/\*

2-axis joystick connected to an Arduino Micro

to output 4 pins, up, down, left & right

If you are using pull down resistors, change all the HIGHs to LOWs and LOWs to HIGH.

This skectch is using pull up resistors.

\*/

String readString;

int UD = 0;

int LR = 0;

/\* Arduino Micro output pins\*/

int DWN = 27;

int UP = 29;

int LEFT = 31;

int RT = 33;

/\* Arduino Micro Input Pins \*/

int IUP=59;//A5

int ILR=63;//A9

int MID = 33; // 33 mid point delta arduino, use 4 for attiny

int LRMID = 0;

int UPMID = 0;

char randomflag = 0;

const int x\_endstopPin\_min = 3; // endstop for x-axis (min)

const int x\_endstopPin\_max = 2; // endstop for x-axis (max)

const int y\_endstopPin\_min = 14; // endstop for y-axis (min)

const int y\_endstopPin\_max = 15; // endstop for y-axis (max)

const int z\_endstopPin = 18; //endstop for z-axis

const int buttonPin = 47; //D47 // the number of the pushbutton pin

const int buttonPin2 = 32; //D32

const int ledPin = 23; // the number of the LED pin

int buttonState = 0; // variable for reading the pushbutton status

int buttonState2 = 0;

int x\_endstopState\_min = 0; //these states will contain value from reading input values

int x\_endstopState\_max = 0;

int y\_endstopState\_min = 0;

int y\_endstopState\_max = 0;

int z\_endstopState = 0;

int XL = 0;

int XR = 0;

int YU = 0;

int YD = 0;

int ZU = 0;

int ZD = 0;

int XL1 = 1000;

int XR1 = 0;

int YU1 = 0;

int YD1 = 1000;

int ZU1 = 0;

int ZD1 = 0;

int XL2 = 0;

int XR2 = 1000;

int YU2 = 0;

int YD2 = 1000;

int ZU2 = 0;

int ZD2 = 0;

int XL3 = 1000;

int XR3 = 0;

int YU3 = 0;

int YD3 = 1000;

int ZU3 = 0;

int ZD3 = 0;

int XL4 = 0;

int XR4 = 1000;

int YU4 = 0;

int YD4 = 1500;

int ZU4 = 0;

int ZD4 = 0;

int XL5 = 0;

int XR5 = 1000;

int YU5 = 1000;

int YD5 = 0;

int ZU5 = 0;

int ZD5 = 0;

int XL6 = 1000;

int XR6 = 0;

int YU6 = 1000;

int YD6 = 0;

int ZU6 = 0;

int ZD6 = 0;

int XL7 = 0;

int XR7 = 1000;

int YU7 = 1000;

int YD7 = 0;

int ZU7 = 0;

int ZD7 = 0;

int XL8 = 1000;

int XR8 = 0;

int YU8 = 1000;

int YD8 = 0;

int ZU8 = 0;

int ZD8 = 0;

int XLEND = 0;

int XREND = 0;

int YUEND = 0;

int YDEND = 0;

int ZUEND = 0;

int ZDEND = 0;

int Done = 1;

int move\_num = 0;

void mover()

{

Done = 0;

XL = XL1;

XR = XR1;

YU = YU1;

YD = YD1;

ZU = ZU1;

ZD = ZD1;

XL1 = XL2;

XR1 = XR2;

YU1 = YU2;

YD1 = YD2;

ZU1 = ZU2;

ZD1 = ZD2;

XL2 = XL3;

XR2 = XR3;

YU2 = YU3;

YD2 = YD3;

ZU2 = ZU3;

ZD2 = ZD3;

XL3 = XL4;

XR3 = XR4;

YU3 = YU4;

YD3 = YD4;

ZU3 = ZU4;

ZD3 = ZD4;

XL4 = XL5;

XR4 = XR5;

YU4 = YU5;

YD4 = YD5;

ZU4 = ZU5;

ZD4 = ZD5;

XL5 = XL6;

XR5 = XR6;

YU5 = YU6;

YD5 = YD6;

ZU5 = ZU6;

ZD5 = ZD6;

XL6 = XL7;

XR6 = XR7;

YU6 = YU7;

YD6 = YD7;

ZU6 = ZU7;

ZD6 = ZD7;

XL7 = XL8;

XR7 = XR8;

YU7 = YU8;

YD7 = YD8;

ZU7 = ZU8;

ZD7 = ZD8;

XL8 = XLEND;

XR8 = XREND;

YU8 = YUEND;

YD8 = YDEND;

ZU8 = ZUEND;

ZD8 = ZDEND;

move\_num++;

return;

}

void setup(){

Serial.begin(9600);

pinMode(ledPin, OUTPUT);

// initialize the pushbutton pin as an input:

pinMode(buttonPin, INPUT);

pinMode(x\_endstopPin\_min, INPUT);

pinMode(x\_endstopPin\_max, INPUT);

pinMode(y\_endstopPin\_min, INPUT);

pinMode(y\_endstopPin\_max, INPUT);

pinMode(z\_endstopPin, INPUT);

pinMode(DWN, OUTPUT);

pinMode(UP, OUTPUT);

pinMode(LEFT, OUTPUT);

pinMode(RT, OUTPUT);

digitalWrite(DWN, HIGH);

digitalWrite(UP, HIGH);

digitalWrite(LEFT, HIGH);

digitalWrite(RT, HIGH);

