

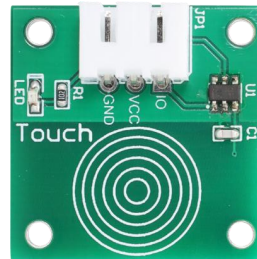
# Project 8-Touch module controls WS2812B RGB light

## 1. project description

Through this project, you can learn how to use ZY -type-c Nano to control the WS2812 RGB module by detecting the input of the touch module. The function of this program is the touch detection module. The RGB light of WS2812 B will gradually turn on the red light, then the touch will gradually turn on the green light, then the touch will gradually turn on the blue light, and so on.

## 2. Introduction to modules

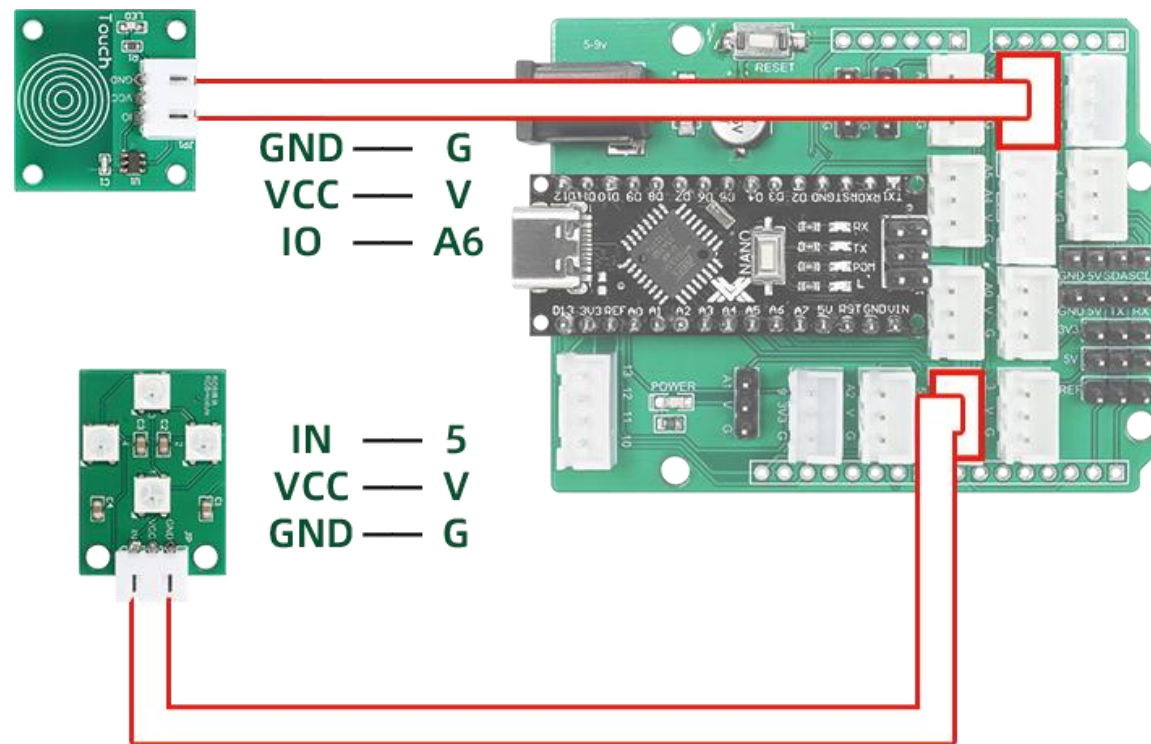
### 2.1 Touch detection module



This module is a capacitive touch switch module based on touch detection IC (TTP223B). Under normal circumstances, the module outputs low level and the mode is low power consumption mode; when touching the corresponding position with a

finger, the module outputs high level and the mode switches to fast mode; when there is no touch for 12 seconds, the mode switches to low power mode again. consumption mode. The module can be installed on the surface of plastic, glass and other non-metallic materials. Do not cover the surface of the module with tissue paper (non-metallic). The direct touch position should be unobstructed . It can be made into a button hidden on the wall, desktop, etc.

