**ECE 346 Midterm Project: Up/Down Counter**

Due: beginning of class Wednesday March 4, 2009, 12:00 noon

**Problem Statement**

You are to build a counter that is controllable by user input from the console.

The counter is required to do the following:

1. The counter counts up or down (From 0-255 in decimal and 00-FF in hexadecimal).
2. The counter will count in either decimal or hexadecimal.
3. The counter resets. After reset the count is 0 and the counter stride is 1.
4. The counter stride is input by the user (0-F).
5. Counting enabled or disabled (paused) by user.
6. When enabled, the counter will advance its count every second (normal mode) or quarter second (fast mode).
7. The speed mode is specified by the user.
8. The current count will be displayed on the  7-segment display.
9. Print any relevant prompts and status/diagnostic information to the console.

The **default values** for the counter are the following:

1. Counting up.
2. Decimal counting.
3. Counter is reset.
4. Stride is 1.
5. Counting is disabled (paused).
6. The speed mode is normal.
7. Initially display a 0 on the 7-segment display.

**Detailed Specifications:**

Upon initial program execution you are to provide a menu to the user on the console. The menu should list all allowable inputs.

* Inputs:

1. **+**or **-** : For counting up or down respectively.
2. **D** or **H** : For counting in decimal or hexadecimal mode.
3. **R** : For resetting the counter
4. **S** : For entering the stride. Once the user presses S, another value from 0-F will be expected for the stride.
5. **E** or **P** : For enabling or disabling (Pausing).
6. **N** or **F** : For normal or fast mode.
7. **Q** : For quitting the program

* Considering the default values the counter should display 0 on the 7-segment display and be paused.
* You are to turn on the DP on 7-segment display, while preserving the current display value, whenever the counting is paused.
* You will need a 7-segment display. Please use the recommended 74LS06 interface. You must drive the display using PortB.
* The output on the 7-Segment display will be displayed in the following way.
  1. If the mode is decimal, then
     + show the hundred value first, wait for 0.1 seconds, then
     + show the ten value next, wait for 0.1 seconds, and
     + show the last digit for the remaining time (0.8s in normal mode or 0.05s in fast mode).
     + Example: If the counter value is 154, then you would display 1 for 0.1 seconds, 5 for 0.1 seconds and 4 for the remaining time.
  2. If the mode is hexadecimal, then
     + show the high nibble first, wait for 0.1 seconds, and
     + show the low nibble next, wait for the remaining time.
     + Example: If the counter value is A3, then you would display A for 0.1 seconds, 3 for the remaining time.
* The timing **must** be interrupt driven and use message passing between the interrupt service routine(s) and the main program. You must use at least one interrupt, but more may be used if desired.
* You may choose any reasonable time base.
* In your report, be sure to include a comprehensive description of the theory of operation, design issues, test and evaluation, your documented program. .LST, .S19, flow diagrams, and memory map. In addition, be sure to include the issues of message passing, interrupt management, timing management, and testing. Include timing diagram(s) describing both interrupt timing and program state sequence.

**Grading**

Your grade will be based on 100 points where 50 points will come from your report and 50 points will be based on the execution of your program. Note that the program you submit will not be debugged.

Therefore, as an additional requirement, you need to clearly specify the operation of your program to the potential user. The first page of your report must be a user guide, a "how to", which describes the **operational** parts of your program and how to execute it, as well as what the user inputs will do to the execution. Among the biggest challenges in this assignment is making efficient use of the limited 6811 memory. If you have not been able to meet all the project requirements, please use this user guide page to clearly state what you were able to accomplish, so that you can get partial credit.

**Honor Code**

This is a pledged assignment. Your work should reflect your individual effort. You should give any help nor should you receive any help from anyone, with the exception of the instructor. Sharing of computer environment problems is allowed. You must include a signed Honor code statement in your report.

**This project must be submitted on time at the beginning of class and no late grace will be allowed.**