WIFI Camera

Configure the WIFI camera network mode.

```
For the first use, it is not recommended to configure it yourself, just keep the
factory settings!
For more detailed information about the WiFi camera, please refer to:
http://www.yahboom.net/study/ROS-WiFi
```

Device connection

Hardware connection

Use Type-B data cable to connect Arduino Uno and computer.

Software connection

Open the "Arduino IDE" software and select the model and serial port number corresponding to the development board.

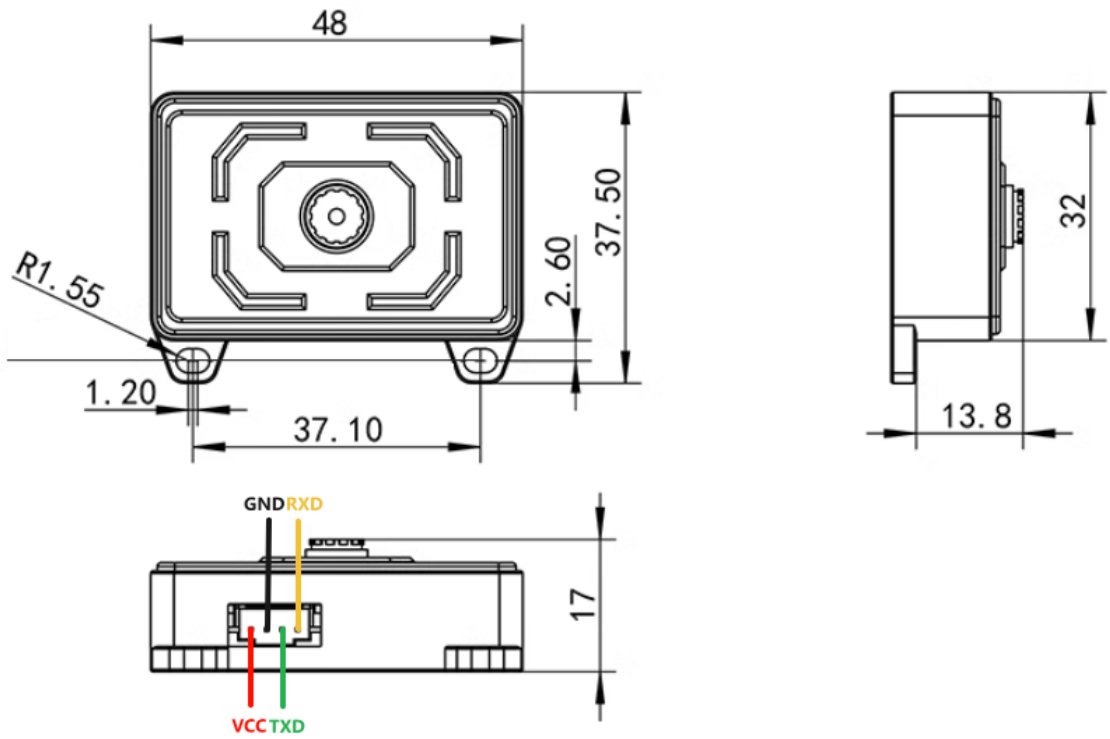
WIFI camera

The WIFI camera module integrates the ESP32-S3 chip and 200W camera. The serial port or serial port module can be used for network configuration to achieve wireless image transmission.

Control principle

We mainly use the WIFI camera module for application, and do not make underlying modifications. We only need to send data in a specified format through the serial port interface of the WIFI camera module to make corresponding configurations.

Control pins



Peripheral module (WIFI camera)	Arduino Uno
VCC	VCC
TXD	RX (0)
RXD	TX (1)
GND	GND

Control instructions

If it is the first time to use, it is not recommended to configure it yourself.

Command	Description	Example	Return value
wifi_mode:	Configure network mode → 0: AP mode 1: STA mode 2: AP+STA mode	wifi_mode:2	ok
sta_ssid:	Name of connected wifi	sta_ssid:Yahboom2	ok
sta_pd:	Password of connected wifi	sta_pd:yahboom890729	ok
ap_ssid:	Name of set hotspot	ap_ssid:Yahboom_ESP32_WIFI	ok
ap_pd:	Password of set hotspot	ap_pd:	ok
sta_ip	Query sta mode ip address	sta_ip	Return own ip address

Command	Description	Example	Return value
ap_ip	Query ap mode ip address	ap_ip	Return own ip address
wifi_reset	Restore wifi configuration to factory settings	wifi_reset	Reset_OK
wifi_ver	Query the version of wifi firmware	wifi_ver	YAHBOOM VerSion:***

Code analysis

Here we only briefly introduce the code content. For detailed code, please refer to the corresponding code file, which is provided in the download area!

