Class:	CPE100L - 1002	Semester:	Spring 2020

Points		Document author:	Kristy Nguyen
		Author's email:	nguyek20@unlv.nevada.edu
		Document topic:	Final Report

Instructor's comments:

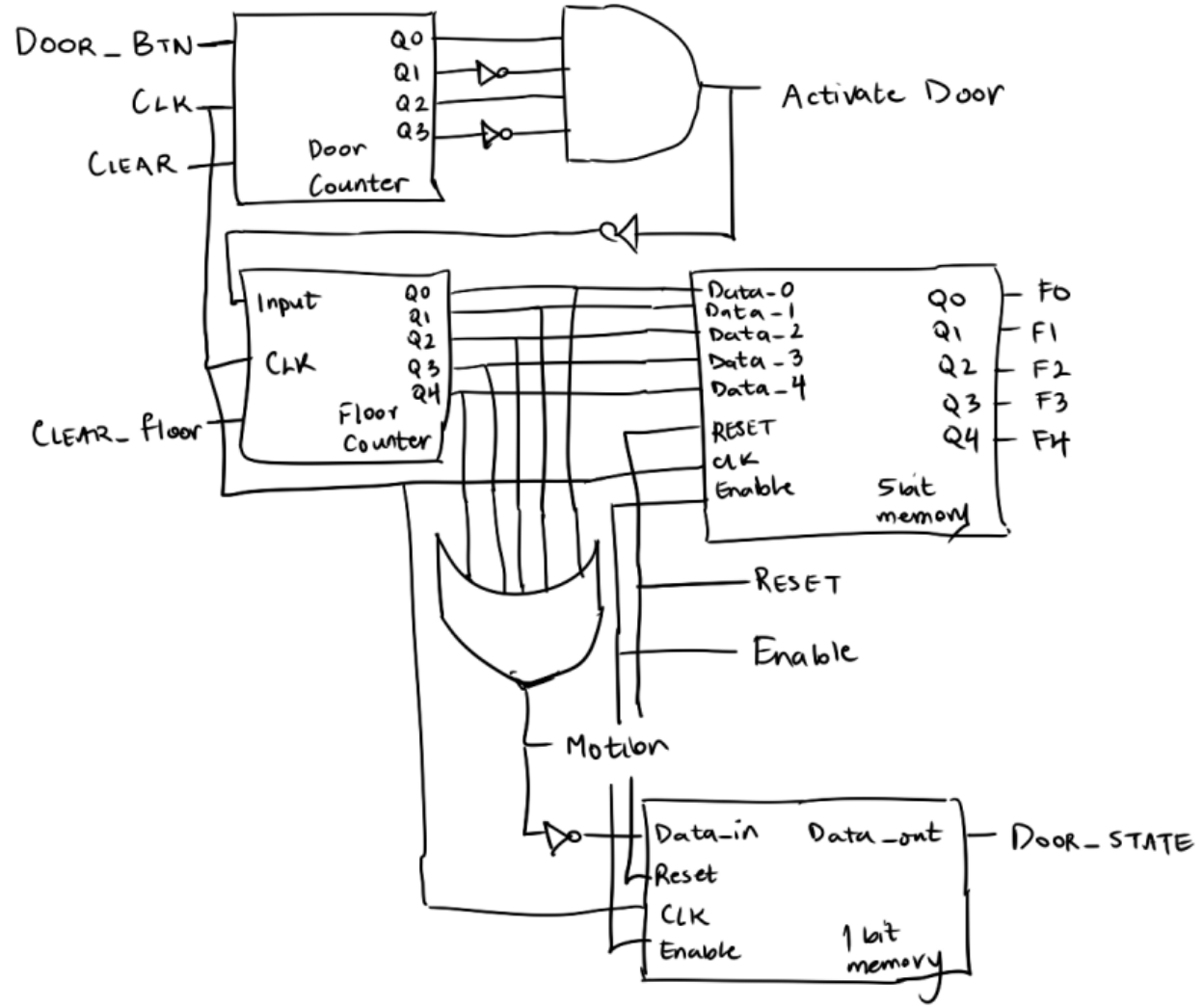
1. **Final Project**
   a. **Description of the Project**
   This project is implementing a digital logic controller for an elevator.

