

# ECE 376 - Homework #5

Keypads in C, Stepper Motors, NeoPixels in C

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

- Valve Position: Input a number from 000 to 200. The stepper motor then goes to that angle and stops.
- Egg Timer: Input a number from 000 to 200 (0 to 20.0 seconds). The stepper motor then moves that many steps then starts stepping back to 000, one step per 100ms.
- Combination Lock: Input a 4-digit number (0000 to 9999). If the number is correct, a door opens (the stepper motor moves to 180 degrees, waits 2 seconds, then goes back to 0 degrees)
- Other

## Stepper Motor Angle Control

1) Give the requirements and flow chart for your program

Input:

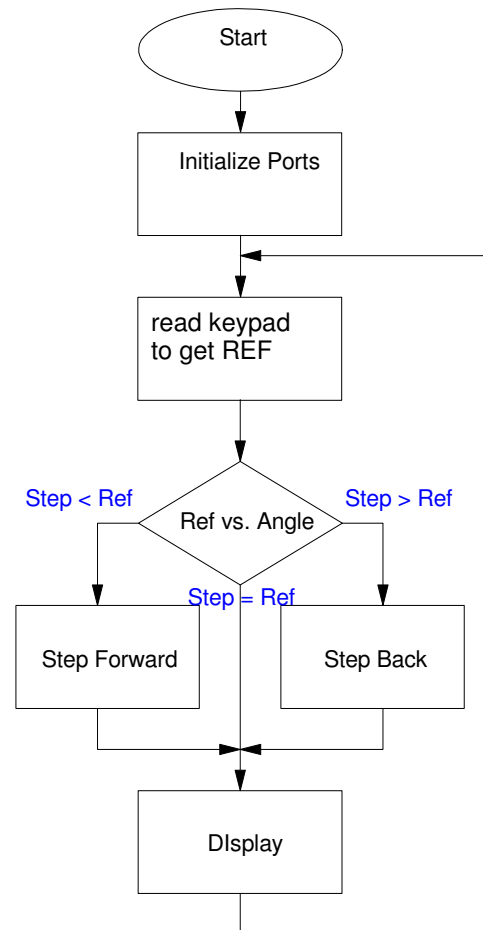
- Keypad: 0..999

Output:

- Stepper Motor

Relationship

- Input a number from 000 to 999 using the keypad
- The stepper motor then moves to that number of steps
- At a rate of 30ms / step, +/- 5ms



## 2) C code and resulting number of lines of assembler

```
void main(void)
{
    unsigned int i, j;
    int TIME, X, RUN, TEMP;
    unsigned char TABLE[4] = {1, 2, 4, 8};
    int REF, STEP;

    TRISA = TRISD = TRISE = 0;
    TRISB = 0xFF;
    TRISC = 0xF8;
    PORTA = PORTB = PORTC = PORTD = PORTE = 0;
    ADCON1 = 15;

    LCD_Init();

    LCD_Move(0,0); for (i=0; i<20; i++) LCD_Write(MSG0[i]);
    LCD_Move(1,0); for (i=0; i<20; i++) LCD_Write(MSG1[i]);
    Wait_ms(100);

    X = 0;
    STEP = 0;

    while(1) {
        TEMP = ReadKey();

        if (TEMP < 10) X = (X*10) + TEMP;

        if (TEMP == 10) {
            REF = X;
            while(REF != STEP) {
                if(STEP < REF) STEP += 1;
                if(STEP > REF) STEP -= 1;
                PORTA = TABLE[STEP % 4];
                LCD_Move(0,8); LCD_Out(REF, 3, 0);
                LCD_Move(1,8); LCD_Out(STEP, 3, 0);
                Wait_ms(200);
            }
            X = 0;
        }

        if (TEMP == 11) {
            X = X / 10;
        }

        LCD_Move(0,8); LCD_Out(X, 3, 0);
        LCD_Move(1,8); LCD_Out(STEP, 3, 0);

        Wait_ms(100);
    }
}
```