- Define the WIFI camera network mode and network parameters

```
// 定义WiFi摄像头网络模式 Define the WiFi camera network mode
#define AP_MODE 0
#define STA_MODE 1
#define STA_AP_MODE 2

// 配置网络模式参数 Set network mode parameters
const char* WIFI_Name = "Yahboom2"; // 设置需要连接的WiFi名称 Set the name of the
WiFi to be connected
const char* WIFI_Password = "yahboom890729";  // 设置需要连接WiFi的密码 Set the
password that you need to connect to WiFi
const char* AP_Name = "Yahboom_ESP32_WIFI"; // 设置自发热点的名称 Set the name of
the spontaneous hotspot
const char* AP_Password = ""; // 设置自发热点的密码 Set the password of the
spontaneous hotspot
const int Mode = STA_AP_MODE; // 设置网络模式 Set network mode
```

- Get WiFi camera version

```
/**
 * @brief 获取wiFi摄像头版本 Get WiFi camera version
 * @param 无 None
 * @retval 无 None
 */
void getVersion(void) {
  Serial.print("wifi_ver");
  delay(500);
}
```

- Restore factory settings

```
/**
 * @brief 恢复出厂设置 Restore factory settings
 * @param 无 None
 * @retval 无 None
 */
void setReset(void) {
  Serial.print("wifi_reset");
  delay(2000);
}
```

- Set the network mode

```
/**
 * @brief 设置网络模式 Set the network mode
 * @param Mode: 网络模式 The network mode
 * @retval 无 None
 */
void setMODE(const int Mode) {
  Serial.print("wifi_mode:" + String(Mode));
  delay(1000);
}
```

- Set WiFi connection information

```
/**
 * @brief 设置wiFi连接信 Set WiFi connection information
 * @param WIFI_Name: WiFi名称 WIFI Name
 * @param WIFI_Password: WiFi密码 WIFI Password
 * @retval 无 None
 */
void SetWIFI(const char* WIFI_Name, const char* WIFI_Password) {
  Serial.print("sta_ssid:" + String(WIFI_Name));
  delay(1000);
  Serial.print("sta_pd:" + String(WIFI_Password));
  delay(5000);
}
```

- Set AP connection information

```
/**
 * @brief 设置自发热点信息 Set AP connection information
 * @param AP_Name: 热点名称 AP Name
 * @param AP_Password: 热点密码 AP Password
 * @retval 无 None
 */
void SetAP(const char* AP_Name, const char* AP_Password) {
  delay(500);
  Serial.print("ap_ssid:" + String(AP_Name));
  delay(1000);
  Serial.print("ap_pd:" + String(AP_Password));
  delay(5000);
}
```

- Get STA ip

```
/**
 * @brief 获取STA模式IP Get STA ip
 * @param 无 None
 * @retval 无 None
 */
void getSTAIP(void) {
  Serial.print("sta_ip");
  delay(500);
}
```

- Get AP ip

```
/**
 * @brief 获取AP模式IP Get AP ip
 * @param 无 None
 * @retval 无 None
 */
void getAPIP(void) {
  Serial.print("ap_ip");
  delay(500);
}
```

- Initialization Code

```
void setup() {
  Serial.begin(115200);                // 初始化串口波特率115200 Initialize serial
communication at 115200 bps
  setMODE(STA_AP_MODE);                // 设置网络模式 Set the network mode
  SetWIFI(WIFI_Name, WIFI_Password);   // 设置WiFi连接信息 Set WiFi connection
information
  SetAP(AP_Name, AP_Password);         // 设置自发热点信息 Set hotspot mode
information
  getSTAIP();                          // 获取STA模式IP Get STA mode IP
  getAPIP();                           // 获取AP模式IP Get AP mode IP
  if (Serial.available() > 0) {
    Serial.print(Serial.readString());  // 打印WiFi摄像头返回的信息 Print the
information returned by the WiFi camera
  }
}
```

- Looping code

```
void loop() {
}
```

Experimental results

After compiling the program successfully, upload the code to the Arduino Uno development board (do not install the WiFi camera module when burning the program, install the WiFi camera module after burning successfully).

After the program starts, the serial port will print the return information of whether we have successfully configured the WiFi camera. If the return information is all ok, it means that the setting is successful. You can cut off the power of the entire car, and then reconnect to the Arduino IDE to view the IP information.

Note: To configure the WiFi camera, you need to connect the battery pack and turn on the expansion board switch, otherwise the WiFi camera will restart, that is, the terminal will show the WiFi camera restart information;

Since the WIFI camera and Arduino Uno use the same serial port, please do not use the Serial.print() function to print data at will, otherwise it will cause the WIFI camera module to misidentify the command!

```
The burning program cannot use other programs to occupy the serial port or an external serial communication module (for example: WiFi camera module), otherwise the program cannot be burned or an error message will be prompted!
```