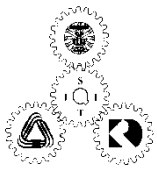


Sirindhorn International Institute of Technology

Thammasat University at Rangsit

School of Information, Computer and Communication Technology

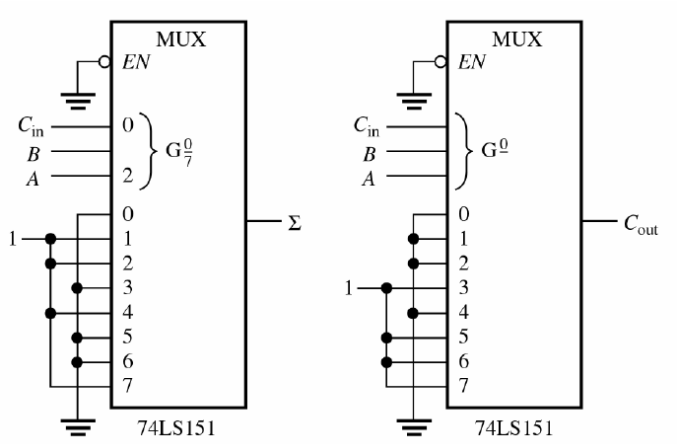
ECS 371: Problem Set 5 Solution

Semester/Year: **1/2009**
Course Title: Digital Circuits
Instructor: Dr. Prapun Suksompong (prapun@siit.tu.ac.th)
Course Web Site: http://www.siit.tu.ac.th/prapun/ecs371/

Due date: Not due

1. In class, we implemented a full-adder using a 3:8 decoder and two OR gates. For this question, implement a full-adder using two 74x151s.

Solution:



In class, we have seen that the output of a MUX is a weighted sum (OR) of all minterms where the weights are the inputs. Here we feed 1 in the inputs that correspond to the minterms that we want.

2. What are the full-adder inputs that will produce each of the following outputs:

(a) $\Sigma = 0, C_{out} = 0$

(b) $\Sigma = 1, C_{out} = 0$

(c) $\Sigma = 1, C_{out} = 1$

(d) $\Sigma = 0, C_{out} = 1$

2.

(a) $A = 0, B = 0, C_{in} = 0$

(b) $A = 1, B = 0, C_{in} = 0$ or $A = 0, B = 1, C_{in} = 0$ or $A = 0, B = 0, C_{in} = 1$

(c) $A = 1, B = 1, C_{in} = 1$

(d) $A = 1, B = 1, C_{in} = 0$ or $A = 0, B = 1, C_{in} = 1$ or $A = 1, B = 0, C_{in} = 1$

3. Determine the outputs of a full-adder for each of the following inputs:

(a) $A = 1, B = 0, C_{in} = 0$

(b) $A = 0, B = 0, C_{in} = 1$

(c) $A = 0, B = 1, C_{in} = 1$

(d) $A = 1, B = 1, C_{in} = 1$

3.

(a) $\Sigma = 1, C_{out} = 0$

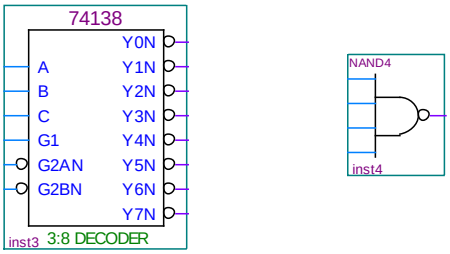
(b) $\Sigma = 1, C_{out} = 0$

(c) $\Sigma = 0, C_{out} = 1$

(d) $\Sigma = 1, C_{out} = 1$

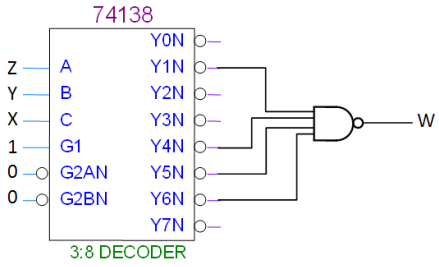
4. Implement the logic function specified in the table below by using only a 74x138 and a NAND gate.

