

```

#include <Servo.h>
#include <PS2X_lib.h>           //for v1.6
#include <AFMotor.h>
AF_DCMotor motor1(3);
AF_DCMotor motor2(4);

#define PS2_DAT  A5
#define PS2_CMD  A4
#define PS2_SEL  A2
#define PS2_CLK  A1
#define pressures false
#define rumble    true

//pin numbers are changeable as long as wiring is done properly

PS2X ps2x;

//Above is all the initial setup for the RC tank's movement

void setup() {
  Serial.begin(57600);

  delay(1); //added delay to give wireless ps2 module some time to startup, before configuring it

  error = ps2x.config_gamepad(PS2_CLK, PS2_CMD, PS2_SEL, PS2_DAT, pressures,
  rumble);

  if (error == 0) {
    Serial.print("Found Controller, configured successful ");
    Serial.print("pressures = ");
    if (pressures)
      Serial.println("true ");
    else
      Serial.println("false");
    Serial.print("rumble = ");
    if (rumble)
      Serial.println("true");
    else
      Serial.println("false");
    Serial.println("Try out all the buttons, X will vibrate the controller, faster as you press
harder;");
    Serial.println("holding L1 or R1 will print out the analog stick values.");
    Serial.println("Note: Go to www.billporter.info for updates and to report bugs.");
  }
}

```

```

}

else if (error == 1)
  Serial.println("No controller found, check wiring, see readme.txt to enable debug. visit
www.billporter.info for troubleshooting tips");

else if (error == 2)
  Serial.println("Controller found but not accepting commands. see readme.txt to enable debug.
Visit www.billporter.info for troubleshooting tips");

else if (error == 3)
  Serial.println("Controller refusing to enter Pressures mode, may not support it. ");

// Serial.print(ps2x.Analog(1), HEX);

type = ps2x.readType();
switch (type) {
  case 0:
    Serial.print("Unknown Controller type found ");
    break;
  case 1:
    Serial.print("DualShock Controller found ");
    break;
  case 2:
    Serial.print("GuitarHero Controller found ");
    break;
  case 3:
    Serial.print("Wireless Sony DualShock Controller found ");
    break;
}
motor1.setSpeed(200);

motor2.setSpeed(200);

}

void loop() {
  // put your main code here, to run repeatedly:

  if (error == 1) //skip loop if no controller found
    return;

```

```
if (type == 1) { //if controller is found

    ps2x.read_gamepad(false, vibrate);

    if (ps2x.ButtonPressed(PSB_CROSS)) {
        motor1.run(BACKWARD);
        motor2.run(BACKWARD);
        //the tank moves backwards
    }
    if (ps2x.ButtonReleased(PSB_CROSS)) {
        motor1.run(RELEASE);
        motor2.run(RELEASE);
        //the tank stops moving
    }

    if (ps2x.Button(PSB_CIRCLE)) {
        motor2.run(FORWARD);
        motor1.run(BACKWARD);
        //the tank rotates right
    }
    if (ps2x.ButtonReleased(PSB_CIRCLE)) {
        motor1.run(RELEASE);
        motor2.run(RELEASE);
        //the tank stops moving
    }

    if (ps2x.Button(PSB_SQUARE)) {
        motor2.run(BACKWARD);
        motor1.run(FORWARD);
        //the tank rotates left
    }
    if (ps2x.ButtonReleased(PSB_SQUARE)) {
        motor1.run(RELEASE);
        motor2.run(RELEASE);
        //the tank stops moving
    }

    if (ps2x.Button(PSB_R1)) {
```

```
    motor1.run(FORWARD);
    //the tank moves it's right motor forward
}

if (ps2x.ButtonReleased(PSB_R1)) {
    motor1.run(RELEASE);
    //the tank stops moving
}

if (ps2x.Button(PSB_L1)) {
    motor2.run(FORWARD);
    //the tank moves it's left motor forward
}

if (ps2x.ButtonReleased(PSB_L1)) {
    motor2.run(RELEASE);
    //the tank stops moving
}

if (ps2x.Button(PSB_R2)) {
    motor1.run(BACKWARD);
    //the tank moves it's right motor backward
}

if (ps2x.ButtonReleased(PSB_R2)) {
    motor1.run(RELEASE);
    //the tank stops moving
}

if (ps2x.Button(PSB_L2)) {
    motor2.run(BACKWARD);
    //the tank moves it's left motor backward
}

if (ps2x.ButtonReleased(PSB_L2)) {
    motor2.run(RELEASE);
    //the tank stops moving
}

if (ps2x.ButtonPressed(PSB_TRIANGLE)) {
    motor1.run(FORWARD);
```

```
motor2.run(FORWARD);  
//the tank moves forward  
}
```

```
if (ps2x.ButtonReleased(PSB_TRIANGLE)) {  
  motor1.run(RELEASE);  
  motor2.run(RELEASE);  
  //the tank stops moving  
  
}
```

```
delay(1);  
  
}
```