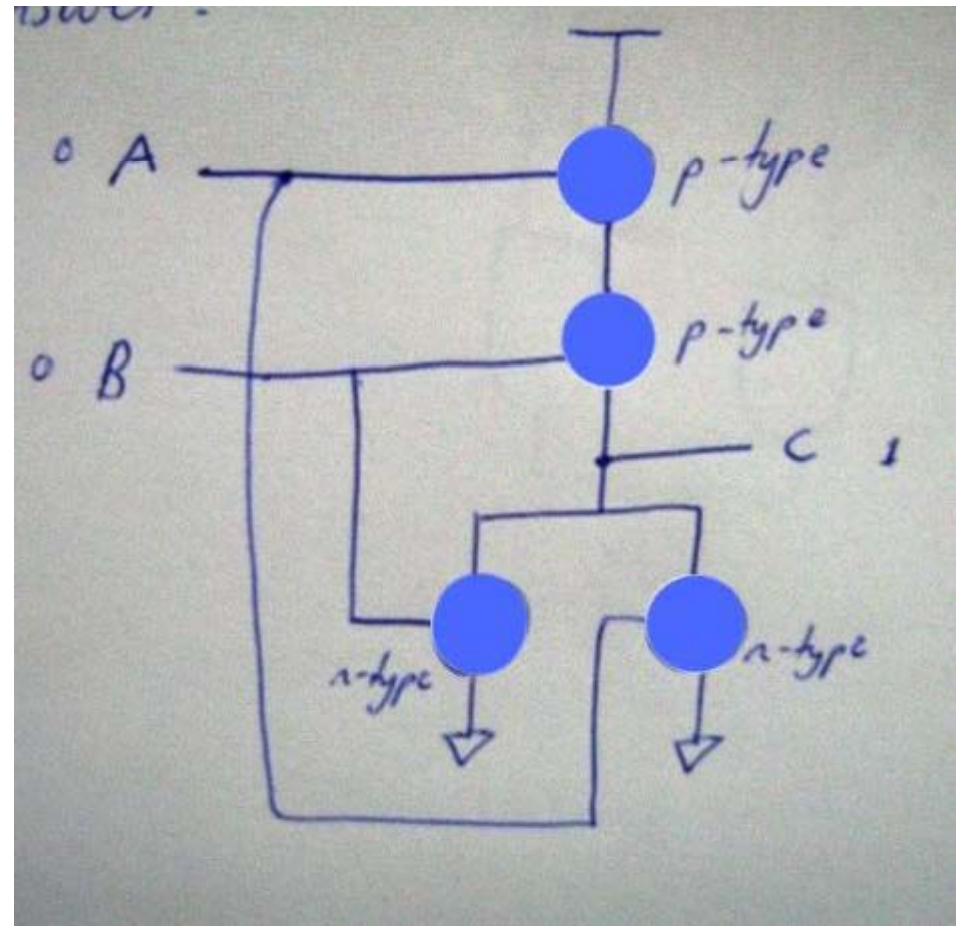
Answer for XOR:

