

ECE 376 - Homework #4

C Programming and LCD Displays. Due Monday, September 26th

1) Determine how many clocks the following C code takes to execute

- Compile and download the code (modify working code and replace the main loop)
- Measure the frequency you see on RC0 (toggles every loop).
 - Use an oscilloscope - or -
 - Connect a speaker to RC0 with a 200 Ohm resistor and measure the frequency with a cell phone app like Piano Tuner
 - RC1 is 1/2 the frequency of RC0, RC2 is 1/4th, RC3 = 1/8th, etc
- The number of clocks it takes to execute each loop is

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

1a) Counting mod 128

```
unsigned char i
while(1) {
    i = (i + 1) % 128;
    if(i == 0) PORTC += 1;
}
```

1b) Counting mod 127

```
unsigned char i
while(1) {
    i = (i + 1) % 127;
    if(i == 0) PORTC += 1;
}
```

1c) Long Integer Addition

```
unsigned long int A, B, C;
unsigned char i;
A = 0x12345678;
B = 0;
while(1) {
    i = (i + 1) % 128;
    if (i == 0) PORTC += 1;
    B = B + A;
}
```

1d) Floating point division

```
float A, B, C;
A = 3.14159265379;
B = 2.718281828;
while(1) {
    i = (i + 1) % 128;
    if(i == 0) PORTC += 1;
    C = A / B;
}
```

\$65 Voting Machine

2) Write a C program which turns your PIC into a voting machine capable of counting up to 65,535 votes per candidate (16-bit numbers):

- The LCD display shows the vote total for 4 candidates: A, B, C, & D
- When RB3 is pressed and released, A increments by 1 (vote for A)
- When RB2 is pressed and released, B increments by 1 (vote for B)
- When RB1 is pressed and released, C increments by 1 (vote for C)
- When RB0 is pressed and released, D increments by 1 (vote for D)
- When RB7 is pressed, all votes are cleared (A=B=C=D=0)

3) How many lines of assembler does your code compile into?

4) Collect data to verify your voting machine works (each press results in one vote for the correct candidate)

\$65 Banjo - 20 Sided Die - Roulette Wheel - Other

Design an embedded system which uses the LCD display and C programming. Some suggestions are

- PIC Banjo. Play notes (G4, C3, G3, B4, D4) when you press buttons RB0..RB4. Display the note you're playing on the LCD display.
- 20-Sided Die: Roll a d20 every time you press RB0 with the roll displayed on the LCD display. When you roll a 20, play 220Hz on RC0 for 1/2 second.
- Roulette Wheel: Roll an 8-sided die when you press RB0. Have a light on PORTC shift left 32+N times at a speed of 100ms/step when you press the button. Beep at 220Hz for 1/10 second each step on RA1. Display the winning number on the LCD display.
- Door Lock: Unlock a door (PORTA turns on for 1 second) when you input the correct key: RB0 - RB1 - RB2 - RB3.
- 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