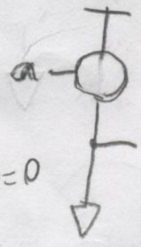
Switches, Transistors, and Logic Gates

CS 350: Computer Organization & Assembler Language Programming

Due Tue Nov 22

A. Why?

- Transistors (electronic switches) operate on binary data represented by voltages.
- Logic gates are the lowest level of hardware that deal with logical values.



B. Outcomes

After this lab, you should be able to

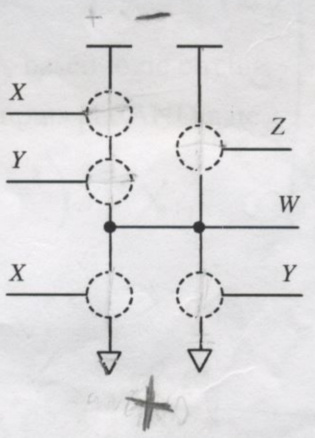
- Be able to read and write simple voltage/current diagrams, transistor-level diagrams, and logic gates-level diagrams.
- Convert between truth tables, logical formulas, and loop-free logic circuits.

C. Problems [50 points total]

1

X	Y	Z	W
0	0	0	1
0	0	1	1
0	1	0	Short
0	1	1	0
1	0	0	short
1	0	1	0
1	1	0	Short
1	1	1	0

1. [16 pts] Study the voltage diagram shown here. Give a table that maps shows (for each configuration of inputs *X*, *Y*, and *Z*): Is power is connected to output *W*? Is *W* is connected to ground? What's the status of the circuit? ($W = 0$ or 1 or we have a short or open circuit?) See the table at the top of page 7 of Lecture 15 for an example.



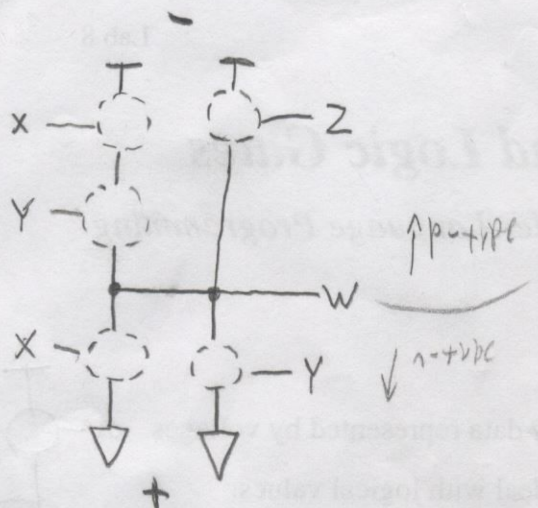
↑ p-type

↓ n-type

2. [6 pts] Modify the voltage diagram for Problem 1 to get a proper CMOS circuit that calculates $W = \overline{X}\ \overline{Y} + \overline{Z}$. Don't modify the p-type transistors. Instead, add / remove / rewrite n-type transistors for *X*, *Y*, and *Z* as necessary.

see back

3. [13 pts] Draw a transistor-level circuit for $Z = \overline{A}\,(\overline{B} + \overline{C})\,\overline{D}$. Be sure to (a) use p-type transistors to connect output *Z* to power iff $\overline{A}\,(\overline{B} + \overline{C})\,\overline{D}$ is true and (b) use n-type transistors to connect *Z* to ground iff $\overline{A}\,(\overline{B} + \overline{C})\,\overline{D}$ is false.



([illegible] p-type)

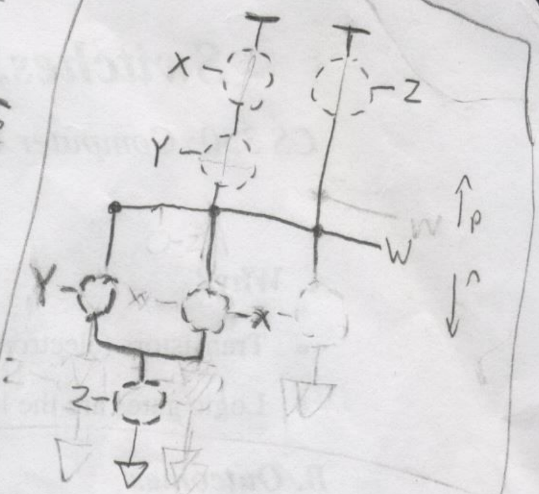
X	Y	Z	W
0	0	0	1
0	0	1	1
0	1	0	Short
0	1	1	0
1	0	0	Short
1	0	1	0
1	1	0	Short
1	1	1	0

