

淘宝店铺

优秀不够，你是否无可替代

知识从未如此性感。烂程序员关心的是代码,好程序员关心的是数据结构和它们之间的关系 --QQ群: 607064330 --本人QQ:946029359 --淘宝 <https://shop411638453.taobao.com/>

随笔 - 811, 文章 - 0, 评论 - 327, 阅读 - 201万

导航

[博客园](#)
[首页](#)
[新随笔](#)
[联系](#)
[订阅](#)
[管理](#)

公告

渡我不渡她 -

Not available

00:00 / 03:41

- 渡我不渡她
- 小镇姑娘
- PDD洪荒之力

加入QQ群

昵称：杨奉武
 园龄：6年2个月
 粉丝：693
 关注：1

搜索

我的标签

8266(88)
 MQTT(50)
 GPRS(33)
 SDK(29)
 Air202(28)
 云服务器(21)
 ESP8266(21)
 Lua(18)
 小程序(17)
 STM32(16)
 更多

随笔分类

Air724UG学习开发(5)
 Android(22)
 Android 开发(8)
 C# 开发(4)
 CH395Q学习开发(17)
 CH573F学习开发(1)
 CH579M物联网开发(12)
 CH579M学习开发(8)
 ESP32学习开发(25)
 ESP8266 AT指令开发(基于STC89C52单片机)(3)
 ESP8266 AT指令开发(基于STM32)(1)
 ESP8266 AT指令开发基础入门篇备份(12)
 ESP8266 LUA脚本语言开发(13)

803-ESP32_SDK开发-手机连接ESP32的热点,使用微信小程序查看摄像头图像(WiFi视频小车,局域网视频监控)

<p><iframe name="ifd" src="https://mnifdv.cn/resource/cnblogs/LearnESP32" frameborder="0" scrolling="auto" width="100%" height="1500"></iframe></p>

ESP32_SDK开发(源码见资料源码)

开发板链接:[ESP32开发板链接](#)

资料源码Git下载链接:<https://github.com/yangfengwu45/learn-esp32.git>

资料源码百度网

盘:<https://pan.baidu.com/s/10SBk0NsvLtJYHpDab9islg> 提取码：25oy

【点击加入乐鑫WiFi模组开发交流群】(群号822685419)https://jq.qq.com/?_wv=1027&k=fXgd3UOo

python虚拟机: [python-3.8.4-amd64.exe](#)

ESP-IDF工具安装器: [esp-idf-tools-setup-2.3.exe](#)

- 基础开源教程:ESP32开发(arduino)
- 基础开源教程:ESP8266:LUA脚本开发
- 基础开源教程:ESP8266 AT指令开发(基于51单片机)
- 基础开源教程:Android学习开发
- 基础开源教程:C#学习开发
- 基础开源教程:微信小程序开发入门篇
- 需要搭配的Android, C#等基础教程如上, 各个教程正在整理。
- [000-ESP32开发板使用说明](#)
- ESP32_SDK开发
- [001-开发环境搭建\(Windows+VSCode\)](#)
- [002-测试网络摄像头\(OV2640\),实现远程视频监控\(花生壳http映射\)](#)
- [003-学习ESP32资料说明](#)
- [004-新建工程模板和创建新的文件](#)
- [005-新建工程补充-通过官方示例创建工程](#)
- [006-关于操作系统-任务,任务堆栈空间,任务的挂起,恢复,删除](#)
- [007-使用缓存管理传递数据](#)
- 基本外设-----
- [101-ESP32管脚说明](#)
- [102-GPIO](#)
- [103-硬件定时器timer](#)
- [104-软件定时器esp_timer](#)
- [105-uart串口,485通信](#)
- [106-SPI](#)
- [107-flash数据存储nvs](#)
- 网络通信-----
- [201-softAP模式配置模组发出的热点](#)
- [202-station模式配置模组连接路由器热点](#)
- [203-softAP+station共存模式](#)
- [204-TCP服务器\(模组AP热点模式,支持多个客户端连接通信\)\(废弃\)](#)
- [205-TCP服务器\(select方式,支持多连接,高速高并发传输\)](#)
-
-
-
-
- 视频传输,监控,直播方案-----

ESP8266 LUA开发基础入门篇
备份(22)
ESP8266 SDK开发(33)
ESP8266 SDK开发基础入门篇
备份(30)
GPRS Air202 LUA开发(11)
HC32F460(华大单片机)物联网
开发(9)
HC32F460(华大单片机)学习开
发(8)
NB-IOT Air302 AT指令和LUA
脚本语言开发(27)
PLC(三菱PLC)基础入门篇(2)
STM32+Air724UG(4G模组)
物联网开发(43)
STM32+BC26/260Y物联网开
发(37)
STM32+CH395Q(以太网)物
联网开发(24)
STM32+ESP8266(ZLESP826
6A)物联网开发(1)
STM32+ESP8266+AIR202/3
02远程升级方案(16)
STM32+ESP8266+AIR202/3
02终端管理方案(6)
STM32+ESP8266+Air302物
联网开发(65)
STM32+W5500+AIR202/30
2基本控制方案(25)
STM32+W5500+AIR202/30
2远程升级方案(6)
UCOSii操作系统(1)
W5500 学习开发(8)
编程语言C#(11)
编程语言Lua脚本语言基础入
门篇(6)
编程语言Python(1)
单片机(LPC1778)LPC1778(2)
单片机(MSP430)开发基础入门
篇(4)
单片机(STC89C51)单片机开发
板学习入门篇(3)
单片机(STM32)基础入门篇(3)
单片机(STM32)综合应用系列
(16)
更多

