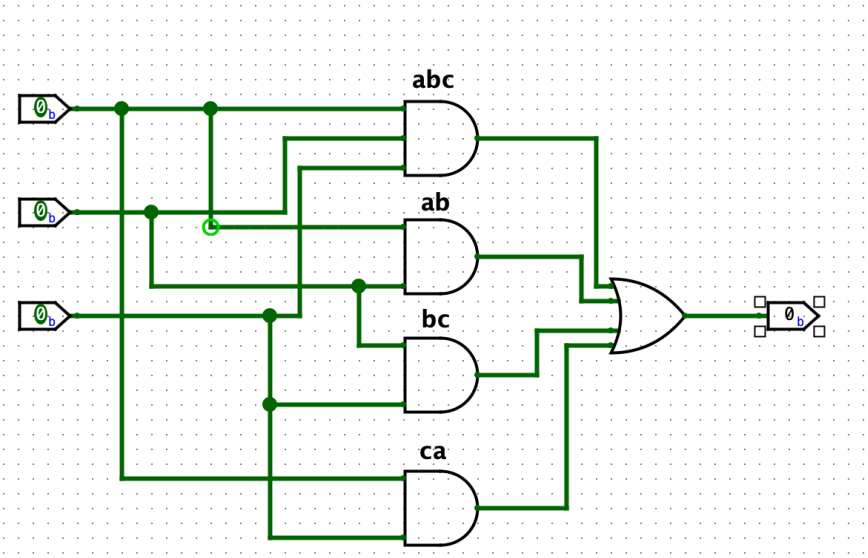
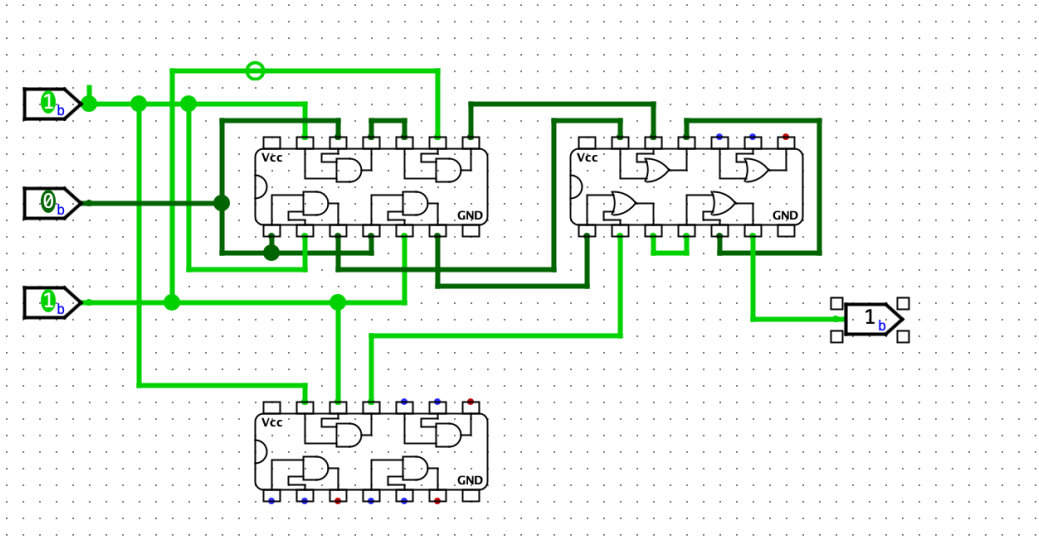
Lab 2

Matthew Peterka 131008310

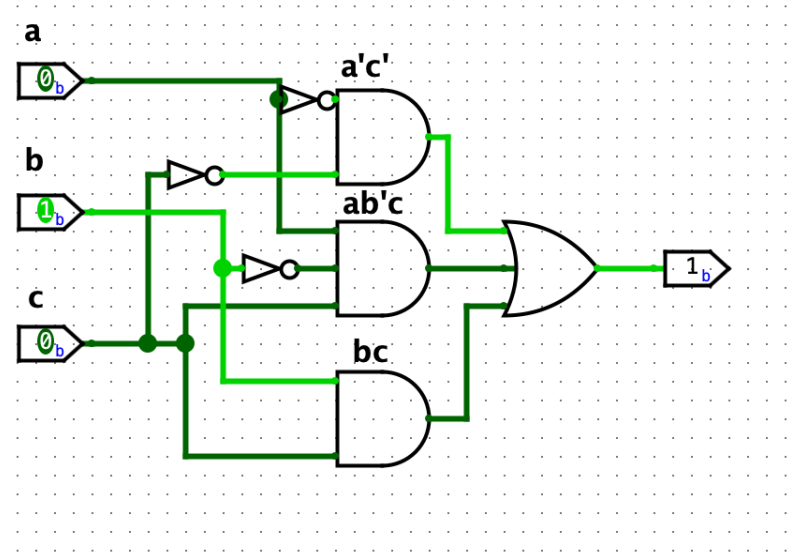
Problem 1.