## 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%)

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

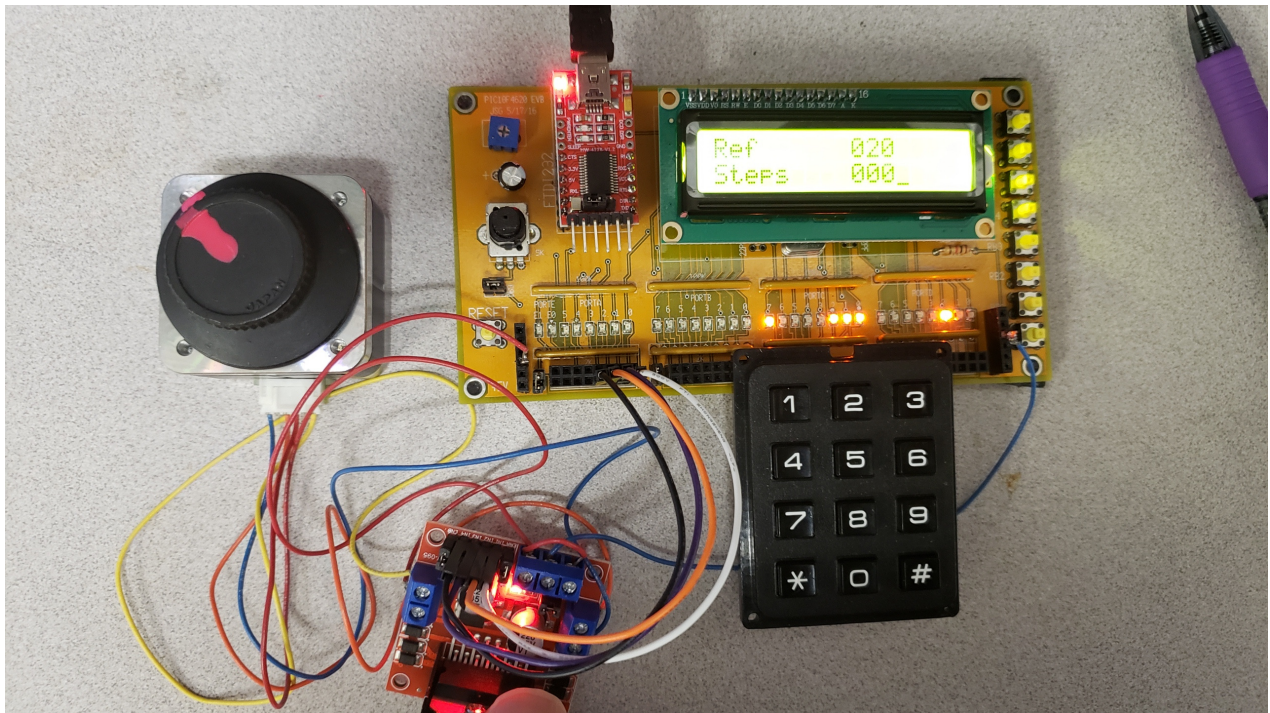
Requirement: Input a number from 000 to 255 using the keypad

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

Requirement: Press #. The stepper goes to that angle

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

4) Demo. Video or in person.



## NeoPixel Flashlight

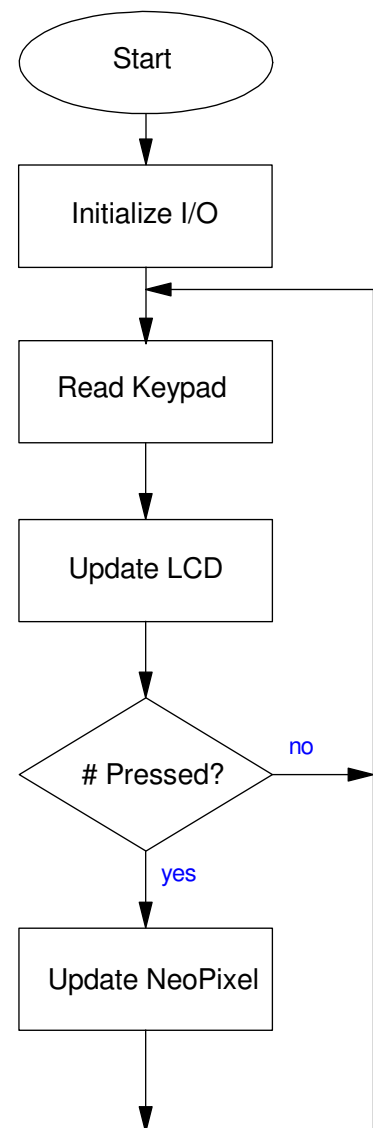
5) Requirements: Specify the inputs / outputs / how they relate.

- 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)

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

Code: Main Loop

```
while(1) {  
    TEMP = ReadKey();  
  
    if (TEMP < 10) X = (X*10) + TEMP;  
  
    if (TEMP == 10) {  
        Y = X;  
        X = 0;  
        NeoPixel_Display(Y, Y, Y);  
        NeoPixel_Display(Y, Y, Y);  
        NeoPixel_Display(Y, Y, Y);  
        NeoPixel_Display(Y, Y, Y);  
        NeoPixel_Display(Y, Y, Y);  
        NeoPixel_Display(Y, Y, Y);  
        NeoPixel_Display(Y, Y, Y);  
        NeoPixel_Display(Y, Y, Y);  
        NeoPixel_Display(Y, Y, Y);  
        NeoPixel_Display(Y, Y, Y);  
        NeoPixel_Display(Y, Y, Y);  
        X = 0;  
    }  
  
    if (TEMP == 11) {  
        X = X / 10;  
    }  
  
    LCD_Move(0,11); LCD_Out(Y, 3, 0);  
    LCD_Move(1,11); LCD_Out(X, 3, 0);  
}  
}
```



## 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%)

7) 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%

8) Demo. Video or in person.

