

Single Analog Thumb PS2 Joystick

Technical Manual Rev 1r0



Single Analog Thumb PS2 Joystick is similar to the 'analog' joysticks on PS2 controllers with two simple potentiometer in 2 - axis, a select button when the joystick is pressed down. Applications like remote control cars, robotic arm controls, and many more.
Compatible in gizDuino and Genuino MCU board.

General Specifications:

Input Supply Voltage: 5VDC

Output: X and Y -Axis

Analog (0 ~ 1023)

Push Button

1 - Not pressed

0 - Pressed

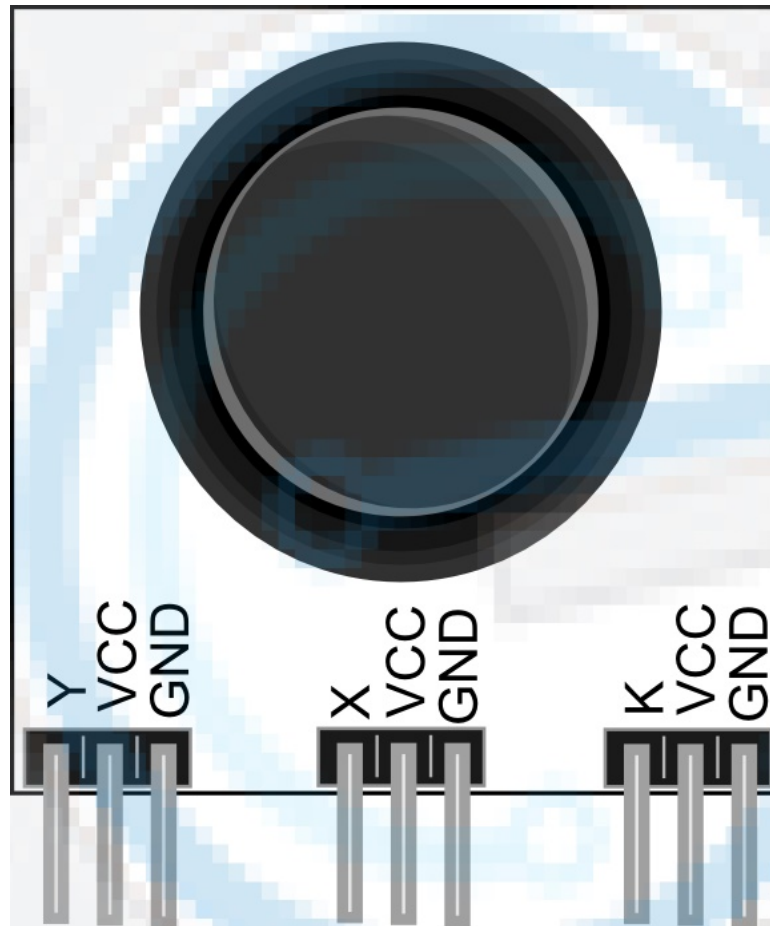


Figure 1. Major parts presentation of PS2 Analog Joystick

Wiring connections

gizDuino	PS2 Joystick
+5V	-----> VCC
GND	-----> GND
A0	-----> X-axis
A1	-----> Y-axis
D3	-----> K (Push button)

