

## 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 - 810, 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(24)  
 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)

## 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)

<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](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(105126)
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\(C# UDP客户端\)](#)
- 
- 

## 说明

这一节是模组采集完一幅图片以后通过UDP把图片数据发送出去。





作为WiFi模块其实和其网络通信有2种方式。

1是连接其无线进行网络通信; 2是控制WiFi连接路由器,在一个路由器下进行网络通信;

因为连接其无线之后IP地址是模块分配的,具有随机性,所以这节测试在同一个路由器下通信;

## 测试(确保自己电脑和WiFi连接同一个路由器)

### 1.先双击运行C#源码中的执行文件

UDPCom > UDPCom > bin > Debug			
	名称	修改日期	类型
我的摄像头照片3	 UDPCom.exe	2021/11/28 12:23	应用程
	 UDPCom.pdb	2021/11/28 12:23	PDB
	 UDPCom.vshost.exe	2021/11/28 14:10	应用程
	 UDPCom.vshost.exe.manifest	2016/7/16 19:44	MAN

### 2.然后点击启动

提示: 192.168.1.93:8888

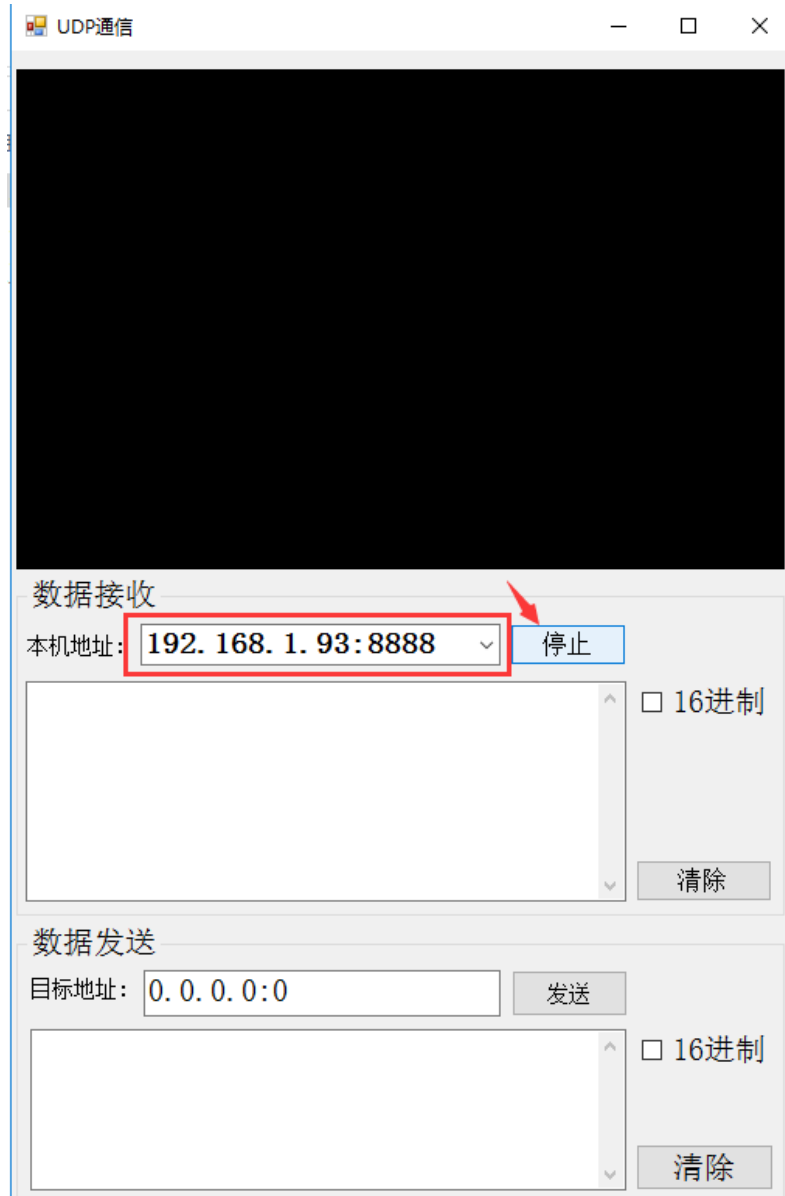
是说该UDP客户端的IP地址是192.168.1.93 监听的端口号是:8888

这个后面要用到。

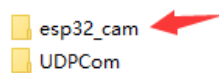
1. 用ESP8266+android,制作自己的WIFI小车(ESP8266篇)(9)
2. C#委托+回调详解(9)
3. 用ESP8266+android,制作自己的WIFI小车(Android 软件)(6)
4. 我的大学四年(6)
5. ESP8266使用详解(AT,LUA,SDK)(6)

#### 最新评论

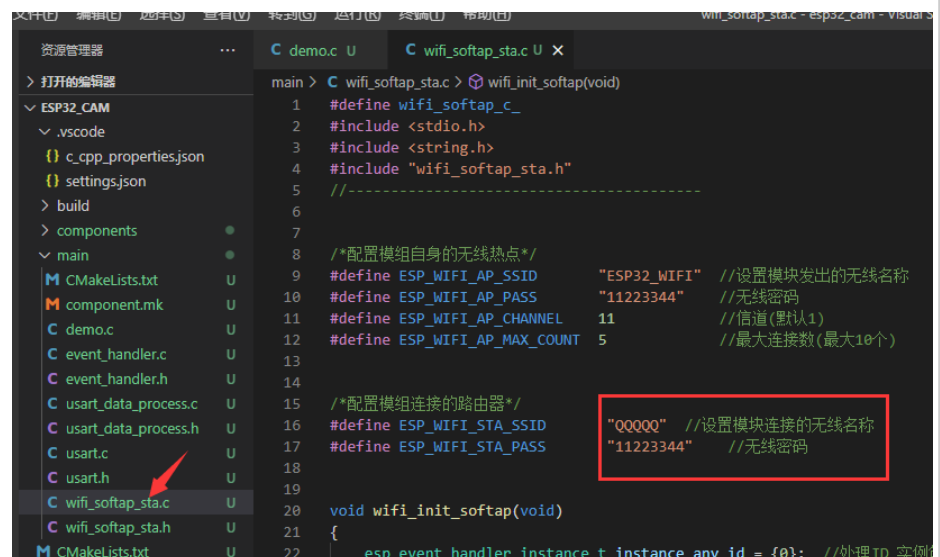
1. Re:2-6-1-视频传输,监控,直播方案-手机连接ESP32的热点,使用微信小程序查看摄像头图像(WiFi视频小车,局域网视频监控)  
赞赞赞,感谢大佬无私奉献  
--SJA2C2A
2. Re:中移动M5311模块使用手册(TCP,MQTT)  
请问你用的usb转ttl是哪种呢,我用的ch340可是开机串口助手没有SIM识别显示  
--夏洛的网娅