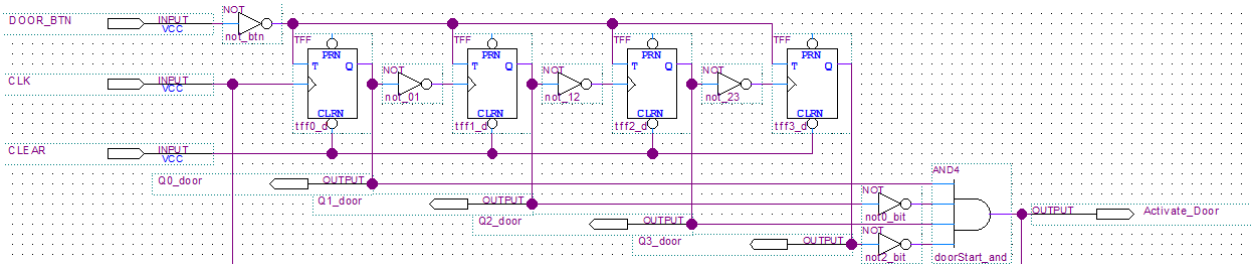
   b. **Goal**
   The goal is to implement a working elevator that operates on 10 floors with a 10 floor buttons to request the destination floor, motion signal to indicate the elevator's motion, output DOOR_STATE to demonstrate the state of the doors (opened or closed), and a DOOR_BTN to open or close the door.

   c. **Background Theory**
   A typical elevator operates on a number of floors, 10 for this project, and has buttons to open/close the door as well as requesting a floor number. After the door opens/closes, it should take some time for the door to close/open (5 CLK cycles). After requesting a floor with 10 floor buttons, the door should close and the elevator should start moving, traveling at the speed of 3 CLK cycles per floor, and not allowing any other button inputs. Once the elevator reaches the destination floor, the door is opened, and the buttons start active again for the next request. There are approximately 12 required inputs and 5 required outputs. There are inputs CLK to drive the logic of the flip-flops, DOOR_BTN to close the door if opened, and FLOOR_BTN1-FLOOR_BTN10 to request the floor. There are outputs Floor0-Floor4 to demonstrate the floor number in BCD form, DOOR_STATE to indicate an open state ('1') or closed state ('0'), and a motion signal to indicate elevator motion ('1' for in motion and '0' for not in motion).