| A | B | C | Z |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 |

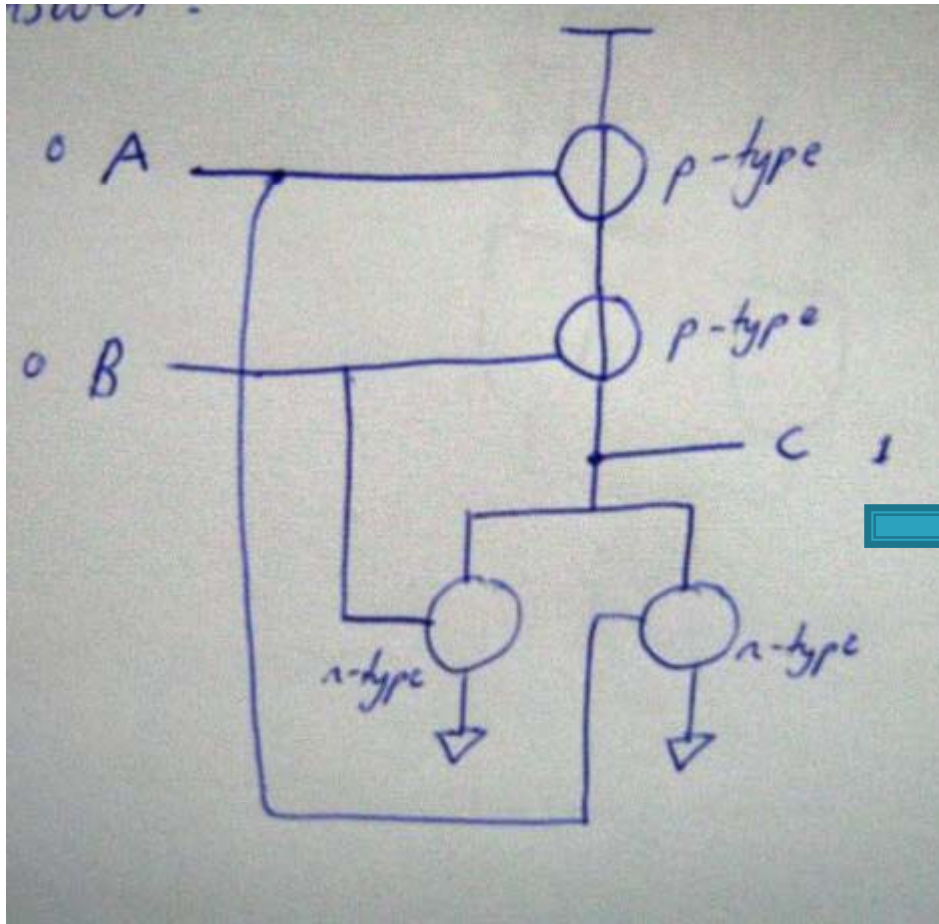


Exercise

- ▶ a) Replace the missing blue circles in the shown circuit with either a wire or no wire to give the output C a logical value of 1.
- ▶ b) Describe a set of inputs that give the output C a logical value of 0. I.e. write out the truth table.
- ▶ c) What type of gate does this circuit represent?



Answer



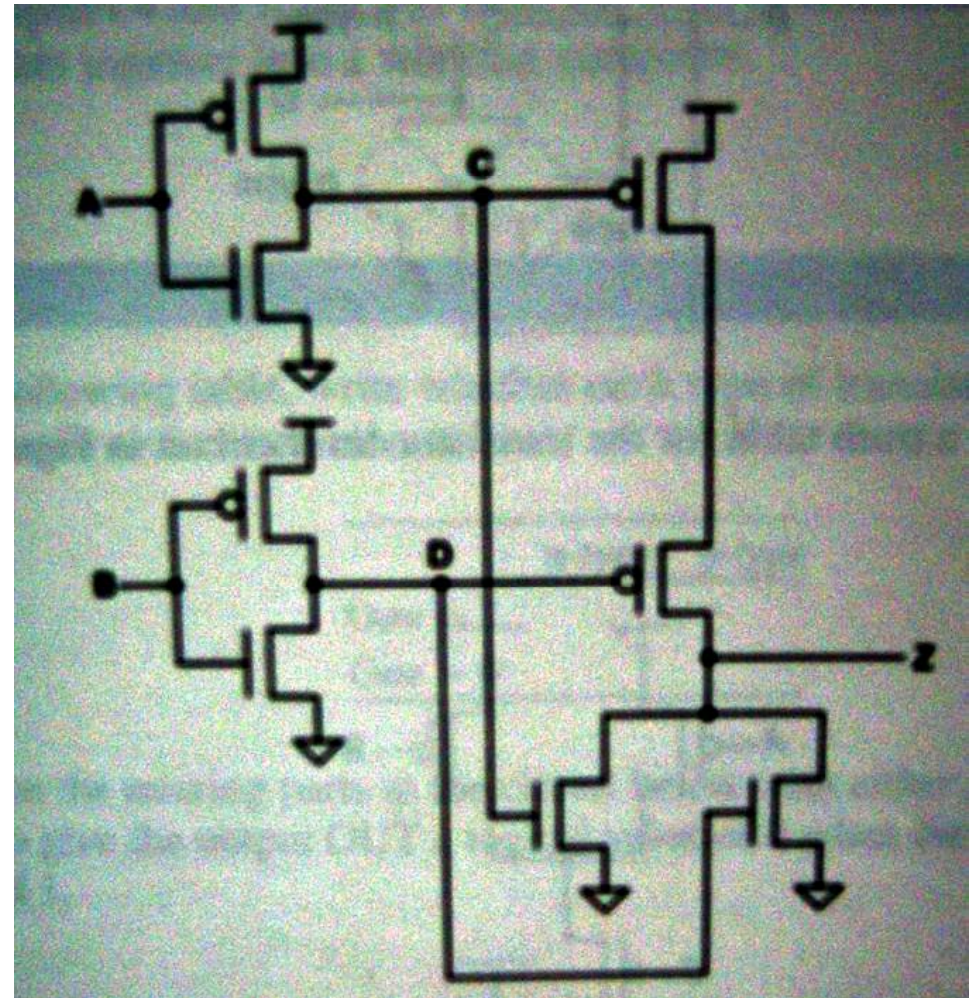
| A | B | C |
|---|---|---|
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 0 |

- ▶ Therefore it represents a NOR gate

Exercise

- ▶ For the transistor-level circuit in the following Figure, fill in the truth table. What is Z in terms of A and B?