### 3.把这节的ESP32代码放到非中文目录,然后使用VS Code打开



#### 4.根据自家的路由器修改(和电脑连接同一个路由器)

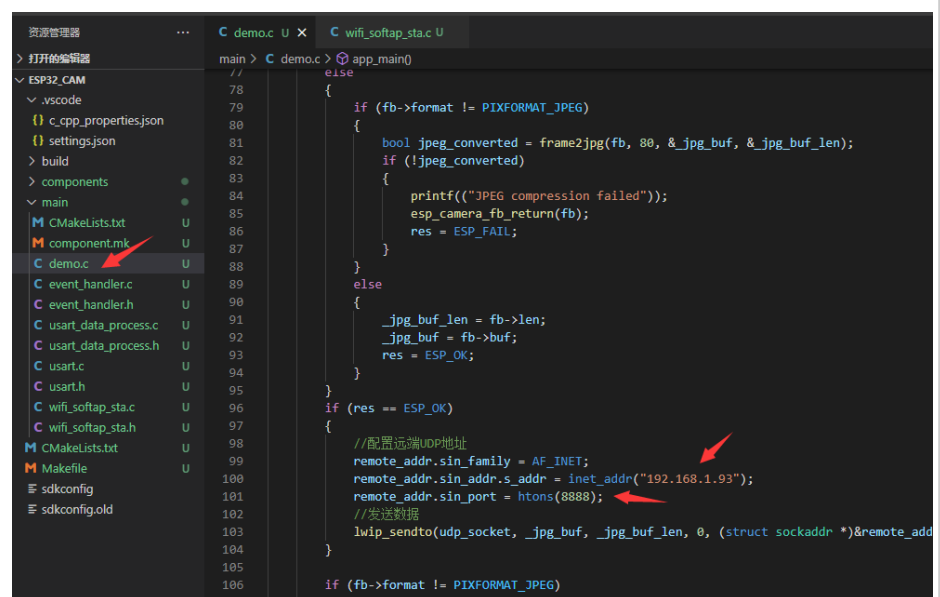


```
main > C wifi_softap_sta.c > wifi_init_softap(void)
1  #define wifi_softap_c_
2  #include <stdio.h>
3  #include <string.h>
4  #include "wifi_softap_sta.h"
5  //-----
6
7
8  /*配置模组自身的无线热点*/
9  #define ESP_WIFI_AP_SSID      "ESP32_WIFI" //设置模块发出的无线名称
10 #define ESP_WIFI_AP_PASS     "11223344" //无线密码
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     "11223344" //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)
21 {
22     esp_event_handler_instance_t instance_any_id = {0}; //处理ID 实例
```

#### 5.设置ESP32模块发送的UDP地址

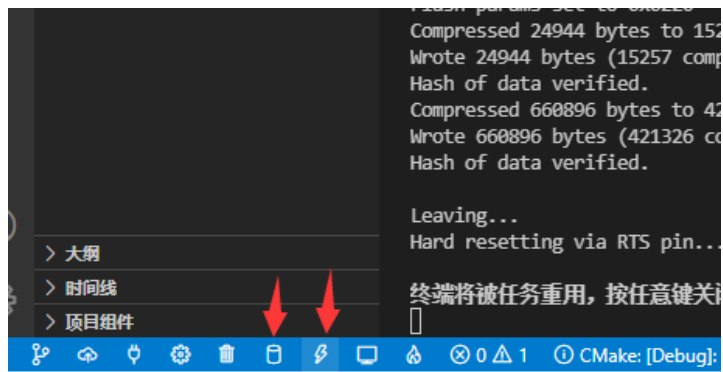
改为上面上位机的地址哈,我上面是192.168.1.93:8888

这个要根据自己的修改....



```
main > C demo.c > app_main()
//
78 else
79 {
80     if (fb->format != PIXFORMAT_JPEG)
81     {
82         bool jpeg_converted = frame2jpg(fb, 80, &_jpg_buf, &_jpg_buf_len);
83         if (!jpeg_converted)
84         {
85             printf(("JPEG compression failed"));
86             esp_camera_fb_return(fb);
87             res = ESP_FAIL;
88         }
89     }
90     else
91     {
92         _jpg_buf_len = fb->len;
93         _jpg_buf = fb->buf;
94         res = ESP_OK;
95     }
96 }
97 if (res == ESP_OK)
98 {
99     //配置远端UDP地址
100     remote_addr.sin_family = AF_INET;
101     remote_addr.sin_addr.s_addr = inet_addr("192.168.1.93");
102     remote_addr.sin_port = htons(8888);
103     //发送数据
104     lwip_sendto(udp_socket, _jpg_buf, _jpg_buf_len, 0, (struct sockaddr *)&remote_addr);
105 }
106 if (fb->format != PIXFORMAT_JPEG)
107 {
```