阅读排行榜

1. ESP8266使用详解(AT,LUA, SDK)(174295)
2. 1-安装MQTT服务器(Windows),并连接测试(105127)
3. 用ESP8266+android,制作自己的WIFI小车(ESP8266篇)(67830)
4. ESP8266刷AT固件与node mcu固件(66903)
5. 有人WIFI模块使用详解(39415)
6. (一)基于阿里云的MQTT远程控制(Android 连接MQTT服务器,ESP8266连接MQTT服务器实现远程通信控制---简单的连接通信)(37063)
7. C#中public与private与static(35919)
8. 关于TCP和MQTT之间的转换(35278)
9. android 之TCP客户端编程(33020)
10. android服务端+eps8266+单片机+路由器之远程控制系统(31685)

推荐排行榜

--

- [801-ESP32\(WiFi\)把采集的摄像头照片数据通过串口输出到串口上位机显示\(C# 串口上位机\)](#)
- [802-ESP32\(WiFi\)把采集的摄像头照片数据通过UDP发送给UDP客户端\(C# UDP客户端\)](#)
- [803-手机连接ESP32的热点,使用微信小程序查看摄像头图像\(WiFi视频小车,局域网视频监控\)](#)
-
-
-
-

说明

这一节的功能是使用手机连接ESP32的热点,然后就可以在微信小程序上查看摄像头图像。

效果展示



为了方便大家伙扩展应用,增加了几个按键.具体介绍接着看哈.

1. 用ESP8266+android,制作自己的WIFI小车(ESP8266篇)(9)
2. C#委托+回调详解(9)
3. 用ESP8266+android,制作自己的WIFI小车(Android 软件)(6)
4. 我的大学四年(6)
5. ESP8266使用详解(AT,LUA, SDK)(6)

最新评论

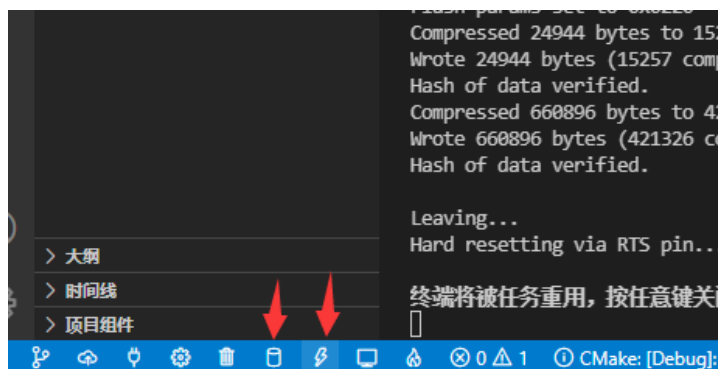
1. Re:2-6-1-视频传输,监控,直播方案-手机连接ESP32的热点,使用微信小程序查看摄像头图像(WiFi视频小车,局域网视频监控)
赞赞赞,感谢大佬无私奉献
--SJA2C2A
2. Re:中移动M5311模块使用手册(TCP,MQTT)
请问你用的usb转ttl是哪一种呢,我用的ch340可是开机串口助手没有SIM识别显示
--夏洛的网娅

测试功能(连接模组热点方式)

1.把这节的ESP32代码放到非中文目录,然后使用VS Code打开



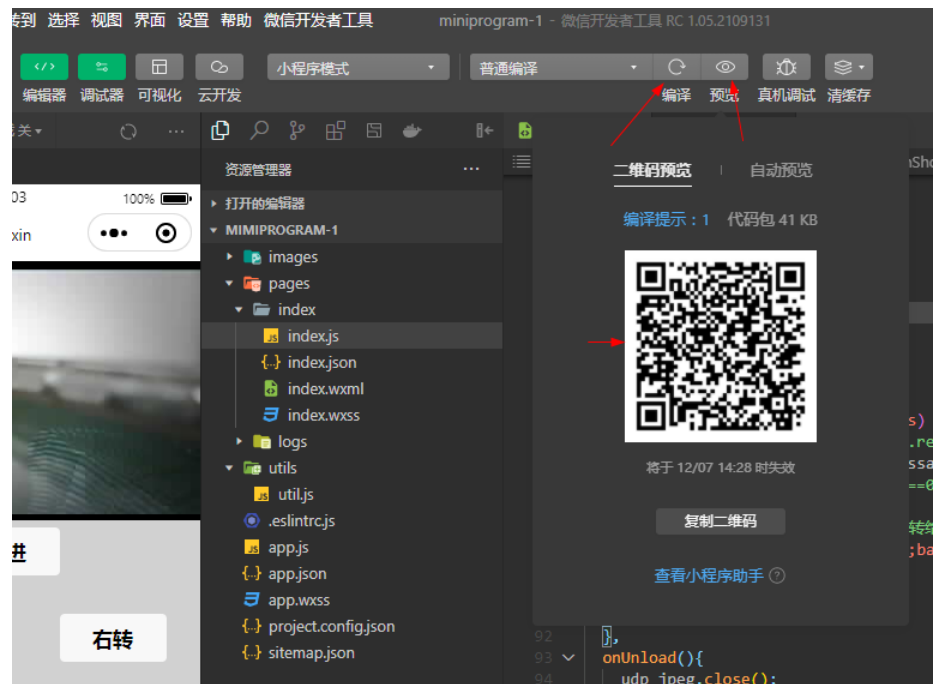
2.编译下载到开发板(第一次编译时间有点长)