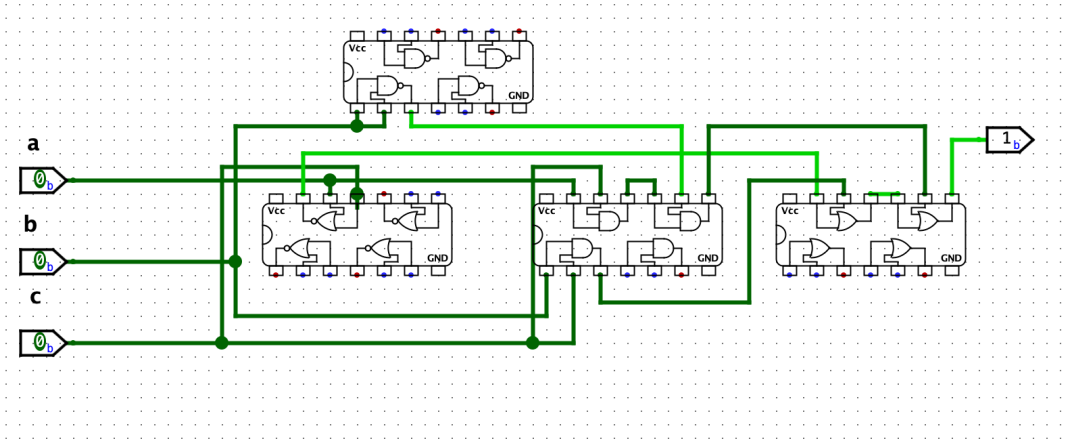
1) Using one and two input gates, draw and verify the digital circuit for the functions. Do not simplify/reduce the equations. Please provide a screenshot of your digital circuit in your lab report.
    a) F