#### 6.编译下载到开发板(第一次编译时间有点长)



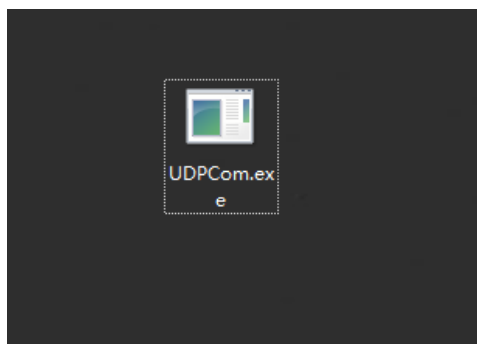
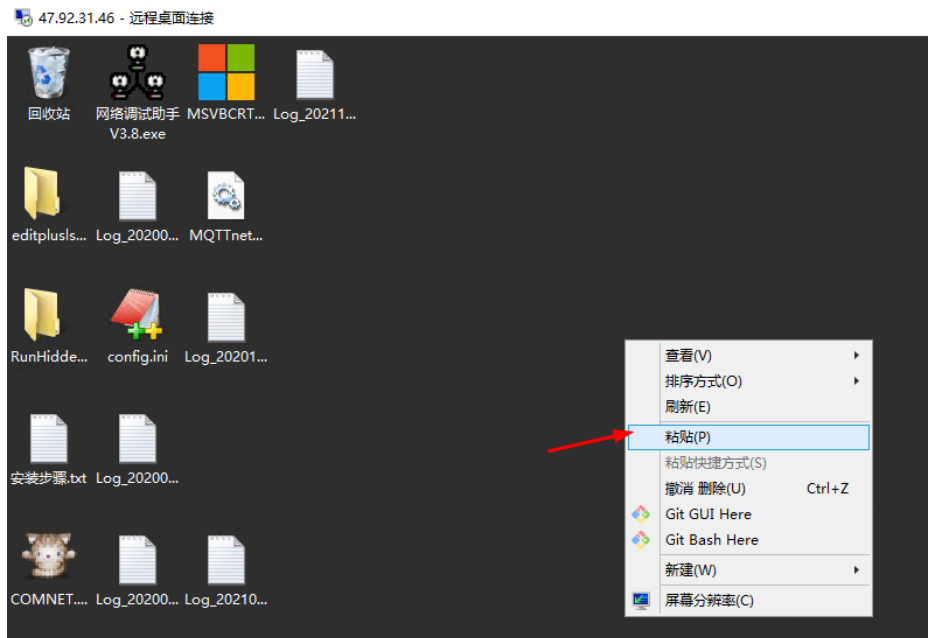
## 7.然后就可以看到图像了



如果有云服务器可以把上位机放到云服务器上查看视频

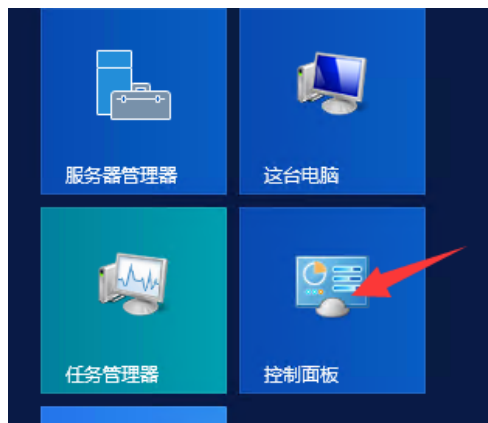
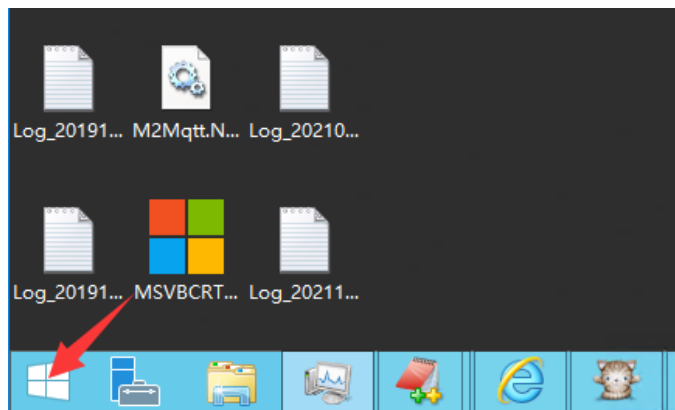
## 1.把软件复制粘贴到云服务器



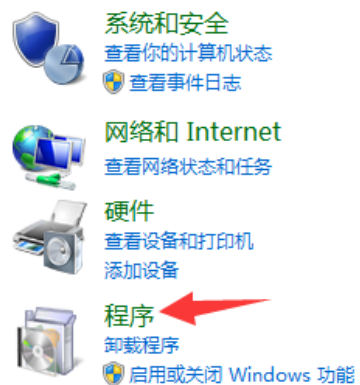


**2.双击运行如果运行不起来出现如下提示(这是确实.net库导致的)...**  
**跟着下面的步骤做,安装net库**





### 调整计算机的设置







## 程序和功能

[卸载程序](#) | [启用或关闭 Windows 功能](#) | [查看已安装的更新](#) |  
[运行为以前版本的 Windows 编写的程序](#) | [如何安装程序](#)



## 默认程序

[更改媒体或设备的默认设置](#) | [始终使用指定的程序打开某种文件类型](#) | [设置默认程序](#)



## Java

添加角色和功能向导

目标服务器  
iZvqdlzfp62noZ

开始之前

开始之前

安装类型

服务器选择

服务器角色

功能

确认

结果

该向导可帮助你安装角色、角色服务或功能。你需要根据组织的计算要求(例如共享文档或托管网站)确定要安装的角色、角色服务或功能。

要删除角色、角色服务或功能,请执行以下操作:  
[启动“删除角色和功能”向导](#)