### 3. Project wiring diagram



#### 4. Download the Arduino code

Confirm that the Adafruit\_NeoPixel library has been added successfully. If it has not been added, please go back to [Project 6](#) to see how to add the library (as shown below).

project6	2023/10/6 16:51	文件夹	
Adafruit_NeoPixel-1.4.0.zip	2023/4/27 14:13	WinRAR ZIP 压缩...	69 KB
Breathing lamp.mp	2023/10/5 14:40	MP 文件	167 KB

Open the project Arduino code file (path: project 8 touch WS2812B\project8\project8.ino)

project8	2023/10/6 17:05	文件夹	
Touch control WS2812B.mp	2023/10/6 15:25	MP 文件	169 KB
项目 8 触控 WS2812B.docx	2023/10/6 15:22	DOCX 文档	1,573 KB

Connect the main control board to the computer using USB, select the board type as Nano, select the newly displayed COM number, click "Download" to start compiling and downloading the program to the main control board.

**Code analysis:**

```

1  #include <Adafruit_NeoPixel.h> //添加ws2812库      Add the ws2812b library
2  #define NUMPIXELS 4           // 定义ws2812 灯数      Number of ws2812b lamps
3  #define RGB_PIN 5             // 定义ws2812引脚5      ws2812b pin definition 5
4  #define KEY_1 20              // 定义触控模块引脚A6    Key pin definition A6
5
6  Adafruit_NeoPixel pixels(NUMPIXELS, RGB_PIN, NEO_GRB + NEO_KHZ800); //实例化灯对象    Creating light objects
7
8  //Variable definition
9  int i;                        //触控值变量      Touch value variable
10 int count_R=0; //R/G/B变量初始值    Initial values for R/G/B variables
11 int count_G=0;
12 int count_B=0;
13 int count_key=0; //触控次数变量      Touch count variable
14 int Press=0; //触控标志变量      Touch flag variables
15
16
17 void setup ()
18 {
19     pinMode(KEY_1, INPUT);
20     pixels.begin(); //初始化库函数    Initialize 2812 library functions
21     pixels.show();
22     Serial.begin(9600);
23     pixels.clear(); //初始清除
24 }
25
26 void loop() //主函数      Principal function
27 {
28     KEY(); //按键函数      Key function
29     RGB(); //亮灯函数      RGB function

```

```

32 void RGB()
33 {
34     switch(count_key)    //第几次触摸    Number of key presses
35     {
36 >     case 1 :            //逐渐亮红灯    The red light is on...
48 >     case 2 :            //逐渐亮绿灯    The green light is on...
60 >     case 3 :            //逐渐亮蓝灯    The blue light is on...
72 >     case 4 :            //亮白灯    The white light is on...
79 >     case 5 :            //灭灯    The light is off...
86     default : break;
87 }
88 }

```

```

90 void KEY()                //触控消抖函数    Key function
91 {
92     i=analogRead(KEY_1);    //A6引脚使用读取模拟方法获取触摸模块数值    Pin A6 uses the read simulation method to obtain the touch module value
93     if((i<300)&&(Press==0)){
94         Press=1;            //按下标志==1    Press the sign
95     }
96     i=analogRead(KEY_1);
97     if((i>1000) &&(Press==1)) //松开触控按钮且按下标志==1    Let go detection
98     {
99         // Serial.print("i:");
100        // Serial.println(i);
101        // Serial.print("count_key:");
102        // Serial.println(count_key);
103        Press=0;
104        count_key++;
105        if(count_key>5)count_key=1;//超过5次回到初始值1    Return to the initial value of 1 more than 5 times

```

## 5. Download Mind+ graphical code

Open the project Mind+code file (path: project 8 touch control WS2812B\Touch control WS2812B.mp)

project8	2023/10/6 17:05	文件夹	
<b>Touch control WS2812B.mp</b>	2023/10/6 15:25	MP 文件	169 KB
项目 8 触控 WS2812B.docx	2023/10/6 15:22	DOCX 文档	1,573 KB

Connect the main control board to the computer with a USB cable and select the newly appeared CH340 serial port COM number. Click "Upload to Device" to complete the code upload.