3.使用微信开发工具打开这节的小程序工程



4.编译,预览, 使用微信扫码安装到手机



5.设置手机连接ESP32的热点



6.然后就可以了



测试功能(局域网,连接路由器方式)

1.设置下模组连接的路由器名称和密码(根据自家的修改哈),然后编译下载到开发板

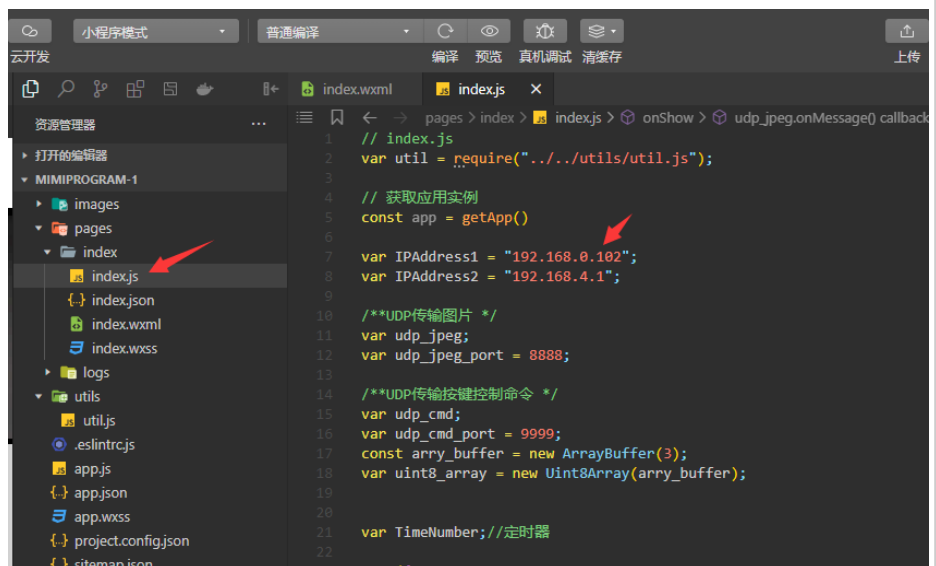
```
资源管理器  ...  C demo.c U  C def.h  C wifi_softap_sta.c U X
> 打开的编辑器
ESP32_CAM
  .vscode
    {} c_cpp_properties.json
    {} settings.json
  > build
  > components
  > main
    M CMakeLists.txt U
    M component.mk U
    C demo.c U
    C event_handler.c U
    C event_handler.h U
    C usart_data_process.c U
    C usart_data_process.h U
    C usart.c U
    C usart.h U
    C wifi_softap_sta.c U
main > C wifi_softap_sta.c > ...
1  #define wifi_softap_c_
2  #include <stdio.h>
3  #include <string.h>
4  #include "wifi_softap_sta.h"
5  //-----
6
7
8  /*配置模组自身的无线热点*/
9  #define ESP_WIFI_AP_SSID      "ESP32_WIFI" //设置模块发出的无线名称
10 #define ESP_WIFI_AP_PASS     "11223344" //无线密码
11 #define ESP_WIFI_AP_CHANNEL  11 //信道(默认1)
12 #define ESP_WIFI_AP_MAX_COUNT 5 //最大连接数(最大10个)
13
14
15 /*配置模组连接的路由器*/
16 #define ESP_WIFI_STA_SSID     "QQQQQ" //设置模块连接的无线名称
17 #define ESP_WIFI_STA_PASS     "11223344" //无线密码
18
19
20 void wifi_init_softap(void)
```

