

```
/*P17310 Latch Arm Control Box
 * *** TO LOAD CODE ONTO ARDUINO MAKE SURE BLUETOOTH
MODULE IS DISCONNECTED
 * This Code is for the Arduinio Inside the Latch
Mechanism Control Box which is the Slave mode Bluetooth
Module
 * THE set up for the control box can be found in the
following webpage
 * http://edge.rit.
edu/edge/P17310/public/Integrated%20System%20Build%20%26
%20Test%20with%20Customer%20Demo
 * This Code will do the following given the following
inputs from the Master Mode Arduino through Bluetooth
if given '1' Power Relay Turns On Direction Relay Turns
On
if given '2' Power Relay Stays On and Direction Relay
Turns Off
if given '3' Both Power and Dirrection Relay Turn Off,
Pressure Sensor Takes a Reading
if given '4' DC Motor CounterClockwise Pulse
if given '5' DC motor Clockwise Pulse
if given '6' Sttepper Motor Rotates and lifts the Latch
Arm Hook, and brings it back down
*/

#include <Wire.h>
#include <Adafruit_MotorShield.h> //Motor Shield library
#include "utility/Adafruit_MS_PWMServoDriver.h" //Motor
Shield library

Adafruit_MotorShield AFMS = Adafruit_MotorShield(0x61);
```

```
//Address to program to the Adafruit Motor Shield
```

```
Adafruit_StepperMotor *myStepperMotor = AFMS.  
getStepper(200, 2); // stepper motor using M3 and M4 on  
the motor shield  
//the stepper motor is 1.8 degrees so one revolution is  
200 steps
```

```
Adafruit_DCMotor *myMotor = AFMS.getMotor(1); //Dc  
motor using M1 on motor shield
```

```
int fsrPin = 0; // the FSR and 1M resistor pulldown  
are connected to a0 for pressure sensor  
int fsrReading; // the analog reading from the FSR  
resistor divider  
int ddirection = 12; //direction relay pin digital 12  
int dpower = 13; //power relay pin digital 13  
int state = 0;
```

```
void setup() {  
  
    // Setup 2 pins as OUTPUT  
    pinMode(ddirection, OUTPUT);  
    pinMode(dpower, OUTPUT);  
  
    digitalWrite(ddirection, LOW); //direction for relay  
system  
    digitalWrite(dpower, LOW); //power for relay system  
    Serial.begin(38400); // Communication rate of the  
Bluetooth module has to match the Slave Mode  
communication Rate
```

```

AFMS.begin();
myStepperMotor->setSpeed(20); // Stepepr Motor 20
rpm
AFMS.begin(); //frequency set at 1.6 Khz
myMotor->setSpeed(55); //DC motor 55 rpm (max value
is 250 min is 0)
}

void loop() {
  if(Serial.available() > 0){ // Checks whether data is
coming from the serial port
    state = Serial.read(); // Reads the data from the
serial port
  }

  if ( state =='1') {
    Serial.println("Relay Allowing for Dome Slit to
Open");
    digitalWrite(ddirection, HIGH); // Turn direction
relay high
    digitalWrite(dpower, HIGH); // turn power relay
high
    state = 0;
  }

  if ( state =='2') {
    Serial.println("Relay Allowing for Dome Slit to
Close");
    digitalWrite(dpower, HIGH); // turn Power Relay
High
  }
}

```

```

    digitalWrite(ddirection, LOW);    // Turn direction
Relay Low
    state = 0;
}

if ( state =='3') {
    digitalWrite(ddirection, LOW);    // Turn Dirrection
Relay Off
    digitalWrite(dpower, LOW);    // Turn off Power Relay
off
    myMotor->run(RELEASE);
        Serial.println("Pressure Sensor for the Dome Slit
Position If Pressure Sensor Anlog Reading less than 500
It is Open, Otherwise it Closed");
for (int x = 0; x<1; x++){//scans the for loop once

    fsrReading = analogRead(fsrPin);    //recieves abalog
reading

    Serial.print("Analog Reading = ");
    Serial.print(fsrReading);    // the raw analog
reading output

    // We'll have a few threshholds, qualitatively
determined
    if (fsrReading > 500) {
        Serial.println(" Dome Slit Closed"); // If
preswsure reading is ablove 500 the Slit is Closed

    } else {

```

```

    Serial.println(" Dome Slit Opened"); //If preswure
reading is ablove 500 the Slit is open
}
Serial.println(); // print black line
delay(10);}
state = 0; //

}
if ( state =='4') {
    Serial.println("DC Motor Forward Motion");
    myMotor->run(BACKWARD); //Runs DC motor in reverse for
200ms
    delay (200);
    myMotor->run(FORWARD); //Runs DC motor forward for
25ms
    delay (25);
    myMotor->run(RELEASE);
    state = 0;
} // wait for a second

if ( state =='5') {
    Serial.println("DC Motor Backward Motion");
    myMotor->run(FORWARD); //Runs DC motor for 255ms
    delay (255);
    myMotor->run(BACKWARD); //Runs DC motor in reverse for
15ms
    delay (15);
    myMotor->run(RELEASE); //Power to DC motor is cut off
    state = 0;
} // wait for a second

```

```
if ( state =='6') {
    Serial.println("Latch Mechanism Stepper Motor");
    myStepperMotor->step(180, FORWARD, DOUBLE); //Stepper
Motor moves forward for 180 steps in double step
    myStepperMotor->release(); //power is cut to Stepper
so it does not heat up while holding position
    delay(3000); //waaits 3 seconds before proceeding
    myStepperMotor->step(180, BACKWARD, DOUBLE);
//Stepper Motor moves Reverse for 180 steps in double
step
    myStepperMotor->release(); //power is cut to Stepper
so it does not heat up while holding position
    delay(2000); // waits 2 seconds to finsh function
    state = 0; //
}
}
```