

Experiment 7 Fall 2012

Count UP/DOWN Timer Using The SPI Subsystem

Due: Week 9 lab Sessions (10/23/2012)

Design and implement a one second interval (and high speed 0.05 sec) count down/up timer system with the following specification:

I. Input:

1. 4X4 Matrix Keypad

- a. Keys 0 - 9 used for single digit numeric input
- b. Assign XIRQ pin for START/STOP
- c. Assign IRQ pin for UP/DOWN
- d. Assign RESET pin to Reset
- e. Assign the F key for High Speed counting (Count every 0.05 sec.)
- f. Assign the A key for Low Speed 1 second counting

II. Output:

Same setup as experiment 5.

1. Use two seven segment displays (with 470 Ohm current limiting resistors.)

The 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. Use them. DO NOT run the seven segment displays with out the current limiting resistors. The displays should be connected to (i.e. driven by) 74HC373 latches. The latches will be driven by a SIPO IC which is driven by the NanoCore12 SPI system pins.

III. Operation:

At power up the display should be blank.

1. Press a single digit key and the least significant digit display (ones place) should display this first digit.
2. Press a single digit key and the ones place digit moves over to the tens place and the new digit move into the ones place. The above step should continue to function in the same manner for more key presses. The system needs to save only the last two keys pressed.
3. Press UP/DOWN to set the count direction
4. Press the START/STOP key and timer should begin to count down (or up) in one second intervals. If START/STOP is pressed again the counting should pause. The counting should resume when START/STOP is pressed again. So, START/STOP is a toggle function.
5. When the count reaches 00 (note: count up sequence is 98,99,00) the display should flash the 00 off and on rapidly 3 times and then the display should go blank.
6. If RESET is pressed at any time the display should go blank and wait for new input as in item 1

above.

Note: If START/STOP is pressed after only a single digit has been pressed then countdown (or up) should commence from the single digit.

You are required to use a "Look-up Table" to store the 7-Segment data for **10 decimal digits**. There can be ten or more data values stored in the look-up table. You then will use Index Addressing to locate the required table data for each digit of the current count.

Your display should count accurately.

We will need to reduce the number of output lines we have been using to make lines available for other purposes. To do this you are required to use the SPI subsystem of the NanoCore12. We will output serial data to a Serial-In-Parallel-Out (SIPO) IC. The output of the SIPO will go to the latches that drive the seven segment displays.

Move your keypad I/O to PortT (PT0-3 for columns and PT4-7 for rows)

We need three pins from PortM for the SPI system so the three other PortM pins can be used for the three required control lines.

Data sheet for the 74HC595 SIPO IC is [here](#).

Some of the special pins on the SIPO:

10 - SCLR' = Serial Clear (active low) Clear all bits of the shift register

11 - SCK = Serial Clock (Shift Clock) Active on rising edge

12 - RCK = Register Transfer Signal. Moves all bits (in parallel) from the Shift register to the Output Latch. Active on rising edge.

13 - G' = Output enable (active Low)

14 - SER = Serial Data Input Line

Experiment 7 is due WEEK 9 (10/23/2012)

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: Mon Oct 15 13:24:59 2012

ECE 367: Experiment 7

Count UP/DOWN Timer Using the SPI Subsystem

Semester: Fall 2012

Name: Kai Zhao

Signature: _____

UIN: 670720413

Due Date: 2012 October 23

Lab Section: T11

TA: Chenjie Tang

User Manual:

This program is a count up/down timer. The user should first supply power to the circuit and then press the reset button to clear the display. On startup, the counter is a slow count up counter that starts at 00. The user has an option of inputting the 2 digits where the counter should begin. The user has an option on pressing 'F' to speed up the counter to pressing 'A' to restore the slow counter speed. The user has the option of pressing the up/down button for the counter to count up or down. When the user is ready, the user should press the start/stop key to start the timer. The user may adjust count direction during the counter, but other options require the timer to be paused before adjusting.

Conclusion:

Yes, my circuit meets the specifications and function properly.

I had trouble with the RC-Schmitt trigger because the up/down button only worked about 80% of the time using the high speed strategy. Therefore, I had to use the only strategy using to 1uF capacitor. I had trouble with the SPI subsystem because I thought we had to display the pulses manually into the 74HC595. I also had trouble with the interrupts because I was not sure how to check if the button was released since XIRQ does not activate on falling edge.

I would do this project differently by check if the value is going to be 00 before ending as opposed to -1 when counting down, so that it exactly matches the count up timer.

The extra feature I have implemented is that the user input in allowed during pauses and that the count direct is adjustable even during the count.

I learned about the SPI subsystem, XIRQ interrupts, Schmitt triggers, using multiple variables, shifting register data, and set-up codes for subsystems and interrupts.