Experiment 6 Fall 2012

Programmable Electronic Combination Lock

In this experiment we will improve upon experiment 5 by adding the ability to program the combination.

Design a system that will function as an programmable electronic combination lock.

The lock must take as input a single decimal digit from a keypad. The user will input four decimal digits and then press an Enter key (also on the keypad). If the correct sequence of decimal digits have been input the lock will open.

Each time the user presses a key the input will NOT be displayed on a 7-segment display if it is a combination under normal use. Instead four different arbitrary images should display (to keep the combination secret.) A 2nd 7-segment should display a L for locked, O for open, and P for program.

The lock combination must be programmable. To program a combination the user will press a "Program Key" and then input a secret password (hard coded). If the secret password is correct then you will be allowed to enter the new 4 digit combination followed by pressing the Program Key. The new combination should now function to open the lock. A key on the keypad should be used to lock the lock.

Design and implement the system with the following specification:

I. Input:

1. 4 X 4 matrix keypad for input. Key 0-9 for standard decimal input and possibly 4 other keys designated for "ENTER", "LOCK", "PROGRAM" and "CLEAR". Other keys may be used for additional features you may wish to implement.

   See this old experiment for keypad interface tips.

2. RESET switch (connected to the RESET NanoCore12 pin) to reset the system.

II. Output:

1. Use one segment displays with 470 Ohm current limiting resistors to display the current input digit as appropriate.
2. Use one segment displays with 470 Ohm current limiting resistors to display the current state of the lock (O,L,P)

   These will be connected to the NanoCore12 using two 8-bit D-TYPE latches.

III. Operation:

1. At power up or RESET pulse (signal that is pulled LOW and then HIGH) the LOCK display should show L (LOCKED) and the input DIGIT display should be blank (no input yet!) Note: at power up or reset the user must program the new combination (as indicated above) before using

the lock.

2. To unlock the user should enter the four digit code and then press the "ENTER" key. If the code is correct the lock should open. (Do we really need an ENTER key? Your choice! It's your lock design!)
3. If the user makes an input error they should press the "CLEAR" key and begin with the first digit. You may implement this in other ways if you wish. It's your lock design!
4. To close the lock press and release a designated key.

This is your design. You may add or reduce features as you see fit. But it must be a programmable combination lock. The secret password for programming the new combination code is required.

Experiment 6 is due WEEK 8 (10/16/2012)

This is a complex project. So, START NOW!

Report: Your lab report must include the following:
1. A cover page with: Experiment Name and number, ECE 367 Spring 2012, Your name, Your UIN, the date submitted, and your TA's name.
2. Your assembler code with your name, course, date, experiment number, program explanation and comments for every line of code. The program explanation should include the NanoCore12 pin assignments and an explanation of the organization of the data table, etc.
3. A logic diagram of the circuit
4. An electric circuit diagram of the complete system
5. A user manual to explain how to use the system.
6. Conclusions: Does your circuit meet the specifications and function properly? What problems did you encounter during the coding or construction of the circuit? How would you do this project differently? What extra features or functionality could you have included? What did you learn from working on this project?

Last modified: Wed Oct 10 00:30:34 2012

ECE 367: Experiment 6

Project: Programmable Electronic Combination Lock

Semester: Fall 2012

Name: Kai Zhao

Signature: _____

UIN: 670720413

Due Date: 2012 October 16

Lab Section: T11

TA: Chenjie Tang

User Manual:

This program is a programmable electronic combination lock. The user should first supply power to the circuit and then press the reset button to lock the combination lock. At start up, the user is in unlocking mode as denoted with a 'L' and must enter the programing mode to set a combination code. Upon entering the programming mode as denoted with a 'P', the user must enter the four digit manufacturing code and press enter before entering the setting mode. Upon entering the setting mode as denoted with a 'P.' the user may set whichever four digits the user will like and then press enter to finish setting the combination code. After setting up the combination code, the program will go back to unlocking mode and the user may attempt to unlock it or reprogram it again.

The manufacturing lock combination is '0', '4', '1', '3'.

Commands:

'A' is for locking to lock, which enters the unlocking mode

'B' is for backspace, which backspaces 1 character in any mode

'C' is for clear, which clears all inputs in any mode

'D' is for programming, which asks the user to manufacturing code before entering the setting mode

'E' is for enter, which is require after 4 digits in every mode

'F' is cheating to enter the setting mode, which allow users to enter a new combination code without the manufacturing code

Conclusion:

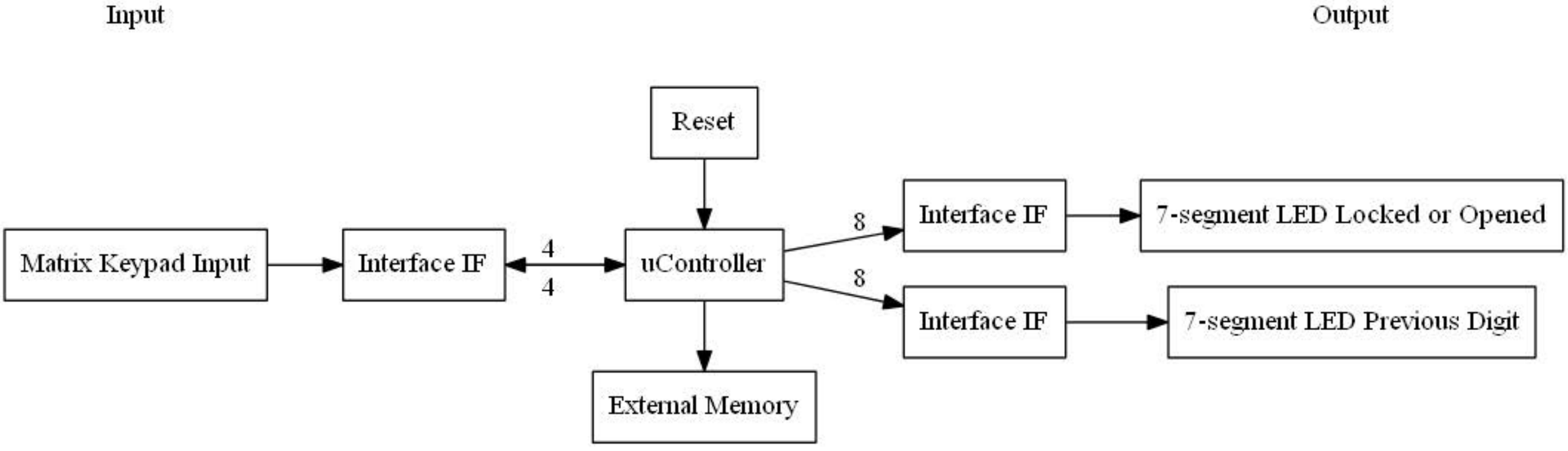
Yes, my circuit meets the specifications and function properly.

I had trouble figuring out why key inputs were not working properly. There was multiple copy and paste errors and I forgot about the release key method.

I would do this project differently by combining the modes as opposed to having 3 separate blocks of code for unlocking mode, programming mode, and setting mode.

The extra feature I have implemented is the lock key, backspace key, clear key, enter key, program key, and set key. How each of these keys work is described in the user manual.

I learned about the keypad, SIP resistor, implementing RAM variables, comparing RAM variables, and a better method of using the branchs.


Input
Output
Reset
Matrix Keypad Input
Interface IF
4
4
uController
8
Interface IF
7-segment LED Locked or Opened
8
Interface IF
7-segment LED Previous Digit
External Memory