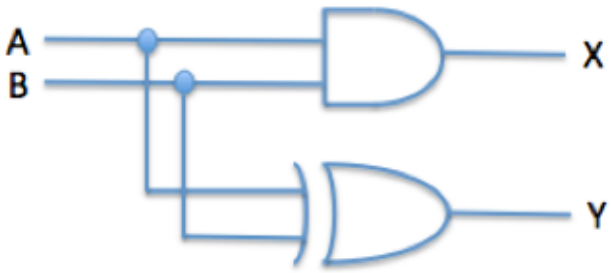
Computer Science 605.611

Problem Set 3 Answers

[For this problem set, use "*" to denote logical AND, "+" to denote logical OR, "^" to denote exclusive-OR and the apostrophe " ' " to denote NOT (e.g., N' means NOT N).]

1. a) (5) Complete the truth table below by filling in the columns for the circuit outputs Y and X as a function of the inputs A and B.



A	B	Y	X
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

b) (5) To what arithmetic functions do the outputs Y and X correspond?

Y corresponds to the single-bit arithmetic sum of A + B and X corresponds to the carry generated when the sum of A+B is computed.

2. (10) Consider the following truth table that defines the output C as a function of the two inputs A & B:

Inputs		Output
A	B	C
0	0	0
0	1	1
1	0	1
1	1	0

Use only Boolean identities (not truth tables) to show that the logical sum of the non-zero minterms for this function is equivalent to the logical product of the zero-maxterms.

The logical sum of the non-zero minterms is:

(A' * B) + (A * B')

The logical product of the zero maxterms is:

(A + B) * (A' + B')

Applying the distributive law to this yields

A*(A' + B') + B*(A' +B') = A*A' + A*B' + B*A' + B*B' = 0 + A*B' + B*A' + 0

= A'*B + A*B'

3. Recall that the modulo function L MOD N is defined as the remainder produced when L is divided by N.

a) (5) Write down a series of MIPS true-op instructions that use the integer divide instruction to produce in register $8, the result of the function $8 MOD $4.

```
div  $8,$4      ;divide $8 by $4, lo reg contains the quotient and hi contains the remainder
mfhi  $8        ; copy remainder from hi into $8
```

b) (5) Write down a single (i.e., one) MIPS logical instruction that produces in register $9 the result of the function $9 MOD 256.

```
andi    $9,$9,255      ; clear all but the low 8 bits in $9
```

4. (5) Consider the circuit:

Write down an equivalent logic function involving only the operators * for AND, + for OR and ‘ for NOT. X = __A’ * B’ +A * B___

5. An encoder generates an output that identifies which one of its inputs is active. The output is in effect the index of the single active input.

a) (5) How many outputs are required for an encoder that has 8 inputs?
Since there are 8 inputs, the index of the active input can range from 0 to 7 (000 to 111 in binary). So three output bits are required.

b) (5) If a decoder like that described in module 3 has 4 outputs, how many inputs should it have? The decoder interprets its inputs as an N-bit number and asserts the one output whose index is given by the N-bit number. Since there are 4 outputs, the input index only ranges from 0 to 3. So two binary inputs are required.

6. (5) Suppose the register $7 contains a single precision IEEE 754 floating point number. Write down a series of MIPS integer instructions that changes the floating point number in $7 into its arithmetic negative.

Inverting the sign bits negates the floating point number. Hence the following pair of instructions will work:

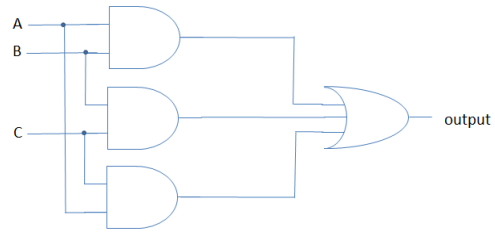
```
lui     $1,0x8000

xori    $7,$7,$1
```

7. (5) How can the same full adders used to generate the sum of two integers N+M be used instead to generate their difference N-M ?

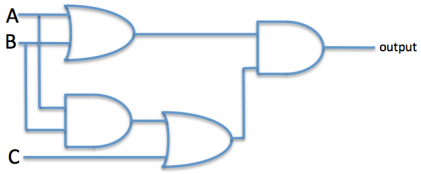
This can be done by adding the two's complement negative of M to N. So the same circuits used for addition can be used for subtraction.

8. a) (5) Write down a logic expression for the output generated by the following circuit:



output = AB + BC + AC

b) (5)Write down a logic expression for the output generated by the circuit below:

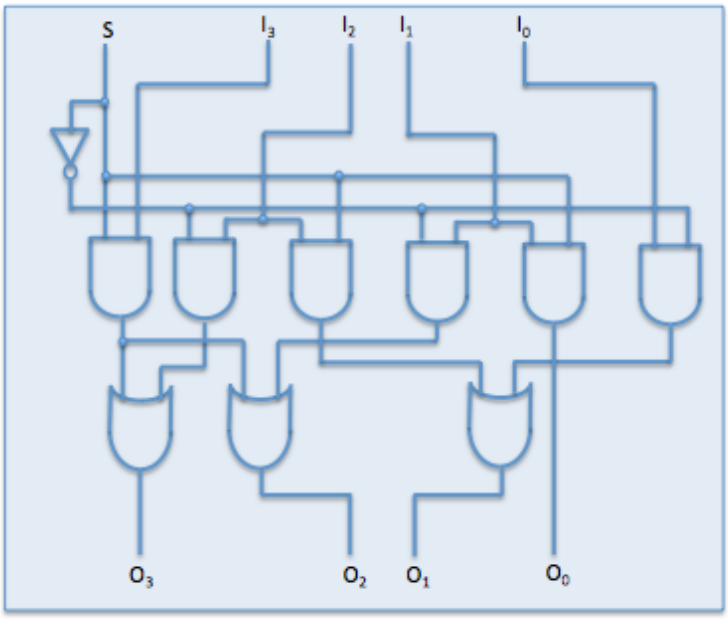


output = (A+B)(AB +C)

c) (5) Use Boolean identities (not truth tables) to show that the expression you obtained for part b) can be transformed into the expression you obtained for part a). That is, the expression for part b) can be rewritten as the expression for part a).

