

ECE 376 - Homework #5

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

NeoPixel Flashlight

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

- Input a number from 0..255 using the keypad
- Press RB0
- The NeoPixel then lights up with a white light at that brightness level (0..255)

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

Code: Main Loop

```
// NeoPixel Flashlight
//
// Input a number from 000 to 255
// Press RB0
// NeoPixel turns on at that brightness

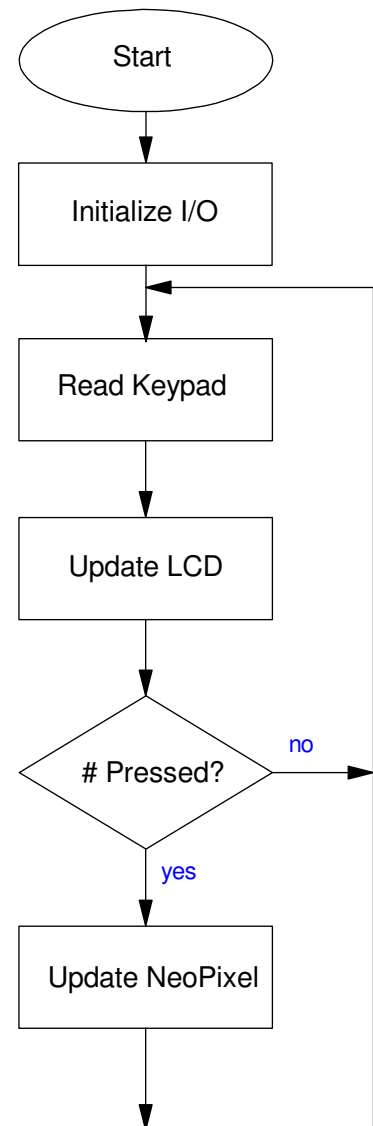
// Global Variables

unsigned char PIXEL @ 0x000;

const unsigned char MSG0[21] = "Flashlight
";
const unsigned char MSG1[21] = "Level:
";

// Subroutine Declarations
#include <pic18.h>

// Subroutines
#include "lcd_portd.c"
```



Compiler Results

Memory Summary:

Program space	used	10E6h (4326)	of	10000h bytes	(6.6%)
Data space	used	2Ch (44)	of	F80h bytes	(1.1%)
EEPROM space	used	0h (0)	of	400h bytes	(0.0%)
ID Location space	used	0h (0)	of	8h nibbles	(0.0%)
Configuration bits	used	0h (0)	of	7h words	(0.0%)

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

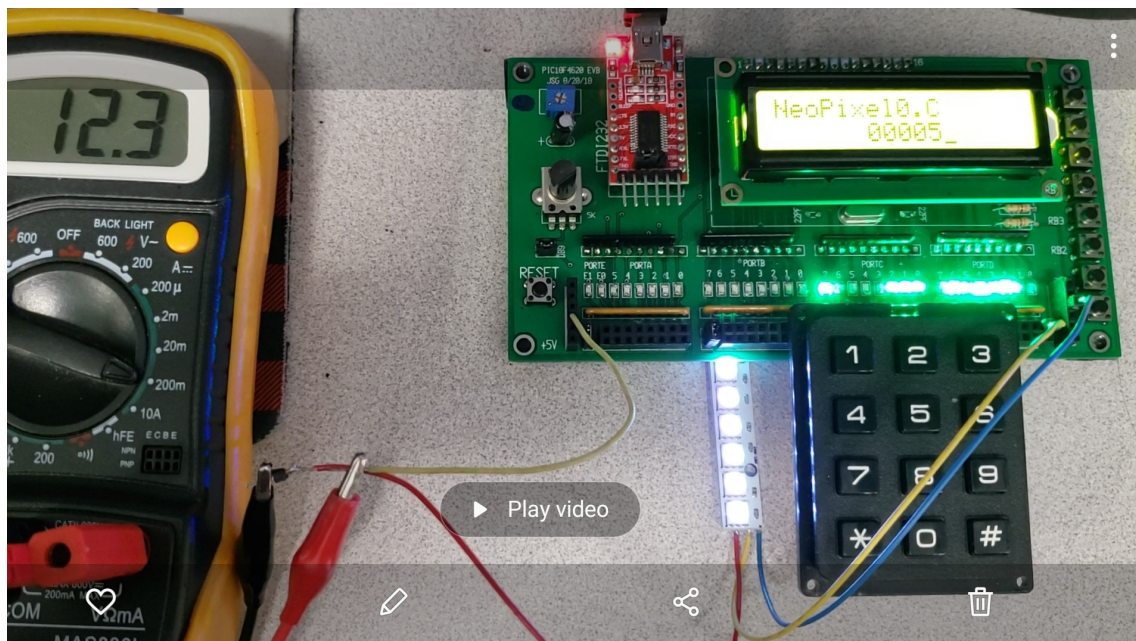
Requirement: Input a number from 000 to 255 using the keypad

- Input 000 (works)
- Input 255 (works)
- Input 123 (works)

Requirement: Press #. The NeoPixel goes to that brightness (255 = 100%)

Input Number	NeoPixels	Current (mA)	% Full Scale theory	% Full Scale measured
0	off	7.1	0%	0.0%
5	dim	12.0	1.9%	1.9%
50		58.9	19.6%	20.48%
100		110.0	39.2%	40.69%
255	really bright	260	100%	100.0%

4) Demo. Video or in person.



