

6.2WiFi 模块联网

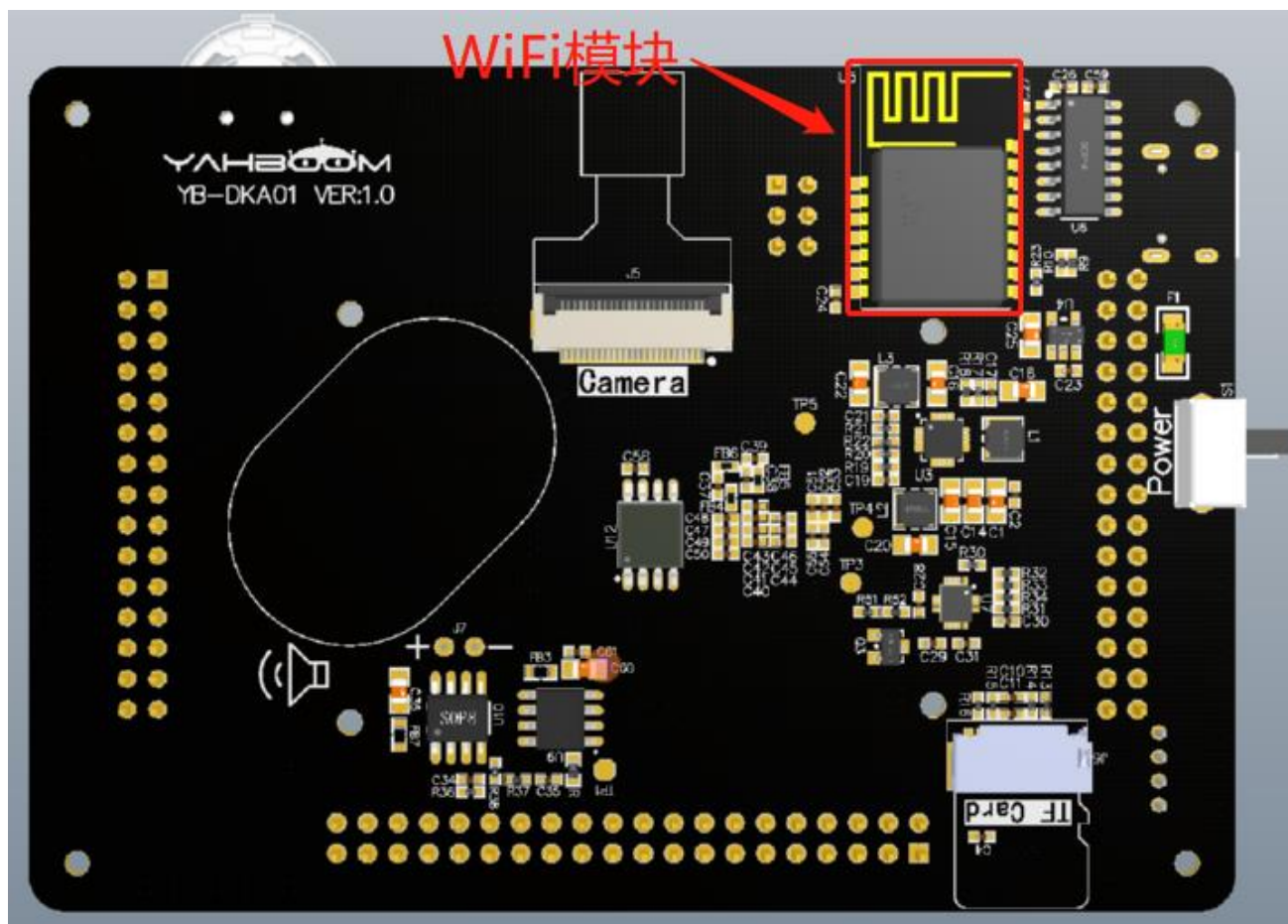
一、实验目的

本节课主要学习 WiFi 模块的连接网络的功能。

二、实验准备

1. 实验元件

WiFi 模块



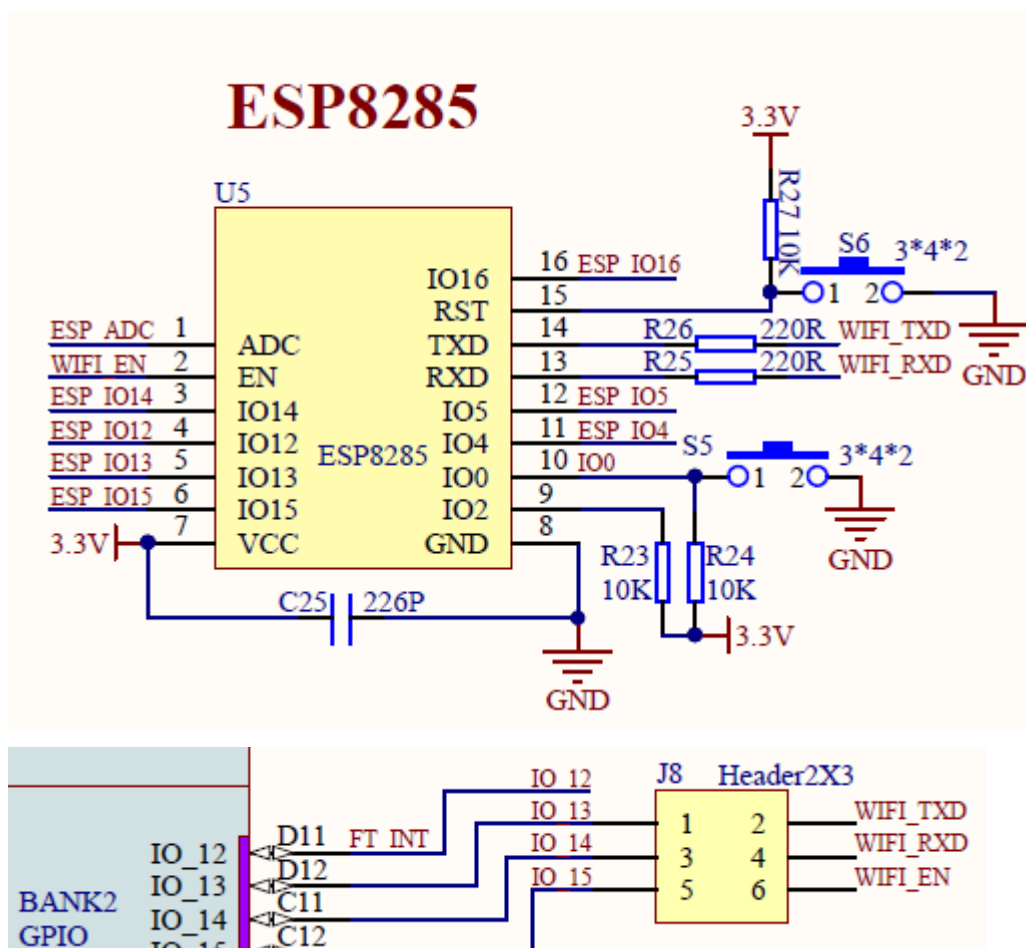
2. 元件特性

K210 开发板自带的 WiFi 模块是 ESP8285 芯片模块，其实 ESP8285 还有更加热门的好兄弟 ESP8266，ESP8285 与 ESP8266 共享同一套 SDK，ESP8285 比 ESP8266

多了一片 1MB FLASH 芯片，并且是 DOUT 模式，所以固件编译和下载时需要设置为 DOUT 模式才可以下载；而且还多了两个 IO 口（GPIO9 和 GPIO10）可以供用户使用。除此之外，ESP8285 还有其他特性，如：体积小、远距离传输、低功耗、可外接天线、耐高温、无线透传、串口转 WiFi 等。

3. 硬件连接

K210 开发板出厂默认已经焊接好 WiFi 模块，WiFi 模块与 K210 之间有三个跳线帽，如果插入跳线帽，则 ESP8285 的 WIFI_TXD 连接 K210 的 IO13，WIFI_RXD 连接 IO14，WIFI_EN 连接 IO15，拔掉跳线帽则不通。