   d. **Schematics, diagrams, etc.**

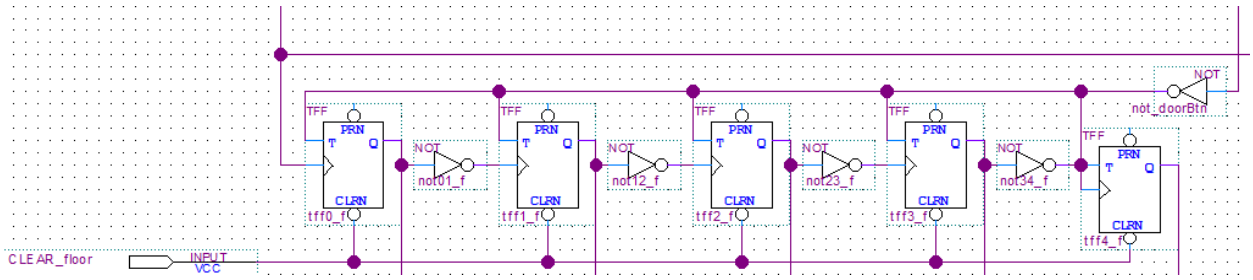
Block diagram





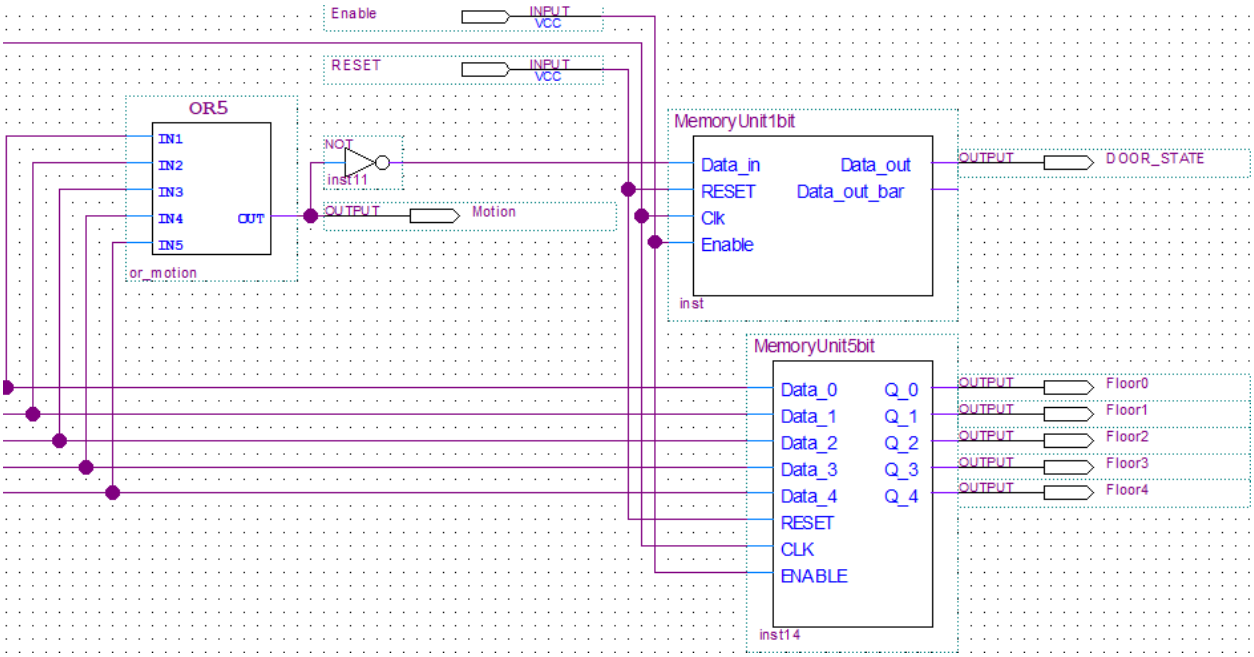
Door Counter

The door counter is a 4-bit counter that counts to 5 in order to activate the door.



Floor Counter

The floor counter is a 5-bit counter that counts the floor number.