b) G



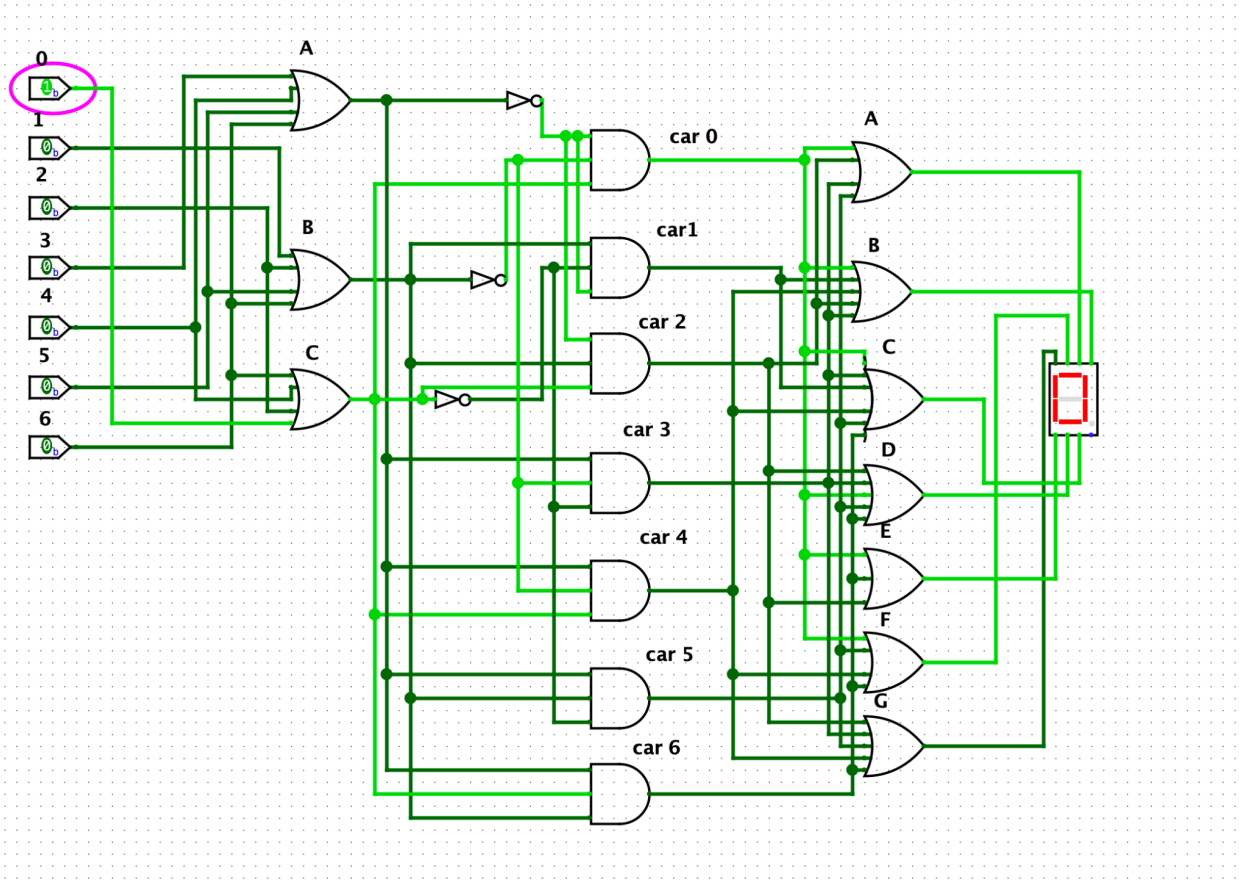


2) What are the names of the 74xx series logic gates that you should use to do the above problem in real-life situations? Explain how you connect input & output using those logic gates to build a digital circuit shown above in real life.
    a) The logic gates that I used to solve the problem were 7400: quad 2-input NAND gate, 7402: quad 2-input NOR gate, 7408: quad 2-input AND gate, and 7432: quad 2-input OR gate. By looking at the inside structure of each gate, you can see where the simple gates are and you can connect two inputs and take an output from 4 different gates in the circuit. Using these outputs, I connected them to other circuits to create one big chain of logical equations that F and G were made of. For example, to create a NOT gate, I used the 7400 circuit and connected the same input twice, creating the NOT gate which was then used in conjunction with the AND gate to get the value of ab'c correctly.

3) Delay Performance:
    a) Function F:
        i) 7408 Gate (x2):
            (1) Tplh, Cl=15pF, Rl=400R, 22 ns
            (2) Tphl, Cl=15pF, Rl=400R, 15 ns
        ii) 7432 Gate:
            (1) Tplh, Cl=15pF, Rl=400R, 22 ns
            (2) Tphl, Cl=15pF, Rl=400R,15 ns
        iii) Adding these up the worst-case delay performance is 66 nanoseconds at the conditions listed.
    b) Function G:
        i) 7400 Gate:
            (1) Tplh, Cl=15pF, Rl=400R, 22 ns
            (2) Tphl, Cl=15pF, Rl=400R,15 ns
        ii) 7402 Gate:
            (1) Tplh, Cl=15pF, Rl=400R, 22 ns
            (2) Tphl, Cl=15pF, Rl=400R,15 ns
        iii) 7408 Gate:
            (1) Tplh, Cl=15pF, Rl=400R, 22 ns
            (2) Tphl, Cl=15pF, Rl=400R, 15 ns
        iv) 7432 Gate:
            (1) Tplh, Cl=15pF, Rl=400R, 22 ns
            (2) Tphl, Cl=15pF, Rl=400R,15 ns
        v) Adding them up the worst-case delay performance is 88 nanoseconds at the conditions listed.

Problem 2:

1. Design Parameters:
    a. Number of switches that will be required
        i. There will be 7 switches since the buttons go from car 0 to car 6.
    b. Number of the bits/wires required in the data bus
        i. In the data bus, there will be 3 bits since it is an 8 to 3 bit encoder.
    c. Size of the encoder and decoder
        i. 3 to 8 bit decoder and an 8 to 3 bit decoder
            1. Although the encoder is 8 bits, one of the bits will not be used since there are not enough switches. The same goes for the decoder, one of the bits will not have to be decoded since there are not enough switches.



2. How it works:

Firstly, there will be seven inputs, one for each subway car. These will then go into three OR gates which is the encoder that will turn these inputs into a 3 bit binary output. Using a truth table for the 8 to 3 bit encoder, we can create equations to represent the OR gates. The equations are as follows, A = car_3 + car_4 + car_5 + car_6, B = car_1 + car_2 + car_5 + car_6, and C = car_0 + car_2 + car_4 + car_6. These equations allow the encoder to convert the input signals into a binary representation of 0's and 1's. Next, the signal goes into the encoder which is built of AND gates as well as NOT gates. The encoder takes in 3 bits and converts those inputs into what car switch is flipped. Using another truth table we can obtain values that the 3 bits convert to. For example, if every switch is off, then the output would be 00000001. This holds for every value past that as well. For car 1 for example, the encoded value would be 001. Then using the decoder it would decode the value as 00000010 meaning that car 1's switch is flipped. To get the 3 input bits, the outputs from the first 3 OR gates are used for this input. Lastly, knowing the car switch that is flipped, the output goes into several OR gates that connect to the 7-segment display for the corresponding number. For example, for 0, the output from the car would go into 6 OR gates since the number 0 on the display is represented by 6 different switches. This process is then repeated for every number 0-6.