2,查看打印的日志,我的模组路由器给分配的ip是: 192.168.0.102

```
问题 输出 终端 调试控制台
I (1570) wifi_init: WiFi RX IRAM OP enabled
I (1680) phy: phy_version: 4500, 0cd6843, Sep 17 2020, 15:37:07, 0, 0
I (1680) wifi:mode : sta (78:e3:6d:11:2d:94) + softAP (78:e3:6d:11:2d:95)
I (1680) wifi:Total power save buffer number: 16
I (1680) wifi:Init max length of beacon: 752/752
I (1690) wifi:Init max length of beacon: 752/752
I (1700) example: Socket created
I (1700) example: Socket bound, port 9999
I (1700) event_handler: WIFI_EVENT_AP_START
I (1700) example: Socket created
I (1710) example: Socket bound, port 8888
I (1820) wifi:ap channel adjust o:11,2 n:6,2
I (1820) wifi:new:<6,2>, old:<11,2>, ap:<6,2>, sta:<6,2>, prof:11
I (1820) wifi:state: init -> auth (b0)
I (1830) wifi:state: auth -> assoc (0)
I (1830) wifi:state: assoc -> run (10)
I (1850) wifi:connected with QQQQQ, aid = 2, channel 6, 400, bssid = 34:96:72:16:9e:42
I (1850) wifi:security: WPA2-PSK, phy: bgn, rssi: -53
I (1850) wifi:pm start, type: 1

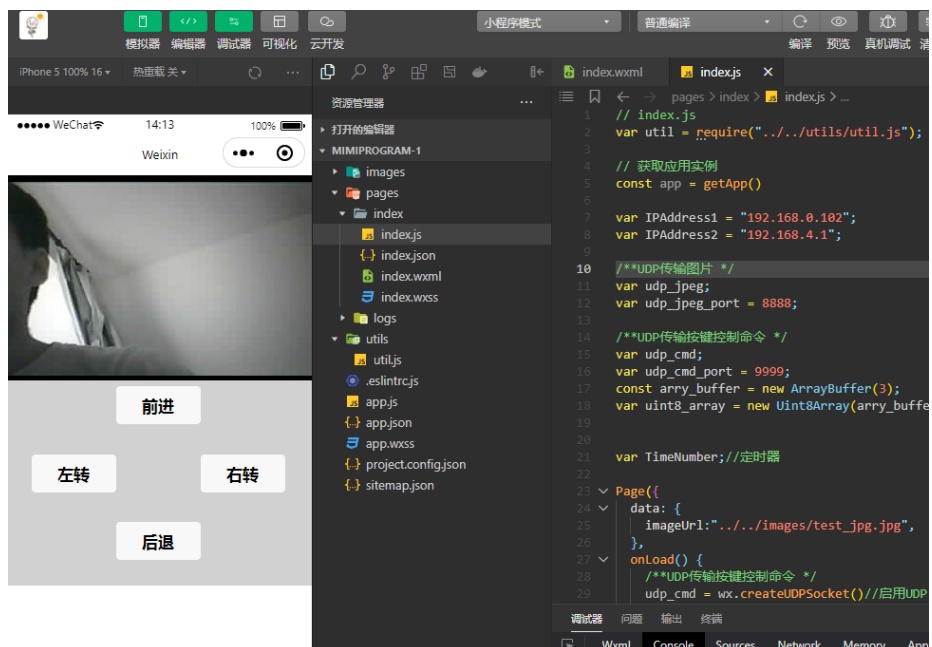
I (1870) wifi:AP's beacon interval = 102400 us, DTIM period = 1
I (2590) esp_netif_handlers: sta ip: 192.168.0.102, mask: 255.255.255.0, gw: 192.168.0.1
I (2590) event_handler: got ip:192.168.0.102
[]
```

3.微信小程序代码里面设置 IPAddress1 为上面的IP地址



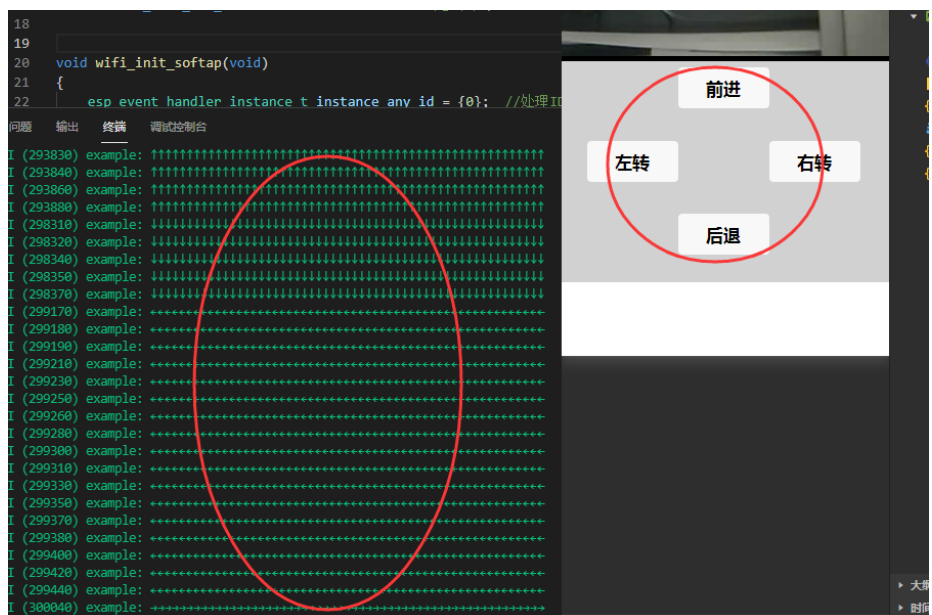
```
云开发 小程序模式 普通编译 编译 预览 真机调试 清缓存 上传
资源管理器
打开的编辑器
MIMIPROGRAM-1
  images
  pages
    index
      index.js
      index.json
      index.wxml
      index.wxss
  logs
  utils
    util.js
    .eslintrc.js
  app.js
  app.json
  app.wxss
  project.config.json
  sitemap.json
index.wxml index.js x
pages > index > index.js > onShow > udp_jpeg.onMessage() callback
1 // index.js
2 var util = require("../utils/util.js");
3
4 // 获取应用实例
5 const app = getApp()
6
7 var IPAddress1 = "192.168.0.102";
8 var IPAddress2 = "192.168.4.1";
9
10 /**UDP传输图片 */
11 var udp_jpeg;
12 var udp_jpeg_port = 8888;
13
14 /**UDP传输按键控制命令 */
15 var udp_cmd;
16 var udp_cmd_port = 9999;
17 const arry_buffer = new ArrayBuffer(3);
18 var uint8_array = new Uint8Array(arry_buffer);
19
20
21 var TimeNumber;//定时器
22
23 Page({
```

