Max Score = 15 points

CS 250 2018 Spring Homework 01 SOLUTION & GRADING GUIDE

This assignment is due at 11:59:00 pm Thursday, January 18, 2018.

Upload your typewritten answer document in either PDF or Word format to Blackboard.
Download from Blackboard to be sure that your upload was successful.
Your last upload that is not marked "LATE" by Blackboard is the upload that will be graded.
There is no "grace" period for late uploads

ADVICE: Upload your solution sufficiently before the deadline to avoid internet congestion issues and server not responding issues.

The policy for all homework assignments this semester is as follows. Please sign, which you may do by typing in your name on the signature line.

In the following I have not represented the work of another person as my own nor have I knowingly or actively assist another person in violating this standard.

(Signed)_____

GRADE: If signature is not present; email gba@purdue.edu with student name.

GRADING SCALE that will be used for homework questions:
0 points earned means no answer given or no meaningful progress made.
1 points earned means a small amount of progress towards an answer is shown.
2 points earned means significant progress towards an answer, but work falls short or ends in the wrong place.
3 points earned (maximum value) means a correct answer or very close to the correct answer.

1. **GRADE [3 pts; 1 point each for correctly defined Zero, Identity, and One functions, do not deduct points for choice of function name.]** Create a single truth table that defines all possible one-input Boolean functions. At the appropriate place in the truth table give your best idea for the name of each function.
   Answer:

Input value	Function name: Zero	Function name: Identity	Function name: Not	Function name: One
0	0	0	1	1
1	0	1	0	1

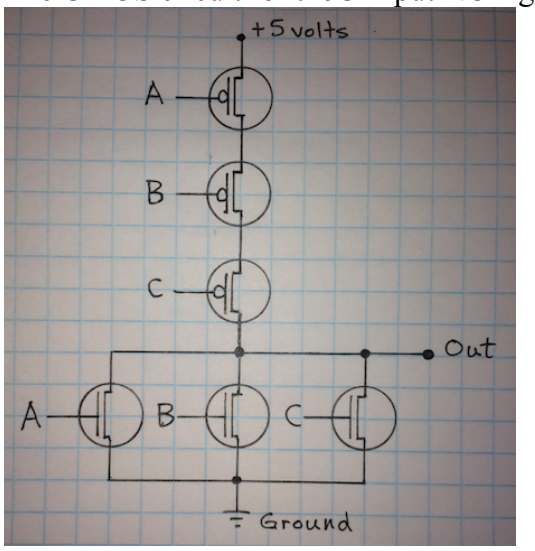
Explanation. With one Boolean input there are two distinct input values: 0 and 1. For each input value there are two choices for defining the result of the function: 0 and 1. Therefore, the number of distinct Boolean function truth tables corresponding to the input possibilities 0 and 1 is $2*2 = 2^2 = 4$. The general formula for the number of distinct k-input Boolean

functions is 2^{2^k}.

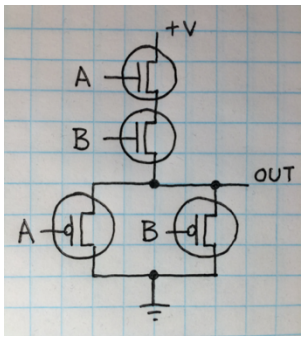
2. **GRADE [3 pts; 1 point for correct truth table, 2 points for correct circuit between Output and +V, 1 point for correct circuit between Output and Ground.]** What is the truth table for the three-input NOR(A,B,C) function? Draw the circuit for the 3-input NOR gate at the level of abstraction of the CMOS transistor. You may hand draw circuit schematics and then scan them into your document here.
Answer: The truth table for 3-input NOR and an explanation of how the circuit will work is:

A	B	C	(A+B+C)'	Conduction path from reference voltage to Out is
0	0	0	1	+V through transistors turned on when all of A, B, and C are 0
0	0	1	0	Ground through transistor turned on by C=1
0	1	0	0	Ground through transistor turned on by B=1
0	1	1	0	Ground through transistors turned on by B and C equal 1
1	0	0	0	Ground through transistor turned on by A=1
1	0	1	0	Ground through transistors turned on by A and C equal 1
1	1	0	0	Ground through transistors turned on by A and B equal 1
1	1	1	0	Ground through transistors turned on by A, B, and C equal 1

The CMOS circuit for the 3-input NOR gate is



3. **GRADE [3 pts for correct schematic]** Using exactly 4 CMOS transistors, design and then draw a schematic for an AND circuit. Comment on the relationship you see between the NAND circuit presented in class and our textbook and the AND circuit that you develop.
Answer: AND circuit looks like this



Explanation: The truth table for AND is

Row number	A	B	(A+B)'
0	0	0	0
1	0	1	0
2	1	0	0
3	1	1	1

The only time a 2-input AND gate should output a high voltage value (a logical 1) is when *both* inputs are 1 (row number 3 in the table). All other input cases (rows 0, 1, and 2) should output a low voltage (a logical 0). For our voltage divider gate design paradigm this means that when the gate output should be high voltage (logic 1) then the path between the gate output and the supply voltage should pass through two MOSFETs in series, one for each input, A and B, and the two MOSFETs should each turn ON with a high (one) input. For the path between the AND gate output and ground (zero) we want either A or B to trigger a conduction path, so we want parallel MOSFETs of the type that turn ON with a 0 input.

Compared to the NAND circuit in the text, the AND circuit uses a MOSFETs in series circuit for the R_1 portion of our lecture slides voltage divider circuit, rather than a parallel MOSFET circuit. The AND also uses a parallel MOSFETs circuit for the R_2 portion of the voltage divider, rather than the series MOSFET circuit of the NAND gate. Finally, the type of MOSFETs used for the R_1 and R_2 portions of the voltage divider are the opposite in the AND circuit of their placement in the NAND circuit. There is a circuit "duality" between NAND and AND: to convert on gate circuit into the other, exchange series and parallel structures and exchange high-active (On) and low-active (Off) MOSFET types. This circuit duality holds true for 3-input NAND and AND as well.

4. Under what conditions does a full adder generate Sum = 1 and Carry out = 0 from Augend, Addend, and Carry in? Show your answer in the form of a table.
Answer:

Augend	Addend	Carry in	Sum	Carry out
1	**0**	**0**	1	0
0	**1**	**0**	1	0
0	**0**	**1**	1	0

5. **[3 points; 1 pt. for each key idea]** What are the three key ideas behind making a circuit behave digitally?
   Answer: (1) The circuit can produce two distinct output voltages, using power supply and ground as two references. (2) The circuit can be built to process any input voltage falling within a reasonable range of power supply voltage or a within a reasonable range of ground voltage the same as if it were one of these ideal values. (3) The circuit can change its output voltage quickly, so intermediate voltage values appear only briefly

6. What principle allows for the simplification of descriptions of hardware or software by omission of detail?
   Answer: Abstraction.

7. What element in pure crystal form was used to make the first transistor?
   Answer: Germanium.

8. **[3 points for an answer using concepts of an infinite set and the finitude of computer hardware]** Why cannot a computer perform addition on the Natural numbers?
   Answer: There are infinitely many Natural numbers, yet all computers are finite, and so cannot represent all Natural numbers, either as an operand nor as a result.