Joystick five-way key Experiment

Introduction to Joystic

The most common joystick application we see is probably the game handle, of course, it is also widely used in a variety of directional control. The following is a miniature Joystick physical map.

Joystick Mechanism

In fact, the joystick structure is very simple, it contains a touch button (the Z axis) and two potentiometers (the X-and Y-axis). The Joystick controls movements depending on two contacts, one of which takes the left and right, the other takes the up and down. The joystick moving decides the positions of the contacts, which is just like the latitude and longitude of the earth, while different positions corresponds to different voltage, then the controller can read different voltage values through the AD sensor, so as to identify specific remote position.

The analog signal are input from the VRx, VRy (the X-and Y-axis) pins of the joystick above.The value of VRx from 0 to 1023 represents the position from left to right, while the value of VRy from 0 to 1023 respectively represents the position from up to down, if the button is not pressed, the two values are all 512 which is the intermediate value.

SW (Z) is a digital signal input pin that connects to the digital port and enables the pull-up resistor.

The value of SW: 1 represents Not Pressed, the 0 represents Pressed.

Aim of Experiment

The aim of this experiment is to control the LED through the joystick, making sure the corresponding 1~5 number LED lights respectively when moving the joystick upper and lower, left and right as well as pressing the joystick.

Component List

◆ Keywish Arduino UNO R3 Mainboard

◆ Breadboard

◆ USB cable

◆ Joystick *1

◆ LED light *5

◆ Current limiting resistor 220Ω*5

Wiring of Circuit

arduino Uno	Joystick
A0	3(X)
A1	4(Y)
A2	5(Z)
8	Led1(enter)
9	Led2(left)
10	Led3(up)
11	Led4(right)
12	Led5(down)
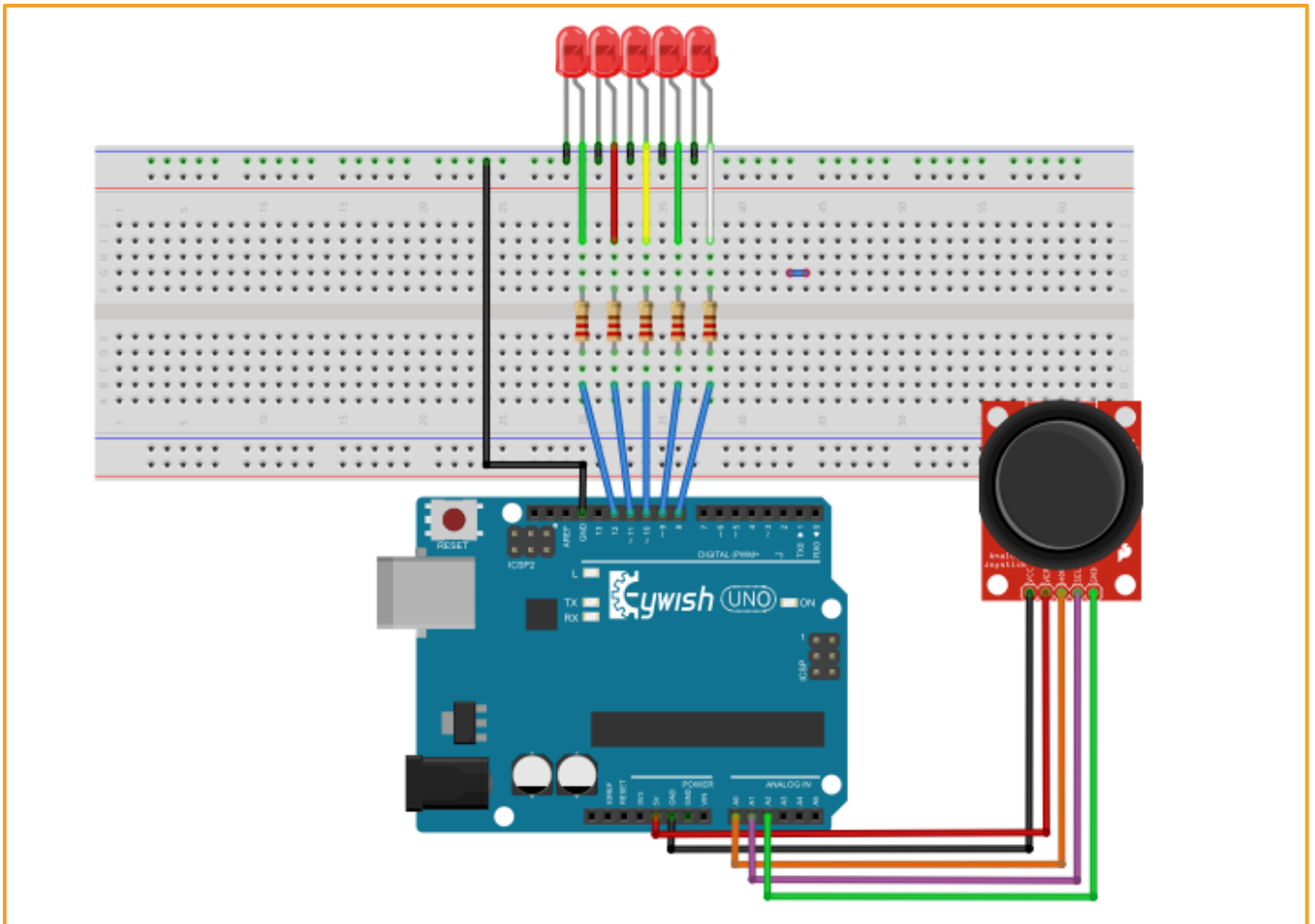
Code

```
#define JOYSTICK_X     A0
#define JOYSTICK_Y     A1
#define JOYSTICK_SW    A2


#define LED_ENTER    8    //enter
#define LED_LEFT     9    //left
#define LED_UP       10   //up
#define LED_RIGHT    11   //right
#define LED_DOWN     12   //down
