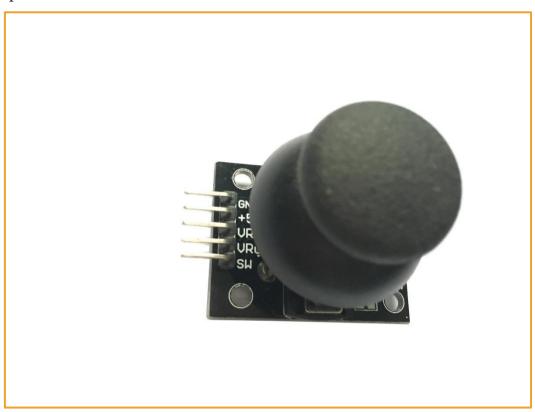


#### Joystick five-way key Experiment

#### 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.

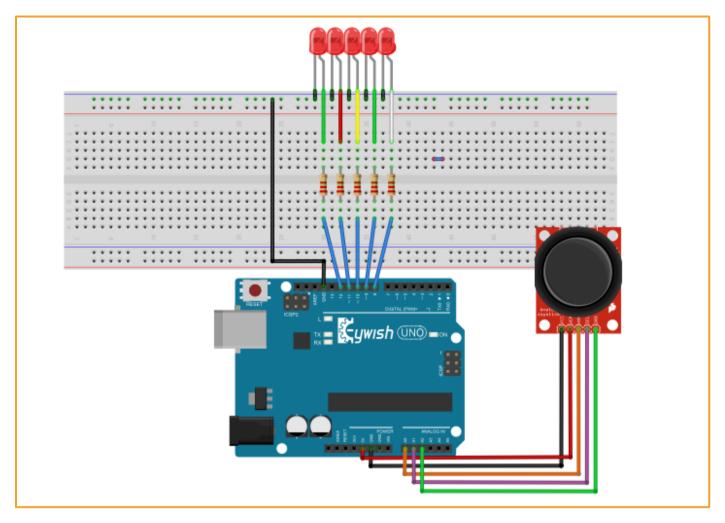
#### **Component List**

- Keywish Arduino UNO R3 Mainboard
- Breadboard
- USB cable
- Joystick \*1
- ♦ LED light \*5
- Current limiting resistor  $220\Omega*5$

# 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;
```



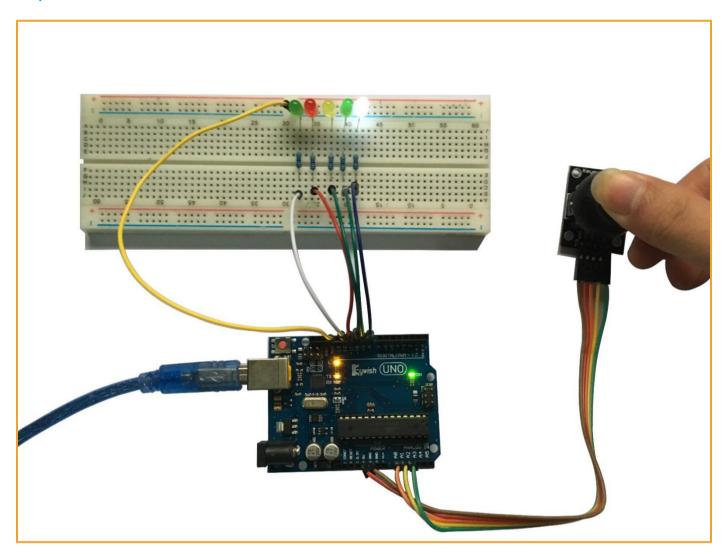
```
void setup()
{
    pinMode(JOYSTICK X, INPUT);
    pinMode(JOYSTICK Y, INPUT);
    pinMode(JOYSTICK SW, INPUT PULLUP);
    pinMode(LED ENTER,OUTPUT);
    pinMode(LED LEFT,OUTPUT);
    pinMode(LED UP,OUTPUT);
    pinMode(LED RIGHT,OUTPUT);
    pinMode(LED DOWN,OUTPUT);
    Serial.begin (115200);
void 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**