2) $W = \bar{X} \cdot \bar{Y} + \bar{Z}$

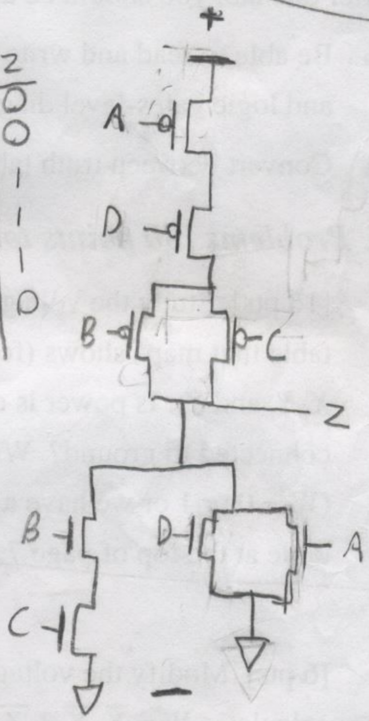
aka

$(\bar{X} \wedge \bar{Y}) \vee \bar{Z}$

X	Y	Z	W
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0



a	b	c	d	z
1	?	?	?	0
?	?	?	1	0
0	0	1	0	1
✓ 0	1	0	0	1
✓ 0	0	0	0	1
✓ 0	1	1	0	0

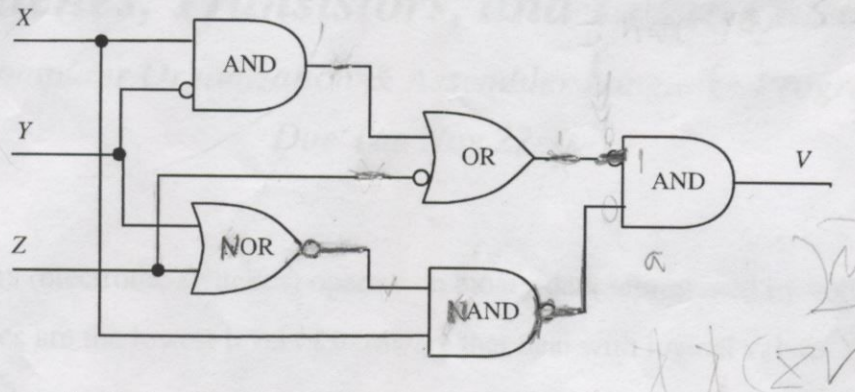


3) $Z = \bar{A}(\bar{B} + \bar{C})\bar{D}$

use p-type to connect Z → − iff Z == 1

use n-type to connect Z → + iff Z == 0

4. [15 = 3*5 pts] Consider the circuit below with inputs *X*, *Y*, *Z* and output *V*.



(a) Translate the logic circuit to get a boolean expression definition for *V*. Give the expression as is (don't manipulate it and give something logically equivalent). Notation: You can use ⊗ and ⊕ for *NAND* and *NOR* (or write them as the *NOT* of an *AND* or *OR*; your choice). You can also use juxtapositioning, ∧, or * for *AND* and ∨ or + for *OR* (your choice).

(b) Give a truth table for *V* (either style of table is okay).

(c) Give the full DNF definition for *V*. Use it to draw a PLA-based logic circuit for *V*. (Feel free to drop unused AND gates, but keep 3 inputs per AND gate.)

$\neg((X\wedge\overline{Y})\vee\overline{Z})\wedge\neg(\neg(Z\overline{\vee}Y)\overline{\wedge}X)$

X	Y	Z	V
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

$\neg((X\wedge\overline{Y})\vee\overline{Z})\wedge\neg((Z\vee Y)\overline{\wedge}X)$

$\ldots\wedge\cancel{\neg}\neg((Z\vee Y)\wedge X)$

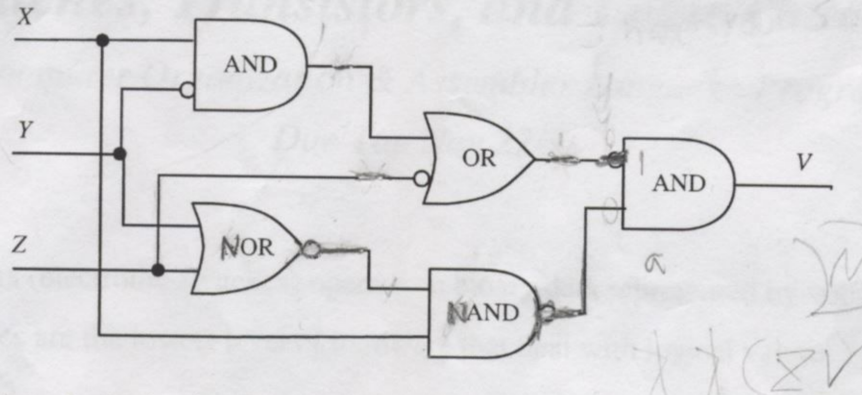
$\neg\neg(Z\vee Y) = Z\vee Y$

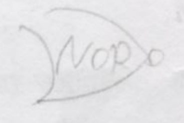
$\boxed{((X\wedge\overline{Y})\vee\overline{Z})\wedge(X\overline{\wedge}(Z\overline{\vee}Y))}$

$\neg((X\wedge\overline{Y})\vee\overline{Z})\wedge((Z\vee Y)\overline{\wedge}X)$

c) $(X\wedge\overline{Y}\wedge Z)\vee(Y\wedge\overline{Z})\vee(\overline{Z}\wedge\overline{X})$

4. [15 = 3*5 pts] Consider the circuit below with inputs *X*, *Y*, *Z* and output *V*.