4.如果电脑和开发板在一个局域网下,电脑上面也可以看到图像了



关于按键

点击前后左右按键,会打印上下左右的箭头,具体看下面的程序介绍



源码说明(ESP32源码)

1.整体说明

程序上启用了两路UDP,一路用来传输图像(8888端口),另一路用来传输按键数据(9999端口)

2.UDP(传输图像)

1,基本的UDP啥的就不说了哈,就是在官方提供的例子上修改的.

2,启动UDP任务

```
main > C demo.c ...
332 //jpeg_quality = 10, //0-100, number means higher quality
333 .fb_count = 2 //if more than one, i2s runs in continuous mode. Use only with JPEG
334 };
335
336 //摄像头初始化
337 //gpio_install_isr_service(ESP_INTR_FLAG_LEVEL1); //设置中断优先级最低
338 err = esp_camera_init(&camera_config);
339 if (err != ESP_OK)
340 {
341     printf("Camera init failed with error 0x%x\r\n", err);
342     return;
343 }
344 setvbuf(stdout, NULL, _IONBF, 0);
345
346 wifi_init_softap(); //初始化模组连接的路由器
347
348 #ifdef CONFIG_EXAMPLE_IPV4
349     xTaskCreate(udp_server_jpeg_task, "udp_server_jpeg_task", 4096, (void*)AF_INET, 5, NULL);
350 #endif
351 #ifdef CONFIG_EXAMPLE_IPV6
352     xTaskCreate(udp_server_jpeg_task, "udp_server_jpeg_task", 4096, (void*)AF_INET6, 5, NULL);
353 #endif
354
355 #ifdef CONFIG_EXAMPLE_IPV4
356     xTaskCreate(udp_server_data_task, "udp_server_data_task", 4096, (void*)AF_INET, 6, NULL);
357 #endif
358 #ifdef CONFIG_EXAMPLE_IPV6
359     xTaskCreate(udp_server_data_task, "udp_server_data_task", 4096, (void*)AF_INET6, 6, NULL);
360 #endif
361
362 while (1)
363 {
364     // printf("hello camera!\r\n");
365     vTaskDelay(100);
366 }
```

3,UDP接收到数据之后判断下是否采集完了一幅图像,如果采集完了,同时接收到命令, 就把图片数据发给客户端

命令是 0x55 0xaa 0x01 0x02

```
C demo.c U x C defh C wifi_softap_sta.c U
main > C demo.c > udp_server_jpeg_task(void *)
125     inet6_ntoa_r(source_addr.sin6_addr, addr_str, sizeof(addr_str) - 1);
126 }
127
128 /*检测是否采集完了摄像头图像*/
129 fb = esp_camera_fb_get();
130 if (!fb)
131 {
132     ESP_LOGI(TAG, "Camera capture failed");
133     res = ESP_FAIL;
134 }
135 else
136 {
137     if (fb->format != PIXFORMAT_JPEG)
138     {
139         bool jpeg_converted = frame2jpeg(fb, 80, &_jpg_buf, &_jpg_buf_len);
140         if (!jpeg_converted)
141         {
142             ESP_LOGI(TAG, "JPEG compression failed");
143             esp_camera_fb_return(fb);
144             res = ESP_FAIL;
145         }
146     }
147     else
148     {
149         _jpg_buf_len = fb->len;
150         _jpg_buf = fb->buf;
151         res = ESP_OK;
152     }
153 }
154 if (res == ESP_OK)
155 {
156     if (len==4)
157     {
158         if (rx_buffer[0]==0x55 && rx_buffer[1]==0xAA && rx_buffer[2]==0x01 && rx_buffer[3]==0x02)
159         {
160             int err = sendto(sock, _jpg_buf, _jpg_buf_len, 0, (struct sockaddr *)&source_addr, sizeof(source_addr));
161             if (err < 0) {
162                 ESP_LOGE(TAG, "Error occurred during sending: errno %d", errno);
163                 break;
164             }
165         }
166     }
167 }
168
169 if (fb->format != PIXFORMAT_JPEG)
170 {
171     free(_jpg_buf);
172 }
173 esp_camera_fb_return(fb); //重新开始采集
```