在继续之前,请确认完成以下任务:

- 管理员帐户使用的是强密码
- 静态 IP 地址等网络设置已配置完成
- 已从 Windows 更新安装最新的安全更新

如果你必须验证是否已完成上述任何先决条件,请关闭向导,完成这些步骤,然后再次运行向导。

单击“下一步”继续。

☐ 默认情况下将跳过此页(S)

< 上一步(B)

下一步(N) >

安装(I)

取消

添加角色和功能向导

目标服务器  
iZvqdlzfp62noZ

选择安装类型

开始之前

安装类型

服务器选择

服务器角色

功能

确认

结果

选择安装类型。你可以在正在运行的物理计算机、虚拟机或脱机虚拟硬盘(VHD)上安装角色和功能。

☒ **基于角色或基于功能的安装**  
通过添加角色、角色服务和功能来配置单个服务器。

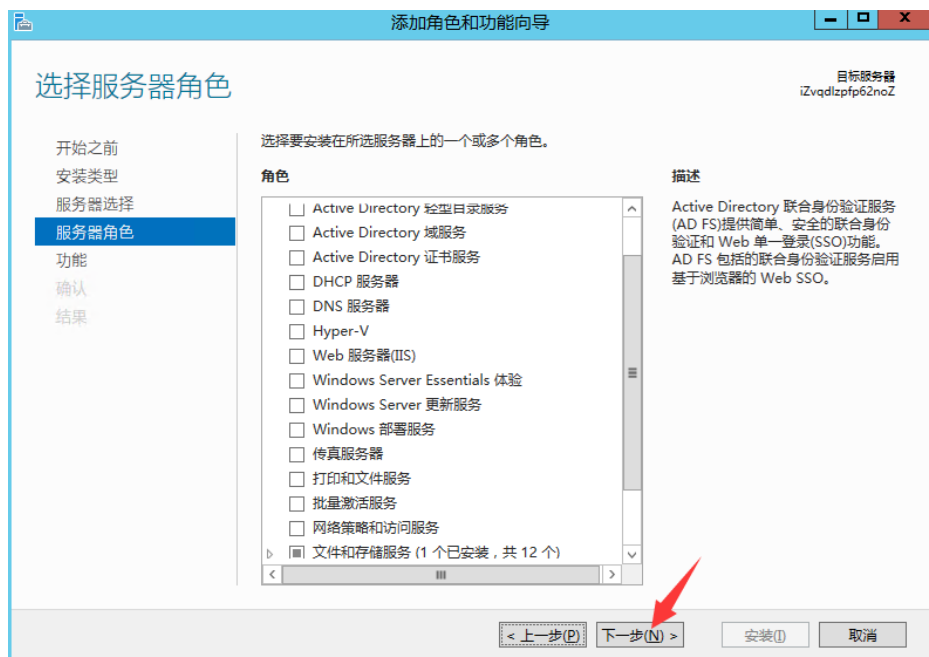
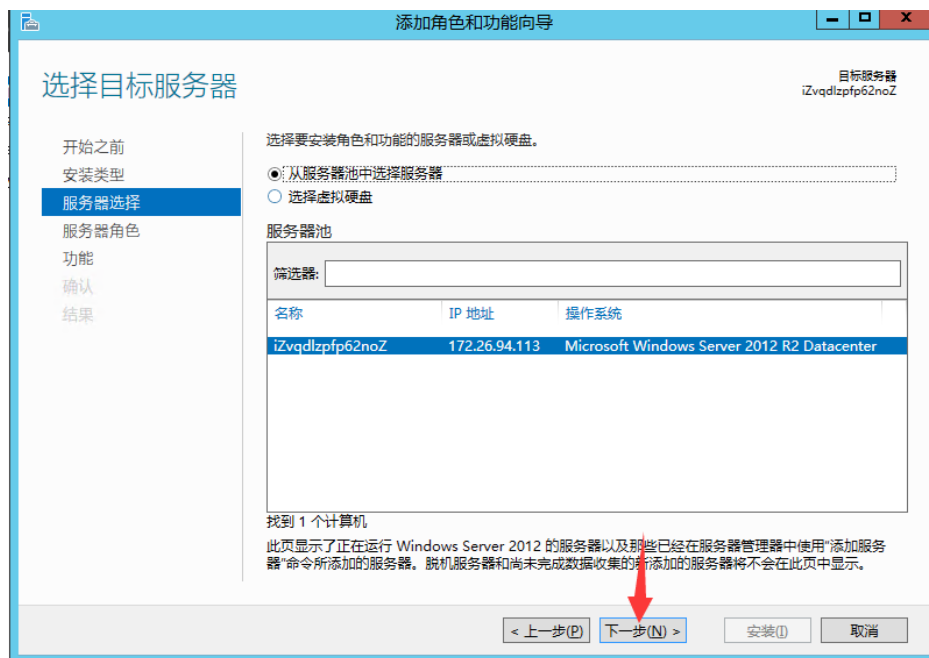
☐ **远程桌面服务安装**  
为虚拟桌面基础结构(VDI)安装所需的角色服务以创建基于虚拟机或基于会话的桌面部署。