//calabrate center

LRMID = analogRead(ILR);

UPMID = analogRead(IUP);

pinMode(38, OUTPUT); //Enable X // old 6

pinMode(54, OUTPUT); //Step X // old 5

pinMode(55, OUTPUT); //Direction // old 4

pinMode(56, OUTPUT);// Enable Y

pinMode(60, OUTPUT);// Step Y

pinMode(61, OUTPUT);// Direction Y

//digitalWrite(38,LOW);

pinMode(62, OUTPUT);// Enable Z

pinMode(46, OUTPUT);// Step Z

pinMode(48, OUTPUT);// Direction Z

}

void loop(){

while (Serial.available()) {

delay(3);

char c = Serial.read();

readString += c;

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*BLUETOOTH BEGIN\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

if (readString.length() >0) {

Serial.println(readString);

if(readString == "UP")

{

randomflag = 1;

}

else if (readString == "off")

{

randomflag = 0;

}

readString="";

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*BLUETOOTH END\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

x\_endstopState\_min = digitalRead(x\_endstopPin\_min); //this obtains the "1" or "0" from endstop signal

x\_endstopState\_max = digitalRead(x\_endstopPin\_max);

y\_endstopState\_min = digitalRead(y\_endstopPin\_min);

y\_endstopState\_max = digitalRead(y\_endstopPin\_max);

z\_endstopState = digitalRead(z\_endstopPin);

buttonState = digitalRead(buttonPin);

buttonState2 = digitalRead(buttonPin2);

UD = analogRead(IUP);

LR = analogRead(ILR);

//z-direction button UP

int buttonvar = buttonState;

int buttonvar2 = buttonState2;

int x\_endstopvar\_min = x\_endstopState\_min;

int x\_endstopvar\_max = x\_endstopState\_max;

int y\_endstopvar\_min = y\_endstopState\_min;

int y\_endstopvar\_max = y\_endstopState\_max;

int z\_endstopvar = z\_endstopState;

if (!buttonvar)

{

// turn LED on:

digitalWrite(48,LOW);//direction

digitalWrite(46,HIGH);//step

digitalWrite(46,LOW);//step

delayMicroseconds(200);

}

else if(buttonvar)

{

digitalWrite(46,LOW);

}

if (!buttonvar2 && z\_endstopvar)

{

// turn LED on:

digitalWrite(ledPin, HIGH);

// Serial.println("LED on");

digitalWrite(48,HIGH);//direction

digitalWrite(46,HIGH);//step

digitalWrite(46,LOW);//step

delayMicroseconds(200);

}

else if(buttonvar || buttonvar2)

{

digitalWrite(46,LOW);

}

if (Done && (move\_num < 9))

{

mover();

}

if ( XR == 0 && XL == 0 && YU == 0 && YD == 0)

{

//Serial.println("TESTEHRFSB");

Done = 1;

}

// y-direction UP

if((((UD < UPMID - MID) || (randomflag ==1)) || (YU > 0)) && y\_endstopvar\_min ){

digitalWrite(DWN, LOW);

digitalWrite(61,LOW);

digitalWrite(UP, LOW);

digitalWrite(60,HIGH);

delayMicroseconds(200);

YU--;

}

else

{

digitalWrite(DWN, HIGH);

digitalWrite(60,LOW);

}

// y-direction DOWN

if(((UD > UPMID + MID) || (YD > 0)) && y\_endstopvar\_max ){

digitalWrite(UP, LOW);

digitalWrite(61,HIGH);

//digitalWrite(DWN, LOW);

digitalWrite(60,HIGH);

delayMicroseconds(200);

YD--;

}

else

{

digitalWrite(UP, HIGH);

digitalWrite(60,LOW);

}

// LEFT-RIGHT

//x-direciton LEFT

if(((LR < LRMID - MID) || (XL > 0)) && x\_endstopvar\_max ){// min

digitalWrite(55,LOW);

//digitalWrite(LEFT, LOW);// LEDS

digitalWrite(54,HIGH);

delayMicroseconds(200);

XL--;

}

else{

digitalWrite(54,LOW);

digitalWrite(LEFT, HIGH);

}

//x-direction RIGHT

if(((LR > LRMID + MID) || (XR > 0)) && x\_endstopvar\_min ){

digitalWrite(55,HIGH);

//digitalWrite(RT, LOW);

digitalWrite(54,HIGH);

delayMicroseconds(200);

XR--;

}else{

digitalWrite(54,LOW);

digitalWrite(RT, HIGH);

}

/\*

//x-direciton LEFT

if(LR < LRMID - MID){

digitalWrite(48,LOW);

digitalWrite(LEFT, LOW);

digitalWrite(46,HIGH);

delayMicroseconds(200);

}

else{

digitalWrite(46,LOW);

digitalWrite(LEFT, HIGH);

}

//x-direction RIGHT

if(LR > LRMID + MID){

digitalWrite(48,HIGH);

digitalWrite(RT, LOW);

digitalWrite(46,HIGH);

delayMicroseconds(200);

}else{

digitalWrite(46,LOW);

digitalWrite(RT, HIGH);

}

\*/

//delay(200);

}