ECE 376 - Homework #5

Keypads in C, Stepper Motors, NeoPixels in C. Due Monday, September 26th

Design an embedded system which uses the keypad and the NeoPixel. Some suggestions are...

- LED Flashlight: Input a number 0..255 on the keypad. Drive the NeoPixel at that brightness level (0..255) as white light (RGB all the same).
- LED Color Flashlight: Input a number 0..255 on the keypad. Set the brightness of RGB by pressing RB2 (R), RB1 (G), or RB2 (B).
- Starter Tree: Input a number on the keypad (N=0..100). When * is pressed, each light on the NeoPixel turns on one at a time with a delay of N*100ms per light.
- Other...

1) Requirements: Specify the inputs / outputs / how they relate.

2) C code, flow chart, and resulting number of lines of assembler

3) Validation: Collect data in lab to verify you met the requirements.

4) Demo. Video or in person.

Design an embedded system which uses the keypad and the stepper motor. Some suggestions are...

- Roulette Wheel: Bet on a winning number with the keypad (0..7). Press RB0. The stepper motor then spins two rotations at 10ms / step and then stops at the winning number (N*25 steps)
- Count-Down Timer: Input how many seconds you want to wait (00 to 99). The stepper motor then turns at a rate of 10ms / step and stops when time is up.
- Stepper Motor Speed Control: Input a number with a keypad (010 to 999) then hit RB0 or RB1. The stepper motor then turns clockwise (RB0) or counter-clockwise (RB1) at on step every N ms.
- Stepper Motor Position Control: Input a number with a keypad (000 to 999) then hit '*'. The stepper motor then turns to that angle (in steps) at a rate of 10ms / step.
- Stepper Motor RPN Calculator. Add, subtract, multiply, and divide using RPN notation. Have the stepper motor point to the answer (# steps = answer in X register)
- Combination Lock: Input the correct password (0000 to 9999) and press '*'. If input correctly, the stepper motor will turn 50 steps (door open), wait 2 seconds, the close (go back to zero steps).
- Other...

5) Requirements: Specify the inputs / outputs / how they relate.

6) C code, flow chart, and resulting number of lines of assembler

7) Validation: Collect data in lab to verify you met the requirements.

8) Demo. Video or in person.