Experiment 5 Fall 2012

Electronic Combination Lock

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

The lock must take as input a single decimal digit in 4-bit BCD format. The user will input a BCD value and then press an Enter key. This will be repeated three times. If the correct sequence of BCD values have been input the lock will open.

Each time the user presses the ENTER key the BCD input will be displayed on a 7-segment display as a decimal digit (0 - 9). The state of the lock will be displayed on a 2nd 7-segment display as L for locked and O for open.

Design and implement the system with the following specification:

I. Input:

1. One line to RESET (close) the lock.
2. One line to implement the ENTER key.
3. Four lines to implement the BCD input.

Use the "rocker switch" (or just wire) and 10K "Pull-up" resistors to create these input lines.
Use IRQ' for the ENTER key and portM for BCD input (PM0-3).
Use the NanoCore12 RESET input to implement the system RESET. No need for a PULL-UP resistor from RESET to Vcc. When RESET is connected to ground the stamp11 will re-start running the code from the Start label location. So, your first code (after setting up the PLL, stack, data direction registers, and any initialization required) should close the lock as specified.

II. Output:

1. Use one segment displays (with 470 Ohm current limiting resistors) to display the current input digit (0-9)
2. Use one segment displays (with 470 Ohm current limiting resistors) to display the current state of the lock (O,L)

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!)
2. Set the 4-bits of BCD input for the first digit input and press and release ENTER. When ENTER is pressed the input digit will display on the 7- segment DIGIT display. Repeat for the next two digits. If the correct sequence is entered the lock should open (Lock display shows O)

3. To close the lock press and release RESET

If the user makes an input error then they should press and release RESET to start over.

Use IRQ' to check for the ENTER key.

Your lock is "HARD CODED." The combination is permanently in the code. The only way to change it will be to re-assemble the code after you change those digits. Please chose 3 random digits for your combination. NO TWO DIGITS CAN BE THE SAME. Allowed: 1-4-2; 3-9-6, 7-0-4; etc. Not allowed: 0-0-0; 2-4-2; 9-8-8; etc.

You are required to use a "Look-up Table" to store the 7-Segment data for decimal digits. You then will use Index Addressing to locate the required table data based on the current BCD input.

The 7-segment displays are Common Cathode type. See this old experiment for the correct connection of the current limiting resistors and display pin-out. You will need 14 470 ohm resistors. You have them in your kits. DO NOT run the seven segment displays without the current limiting resistors. The displays should be connected to (i.e. driven by) 74HC373 8-bit D_TYPE latches. The latches will be driven by NanoCore12 output pins. Use PM4 and PM5 for latch enable (LE) control signals. The latch tri-state output controls (OE)' are active LOW. So, tie them to ground. We do want output from the latch and not High-Z! Here is the 74HC373 data sheet. The pins are defined on page 1.

Experiment 5 is due WEEK 7 (10/09/2012)

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: Tue Oct 2 18:16:34 2012

ECE 367: Experiment 5

Project: Electronic Combination Lock

Semester: Fall 2012

Name: Kai Zhao

Signature: _____

UIN: 670720413

Due Date: 2012 October 9

Lab Section: T11

TA: Chenjie Tang

User Manual:

To use this microprocessor, first supply power by USB and press the reset button to lock the combination lock. When the lock is locked, the upper 7-segment LED should display 'L' for locked. Then the user may use the switches to enter a digit in binary. After the digit is prepared, the user should press the interrupt button and the lower 7-segment LED display will display the user's input. If the user gets all 3 digits correct, then the upper 7-segment LED display should open by displaying an 'O'. The lock combination is '4', '5', '1'.

Conclusion:

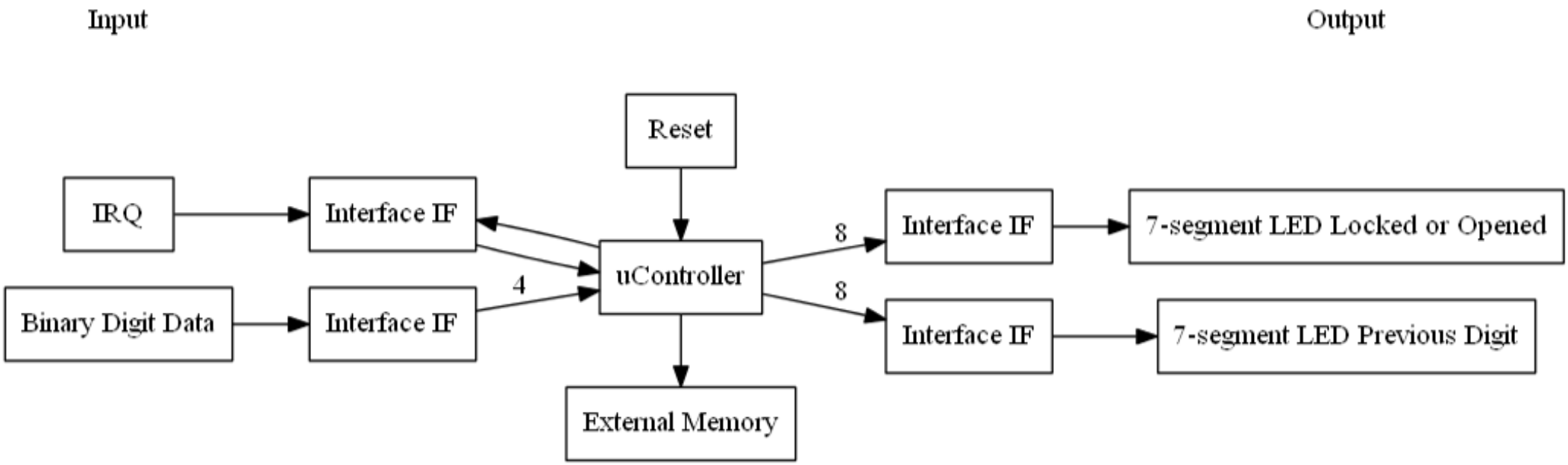
Yes, my circuit meets the specifications and function properly.

I had trouble figuring out how to use interrupts because I thought I was suppose the write the main portions of my code in ISR_IRQ as opposed to just using it to complement the flag. I also had trouble with the Schmidt Trigger because the circuits found only shows one use of the Schmidt trigger and every example of the Schmidt trigger used different resistance and capacitance values. I also had trouble using variables because I was not aware that registers X and Y does not really load the variable values and that I had to use the accumulators.

I would do this project differently by using a loop for each input as opposed to using 3 sets of BRSET to reduce the repetitive of my code.

The extra feature I have implemented is the visual LED for my switch for binary input and a decimal when the user have inputted 3 digits already to let users know that the lock is permanently locked until the reset button is pressed.

I learned about the latch enable of the 74HC373 IC, differences between the accumulators and other registers, showing inputs with LEDs, dummy methods to call long branch in case methods are out of range of 128, interrupts, and interrupt vectors.


Input
Output
Reset
IRQ
Interface IF
uController
8
Interface IF
7-segment LED Locked or Opened
Binary Digit Data
Interface IF
4
8
Interface IF
7-segment LED Previous Digit
External Memory