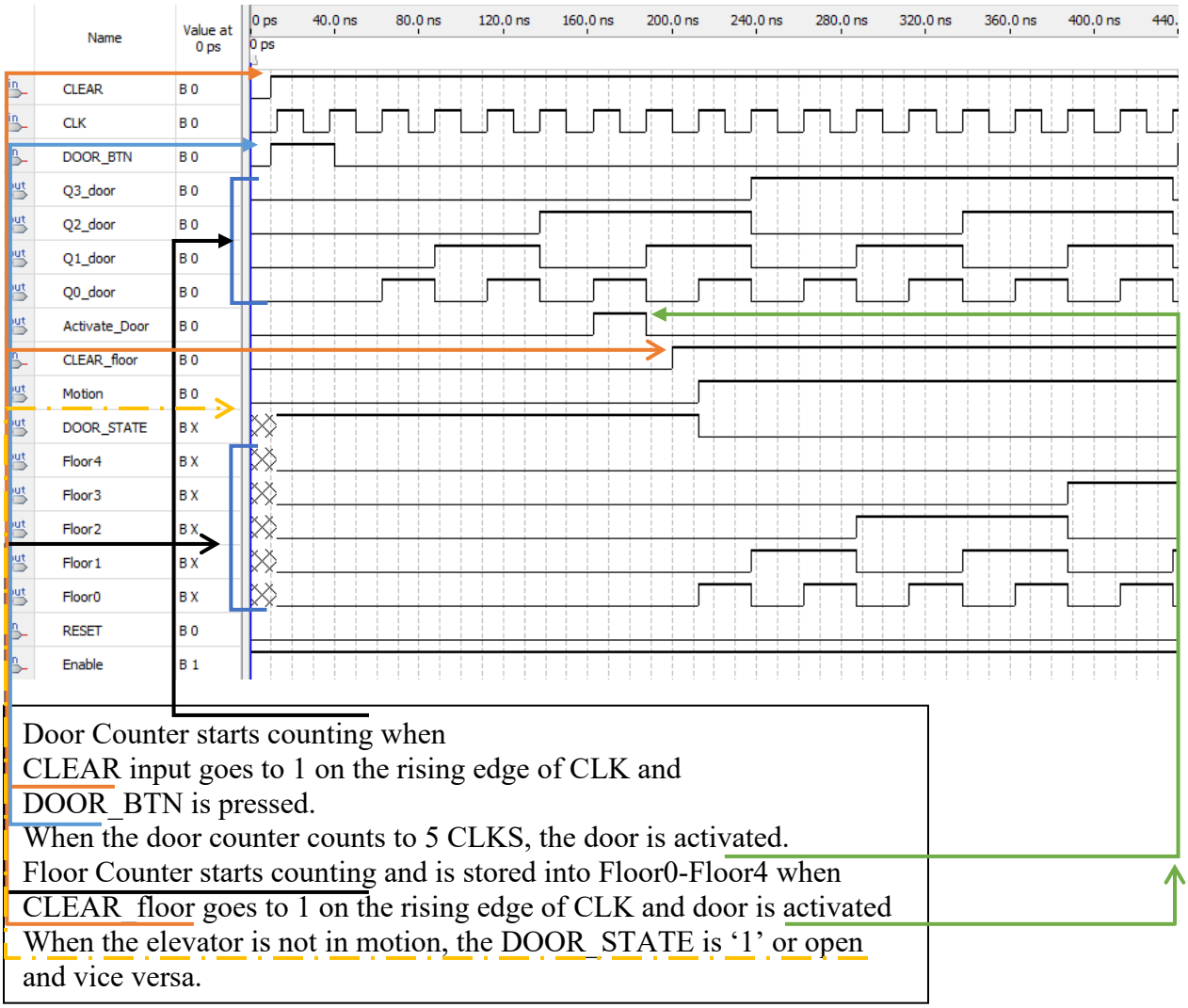


In this schematic screenshot, the outputs from the floor counter activate the motion signal. In other words, when the floor counter starts counting, the motion signal should be activated or '1'. When the motion signal is not '1', the memory unit stores the DOOR_STATE, which should be '0' or closed and vice versa. In addition, the outputs of the floor counter should be stored and demonstrated through the outputs Floor0-Floor4 in BCD.

e. Circuit Operation

There should be no circuit operation since there is no equipment to operate the circuit with. If there were to be a circuit, I would assume that the use of the Altera DE0 board would be useful in programming the memory, utilizing the buttons, and demonstrating the count like an elevator through the LEDs.

f. Simulation Results



g. Encountered Problems and How They Were Solved

I tried my best to do what I can, but I was only able to implement part of the elevator. I was only able to demonstrate that the DOOR_STATE is open when the elevator is not in motion and vice versa. I demonstrated that the floor counter stores the floor numbers through a 5 bit memory unit and starts counting once the door counter counts to 5 CLKS. The activate door output that I have simply acts like a button but I manipulated the CLEAR input to be after the Activate Door output to make it seem like the counter counts after the Activate Door acts like a DOOR_BTN to allow the elevator to count. I was able to demonstrate the DOOR_STATE through inverting the motion and inputting it through the 1 bit memory unit to show that the elevator opens when the elevator is not in motion. I did not solve the encountered problems and was unable to demonstrate the FLOOR_BTN1-FLOOR_BTN10 completing the floor requests as well as the other operations that would require the door to open and close; however, I did

my best by gathering the notes that I had from Smit and my own interpretations and pictures to show as many parts of the elevator as I could.

h. Any information regarding the project that might be interesting…

I found it interesting that when the elevator is motion, button requests are not allowed. In a real elevator, I believe the users could still input floor requests with the buttons to reach certain floor destinations. Whether this makes it more complex or less complex to implement is really interesting to me.

i. Conclusions

I reached the conclusion that a CLK input signal is necessary for many different operations, including the memory bit units that store information. Though I had not fully implemented the elevator, I understand that input signals can be put through combinational or sequential logic to do great things such as designing an elevator that could travel at different speeds.