### Programming analysis:

Click "Extension" in the lower left corner, and then select the main control board type as Nano.



Add the WS2812 RGB light library file : click the "Display" type and select the WS2812 RGB light

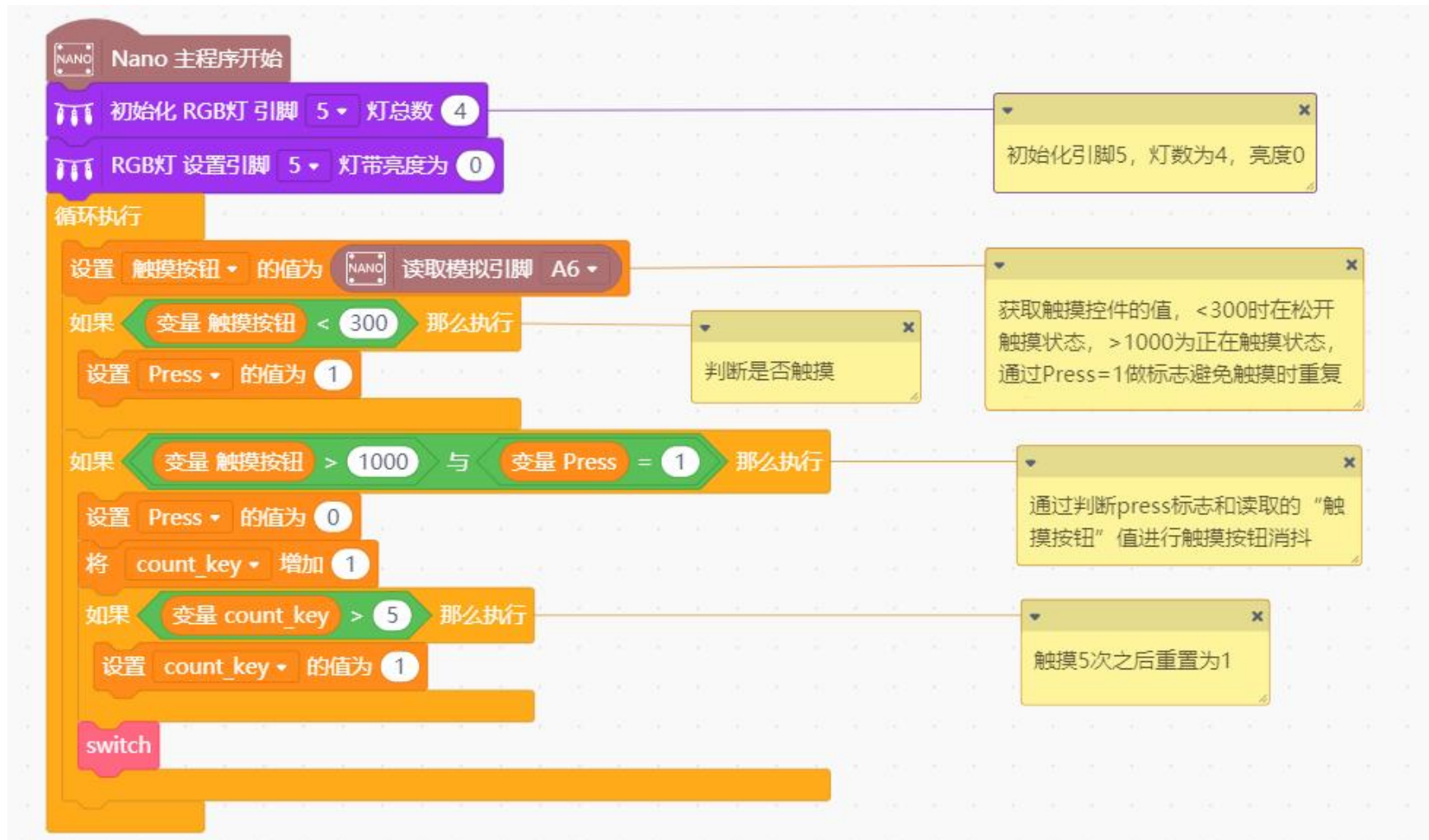


After the addition is successful, you can see that there are two more categories in the programming block column on the left: Nano and "Display"





The complete programming is as follows:





5 conditional judgment statements constitute the switch function

The image shows a Scratch script for a switch function. It starts with a '定义 switch' (Define switch) block. The script then sets the '亮度' (Brightness) variable to 0. It uses a series of '如果' (If) conditional statements to control the RGB lights based on the 'count\_key' variable. The first '如果' block checks if 'count\_key' is 1. If true, it enters a '重复执行 40 次' (Repeat 40 times) loop. Inside the loop, it waits for 0.01 seconds, increases the '亮度' by 5, and then sets the RGB light at pin 5 to the current '亮度' value. The second '如果' block checks if 'count\_key' is 2. If true, it enters another '重复执行 40 次' loop, where it waits for 0.01 seconds, increases the '亮度' by 5, and sets the RGB light at pin 5 to the current '亮度' value. The script continues with more '如果' blocks, each corresponding to a different 'count\_key' value and a different color for the RGB light.

定义 switch

设置 亮度 的值为 0

如果 变量 count\_key = 1 那么执行