2.UDP(按键数据)

我这里只是打印了下,具体实施大家伙可以做.

```
C demo.c U x C def.h C wifi_softap_sta.c U
main > C demo.c > udp_server_jpeg_task(void *)
238 // Error occurred during receiving
239 if (len < 0) {
240     ESP_LOGE(TAG, "recvfrom failed: errno %d", errno);
241     break;
242 }
243 // Data received
244 else {
245     // Get the sender's ip address as string
246     if (source_addr.sin6_family == PF_INET) {
247         inet_ntoa_r(((struct sockaddr_in *)&source_addr)->sin_addr.s_addr, addr_str, sizeof(addr_str) - 1);
248     } else if (source_addr.sin6_family == PF_INET6) {
249         inet6_ntoa_r(source_addr.sin6_addr, addr_str, sizeof(addr_str) - 1);
250     }
251
252     if(len==3)
253     {
254         if (rx_buffer[0]==0x55 && rx_buffer[1]==0xAA)
255         {
256             if (rx_buffer[2]==0x00)
257             {
258                 /* 停止 */
259             }
260             else if(rx_buffer[2]==0x01)
261             {
262                 ESP_LOGI(TAG, "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!");
263             }
264             else if(rx_buffer[2]==0x02)
265             {
266                 ESP_LOGI(TAG, "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!");
267             }
268             else if(rx_buffer[2]==0x03)
269             {
270                 ESP_LOGI(TAG, "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!");
271             }
272             else if(rx_buffer[2]==0x04)
273             {
274                 ESP_LOGI(TAG, "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!");
275             }
276         }
277     }
278 }
279
280
281 if (sock != -1) {
282     ESP_LOGE(TAG, "Shutting down socket and restarting...");
283     shutdown(sock, 0);
284 }
```

源码说明(微信小程序源码)

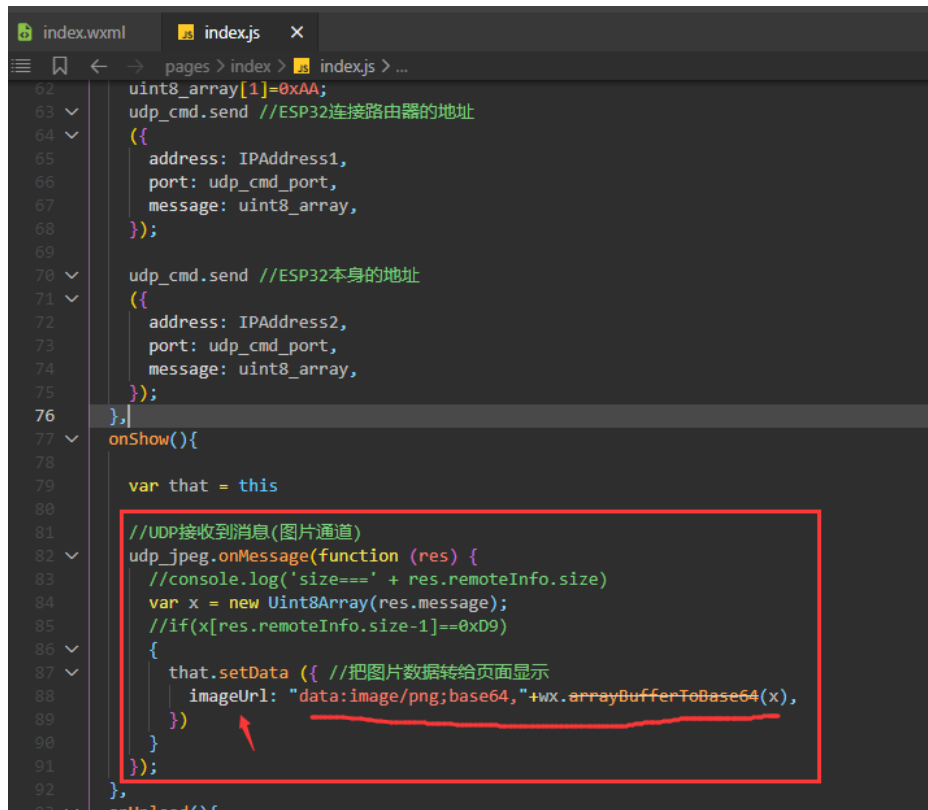
1.创建UDP

```
1 // index.js
2 var util = require("../utils/util.js");
3
4 // 获取应用实例
5 const app = getApp()
6
7 var IPAddress1 = "192.168.0.102";
8 var IPAddress2 = "192.168.4.1";
9
10 /**UDP传输图片 */
11 var udp_jpeg;
12 var udp_jpeg_port = 8888;
13
14 /**UDP传输按键控制命令 */
15 var udp_cmd;
16 var udp_cmd_port = 9999;
17 const array_buffer = new ArrayBuffer(3);
18 var uint8_array = new Uint8Array(array_buffer);
19
20
21 var TimeNumber;//定时器
22
23 Page({
24   data: {
25     imageUrl: "../images/test_jpg.jpg",
26   },
27   onLoad() {
28     /**UDP传输按键控制命令 */
29     udp_cmd = wx.createUDPSocket();//启用UDP
30     udp_cmd.bind()
31
32     /**UDP传输图片 */
33     udp_jpeg = wx.createUDPSocket();//启用UDP
34     udp_jpeg.bind()
35     /**定时器 */
36     try { clearInterval(TimeNumber); } catch (e) { }
37     TimeNumber = setInterval(this.TimeNumberCallBack, 10, "null");
38   },
39   /**
```