Analog Inputs

5) Determine how long it takes to do an A/D conversion with a PIC processor

```
void main(void)
{
    TRISC = 0;
    ADCON1 = 0x0F;

    // Turn on the A/D input
    TRISA = 0xFF;
    TRISE = 0x0F;
    ADCON2 = 0x95;
    ADCON1 = 0x07;
    ADCON0 = 0x01;

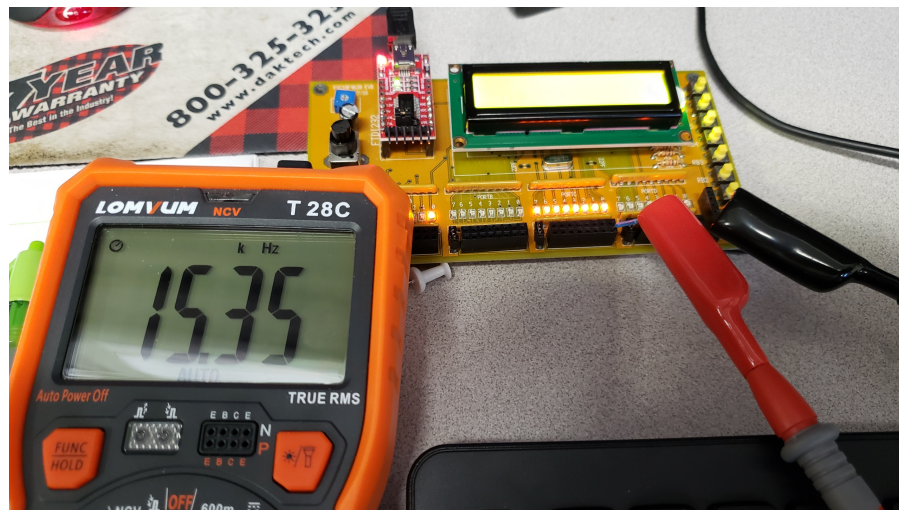
    while(1) {
        A2D = A2D_Read(0);
        PORTC = PORTC + 1;
    }
}
```

$f = 15.35\text{kHz}$

$$N = \left(\frac{10,000,000}{2 \cdot Hz} \right) = 325.73$$

It takes about 32.57us to do an A/D read

(a little less if you take the time to count into account)



6) Assume the A/D reads 275 for the following circuit.

- What is the voltage, V_x ?
- What is the resistance, R_t ?
- What is the temperature?

Assume

$$R_t = 1000 \cdot \exp\left(\frac{3905}{T+273} - \frac{3905}{298}\right) \Omega$$

V_x is proportional to the A/D reading

$$V_x = \left(\frac{275}{1023}\right) 5.00V$$

$$V_x = 1.3441V$$

R_t comes from voltage division:

$$V_x = \left(\frac{R_t}{R_t + 1000}\right) 5V$$

$$R_t = \left(\frac{V_x}{5 - V_x}\right) 1000\Omega$$

$$R_t = 367.647\Omega$$

Temperature comes from the thermistor equation

$$367.647\Omega = 1000 \cdot \exp\left(\frac{3905}{T+273} - \frac{3905}{298}\right) \Omega$$

$$T = 49.63C$$

Stepper Motor Angle Control

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

Input:

- Analog Input: 0..255

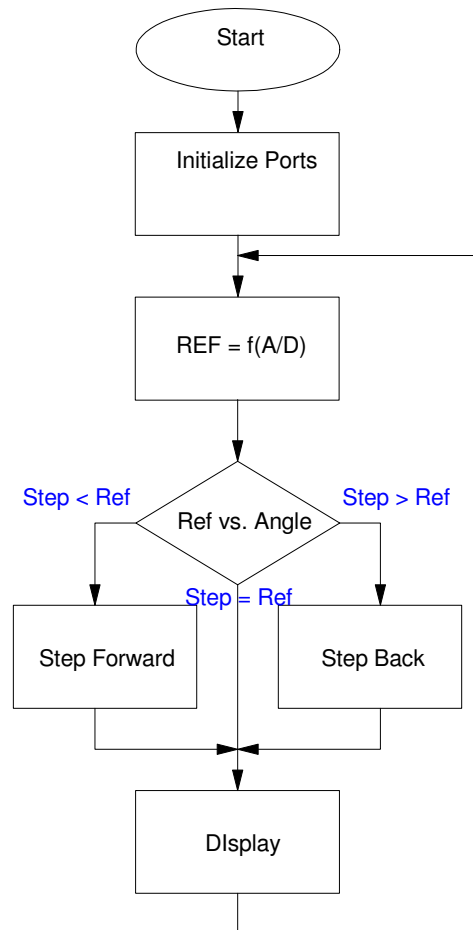
Output:

- Stepper Motor

Relationship

- Input a number from 000 to 255 using the analog input
- The stepper motor then moves to that number of steps
- At a rate of 30ms / step, +/- 5ms

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



< code >

Compilation Results

Memory Summary:

Program space	used	B1Ah (2842)	of 10000h bytes	(4.3%)
Data space	used	2Dh (45)	of F80h bytes	(1.1%)
EEPROM space	used	0h (0)	of 400h bytes	(0.0%)
ID Location space	used	0h (0)	of 8h nibbles	(0.0%)
Configuration bits	used	0h (0)	of 7h words	(0.0%)

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

Requirement: Input numbers 000 to 255 with the analog input

- Knob all the way left: 000 (works)
- Knob all the way right (255) (works)
- Knob in the middle (126) (works)

Requirement: The motor goes to that angle

Input	Went To..
0	0
50	50 steps (90 degrees)
100	100 steps (180 degrees)
200	200 steps (360 degrees)

Requirement: At a rate of 30ms / step, +/- 5ms

- 255 steps took 7.47 seconds (using stopwatch)
- Time = 29.3ms / step

10) Demo. Video or in person.

