

**Instructions:**

- Print your name in the space provided below.
- This examination is closed book and closed notes, aside from the permitted one-page formula sheet. No calculators or other computing devices may be used.
- Answer each question in the space provided. If you need to continue an answer onto the back of a page, clearly indicate that and label the continuation with the question number.
- If you want partial credit, justify your answers, even when justification is not explicitly required.
- There are 5 questions, priced as marked. The maximum score is 100.
- When you have completed the test, sign the pledge at the bottom of this page and turn in the test.
- Note that either failing to return this test, or discussing its content with a student who has not taken it is a violation of the Honor Code.

Do not start the test until instructed to do so!

Name _____
printed

Pledge: On my honor, I have neither given nor received unauthorized aid on this examination.

signed

1. Consider the following integer X, represented in 32-bit 2's complement form:

1001 0000 1001 0000 1001 0000 1001 0000

- a) [10 points] What is the representation for $-X$?

0110 1111 0110 1111 0110 1111 0111 0000

- b) [10 points] Would it be more efficient in hardware to find the negative of an integer if the integer were represented in sign-magnitude form or in 2's complement? Why? Be precise.

Negating in sign-magnitude requires inverting a single bit. Negating in 2's complement requires inverting all the bits and then adding 1, OR finding the rightmost 1 and inverting all the bits to its left.

Either way, more time would be required to negate a 2's complement integer, since you either have to perform an addition operation, or you have to actually determine where the rightmost 1 is.

It's also clear that either way, more circuitry would be required to negate a 2's complement integer.

-
2. [10 points] Suppose an adder for 32-bit 2's complement integers was implemented using the ripple-carry approach discussed in class. Assume that the gate delay for AND and OR gates is t , and that the gate delay for an inverter is negligible. In terms of t , what delay would occur between setting the inputs to the adder and the sum bits of the output becoming stable on the correct result? You may neglect wire delays. Justify your conclusion.

There would be a separate "three-fourths" adder circuit for each bit; each would require two levels of gates (ignoring the inverters).

The adder for bit j would not receive the correct carry-in value until the adder for bit $j-1$ has stabilized, so the adder for bit j could not produce a stable, definitely correct output in less than the time for $2j$ gate delays.

So, the final sum could not be guaranteed in less than $64t$ time units.

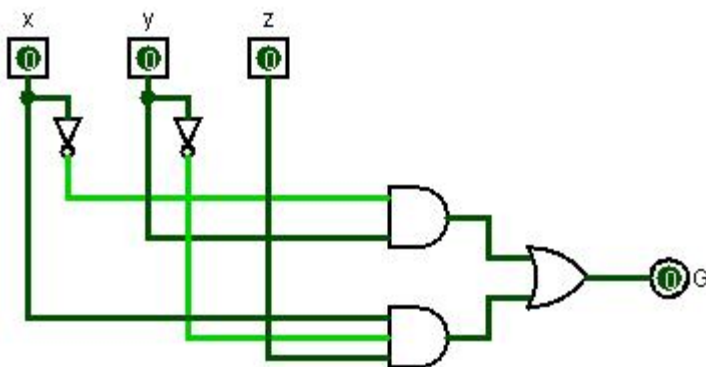
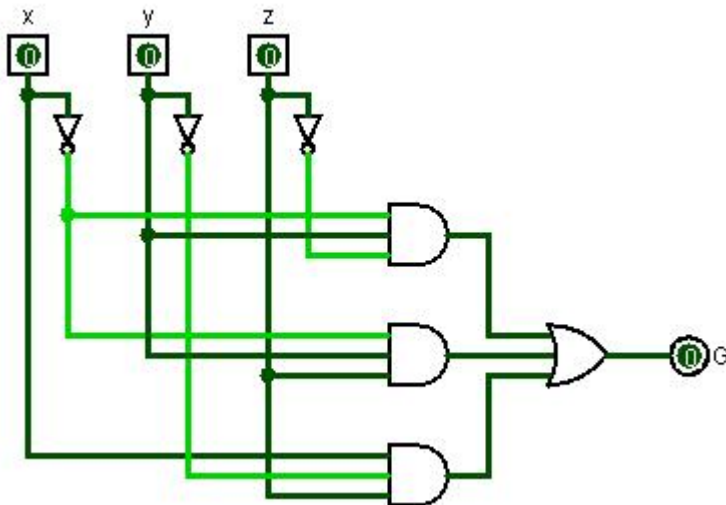
3. The following table defines a Boolean function G:

x	y	z	G
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	0

a) [15 points] Write the sum-of-products formula for the function G.