Input			Output
X	Y	Z	W
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0



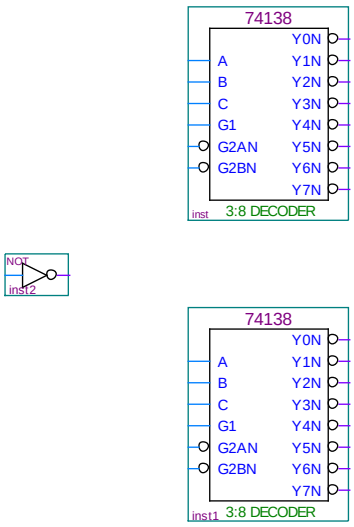
Your score depends strongly on your explanation of your answer. Zero score may be given even for a correct answer if the explanation is incomplete.

Solution:



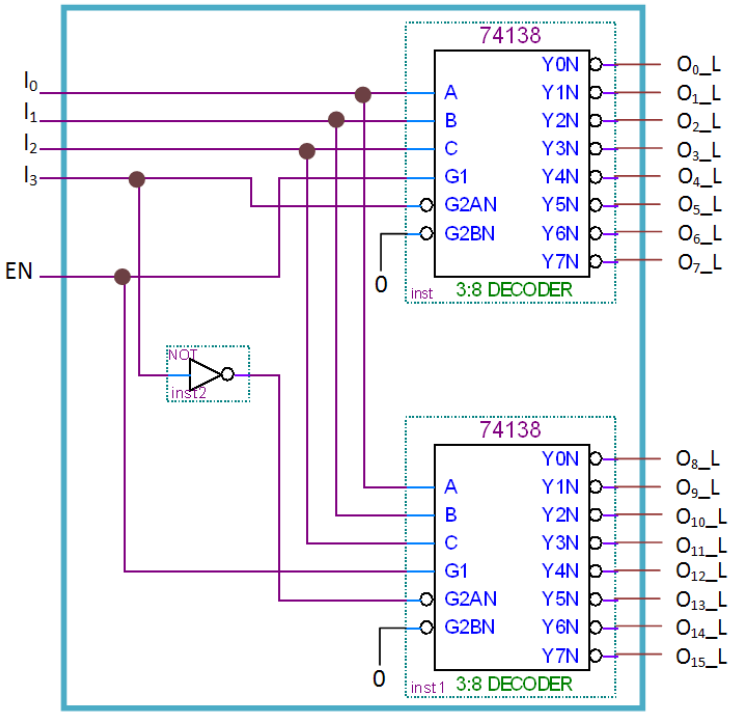
We have seen in class that the decoder generates all possible minterms. Therefore, a decoder can be used to implement any SOP by OR-ing appropriate outputs (minterms) of the decoder. In this case, the outputs of the decoder are active-LOWs. So, instead of the OR gate, we need a negative-OR gate where the “negative” is needed to cancel the bubbles on the outputs of the decoder. Of course, negative-OR gate is the same as the NAND gate.

5. Construct a 4:16 decoder with an active-HIGH enable (EN) and active-LOW outputs from two 74x138 decoders and one NOT gate. Label the inputs of the 4:16 decoder by I_3 I_2 I_1 I_0 where I_3 is the MSB. Label the outputs of the 4:16 decoder by O_{15} O_{14} Q_{13} O_{12} … O_1 O_0.



Again, your score depends strongly on your explanation of your answer. Zero score may be given even for a correct answer if the explanation is incomplete.

Solution:



We connect the EN line to G1 of both decoders. Because G1 is active-HIGH. EN = 1 will enable both decoders and EN = 0 will disable both decoders.

The lower significant bits (I_2 I_1 I_0) of the inputs are connected to both decoders. We use the MSB to control which decoder to be enabled to get the correct output. We have seen in class that the upper decoder should get $\overline{I_3}$ which, in this case, is achieved by connecting I_3 to the active-LOW enable (G2AN) of the upper decoder.

Furthermore, we have seen in class that the lower decoder should get a direct connection to I_3. In this case, we pass I_3 through the NOT gate to cancel the effect of the bubble on the active-LOW enable (G2AN) of the lower decoder.

The unused enable lines (G2BN) of both decoders need to be asserted. Otherwise, the decoders will always be disabled. To make them active, the G2BN lines are fed with 0 because they are active-LOW.