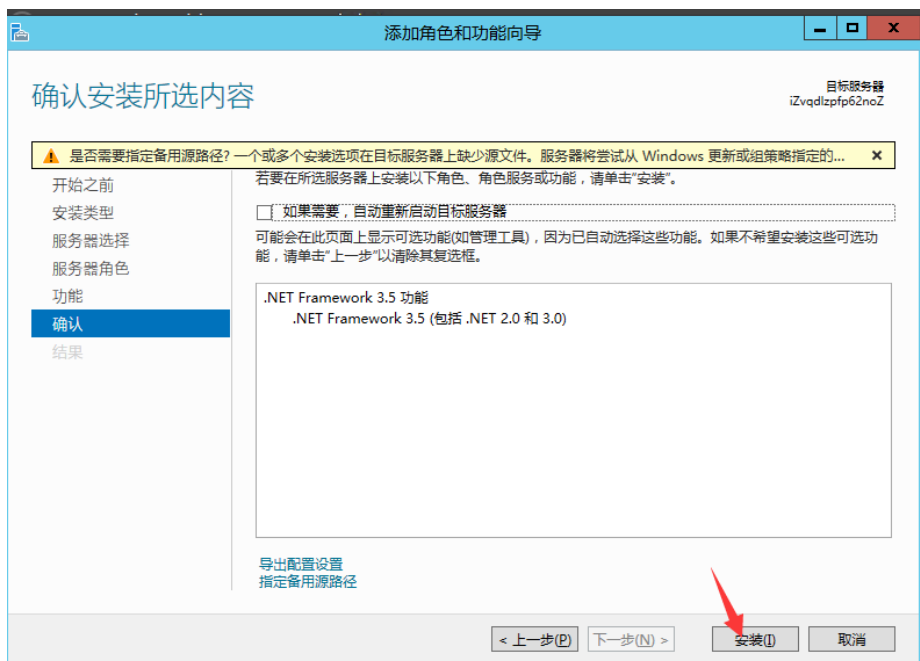
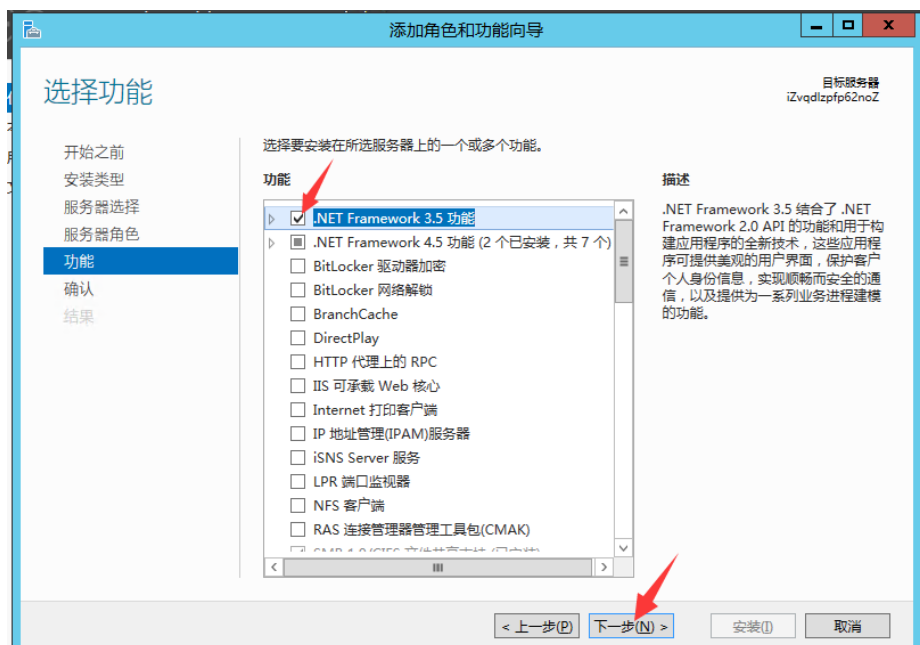
< 上一步(B)

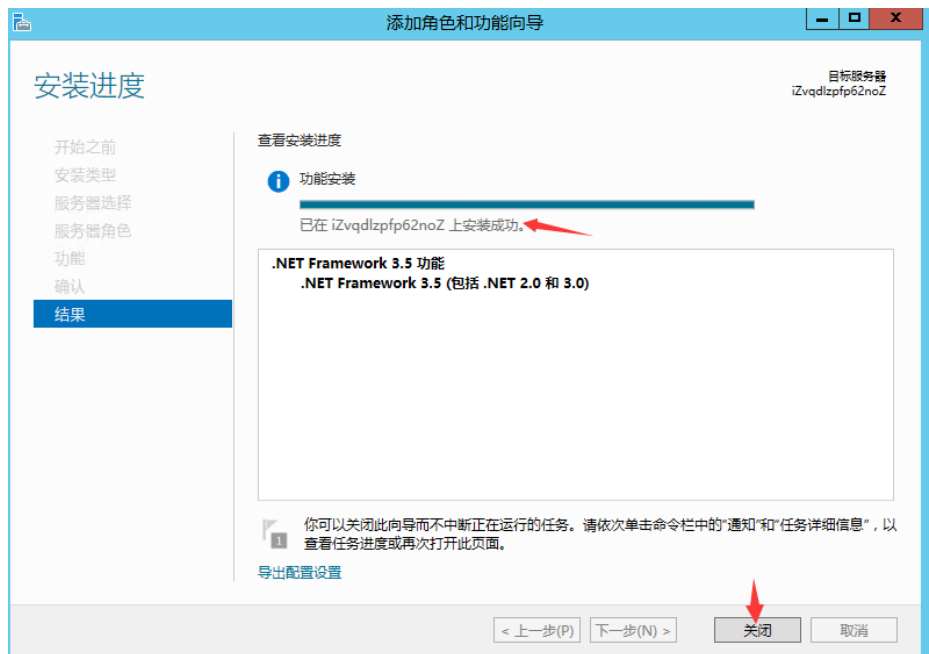
下一步(N) >

安装(I)