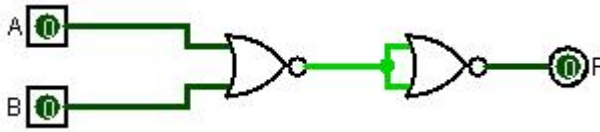
$$\begin{aligned}
 G &= \bar{x} \cdot y \cdot \bar{z} + \bar{x} \cdot y \cdot z + x \cdot \bar{y} \cdot z \\
 &= \bar{x} \cdot y \cdot (\bar{z} + z) + x \cdot \bar{y} \cdot z \\
 &= \bar{x} \cdot y + x \cdot \bar{y} \cdot z
 \end{aligned}$$

b) [10 points] Draw the corresponding circuit to compute G. (You may simplify the expression for G or not, as you prefer.)



4. [20 points] For each circuit below, give the simplest Boolean expression for the output function computed by the circuit. Justify your answers.

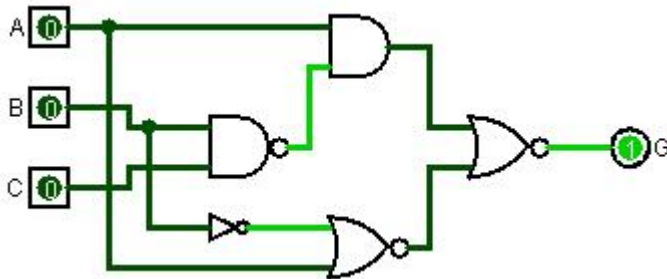
a) [10 points]



$$\begin{aligned} F &= \overline{\overline{A+B} + \overline{A+B}} \\ &= \overline{\overline{A+B}} \\ &= A+B \end{aligned}$$

So, the circuit implements the OR function using only NOR gates.

b) [15 points]



$$\begin{aligned} G &= A \cdot (\overline{B \cdot C}) + \overline{\overline{B}} + A \\ &= A \cdot (\overline{B \cdot C}) \cdot \overline{\overline{B}} + A \\ &= (\overline{A} + \overline{\overline{B \cdot C}}) \cdot (\overline{B} + A) \\ &= (\overline{A} + B \cdot C) \cdot (A + \overline{B}) \\ &= \overline{A} \cdot A + \overline{A} \cdot \overline{B} + A \cdot B \cdot C + B \cdot \overline{B} \cdot C \\ &= 0 + \overline{A} \cdot \overline{B} + A \cdot B \cdot C + 0 \cdot C \\ &= \overline{A} \cdot \overline{B} + A \cdot B \cdot C \end{aligned}$$

5. [20 points] Use the algebra of Boolean expressions to prove the Boolean equation given below is correct. You may use any of the rules given on the supplement to this test. If you want to use any other rules, you must prove them as well. You must justify each step by referring to the rule used; it's acceptable to combine steps but you must list every rule that is used.

$$a \cdot b = a \cdot \overline{\overline{b}}$$

$$\begin{aligned} a \cdot \overline{\overline{b}} &= a \cdot (\overline{\overline{a + b}}) && \text{DeMorgan's Law} \\ &= a \cdot (\overline{\overline{a}} + \overline{\overline{b}}) && \text{Double-negation Law} \\ &= a \cdot \overline{\overline{a}} + a \cdot \overline{\overline{b}} && \text{Distributive Law} \\ &= 0 + a \cdot b && \text{Complements Law} \\ &= a \cdot b && \text{Boundedness Law} \end{aligned}$$