

Experiment no: 10

Name of the experiment, Interfacing servo motor with PIC microcontroller.

~~Theor~~
Objective(s):

1. To use how use PIC microcontroller.
2. To know the how control servo motor using PIC.

Theory Servo motor uses error sensing negative feedback to control the precise angular position. Servos are used for precise positioning in robotic arms, legs, RC Aeroplanes, Helicopters etc. Please read the article servo motor for more information about its working and construction. Hobby servo motors have three wires, two of them (RED and BLACK) are used to give power and the third one is used to give control signals. Servo can be easily be controlled using micro-controllers using pulse width modulated (PWM) signals on the control wire. Here we are using a servo whose angular rotation is limited to $0-180^\circ$. We can control the exact angular position by using a pulse, whose width varying from 1 millisecond to 2 millisecond on the control wire. The actual behaviour of a particular motor depends upon its

manufacture, please refer the datasheet of the particular motor for that.

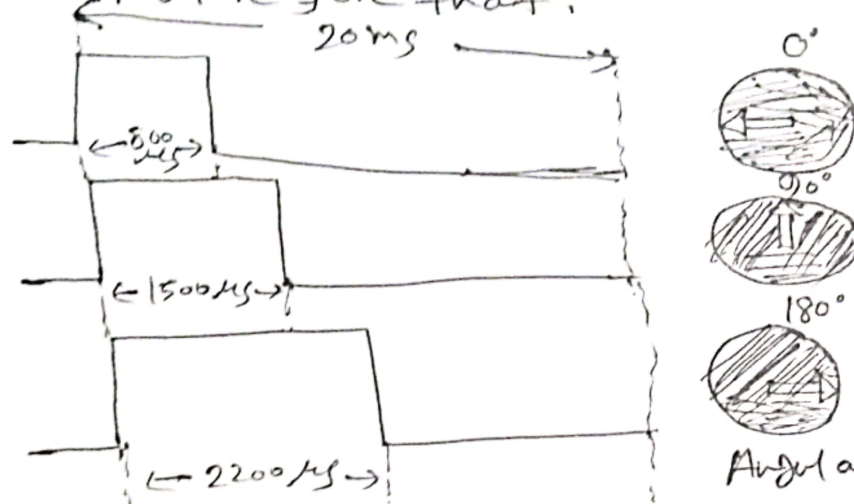


fig: working mechanism of servomotor

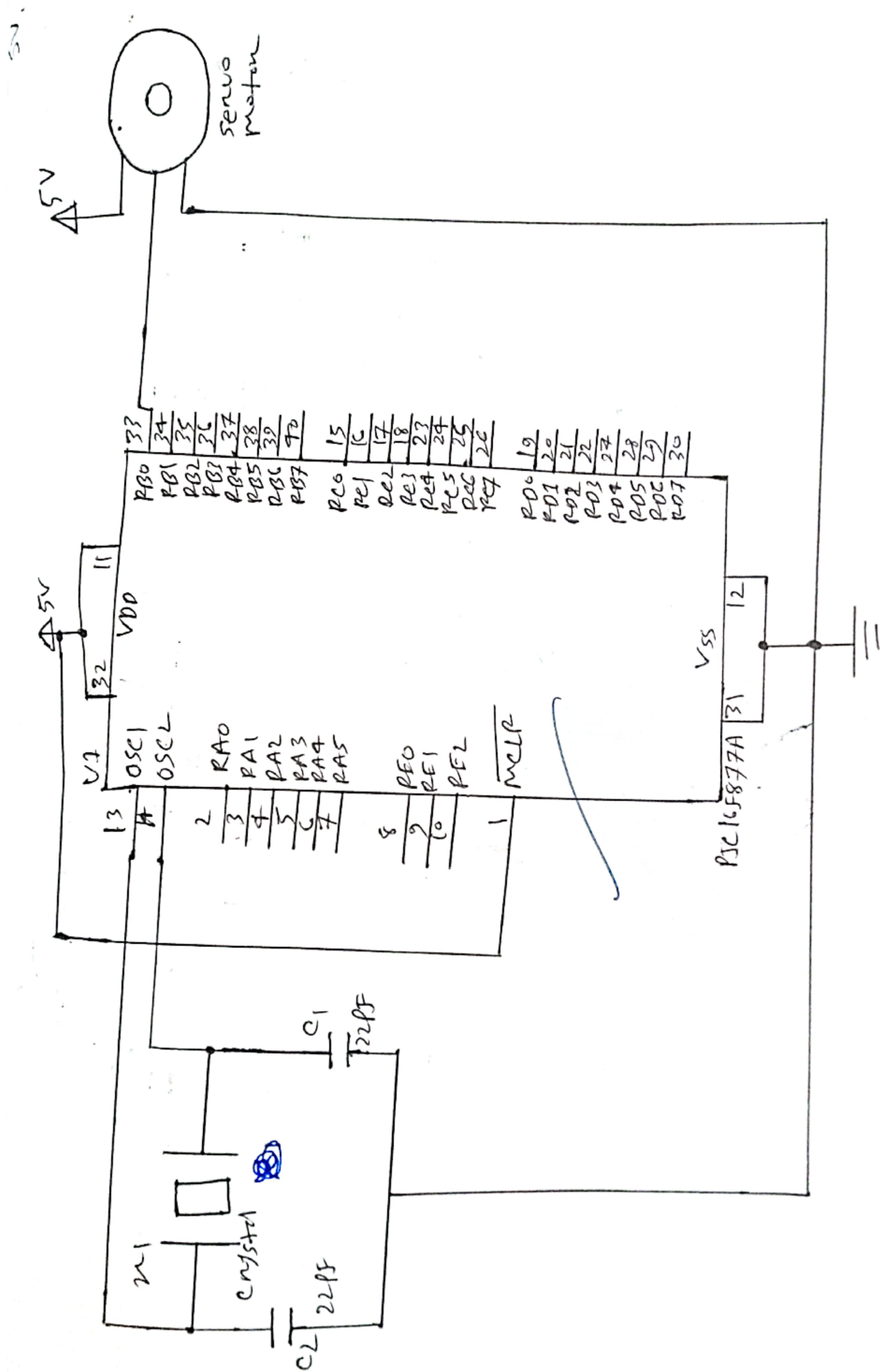


fig: DC motor speed control using PWM
 fig1 Interfacing servo motor with PIC microcontroller.

Source Code 1

```
void servoRotate0C()
```

```
{ unsigned int i;  
  for (i=0; i<50; i++)
```

```
  {  
    PORTB.F0 = 1;  
    Delay_us (800);  
    PORTB.F0 = 0;  
    Delay_us (19200);
```

```
  }  
}  
void servoRotate90C()
```

```
{ unsigned int i;  
  for (i=0; i<50; i++)
```

```
  {  
    PORTB.F0 = 1;  
    Delay_us (1500);  
    PORTB.F0 = 0;  
    Delay_us (18500);
```

```
  }  
}  
void servoRotate180C()
```

```
{ unsigned int i;  
  for (i=0; i<50; i++)
```

```
  {  
    PORTB.F0 = 1;  
    Delay_us (2200);  
    PORTB.F0 = 0;  
    Delay_us (17800);
```

```
  }  
}  
void main()
```

```
{ TRISB = 0;
```

```
do  
{  
  servoRotate 0();  
  Delay_ms (2000);  
  servoRotate 90();  
  Delay_ms (2000);  
  servoRotate 180();  
} while (1);  
}
```

