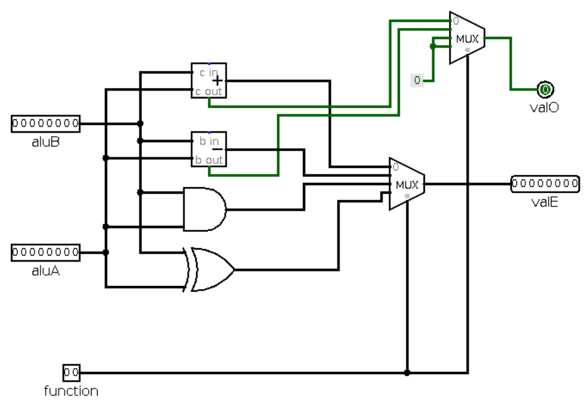
CPSC 121 Midterm 1
Tuesday, October 15th, 2013

[1] 1. Do you want your tutorial attendance to count towards your grade?

If you answered 'YES', then 1% of your final course mark will be based on your tutorial attendance (# attended divided by total number minus 2, but no more than full credit). Online quizzes will be worth 5%. If you answered 'NO', then tutorial attendance is worth nothing in your mark, and online quizzes count for 6% of your final course mark.

[6] 2. Short Answers

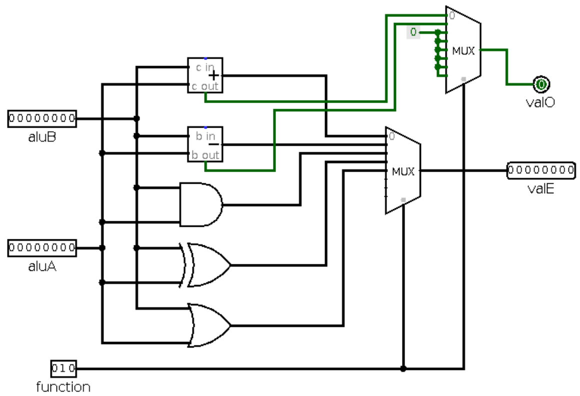
[3] a. Recall the arithmetic and logic unit from lab #4:



This circuit can not compute the OR of its two inputs (i.e. $aluA \vee aluB$). Describe briefly the changes you would need to make to the circuit to add the OR function to it. You are not allowed to remove one of the functions it currently computes to make room for OR.

Solution : You would need to (1) add an OR gate connected to the inputs `aluA` and `aluB`, (2) modify the two multiplexers to have 8 inputs (instead of 4) and a 3-bit `select` input (instead of a 2-bit `select` input), and (3) increase the width of the `function` input to 3 bits from 2.

The modified ALU is shown below:

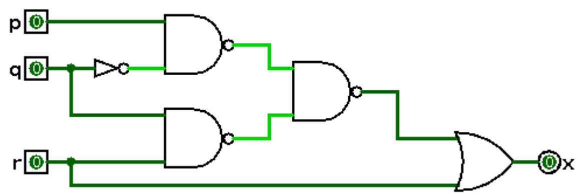


[3] b. How could you determine if the following is a valid rule of inference, knowing that it can not be proved by applying the rules listed on the lectures slides and/or in Epp ?

$$\begin{array}{l} (p \rightarrow q) \wedge (r \rightarrow s) \\ p \vee r \\ \hline \therefore q \vee s \end{array}$$

Solution : You could verify that the proposition $((p \rightarrow q) \wedge (r \rightarrow s) \wedge (p \vee r)) \rightarrow (q \vee s)$ is a tautology using a truth table.

[9] 3. Consider the following circuit:



[3] a. Write an unsimplified boolean algebraic expression for the output of this circuit.

Solution : The unsimplified boolean expression corresponding to the circuit is

$$x : \sim(\sim(p \wedge \sim q) \wedge \ \sim(q \ \wedge \ r)) \ \vee \ r$$

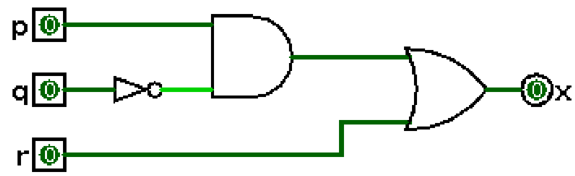
[4] b. Using a sequence of logical equivalences, simplify this expression as much as possible.