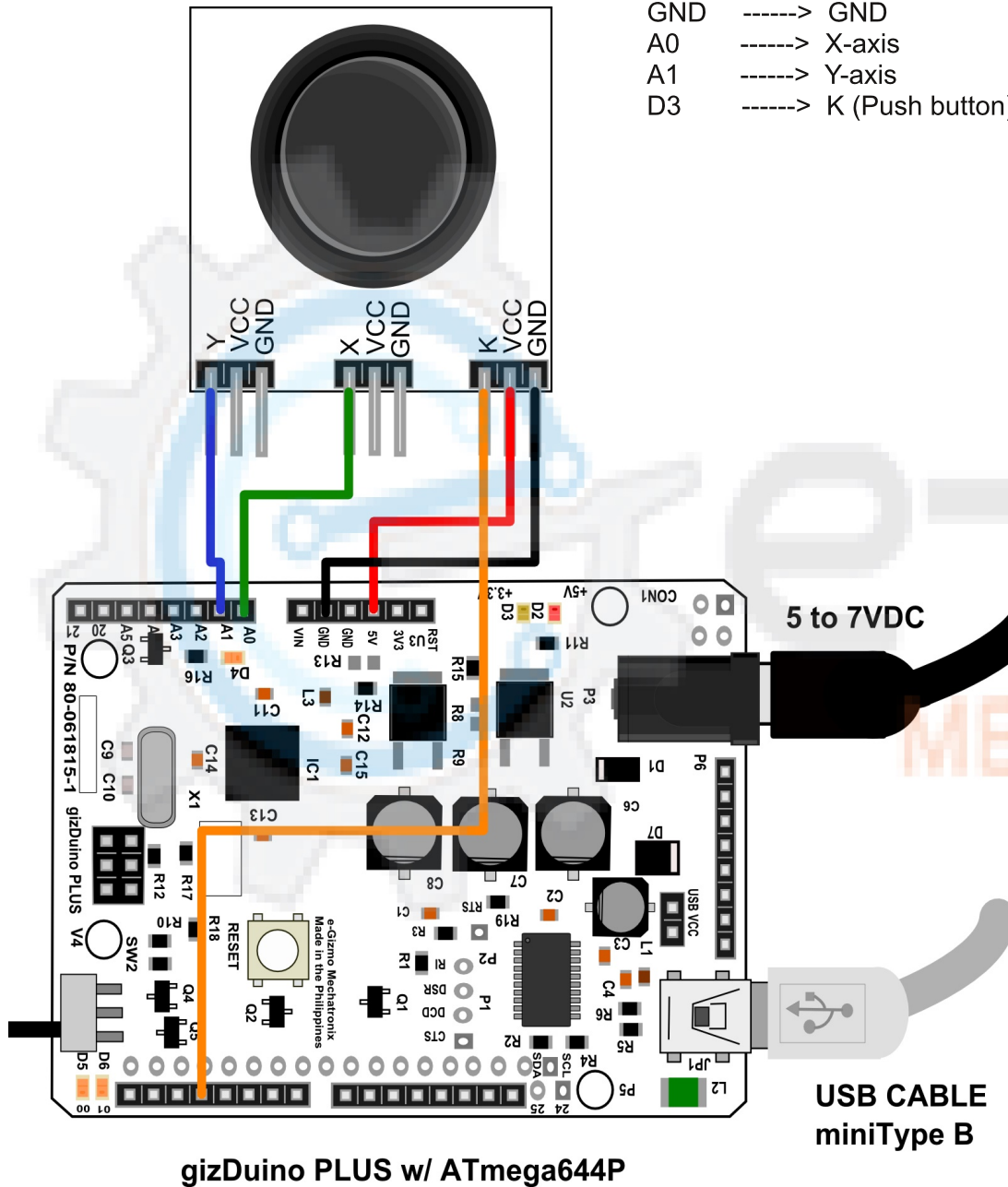


Figure 2. Sample connections

Upload this code to the gizDuino PLUS Microcontroller.
then Open the Serial Monitor to see the output value of
X and Y - axis and Push button when pressed.

```
//*****//
//      Single Thumb Analog      //
//      PS2 Joystick Controller   //
//                               //
// A sample sketch for reading the analog //
// X and Y - axis input pin and push button of //
// PS2 Controller. The out value of X and Y is //
// 0 - 1023 while in pressing down is 1. //
// Compatible in all gizDuino boards. //
//                               //
// by e-Gizmo Mechatronix Central //
// http://www.e-gizmo.com //
//                               //
//*****//
```

```
int X= A0; // X - AXIS INPUT PIN
int Y = A1; // Y - AXIS INPUT PIN
int K = 3; // PUSH BUTTON
```

```
void setup ()
{
  Serial.begin (9600);
  pinMode (K, INPUT); // K PIN SET AS INPUT
}

void loop ()
{
  int X_VALUE, Y_VALUE, BUTTON_VALUE;

  //READING THE VALUE
  X_VALUE = analogRead (X);
  Y_VALUE = analogRead (Y);
  BUTTON_VALUE = digitalRead (K);
```

```
//PRINTING THE OUTPUT VALUE
Serial.print("X = ");
Serial.println (X_VALUE, DEC);
```

```
Serial.print ("Y = ");
Serial.println (Y_VALUE, DEC);
```

```
Serial.print("BUTTON IS ");
if (BUTTON_VALUE == HIGH){
  Serial.println ("NOT PRESSED");
}
else{
  Serial.println ("PRESSED");
}
```

```
delay (500);
}
```

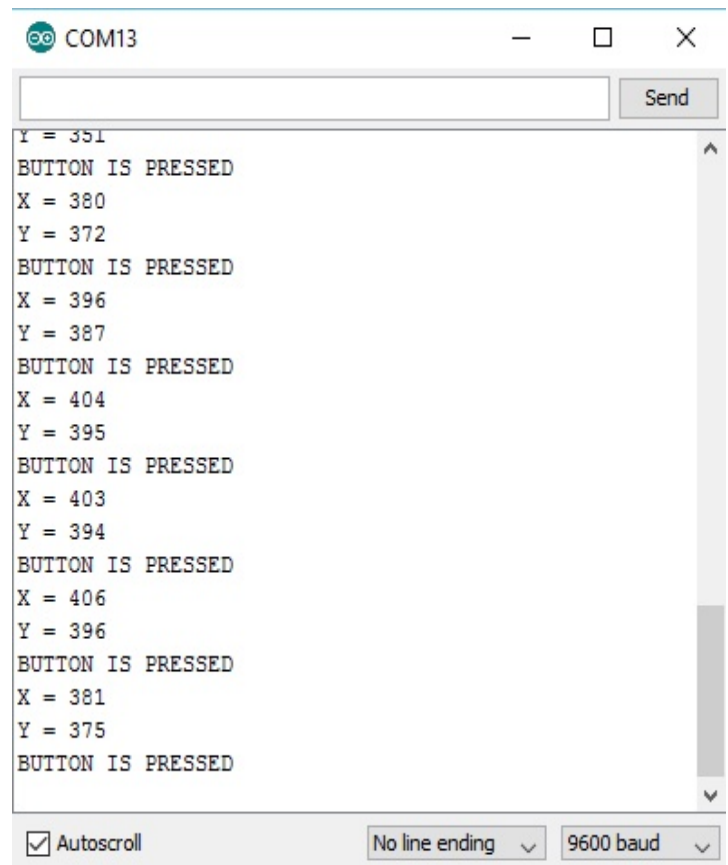


Figure 3. Serial Monitor