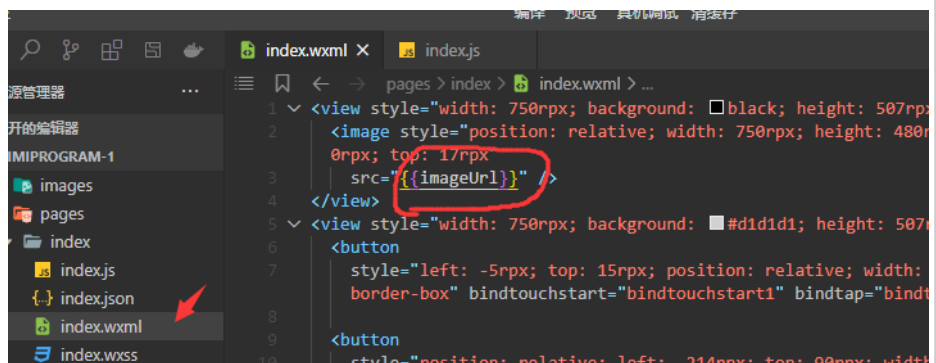
2.使用定时器轮训发送数据

```
index.wxml  index.js  x
pages > index > index.js > ...
34  udp_jpeg.bind()
35  /**定时器 */
36  try { clearInterval(TimeNumber); } catch (e) { }
37  TimeNumber = setInterval(this.TimeNumberCallBack, 10, "null");
38  },
39  /**
40   * 定时器回调函数
41   */
42  TimeNumberCallBack: function (arg) {
43    /**发送命令获取图像数据 */
44    var udp_data = util.StringToHex("55 AA 01 02");
45    udp_jpeg.send //ESP32连接路由器的地址
46    ({
47      address: IPAddress1,
48      port: udp_jpeg_port,
49      message: udp_data
50    });
51
52    udp_jpeg.send //ESP32本身的地址
53    ({
54      address: IPAddress2,
55      port: udp_jpeg_port,
56      message: udp_data
57    });
58
59
60    /**发送按键状态 */
61    uint8_array[0]=0x55;
62    uint8_array[1]=0xAA;
63    udp_cmd.send //ESP32连接路由器的地址
64    ({
65      address: IPAddress1,
66      port: udp_cmd_port,
67      message: uint8_array,
68    });
69
70    udp_cmd.send //ESP32本身的地址
71    ({
72      address: IPAddress2,
73      port: udp_cmd_port,
74      message: uint8_array,
75    });
76  },
77  onShow(){
78
```

3.接收图片数据并把图片数据流添加到image控件



```
62 uint8_array[1]=0xAA;
63 udp_cmd.send //ESP32连接路由器的地址
64 ({
65   address: IPAddress1,
66   port: udp_cmd_port,
67   message: uint8_array,
68 });
69
70 udp_cmd.send //ESP32本身的地址
71 ({
72   address: IPAddress2,
73   port: udp_cmd_port,
74   message: uint8_array,
75 });
76
77 onShow(){
78
79   var that = this
80
81   //UDP接收到消息(图片通道)
82   udp_jpeg.onMessage(function (res) {
83     //console.log('size=='+ res.remoteInfo.size)
84     var x = new Uint8Array(res.message);
85     //if(x[res.remoteInfo.size-1]==0x09)
86     {
87       that.setData ({ //把图片数据转给页面显示
88         imageUrl: "data:image/png;base64,"+wx.arrayBufferToBase64(x),
89       })
90     }
91   });
92
93   onUnload(){
```



```
1 <view style="width: 750rpx; background: black; height: 507rpx;
2   <image style="position: relative; width: 750rpx; height: 480rpx; top: 17rpx;
3     src="{{imageUrl}}" />
4 </view>
5 <view style="width: 750rpx; background: #d1d1d1; height: 507rpx;
6   <button
7     style="left: -5rpx; top: 15rpx; position: relative; width:
8     border-box" bindtouchstart="bindtouchstart1" bindtap="bind
9   <button
10    style="position: relative; left: -214rpx; top: 90rpx; width:
```