Solution :

$$\begin{array}{rcll} x & \equiv & (\sim\sim(p \wedge \sim q) \vee \ \sim\sim(q \ \wedge \ r)) \ \vee \ r & \text{by DeMorgan's law} \\ & \equiv & ((p \wedge \sim q) \ \vee \ (q \ \wedge \ r)) \ \vee \ r & \text{by the double negative law} \\ & \equiv & (p \wedge \sim q) \ \vee \ ((q \ \wedge \ r) \ \vee \ r) & \text{by the associative law} \\ & \equiv & (p \wedge \sim q) \ \vee \ r & \text{by the absorption law} \end{array}$$

[2] c. Finally, draw the circuit corresponding to the simplified expression. Hint: you should only need three gates.

Solution :



[11] 4. A political sciences major is studying the political deeds (and misdeeds) before an election in the country of Borduria. He learns the following facts:

- Either Syldavia is a much nicer country to visit than Borduria, or the bordurian secret police planted stolen art in someone's bedroom.
- If colonel Sponsz had a rival thrown in jail on a fake charge, then the bordurian secret police planted stolen art in someone's bedroom.
- Marshal Pleksy-gladz is proud of his facial hair, and either colonel Sponsz had a rival thrown in jail on a fake charge or general Tapioca likes bananas.
- If Boris is a friend of the king of Syldavia, then colonel Sponsz did not have a rival thrown in jail on a fake charge and general Tapioca does not like bananas.
- If Syldavia is not a much nicer country to visit than Borduria, then either Boris is a friend of the king of Syldavia or colonel Sponsz did not have a rival thrown in jail on a fake charge.
- Either colonel Sponsz had a rival thrown in jail on a fake charge or Boris is a friend of the king of Syldavia.

Note: Syldavia is a country that borders Borduria.

[3] (a) Rewrite each of these statements using propositional logic. Make sure to define the propositions you are using (that is, state something like: "s: Syldavia is a much nicer country to visit than Borduria.' before writing the proposition $\sim s$). You can abbreviate these (e.g. "s: Syldavia is a . . . Borduria") if you wish.

Solution : We will use the following propositions (I have ordered them alphabetically, but there was no need for you to do that):

b : Boris is a friend of the king of Syldavia.
c : Colonel Sponsz had a rival thrown in jail on a fake charge.
g : General Tapioca likes bananas.
m : Marshal Pleksy-gladz is proud of his facial hair.
p : The bordurian secret police planted stolen art in someone's bedroom.
s : Syldavia is a much nicer country to visit than Borduria.

Given that, the statements can be rewritten as:

1. $s \vee p$
2. $c \rightarrow p$
3. $m \wedge (c \vee g)$
4. $b \rightarrow (\sim c \wedge \sim g)$
5. $\sim s \rightarrow (b \vee \sim c)$
6. $c \vee b$

[8] (b) Using your answer from part (a), known logical equivalences, and the rules of inference, prove that Syldavia is a much nicer country to visit than Borduria (note: there is additional space at the top of the next page).

Solution : Here is one possible proof:

7.	$c \vee g$	specialization from 3.
8.	$\sim\sim(c \vee g)$	double negative law from 7.
9.	$\sim(\sim c \wedge \sim g)$	De Morgan's law from 8.
10.	$\sim b$	modus tollens from 4 and 9.
11.	c	elimination from 6 and 10.
12.	$\sim b \wedge c$	conjunction from 10 and 11.
13.	$\sim\sim(\sim b \wedge c)$	double negative law from 12.
14.	$\sim(\sim\sim b \vee \sim c)$	De Morgan's law from 13.
15.	$\sim(b \vee \sim c)$	double negative law from 14.
16.	$\sim\sim s$	modus tollens from 5 and 15.
17.	s	double negative law from 16.

[6] 5. The Ant-el 9099 CPU uses 9 bit integers and represents signed integers using two's complement.

[2] a. What sequence of bits would the Ant-el 9099 CPU use to represent the integer -215? Hint: $215 = 128 + 64 + 16 + 4 + 2 + 1$.

Solution : Since $215_{10} = 011010111_2$, it follows that $-215_{10} = 100101000_2 + 1 = 100101001$.

[2] b. The Ant-el then adds -80 to **your** answer from part (a). Knowing that $-80_{10} = 110110000_2$, what will be the decimal value of the sum ?

Solution : The sum is $100101001 + 110110000 = 011011001$ in binary, which is $128 + 64 + 16 + 8 + 1 = 217$.

[2] c. Why is the sum of these two negative integers on the Ant-el 9099 positive?

Solution : It is negative because $-215 - 80 = -295$, which is outside the range of values that can be represented with a 9-bit signed integer $(-256 \ldots 255)$.