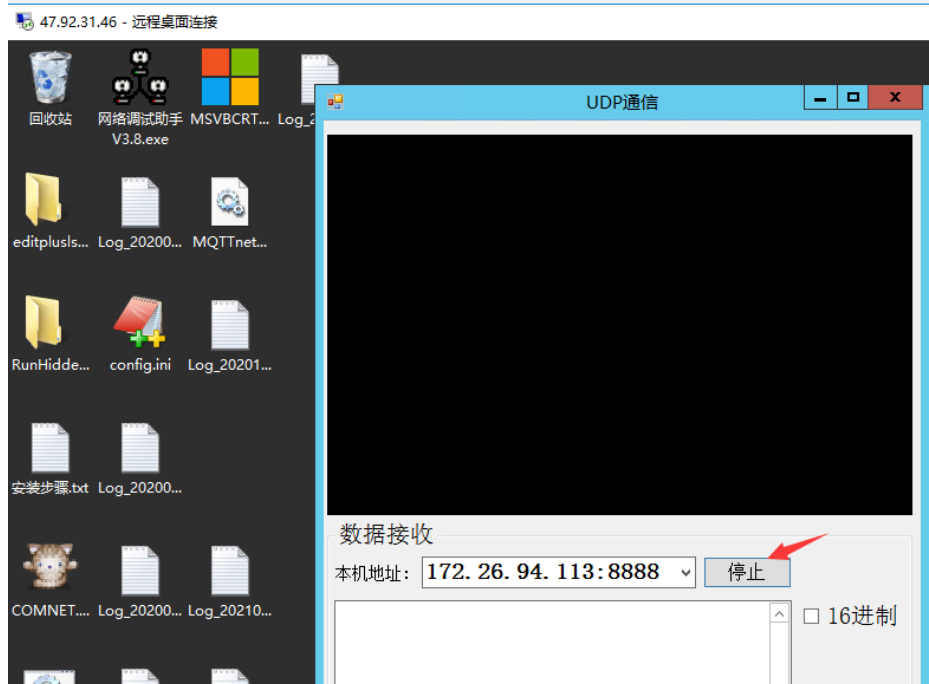
取消





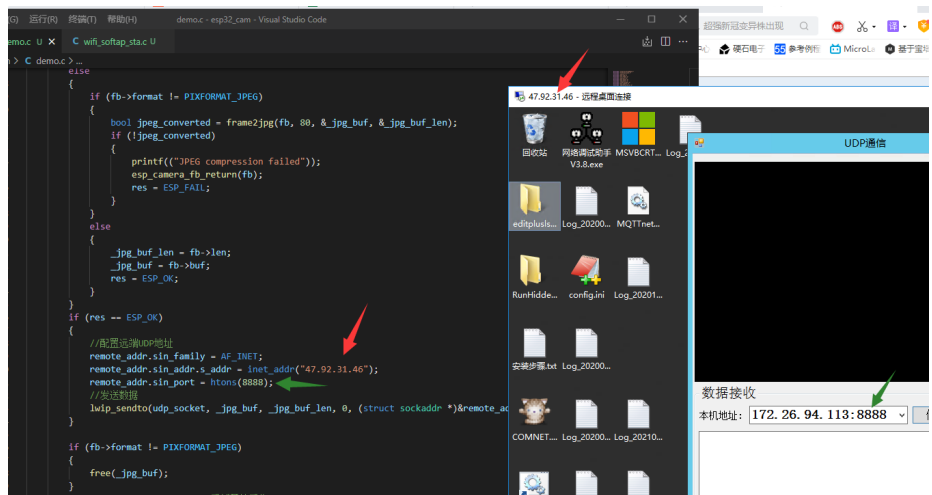


### 3.双击运行

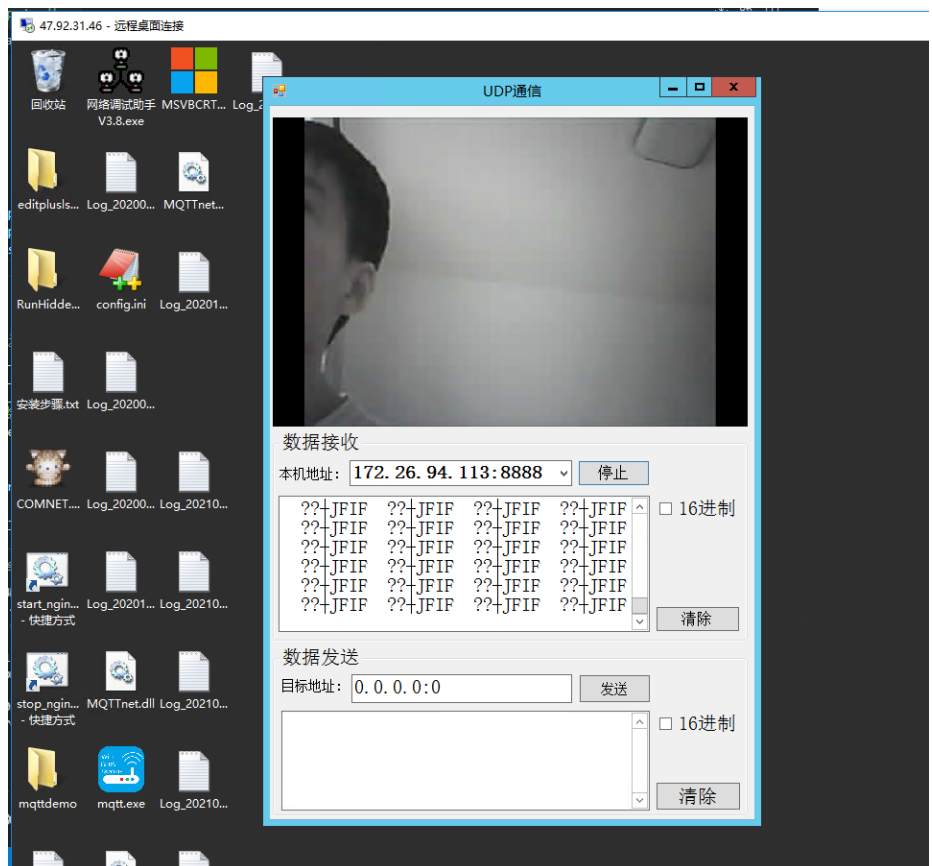


### 4.改一下ESP32的UDP发送地址

注意哈, IP地址要填写服务器的IP地址(外网IP)



