23I-6006
23I-6012
23I-6077
23I-6009

Lab 11

Motor code:

```
#include <xc.h>

#define _XTAL_FREQ 20000000UL   // 20 MHz crystal

// CONFIGURATION BITS
#pragma config FOSC = HS
#pragma config WDTE = OFF
#pragma config PWRTE = ON
#pragma config BOREN = ON
#pragma config LVP = OFF
#pragma config CPD = OFF
#pragma config WRT = OFF
#pragma config CP = OFF

// ----------------------------------------
// Custom delay in microseconds
// ----------------------------------------
void delay_us_custom(unsigned int us)
{
   while(us--)
   {
      __delay_us(1);
   }
}

// ----------------------------------------
// Servo pulse output 0°–180°
// ----------------------------------------
void servo_write(unsigned int angle)
{
   unsigned int pulse;   // 500 to 2500 microseconds
   pulse = 500 + (angle * 2000UL) / 180UL;

   // HIGH pulse
   PORTBbits.RB0 = 1;
```

```
  delay_us_custom(pulse);

  // LOW for rest of 20 ms frame
  PORTBbits.RB0 = 0;
  delay_us_custom(20000
```

- pulse);
  }

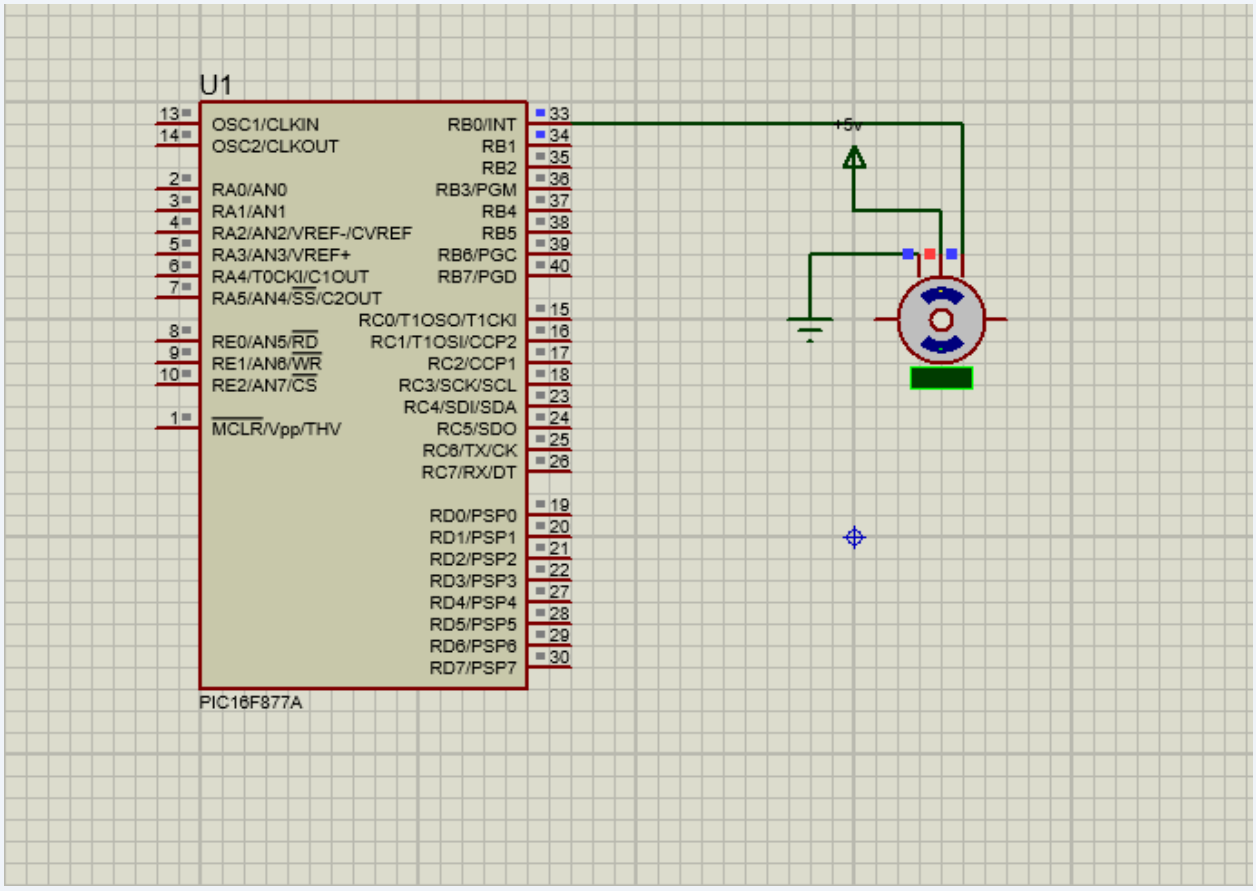
```
void main(void)
{
  // Configure RB0 as output
  TRISBbits.TRISB0 = 0;
  PORTBbits.RB0 = 0;

  unsigned int angle = 0;
  int step = 1;   // +1 = sweep up, -1 = sweep down

  while(1)
  {
    // Send one frame at current angle
    servo_write(angle);

    // Update angle gradually
    angle += step;

    // Reverse direction at limits
    if(angle >= 180) step = -1;
    if(angle == 0)   step = +1;
  }
```


U1
13
14
2
3
4
5
6
7
8
9
10
1
OSC1/CLKIN
OSC2/CLKOUT
RA0/AN0
RA1/AN1
RA2/AN2/VREF-/CVREF
RA3/AN3/VREF+
RA4/T0CKI/C1OUT
RA5/AN4/SS/C2OUT
RE0/AN5/RD
RE1/AN6/WR
RE2/AN7/CS
MCLR/Vpp/THV
33
34
35
36
37
38
39
40
15
16
17
18
23
24
25
26
19
20
21
22
27
28
29
30
RB0/INT
RB1
RB2
RB3/PGM
RB4
RB5
RB6/PGC
RB7/PGD
RC0/T1OSO/T1CKI
RC1/T1OSI/CCP2
RC2/CCP1
RC3/SCK/SCL
RC4/SDI/SDA
RC5/SDO
RC6/TX/CK
RC7/RX/DT
RD0/PSP0
RD1/PSP1
RD2/PSP2
RD3/PSP3
RD4/PSP4
RD5/PSP5
RD6/PSP6
RD7/PSP7
PIC16F877A
+5v

```
#include <xc.h>
```

```
#define _XTAL_FREQ 20000000UL // 20 MHz crystal

// CONFIGURATION BITS
#pragma config FOSC = HS
#pragma config WDTE = OFF
#pragma config PWRTE = ON
#pragma config BOREN = ON
#pragma config LVP = OFF
#pragma config CPD = OFF
#pragma config WRT = OFF
#pragma config CP = OFF

void main(void)
{
// Configure RB2 as output for buzzer
TRISBbits.TRISB2 = 0; // RB2 = output
PORTBbits.RB2 = 0; // buzzer initially OFF

while(1)
{
// Turn buzzer ON
PORTBbits.RB2 = 1;
__delay_ms(100);
// Turn buzzer OFF
PORTBbits.RB2 = 0;
__delay_ms(4000); // buzzer OFF for 1 second
}
}
```