Oregon TECH

Computer Systems Engineering Technology
CST 345 – HW/SW Co-Design

Midterm – Takehome Test
Winter 2015
Instructor: Troy Scevers
Possible Points: 100

Name_____
Due Date: Monday, February 16th @ 5pm

Problem 1

System partitioning is an integral step in embedded system co-design methodology. Assignment of an operation to software or hardware determines the delay in that operation. Efficient design of an embedded system entails optimization of this division of labor between software and hardware. Consider the design of a door lock, such as the PV-119 has.

The input combination is a sequence of four – 4-bit digits. You have a keypad with 10 digits of numeric information along with a cancel (# for our Keypads use C) and an enter switch (* for our keypads use E). Press the first number of the combination, then the second, then third and fourth. Finally, press the enter (*) switch to open the door. If you have successfully entered the appropriate four numbers, a signal goes high to unlock the door (in your case, light one of the LEDs). The correct combinations can be encoded as part of the design. You must have at least 5 correct combinations to unlock the door. If the combination you entered is wrong it will flash a second LED for a fixed amount of time (2 seconds), leaving the door locked. This indicates that you have incorrectly entered the combination.

You will also want a signal to reset the lock. The (# for our Keypads use C) switch can generate this signal. If the reset signal goes high, the door locks and clears the current sequence. You will also need a time out after a given period of time has passed (say 5 seconds) to clear the sequence if in the middle and relock the door (There should be an LED indicating whether the door is locked or not).

Your assignment is as follows.

1. Formalize the design by determining how you will partition the hardware and software for this design. Describe what hardware must be developed and what software is to be written. Justify any design decisions that you make.
2. Code the hardware design in Verilog (or VHDL) and simulate it. Give a justification for your choice of test sequence. Why does it do a good job of testing the design?
3. Develop a flow chart (or pseudocode) for the software for this design. Write the code and simulate it with the fidex software simulator.
4. Design the complete system using the Picoblaze processor with all glue logic needed to communicate between the hardware and software for this design.

Problem 2

You are to design a simple countdown timer that is sort of like the cooking timer you might find in many kitchens. Using the keypad from the last lab, you will set the input time in minutes (up to 9 minutes and must be a valid time). If the “CLEAR” key (C on the keypad) is hit it will cause the display to clear so that you can reenter a new valid time. If the “ENTER” key (E on the Keypad) is hit it will cause the timer to begin counting down. The timer starts with the “ENTER” input switch that starts counting the time down until it reaches “0”. The other keys on the keypad will just cause the display to hold the present data except for the “CLEAR” key.

It will count down the time (in seconds) until time has elapsed, when an alarm will goes off (indicating time has expired). The alarm will go off for 5 seconds, while flashing the Led’s on the display before it allows another new input from the keyboard. It then waits for new data to be input.

Your assignment is as follows.

1. Formalize the design by determining how you will partition the hardware and software for this design. Describe what hardware must be developed and what software is to be written. Justify any design decisions that you make.
2. Code the hardware design in Verilog (or VHDL) and simulate it. Give a justification for your choice of test sequence. Why does it do a good job of testing the design?
3. Develop a flow chart (or pseudocode) for the software for this design. Write the code and simulate it with the fidex software simulator.
4. Design the complete system using the Picoblaze processor with all glue logic needed to communicate between the hardware and software for this design.

Deliverables

1. Each problem will need to be demonstrated to the instructor running on the Nexys 3 development board.
2. Formal Lab writeup as done in Lab4 covering both problems.
3. Zip up all design files and upload to blackboard