[6] 6. Given the following definitions:

- P: the set of people.
- C: the set of the kinds of creatures.
- *Likes*(x, y): person x likes creature y.
- *AfraidOf*(x, y): person x is afraid of creature y.

translate each of the following English statements into predicate logic. For instance, the statement "Paul likes dogs" would be translated as *Likes*$(Paul, dogs)$.

[3] a. One or more people are afraid of ghosts and witches, but are not afraid of (and like) black cats.

Solution : $\exists p \in P, \textit{AfraidOf}(p, \text{ghosts}) \wedge \textit{AfraidOf}(p, \text{witches}) \wedge \sim\textit{AfraidOf}(p, \text{black cats}) \wedge \textit{Likes}(p, \text{black cats})$.

[3] b. People do not like creatures they are afraid of.

Solution : $\forall p \in P,\ \forall c \in C, \textit{Afraid}(p, c) \rightarrow \sim\textit{Likes}(p, c)$

[3] 7. Using the same definitions as in the previous question, translate

$$\forall p \in P, \forall q \in P, \exists c \in C, \mathit{Likes}(p, c) \wedge \mathit{Likes}(q, c)$$

into English.

Solution : Any two people have a creature that they both like in common.

[8] 8. Design a circuit that takes as input a four bit unsigned integer $b_3b_2b_1b_0$ and outputs the integer part of its square root as a two bit unsigned integer x_1x_0. For instance, if the input was 4, then the output would be 2, and if the input was 8 then the output would also be 2. Hint: think about the outputs individually; your circuit should not attempt to "compute" the square root mathematically!

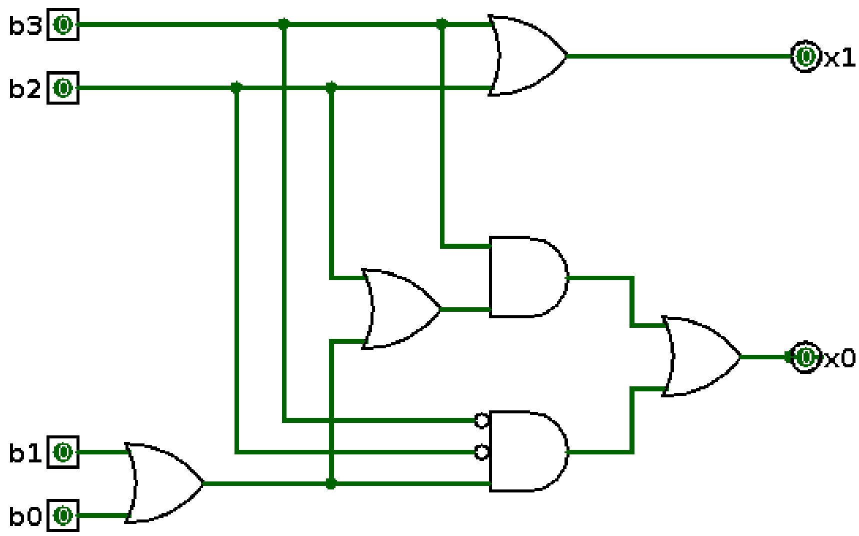
Solution : We start by writing out the truth table for the circuit:

Input	b_3	b_2	b_1	b_0	Output	x_1	x_0
0	0	0	0	0	0	0	0
1	0	0	0	1	1	0	1
2	0	0	1	0	1	0	1
3	0	0	1	1	1	0	1
4	0	1	0	0	2	1	0
5	0	1	0	1	2	1	0
6	0	1	1	0	2	1	0
7	0	1	1	1	2	1	0
8	1	0	0	0	2	1	0
9	1	0	0	1	3	1	1
10	1	0	1	0	3	1	1
11	1	0	1	1	3	1	1
12	1	1	0	0	3	1	1
13	1	1	0	1	3	1	1
14	1	1	1	0	3	1	1
15	1	1	1	1	3	1	1

We can then observe the following patterns:

- Output x_1 is 1 whenever the input is greater than or equal to 4. That is, x_1 is 1 when either b_3 is 1, or b_2 is 1 (or both).
- Output x_0 is 1 when the input is either less than 4 (but not 0), or larger than 8. The first condition can be written as $\sim b_3 \wedge \sim b_2 \wedge (b1 \vee b_0)$, while the second condition can be expressed as $b_3 \wedge (b_2 \vee b_1 \vee b_0)$.

These give us the following circuit:


b3
b2
b1
b0
x1
x0