| A | B | C | D | Z |
|---|---|---|---|---|
| | | | | |
| | | | | |
| | | | | |
| | | | | |

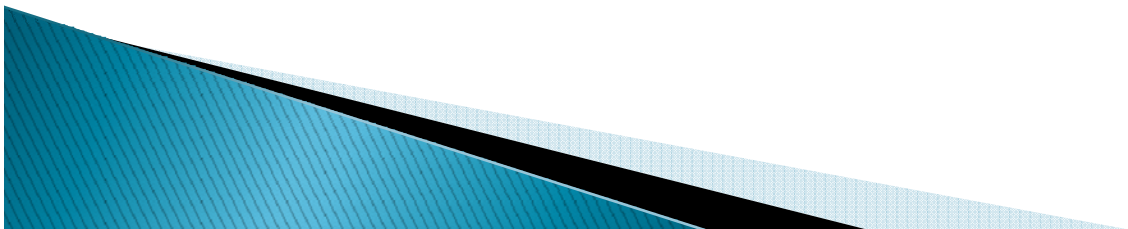


Answer

- ▶ Truth table:

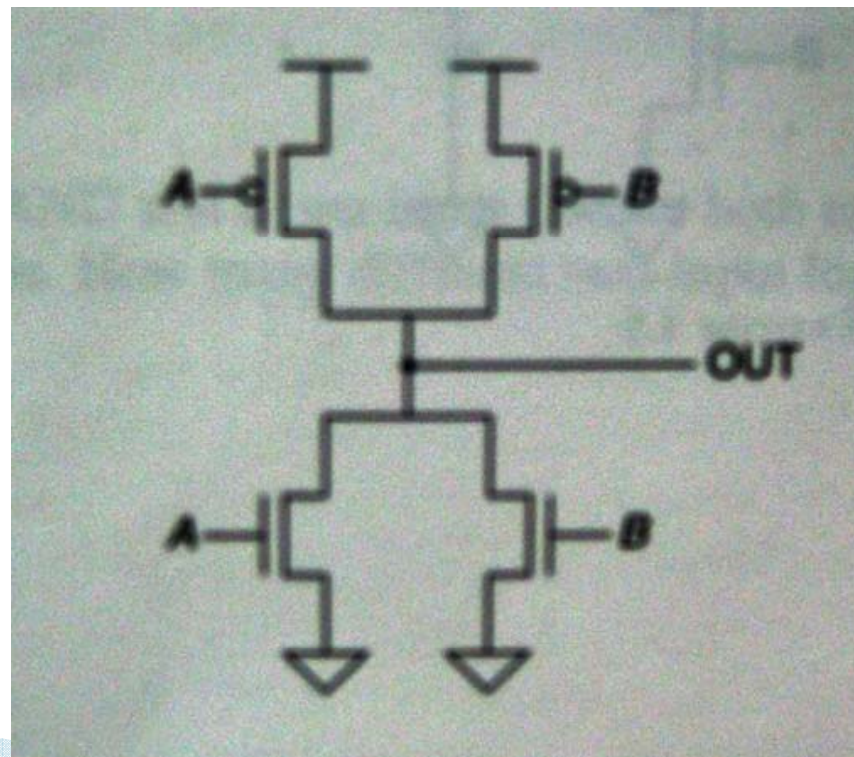
| A | B | C | D | Z |
|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 1 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 | 1 |

- ▶ Therefore $Z = A \text{ AND } B$



Exercise

- ▶ What is the flaw in the following circuit? Analyse it by working through the truth table and seeing which inputs show up the flaw.

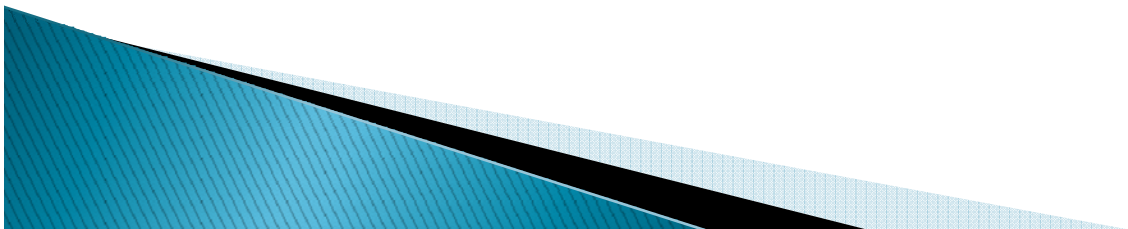


Answer

- ▶ The circuit will short-circuit for certain inputs.

| A | B | Output |
|---|---|--------------------|
| 0 | 0 | 1 |
| 0 | 1 | 1 (short circuit!) |
| 1 | 0 | 1 (short circuit!) |
| 1 | 1 | 0 |

- ▶ Why?



Finish off assignment helps

- ▶ Have you not finished your assignment 2?
- ▶ You still have questions remaining on your assignment...