$\neg(Z \vee Y)$

(a) Translate the logic circuit to get a boolean expression definition for *V*. Give the expression as is (don't manipulate it and give something logically equivalent). Notation: You can use ⊗ and ⊕ for *NAND* and *NOR* (or write them as the *NOT* of an *AND* or *OR*; your choice). You can also use juxtapositioning, ∧, or * for *AND* and ∨ or + for *OR* (your choice).

(b) Give a truth table for *V* (either style of table is okay).

(c) Give the full DNF definition for *V*. Use it to draw a PLA-based logic circuit for *V*. (Feel free to drop unused AND gates, but keep 3 inputs per AND gate.)

$$\neg((X \wedge \overline{Y}) \vee \overline{Z}) \wedge \neg(\neg(Z \mathbin{\overline{\vee}} Y) \mathbin{\overline{\wedge}} X)$$

X	Y	Z	V
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

$$\neg((X \wedge \overline{Y}) \vee \overline{Z}) \wedge \neg((Z \vee Y) \mathbin{\overline{\wedge}} X)$$

$$\ldots \wedge \cancel{\neg}\neg((Z \vee Y) \wedge X)$$

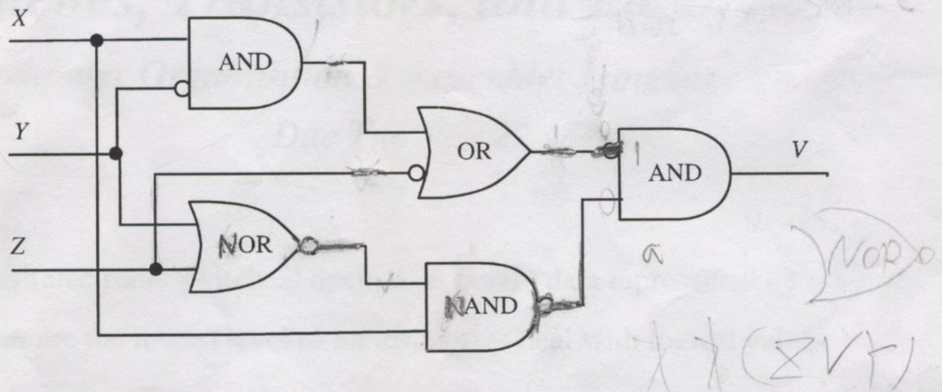
$$\neg\neg(Z \vee Y) = Z \vee Y$$

$$\boxed{((X \wedge \overline{Y}) \vee \overline{Z}) \wedge (X \mathbin{\overline{\wedge}} (Z \mathbin{\overline{\vee}} Y))}$$

$$\neg((X \wedge \overline{Y}) \vee \overline{Z}) \wedge ((Z \vee Y) \mathbin{\overline{\wedge}} X)$$

c) $(X \wedge \overline{Y} \wedge Z) \vee (Y \wedge \overline{Z}) \vee (\overline{Z} \wedge \overline{X})$

4. [15 = 3*5 pts] Consider the circuit below with inputs *X*, *Y*, *Z* and output *V*.



(a) Translate the logic circuit to get a boolean expression definition for *V*. Give the expression as is (don't manipulate it and give something logically equivalent). Notation: You can use ⊗ and ⊕ for *NAND* and *NOR* (or write them as the *NOT* of an *AND* or *OR*; your choice). You can also use juxtapositioning, ∧, or * for *AND* and ∨ or + for *OR* (your choice).

(b) Give a truth table for *V* (either style of table is okay).

(c) Give the full DNF definition for *V*. Use it to draw a PLA-based logic circuit for *V*. (Feel free to drop unused AND gates, but keep 3 inputs per AND gate.)

$\neg((X \wedge \bar{Y}) \vee \bar{Z}) \wedge \neg(\neg(Z \bar{\vee} Y) \bar{\wedge} X)$

X	Y	Z	V
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

$\neg((X \wedge \bar{Y}) \vee \bar{Z}) \wedge \neg((Z \vee Y) \bar{\wedge} X)$

$\ldots \wedge \cancel{\neg}\neg((Z \vee Y) \wedge X)$

$\neg\neg(Z \vee Y)$

$=$

$Z \vee Y$

$\boxed{((X \wedge \bar{Y}) \vee \bar{Z}) \wedge (X \bar{\wedge} (Z \bar{\vee} Y))}$

$\neg((X \wedge \bar{Y}) \vee \bar{Z}) \wedge ((Z \vee Y) \bar{\wedge} X)$

c) $(X \wedge \bar{Y} \wedge Z) \vee (Y \wedge \bar{Z}) \vee (\bar{Z} \wedge \bar{X})$