output = (A+B)(AB +C) = A(AB+C) + B(AB+C)
= AAB + AC + BAB + BC = AAB + AC + ABB + BC = AB + AB + AC + BC
= AB + BC + AC

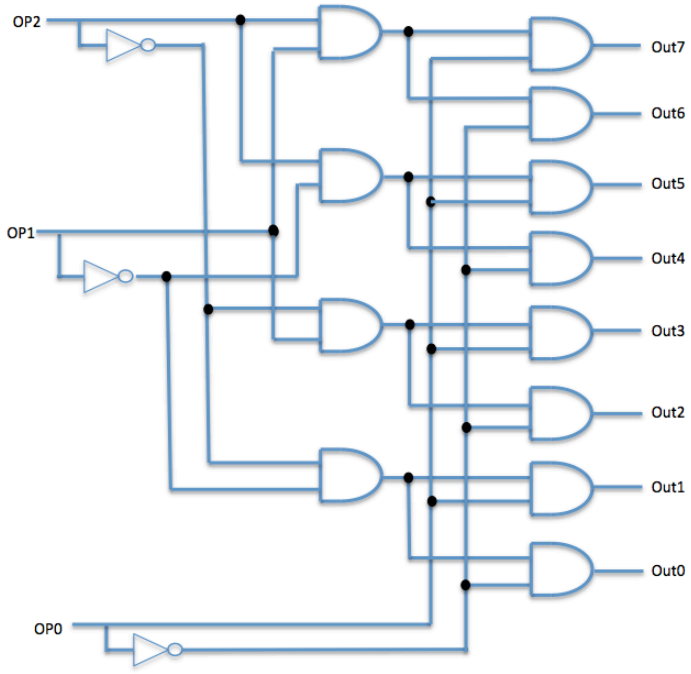
9. Consider the following circuit:



a) (5) If S=0 show the outputs if initially I_3, I_2, I_1 I_0 = 0101.
O_3 = ___1___
O_2 = ___0___
O_1 = ___1___
O_0 = ___ 0___

b) (5) If S=1 show the outputs if initially I_3, I_2, I_1 I_0 = 1110.
O_3 = ___1___
O_2 = ___1___
O_1 = ___1___
O_0 = __ 1____

10. (10) Consider a CPU that uses 3-bit opcodes. Draw a logic circuit containing only discrete NOT gates together with AND each of which uses only 2 inputs; the circuit should implement a decoder for the 3-bit opcode. Your circuit should contain the minimum number of NOT gates and the minimum number of 2-input AND gates.
A minimum of 3 inverters are needed (one for each of the 3 inputs).
Four AND gates are required to generate (Op2*Op1), (Op2*Op1'), (Op2'*Op1) and (Op2'*Op1'). Four more AND gates can be used to AND each of these terms with OP0; another four AND gates can be used to AND each of these with OP0'. So a total of 4+4+4 = 12 AND gates are required. The corresponding circuit is shown below:



11. (5) The following boolean expression defines Z as a function of four inputs A, B, C and D:
Z = A*D*(B+C) + B*C*(A + D)

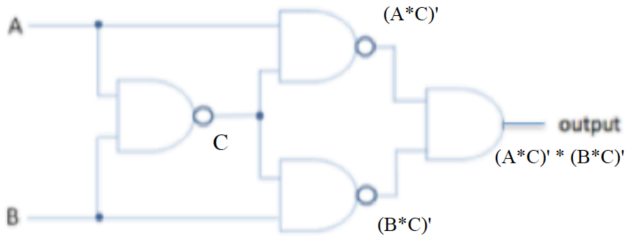
where "*" denotes a logical AND operator and "+" denotes a logical OR operator. Which of the following types or circuits best matches this function? Explain why.

I. Comparator (Z=1 if all the inputs are equal)
II. Majority vote (Z=1 if more than half of the inputs are set)

This is a majority vote function since the output is 1 only if at least 3 of the 4 inputs are 1. That is, if the majority of the inputs are 1. Otherwise the output is 0.

A comparator outputs 1 if all of the inputs are the same and outputs 0 if the inputs differ. This circuit outputs 1 even if, for example, B=0 and the other inputs are all 1. Hence it is not a comparator

12. (5) Consider the logic circuit shown below:



Which one of the following individual logic gates is equivalent to this circuit (i.e., generates the same output for the same inputs)?

a) AND
b) OR
c) NOR
d) NAND
e) XNOR
f) XOR

Let C=(A*B)'
output = (A*C)' * (B*C)' = (A' + C') * (B' + C') = A' * (B' + C') + C' * (B' + C')
= A' * B' + A'*C' + C' * B' + C' * C' = A' * B' + A' * C' + C' * B' + C'
= A' * B' + A' * C' + C' * (B' + 1) = A' * B' + A' * C' + C' = A' * B' + C' * (A'+1)
= A' * B' + C' = A' * B' + A * B = A XNOR B by definition. So the answer is e)