5.主意哈,保证WiFi模块是连接的可以上网的路由器, 然后重新烧录, 然后就可以了



## 源码说明(ESP32源码)

1,初始化配置摄像头输出的是JPEG格式的图片数据; 初始化WiFi热点; 初始化UDP

```
main > C demo.c > app_main()
114
115
116 void app_main()
117 {
118     //初始化 NVS(配置WiFi的参数存储需要用到NVS)
119     esp_err_t err = nvs_flash_init();
120     if (err == ESP_ERR_NVS_NO_FREE_PAGES || err == ESP_ERR_NVS_NEW_VERSION_FOUND) {
121         ESP_ERROR_CHECK(nvs_flash_erase());
122         err = nvs_flash_init();
123     }
124     ESP_ERROR_CHECK(err);
125     //初始化内部的lwip
126     ESP_ERROR_CHECK(esp_netif_init());
127     //创建系统事件任务
128     ESP_ERROR_CHECK(esp_event_loop_create_default());
129
130     camera_config_t camera_config = { ...
131
132     //摄像头初始化
133     //gpio_install_isr_service(ESP_INTR_FLAG_LEVEL1); //设置中断优先级最低
134     err = esp_camera_init(&camera_config);
135     if (err != ESP_OK)
136     {
137         printf("Camera init failed with error 0x%x\r\n", err);
138         return;
139     }
140     setvbuf(stdout, NULL, _IONBF, 0);
141
142     wifi_init_softap(); //初始化模组连接的路由器
143
144     udp_init_fun(); //初始化UDP
145     while (1)
146     {
147         jpg_stream();
148         // printf("hello camera!\r\n");
149         // vTaskDelay(100);
150     }
151 }
152
```

```
/*初始化本地UDP*/
void udp_init_fun(void)
{
    struct sockaddr_in addr;
    udp_socket = lwip_socket(AF_INET, SOCK_DGRAM, 0);
    if (udp_socket < 0)
    {
        printf("socket创建失败\r\n");
        while(1); //等待重启
    }
    addr.sin_family = AF_INET;
    int err = lwip_bind(udp_socket, (struct sockaddr *)&addr, sizeof(addr));
    if (err < 0)
    {
        printf("bind失败\r\n");
        while(1); //等待重启
    }
}
```

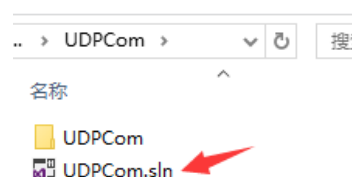
2,在任务里面只要判断采集了一副图片,就把图片数据发送到UDP.

```
C demo.c U X C wifi_softap_sta.c U
main > C demo.c > app_main()
109     }
170     setvbuf(stdout, NULL, _IONBF, 0);
171
172     wifi_init_softap();//初始化模组连接的路由器
173
174     udp_init_fun();//初始化UDP
175     while (1)
176     {
177         jpg_stream();
178         // printf("hello camera!\r\n");
179         // vTaskDelay(100);
180     }
181 }
182
```

```
main > C demo.c > app_main()
09     while (true)
10     {
170         fb = esp_camera_fb_get();
171         if (!fb)
172         {
173             printf(("Camera capture failed"));
174             res = ESP_FAIL;
175         }
176         else
177         {
178             if (fb->format != PIXFORMAT_JPEG)
179             {
180                 bool jpeg_converted = frame2jpg(fb, 80, &_jpg_buf, &_jpg_buf_len);
181                 if (!jpeg_converted)
182                 {
183                     printf(("JPEG compression failed"));
184                     esp_camera_fb_return(fb);
185                     res = ESP_FAIL;
186                 }
187             }
188             else
189             {
190                 _jpg_buf_len = fb->len;
191                 _jpg_buf = fb->buf;
192                 res = ESP_OK;
193             }
194         }
195         if (res == ESP_OK)
196         {
197             //配置远端UDP地址
198             remote_addr.sin_family = AF_INET;
199             remote_addr.sin_addr.s_addr = inet_addr("192.168.1.93");
200             remote_addr.sin_port = htons(8888);
201             //发送数据
202             lwip_sendto(udp_socket, _jpg_buf, _jpg_buf_len, 0, (struct sockaddr *)&remote_addr, sizeof(remote_addr));
203         }
204         if (fb->format != PIXFORMAT_JPEG)
205         {
206             free(_jpg_buf);
207         }
208         esp_camera_fb_return(fb);//重新开始采集
209     }
210 }
211
212
213
```

## 源码说明(C#源码)

### 1.打开工程



## 2.启动程序的时候获取本机的IP地址

```
m1.cs x Form1.cs [设计]
UDPCom UDPCom.Form1
70
71
72 1 个引用|0 项更改|0 名作者, 0 项更改
73 private void Form1_Load(object sender, EventArgs e)
74 {
75     getIPAddress();
76 }
77
78 /// <获取本机 IP 地址>
79 ///
80 /// </summary>
81 /// </returns></returns>
82 1 个引用|0 项更改|0 名作者, 0 项更改
83 private void getIPAddress()
84 {
85     IPAddress[] hostipspool = Dns.GetHostAddresses("");
86     comboBoxIPAddress.Items.Clear();
87     foreach (IPAddress ipa in hostipspool)
88     {
89         if (ipa.AddressFamily == AddressFamily.InterNetwork)
90         {
91             //comboBoxIPAddress.Items.Add(ipa.ToString());
92             comboBoxIPAddress.Items.Add(ipa.ToString() + ":" + SocketPort);
93             comboBoxIPAddress.SelectedIndex = comboBoxIPAddress.Items.Count > 0 ? 0 : -1;
94         }
95     }
96 }
97
98 1 个引用|0 项更改|0 名作者, 0 项更改
99 private void textBoxSend_TextChanged(object sender, EventArgs e)
100 {
101 }
```

