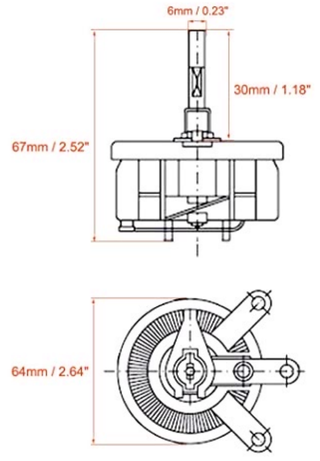
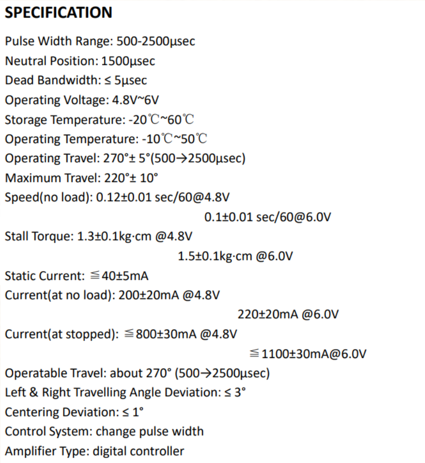
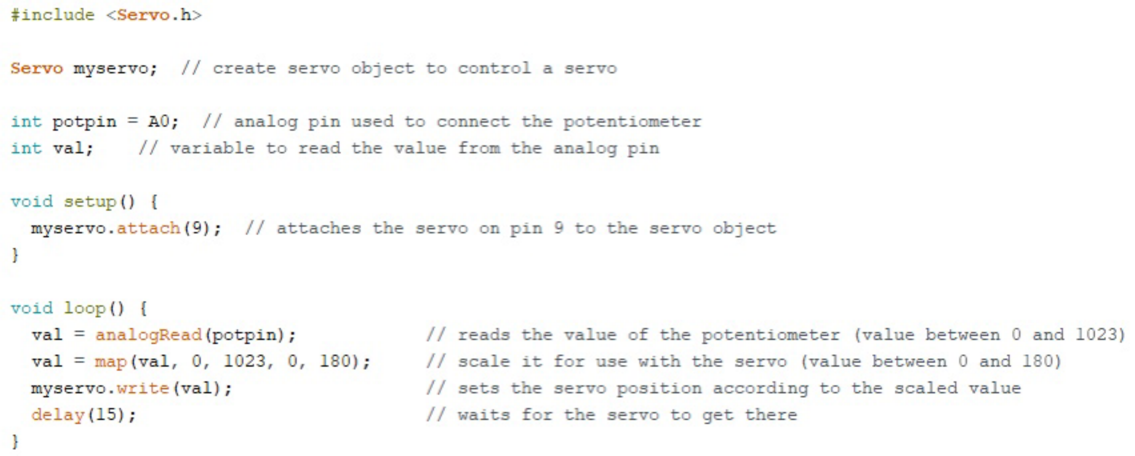
Controllable Load Resistance:
Pair of Servo motor (4.8V 270 degrees rotation servo) + Rotary potentiometer (50W 100 Ohm)
Arduino IDE - PWM signal control motor to rotate and stop at specific angle.
(Careful: the 360 degrees continuous servo will spin and never stop
20ms period 50HZ: 1.5ms stop; 1ms counterclockwise full speed; 2ms clockwise full speed.)
20ms period 50HZ: 1.5ms stop at middle position; 0.5~2.5ms rotate in 270 degrees.




SPECIFICATION

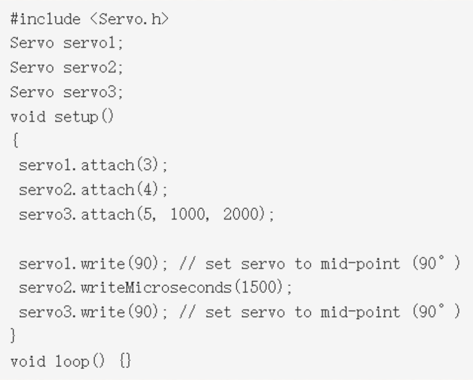
Pulse Width Range: 500-2500μsec
Neutral Position: 1500μsec
Dead Bandwidth: ≤ 5μsec
Operating Voltage: 4.8V~6V
Storage Temperature: -20℃~60℃
Operating Temperature: -10℃~50℃
Operating Travel: 270°± 5°(500→2500μsec)
Maximum Travel: 220°± 10°
Speed(no load): 0.12±0.01 sec/60@4.8V
0.1±0.01 sec/60@6.0V
Stall Torque: 1.3±0.1kg·cm @4.8V
1.5±0.1kg·cm @6.0V
Static Current: ≦40±5mA
Current(at no load): 200±20mA @4.8V
220±20mA @6.0V
Current(at stopped): ≦800±30mA @4.8V
≦1100±30mA@6.0V
Operatable Travel: about 270° (500→2500μsec)
Left & Right Travelling Angle Deviation: ≤ 3°
Centering Deviation: ≤ 1°
Control System: change pulse width
Amplifier Type: digital controller

Arduino servo example code:
Use potentiometer to control servo motor angle position.



```
#include <Servo.h>

Servo myservo;  // create servo object to control a servo

int potpin = A0;  // analog pin used to connect the potentiometer
int val;    // variable to read the value from the analog pin

void setup() {
  myservo.attach(9);  // attaches the servo on pin 9 to the servo object
}

void loop() {
  val = analogRead(potpin);            // reads the value of the potentiometer (value between 0 and 1023)
  val = map(val, 0, 1023, 0, 180);     // scale it for use with the servo (value between 0 and 180)
  myservo.write(val);                  // sets the servo position according to the scaled value
  delay(15);                           // waits for the servo to get there
}
```

For this project, the servo motor mounted with a rotary potentiometer, the angle position is received by deserialization of a Jason message. This time, the rotation range is 270 degrees. The datasheet for the 270 degree motor: pulse width range 0.5~2.5ms and 5us resolution; 50Hz 20ms period.
But entire range of servo rotate angle is 180 steps, which will lose rotate resolution.
Better to use myservo.attach(3, 500, 2500);   //states the min and max setting are 500us and 2500us
                    mservo.writeMicroseconds(500); //sets the servo position at 0 degree
Minimum increment 5us



```
#include <Servo.h>
Servo servo1;
Servo servo2;
Servo servo3;
void setup()
{
 servo1.attach(3);
 servo2.attach(4);
 servo3.attach(5, 1000, 2000);

 servo1.write(90); // set servo to mid-point (90° )
 servo2.writeMicroseconds(1500);
 servo3.write(90); // set servo to mid-point (90° )
}
void loop() {}
```