重复执行 40 次

等待 0.01 秒

将 亮度 增加 5

RGB灯 设置引脚 5 灯带亮度为 变量 亮度

RGB灯 引脚 5 灯号 0 到 4 显示颜色 红色

如果 变量 count\_key = 2 那么执行

重复执行 40 次

等待 0.01 秒

将 亮度 增加 5

RGB灯 设置引脚 5 灯带亮度为 变量 亮度

RGB灯 引脚 5 灯号 0 到 4 显示颜色 绿色

判断触摸按键第几次(count\_key) 触摸, 从而显示对应的颜色。

亮度每次增加5, 执行40次RGB灯逐渐亮起, 亮度由0增加到200, 延时为0.01。注意: 重复次数越多或延时越大就会影响程序执行速度, 要适当设置。

## Programming Tips: Custom Functions

Click the "Function" option > Customize the module and add the function name

The image shows a programming interface with a left sidebar and a main workspace. In the sidebar, the '函数' (Function) category is highlighted with a red box. The main workspace displays a script for a Nano microcontroller. The script includes a 'Nano 主程序开始' (Nano Main Program Start) block, followed by a '读取数字引脚 2' (Read Digital Pin 2) block, and a '读取模拟引脚 A0' (Read Analog Pin A0) block. A '自定义模块' (Custom Module) block is also present. The workspace also shows a 'switch' block in the '控制' (Control) category. Below the workspace, a dialog box titled '添加一个自定义模块' (Add a Custom Module) is open, showing a 'switch' block with a red box around it.

自定义模块

switch

Nano

Nano 主程序开始

引脚操作

读取数字引脚 2

读取模拟引脚 A0

设置 触摸按钮 的值为 读取模拟引脚 A6

如果 变量 触摸按钮 < 300 那么执行

设置 Press 的值为 1

如果 变量 触摸按钮 > 1000 与 变量 Press = 1 那么执行

设置 Press 的值为 0

将 count\_key 增加 1

如果 变量 count\_key > 5 那么执行

判断是否触摸

添加一个自定义模块

switch

The implementation of the function is spliced below the "Definition"



Just drag it out when calling