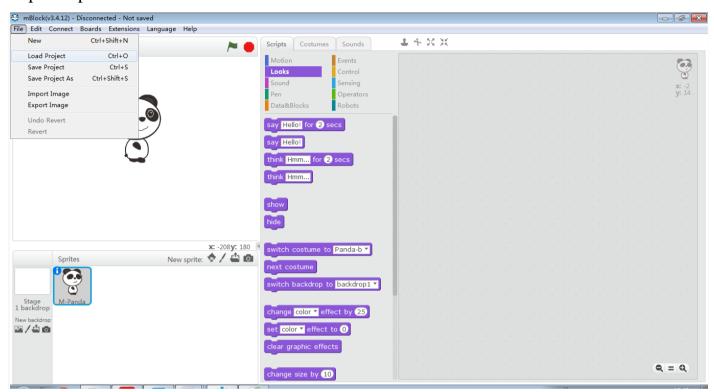
# MBlock graphical programming program

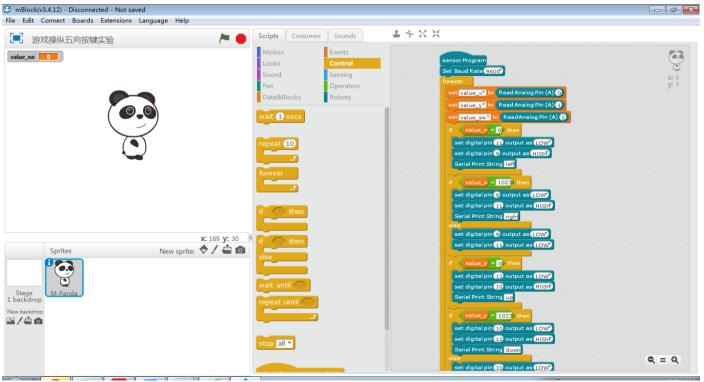
MBlock writes five-way keystroke program as shown in the figure below:

```
sensor Program
Set Baud Rate 9600
  set_value_x to Read Analog Pin (A) 0
  set_value_y to Read Analog Pin (A) 1
  set_value_sw v to ReadAnalog Pin (A) 2
  if 🦪 value_x = 0 then
    set digital pin 11 output as LOW
    set digital pin 9 output as HIGH
    Serial Print String left
       value_x = 1023 then
    set digital pin 9 output as LOW
    set digital pin 11 output as HIGH
    Serial Print String right
    set digital pin 9 output as LOW
    set digital pin 11 output as LOW
  if 🤻 value_y) = 0> then
    set digital pin 12 output as LOW
    set digital pin 10 output as HIGH
    Serial Print String up
  if 🧹 value_y = 1023> then
    set digital pin 10 output as LOW
    set digital pin 12 output as HIGH
    Serial Print String down
  set digital pin 10 output as LOW
  set digital pin 12 output as LOW
     value_sw = 0 then
  set digital pin 8 output as HIGH
  Serial Print String enter
  set digital pin 8 output as LOW
 wait 0.1 secs
```



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:

```
Declare value_x as int value
Declare value_y as int value
Declare value_sw as int value
pin A0 v set input v

pin A1 v set input v

pin A2 v set INPUT_PULLUP v

value_x AnalogRead pin A0 v

value_y AnalogRead pin A1 v

value_sw AnalogRead pin A2 v
```

```
value_x • V 0
    Digital write 11 v set low v
    Digital write 9 v set high v
    Serial v println / " left »
         value_x - 1023
    Digital write 11 v set high v
    Digital write 9 v set low v
    Serial v println | " right "
   Digital write (11 v set (low v
    Digital write 9 v set 10w v
         value_y - V 0
    Digital write 12 v set low v
    Digital write 10 v set high v
    Serial v println | " up "
       value_y - V 1023
    Digital write 12 v set high v
    Digital write 10 v set low v
    Digital write 12 v set low v
    Digital write 10 v set low v
       value_sw - v 0
    Digital write 8 v set high v
    Serial v println | " enter "
   Digital write ( 8 v set ( low v
Delay ms 🔻 100
```



# MagicBlock graphical programming