 int value_x,value_y,value_sw;
```

```
void setup()
{
    pinMode(JOYSTICK_X, INPUT);
    pinMode(JOYSTICK_Y, INPUT);
    pinMode(JOYSTICK_SW, INPUT_PULLUP);
    pinMode(LED_ENTER,OUTPUT);
    pinMode(LED_LEFT,OUTPUT);
    pinMode(LED_UP,OUTPUT);
    pinMode(LED_RIGHT,OUTPUT);
    pinMode(LED_DOWN,OUTPUT);
    Serial.begin(115200);
}
void loop()
{
    value_x=analogRead(JOYSTICK_X);
    value_y=analogRead(JOYSTICK_Y);
    value_sw=digitalRead(JOYSTICK_SW);
    if(value_x==0)
    {
        digitalWrite(LED_RIGHT,LOW);
        digitalWrite(LED_LEFT,HIGH);
    }else if(value_x==1023)
    {
        digitalWrite(LED_LEFT,LOW);
        digitalWrite(LED_RIGHT,HIGH);
    }else
    {
        digitalWrite(LED_LEFT,LOW);
        digitalWrite(LED_RIGHT,LOW);
    }
```

```
if(value_y==0)
{
    digitalWrite(LED_DOWN,LOW);
    digitalWrite(LED_UP,HIGH);
}
else if(value_y==1023)
{
    digitalWrite(LED_UP,LOW);
    digitalWrite(LED_DOWN,HIGH);


}
else if(value_y==1023)
{
    digitalWrite(LED_UP,LOW);
    digitalWrite(LED_DOWN,HIGH);
}else
{
    digitalWrite(LED_UP,LOW);
    digitalWrite(LED_DOWN,LOW);
}
if(value_sw == 0 )
{
    digitalWrite(LED_ENTER,HIGH);
}
else
{
    digitalWrite(LED_ENTER,LOW);
}
delay(100);
```
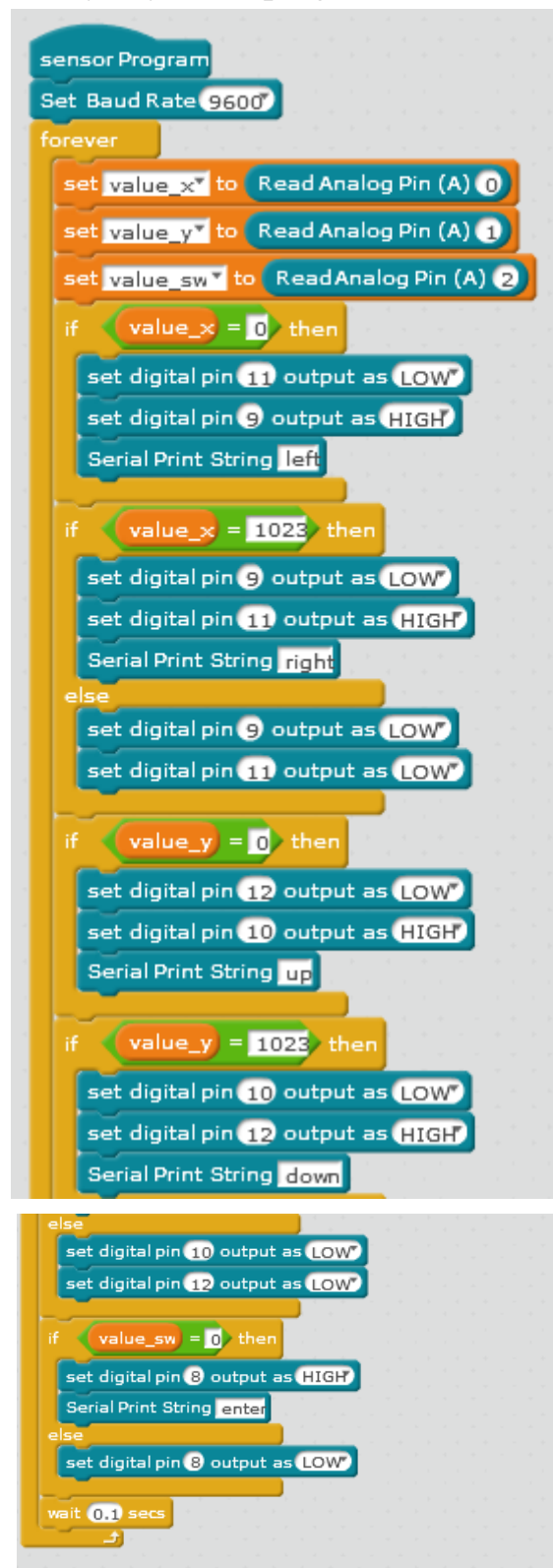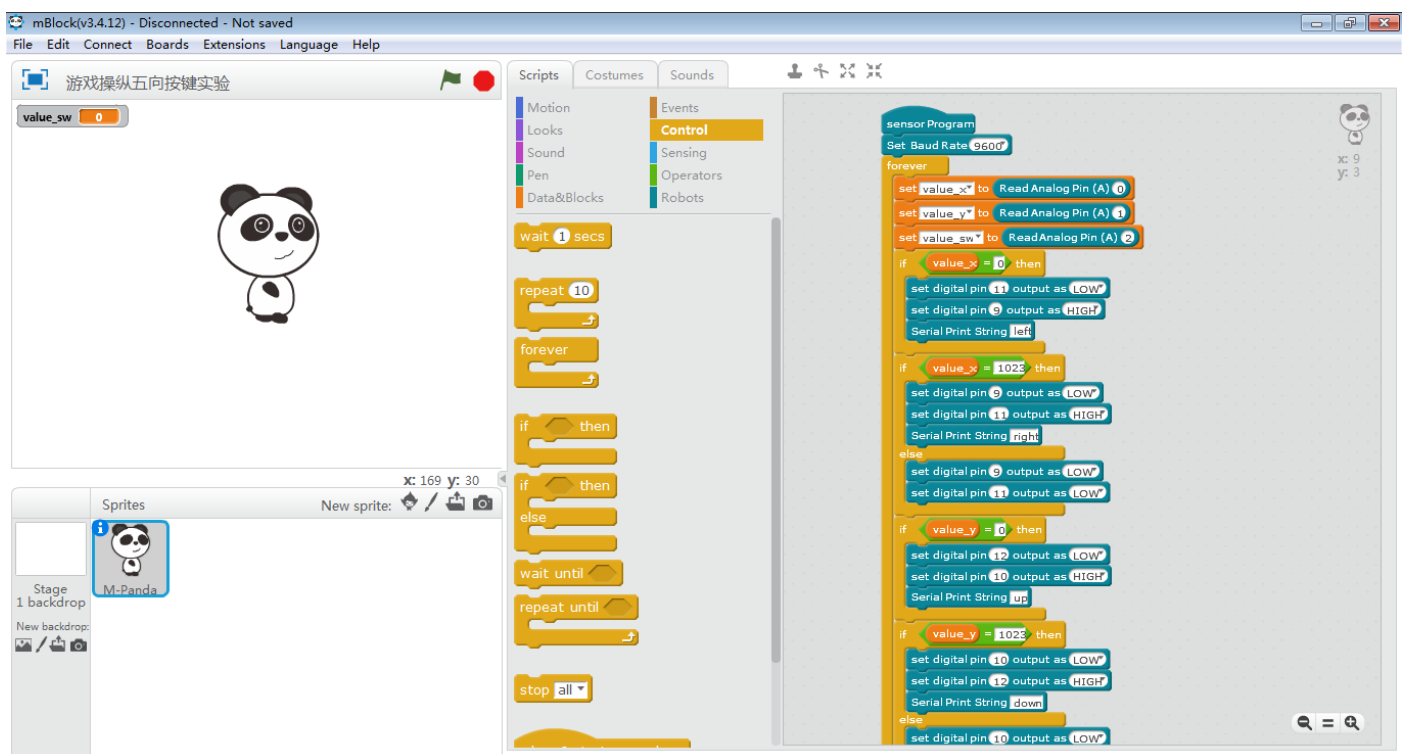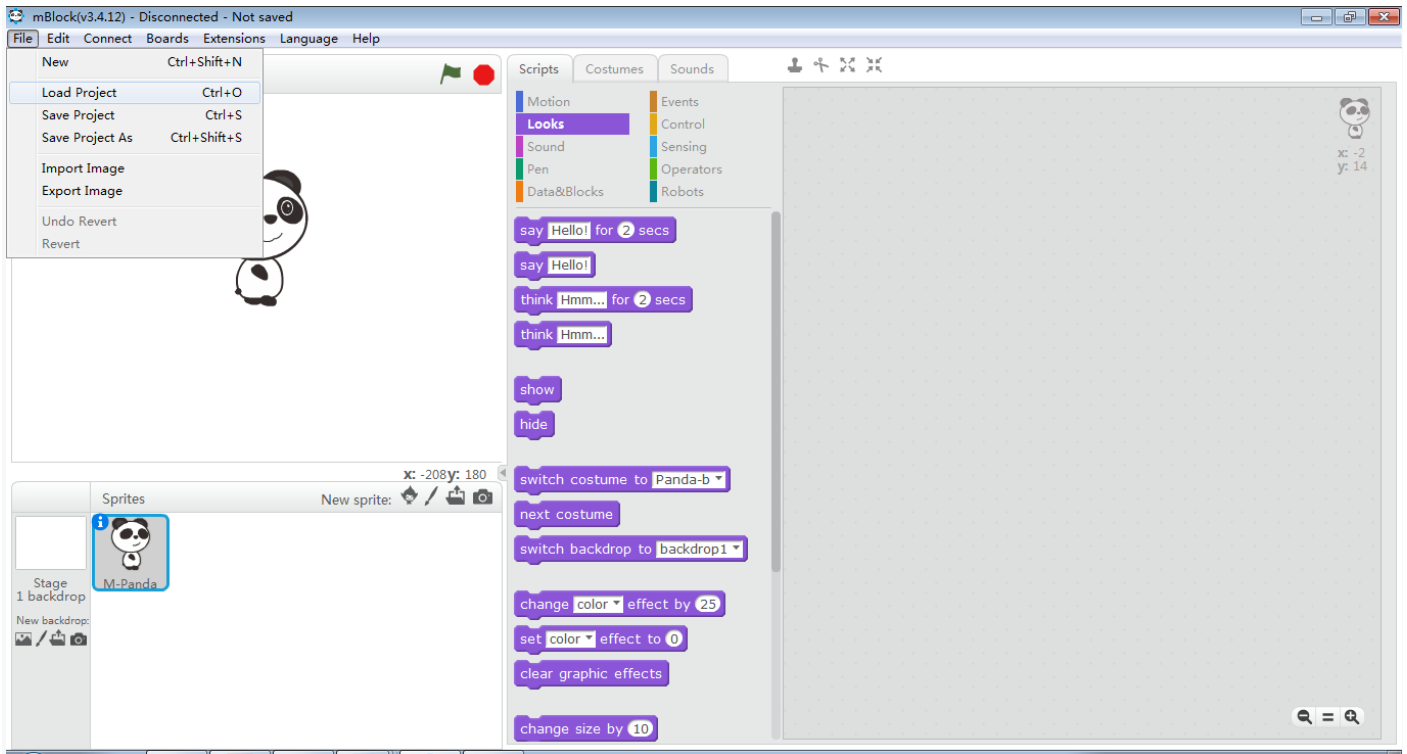
Experiment Result
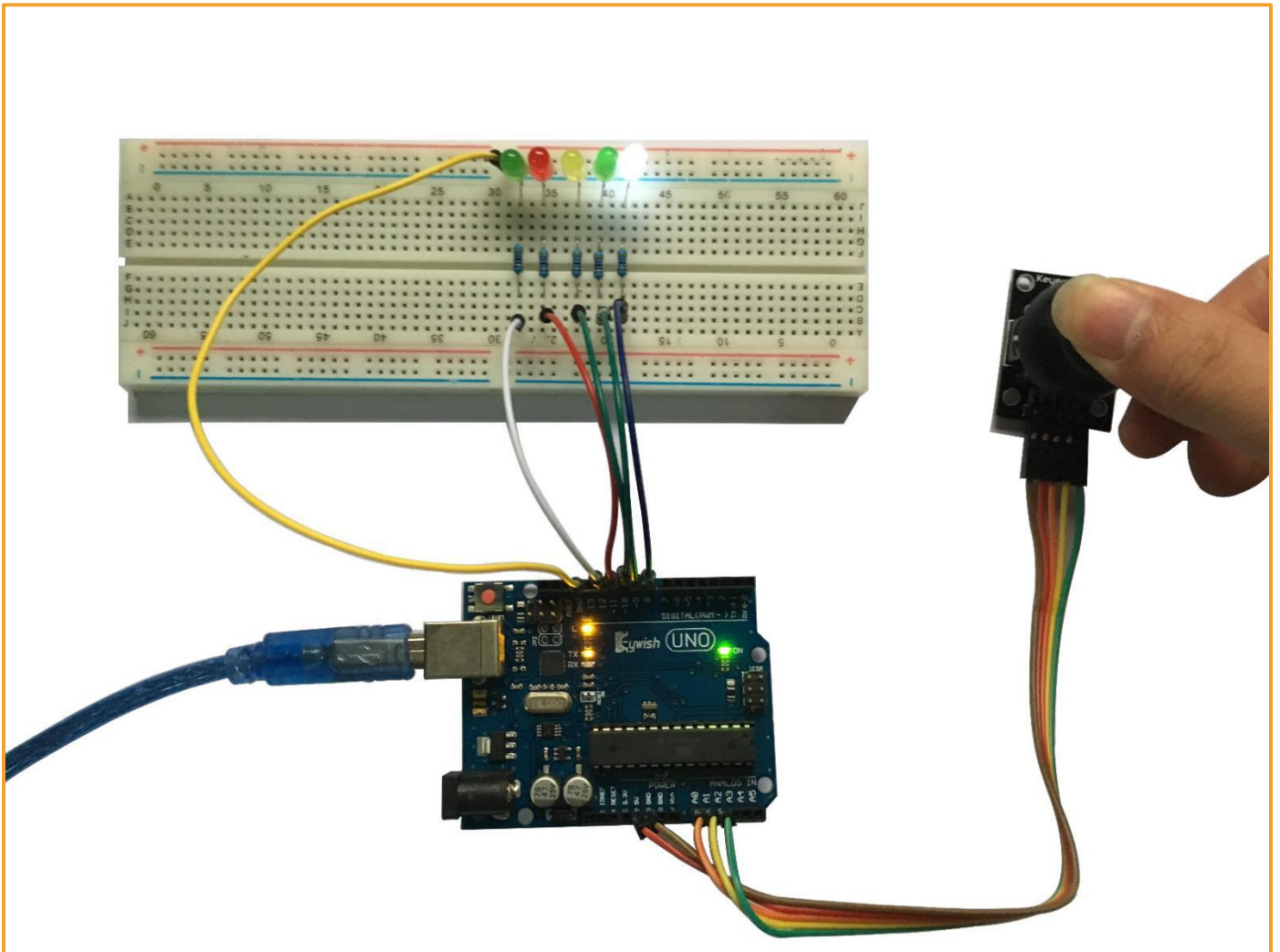
MBlock graphical programming program

