Blinker APP配置：

进入主界面-->点击右上角加号-->点击独立设备-->点击WiFi接入-->点击阿里云（可尝试其他选项）-->记录下Secret Key-->点击返回我的设备

进入主界面-->点击新添加的设备-->点击右上角的铅笔图标-->在下方一行选项中找到“颜色”并点击-->点击界面上新添加的模块-->将数据键名改为“RGBKey”-->点击右上角的保存-->再点击右上角像锁一样的图标

PART A：

功能实现：将以下代码编译下载（将之前记录下的Secret Key、即将要连接的WiFi名称和秘密写到对应的位置）。打开串口监视器，波特率调至115200，Blinker APP中打开之前新建的设备，然后点击圆形区域内的任意颜色，在串口监视器中即可看到所点击颜色的RGB值，拉取圆形下方的进度条，在串口监视器中即可看到颜色的亮度值也会随着变化。PART A部分验证成功。

#define BLINKER\_WIFI

#include <Blinker.h>

const char auth[] = "your own Secret Key";

const char ssid[] = "your wifi name ";

const char pswd[] = "your wifi password";

BlinkerRGB RGB1("RGBKey");

void rgb1\_callback(uint8\_t r\_value, uint8\_t g\_value,

uint8\_t b\_value, uint8\_t bright\_value)

{

Serial.print("R:");

Serial.print(r\_value);

Serial.print(", G:");

Serial.print(g\_value);

Serial.print(", B:");

Serial.print(b\_value);

Serial.print(", bright:");

Serial.println(bright\_value);

}

void setup() {

Serial.begin(115200);

BLINKER\_DEBUG.stream(Serial);

Blinker.begin(auth, ssid, pswd);

RGB1.attach(rgb1\_callback);

}

void loop() {

Blinker.run();

}

程序讲解：程序Blinker.begin(auth, ssid, pswd)用于连接网络等一些初始化操作，程序BlinkerRGB RGB1("RGBKey")的功能为注册一个小组件RGB1，并利用RGB1.attach(rgb1\_callback)为其注册一个回调函数rgb1\_callback()，当手机APP中有点击时，就会进入到这个回调函数中，并将所点击的颜色值和亮度值传入到该函数中，并在函数内部将收到的值打印到串口监视器上。

PART B:

硬件连接：将RGB灯的RGB三个引脚分别接到板子上的13、12、14引脚上，具体13、12、14引脚对应到板子的哪个引脚可参照图片“nodemcu引脚图.PNG”，将RGB灯的GND接到板子的GND即可。

功能实现：将以下代码编译下载（将之前记录下的Secret Key、即将要连接的WiFi名称和秘密写到对应的位置）。打开串口监视器，进入到Blinker APP中之前新建的设备，点击圆形区域中的任意颜色，并将圆形下边的进度条拉到某个位置，此时串口监视器会打印收到的R、G、B值和亮度值，同时小灯也会依据收到的颜色值和亮度值发出对应的光，即实验成功。

#define BLINKER\_WIFI

#include <Blinker.h>

const char auth[] = "your own Secret Key";

const char ssid[] = "your wifi name ";

const char pswd[] = "your wifi password";

const int PIN\_LED\_RED = 13;

const int PIN\_LED\_GREEN = 12;

const int PIN\_LED\_BLUE = 14;

BlinkerRGB RGB1("RGBKey");

void rgb1\_callback(uint8\_t r\_value, uint8\_t g\_value,

uint8\_t b\_value, uint8\_t bright\_value)

{

Serial.print("R:");

Serial.print(r\_value);

Serial.print(", G:");

Serial.print(g\_value);

Serial.print(", B:");

Serial.print(b\_value);

Serial.print(", bright:");

Serial.println(bright\_value);

analogWrite(PIN\_LED\_RED, r\_value\*bright\_value\*0.041);

analogWrite(PIN\_LED\_GREEN, g\_value\*bright\_value\*0.041);

analogWrite(PIN\_LED\_BLUE, b\_value\*bright\_value\*0.041);

}

void setup() {

Serial.begin(115200);

BLINKER\_DEBUG.stream(Serial);

pinMode(PIN\_LED\_RED, OUTPUT);

pinMode(PIN\_LED\_GREEN, OUTPUT);

pinMode(PIN\_LED\_BLUE, OUTPUT);

Blinker.begin(auth, ssid, pswd);

RGB1.attach(rgb1\_callback);

}

void loop() {

Blinker.run();

}

程序讲解：变量PIN\_LED\_RED、PIN\_LED\_GREEN、PIN\_LED\_BLUE用来配置RGB灯的三个引脚所对应板子上要接的引脚；RGB1为注册的小组件，并利用RGB1.attach(rgb1\_callback)为其注册一个回调函数rgb1\_callback()，当手机APP上有点击到颜色时，会进入到回调函数rgb1\_callback()中，并将点击的RGB值、亮度值传入到该函数中，在函数里边把收到的颜色亮度值打印出来，并利用上一节相关知识，即利用函数analogWrite()将收到的颜色、亮度值输出到RGB灯上。