## 3.点击启动按键

```
Form1.cs x Form1.cs [设计]
UDPCom UDPCom.Form1
100
101 1 个引用|0 项更改|0 名作者, 0 项更改
102 private void button1_Click(object sender, EventArgs e)
103 {
104     if (button1.Text=="启动")
105     {
106         try
107         {
108             // 创建负责监听的套接字, 注意其中的参数;
109             socketWatch = new Socket(AddressFamily.InterNetwork, SocketType.Dgram, ProtocolType.Udp);
110             // 获得文本框中的IP对象;
111             string ListenAddress = comboBoxIPAddress.Text.Trim();
112             string[] ListenAddressTemp = ListenAddress.Split(':');
113             IPAddress address = IPAddress.Parse(ListenAddressTemp[0]);
114
115             // 创建包含ip和端口号的网络节点对象;
116             IPEndPoint endPoint = new IPEndPoint(address, int.Parse(ListenAddressTemp[1]));
117
118             try
119             {
120                 // 将负责监听的套接字绑定到唯一的ip和端口上; 创建UDP
121                 socketWatch.Bind(endPoint);
122             }
123             catch (SocketException se)
124             {
125                 MessageBox.Show("异常: " + se.Message);
126                 return;
127             }
128             button1.Text = "停止";
129
130             threadReadData = new Thread(ReadDataMethod);
131             threadReadData.IsBackground = true;
132             threadReadData.Start();
133             // 启动UDP接收数据任务
134         }
135         catch (Exception e1)
136         {
137             MessageBox.Show(e1.ToString());
138         }
139     }
140     else
141     {
142         try
143         {
144             socketWatch.Close();
145         }
146         catch (Exception)
147         {
148             {}
149         }
150         button1.Text = "启动";
151     }
152 }
```



## 4.接收并持续显示图片数据

```
orm1.cs  Form1.cs [设计]
UDPCom
UDPCom.Form1
@_getIPAddress()

151  /// <summary>
152  /// 接收数据
153  /// </summary>
154  void ReadDataMethod()
155  {
156      while (button1.Text=="停止") // 持续不断的监听客户端的连接请求;
157      {
158          try
159          {
160              EndPoint point = new IPEndPoint(IPAddress.Any, 0); //用来保存发送方的ip和端口号
161              byte[] buffer = new byte[1024000];
162              int length = socketWatch.ReceiveFrom(buffer, ref point); //接收数据报
163              byte[] buff = new byte[length];
164              Array.Copy(buffer, buff, length); //拷贝数据
165
166              if (length > 2)
167              {
168                  /*判断接收到了图片数据头*/
169                  if ((buff[0] & 0xff) == 0xff && (buff[1] & 0xff) == 0xd8 && (buff[2] & 0xff) == 0xff)
170                  {
171                      image_data_start = true;
172                      image_buff_cnt = 0;
173                  }
174              }
175
176              if (image_data_start) //开始接收数据
177              {
178                  /*数据不得超过缓存数组*/
179                  if (image_buff_cnt < image_buff.Length - length)
180                  {
181                      Buffer.BlockCopy(buff, 0, image_buff, image_buff_cnt, length);
182                      image_buff_cnt = image_buff_cnt + length;
183                  }
184                  else
185                  {
186                      image_buff_cnt = 0;
187                      image_data_start = false;
188                  }
189              }
190
191              /*接收到图片结尾*/
192              if (image_buff_cnt > 2 && (image_buff[image_buff_cnt - 1] & 0xff) == 0xd9 && (image_buff[image_buff_cnt - 2] & 0xff) == 0xff)
193              {
194                  try
195                  {
196                      MemoryStream ms = new MemoryStream(image_buff, 0, image_buff_cnt);
197                      Image img = Image.FromStream(ms, true);
198                  }
199              }
200          }
201          catch { }
202      }
203  }
```

UDP接收数据

判断图片数据,显示图片

```
orm1.cs  Form1.cs [设计]
UDPCom
UDPCom.Form1
@_getIPAddress()

196  {
197      image_buff_cnt = 0;
198      image_data_start = false;
199  }
200
201  /*接收到图片结尾*/
202  if (image_buff_cnt > 2 && (image_buff[image_buff_cnt - 1] & 0xff) == 0xd9 && (image_buff[image_buff_cnt - 2] & 0xff) == 0xff)
203  {
204      try
205      {
206          MemoryStream ms = new MemoryStream(image_buff, 0, image_buff_cnt);
207          Image img = Image.FromStream(ms, true);
208          image_data_start = false;
209
210          Invoke((new Action() =>
211          {
212              if (checkBox1.Checked)
213              {
214                  pictureBox1.Refresh();
215                  pictureBox1.Image = img; //显示图片
216              }
217          }));
218      }
219      catch (Exception)
220      { }
221  }
222
223  /*把接收的数据显示*/
224  Invoke((new Action() =>
225  {
226      if (checkBox1.Checked)
227      {
228          textBoxDataRes.AppendText(byteToHexStr(buff));
229      }
230      else
231      {
232          textBoxDataRes.AppendText(Encoding.Default.GetString(buff)); //对数据进行追加显示数据
233      }
234  }));
235  }
236  catch (Exception)
237  { }
238  }
```

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: [801-ESP32\\_SDK开发-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\(C# 串口上位机\)](#)

posted on 2021-12-10 12:53 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



单片机,物联网,上位机,...

扫一扫二维码, 入群聊。