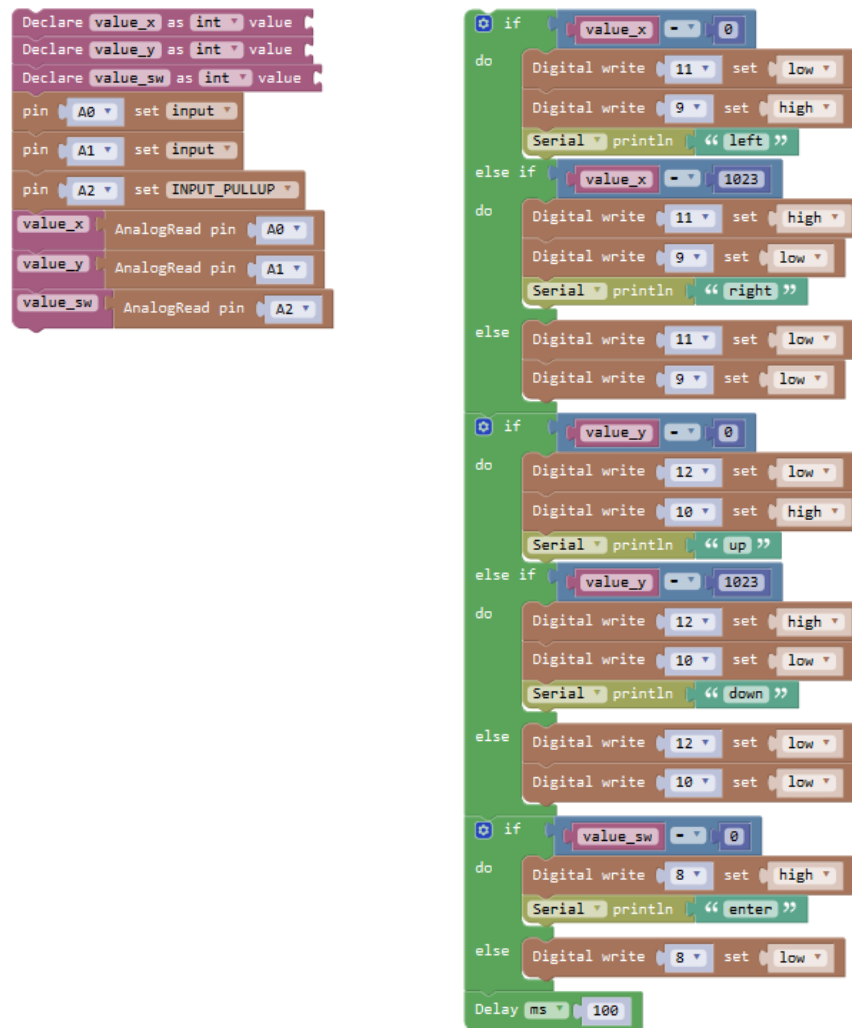MBlock writes five-way keystroke program as shown in the figure below:

You can also open the program file directly with mblock, which is a. Sb2 file. Here are the steps to open it:

Mixly graphical programming

Mixly programmed the five-way key program as shown in the figure below:

MagicBlock graphical programming

如果　获取变量 X 的值 ＝ 0 　那么
　数字写 11 ▼ LOW ▼
　数字写 9 ▼ HIGH ▼
　串口 Serial ▼ 打印字符(自动换行) left

如果　获取变量 X 的值 ＝ 1023 　那么
　数字写 11 ▼ HIGH ▼
　数字写 9 ▼ LOW ▼
　串口 Serial ▼ 打印字符(自动换行) right
否则
　数字写 11 ▼ LOW ▼
　数字写 9 ▼ LOW ▼

如果　获取变量 Y 的值 ＝ 0 　那么
　数字写 12 ▼ LOW ▼
　数字写 10 ▼ HIGH ▼
　串口 Serial ▼ 打印字符(自动换行) up