4.关于按键

按键数据的前两个数据是固定的: 0x55 0xaa

后面一个数据是根据按键状态改变的,

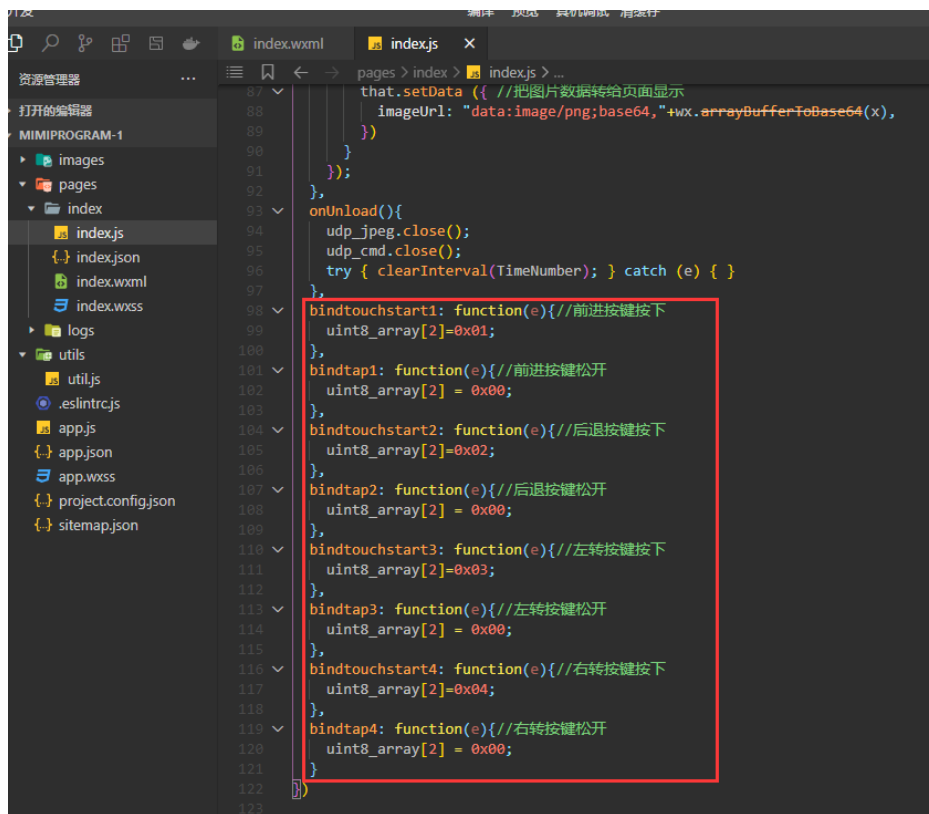
按下前进(0x01),

按下后退(0x02),

按下左转(0x03),

按下右转(0x04),

凡是松开哪一个按键最后一个数据都会变为 0x00



分类: ESP32学习开发

好文要顶

关注我

收藏该文



杨奉武

关注 - 1

粉丝 - 693

0

0

« 上一篇: [802-ESP32_SDK开发-ESP32\(WiFi\)把采集的摄像头照片数据通过UDP发送给UDP客户端\(C# UDP客户端\)](#)

posted on 2021-12-10 13:00 杨奉武 阅读(0) 评论(0) 编辑 收藏 举报

[刷新评论](#) [刷新页面](#) [返回顶部](#)

发表评论

编辑

预览

B



</>

“



支持 Markdown

☒ 自动补全

提交评论

退出

[Ctrl+Enter]快捷键提交

【推荐】跨平台组态\工控\仿真\CAD 50万行C++源码全开放免费下载！

【推荐】华为 HMS Core 线上 Codelabs 挑战赛第4期，探索 “智” 感生活

编辑推荐：

- 如何在 ASP.NET Core 中构建轻量级服务
- 理解ASP.NET Core - 模型绑定&验证
- [翻译].NET 6 中的 dotnet monitor
- .NET Core 如何配置 TLS Cipher (套件) ？
- 记一次 .NET 某智能服装智造系统 内存泄漏分析

最新新闻：

- “内忧外患”，薄荷健康危局乍现 (2021-12-10 11:40)
- 小米手机卖不动了：利润低、创新弱，8000家门店也没用 (2021-12-10 11:34)
- 联想控股内网发声明：2009年联想控股29%股权转让合法依规 (2021-12-10 11:30)
- B站笼络中年人 (2021-12-10 11:27)
- 70后的童年记忆：任天堂“红白机FC之父” 上村雅之去世 (2021-12-10 11:21)
- » 更多新闻...

历史上的今天：

2020-12-10 2-STM32+BC26/260Y基本控制篇-整体运行测试-APP扫码绑定BC26,并通过MQT...
2019-12-10 ESP8266 AT指令开发(基于STC89C52(58)单片机): 硬件使用说明
2016-12-10 备用电源

Powered by:

博客园

Copyright © 2021 杨奉武

Powered by .NET 6 on Kubernetes



单片机,物联网,上位机,...

扫一扫二维码, 入群聊。