三、实验原理

WiFi 模块内部集成一个 ESP8285 芯片，可以通过 AT 指令的方式连接搜索到

的 WiFi 信号。通过 K210 的串口传输数据的功能，把串口助手接收到的数据传给 WiFi 模块，WiFi 模块根据 AT 指令的内容，查找附近的 WiFi 信号，匹配密码，这样就可以连接到 WiFi 路由器上。

四、实验过程

1. 首先根据上面的硬件连接引脚图，K210 的硬件引脚和软件功能使用的是 FPIOA 映射关系。IO4 和 IO5 是 K210 开发板的 USB 串口引脚，所以也要初始化。USB 串口使用的是串口 3，而 WiFi 模块的串口使用的是串口 1。

```
/******HARDWARE-PIN*****  
// 硬件IO口，与原理图对应  
#define PIN_UART_USB_RX      (4)  
#define PIN_UART_USB_TX      (5)  
  
#define PIN_UART_WIFI_RX     (13)  
#define PIN_UART_WIFI_TX     (14)  
  
/******SOFTWARE-GPIO*****  
// 软件GPIO口，与程序对应  
#define UART_USB_NUM          UART_DEVICE_3  
  
#define UART_WIFI_NUM          UART_DEVICE_1  
  
/******FUNC-GPIO*****  
// GPIO口的功能，绑定到硬件IO口  
#define FUNC_UART_USB_RX      (FUNC_UART1_RX + UART_USB_NUM * 2)  
#define FUNC_UART_USB_TX      (FUNC_UART1_TX + UART_USB_NUM * 2)  
  
#define FUNC_UART_WIFI_RX     (FUNC_UART1_RX + UART_WIFI_NUM * 2)  
#define FUNC_UART_WIFI_TX     (FUNC_UART1_TX + UART_WIFI_NUM * 2)
```

```
void hardware_init(void)
{
    /* USB串口 */
    fpioa_set_function(PIN_UART_USB_RX, FUNC_UART_USB_RX);
    fpioa_set_function(PIN_UART_USB_TX, FUNC_UART_USB_TX);

    /* WIFI模块串口 */
    fpioa_set_function(PIN_UART_WIFI_RX, FUNC_UART_WIFI_RX);
    fpioa_set_function(PIN_UART_WIFI_TX, FUNC_UART_WIFI_TX);
}
```

2. 初始化串口的配置，波特率设置为 115200，串口数据宽度为 8 位，停止位 1 位，不使用奇偶校验。

```
// 初始化USB串口，设置波特率为115200
uart_init(UART_USB_NUM);
uart_configure(UART_USB_NUM, 115200, UART_BITWIDTH_8BIT, UART_STOP_1, UART_PARITY_NONE);

/* 初始化WiFi模块的串口 */
uart_init(UART_WIFI_NUM);
uart_configure(UART_WIFI_NUM, 115200, UART_BITWIDTH_8BIT, UART_STOP_1, UART_PARITY_NONE);
```

3. 开机的时候发送 “hello yahboom!”，提示已经开机完成。

```
/* 开机发送hello yahboom! */
char *hello = {"hello yahboom!\n"};
uart_send_data(UART_USB_NUM, hello, strlen(hello));
```

4. 最后循环接收串口的数据，如果 WiFi 模块有数据传到 K210 芯片，K210 会把数据通过串口传输到电脑上显示；如果电脑上的串口助手发送数据到 K210 芯片，K210 也会把数据经过 WiFi 模块的串口发送给 WiFi 模块。

```
char recv = 0, send = 0;

while (1)
{
    /* 接收WIFI模块的信息 */
    if(uart_receive_data(UART_WIFI_NUM, &recv, 1))
    {
        /* 发送WiFi的数据到USB串口显示 */
        uart_send_data(UART_USB_NUM, &recv, 1);
    }

    /* 接收串口的信息，并发送给WiFi模块 */
    if(uart_receive_data(UART_USB_NUM, &send, 1))
    {
        uart_send_data(UART_WIFI_NUM, &send, 1);
    }
}
return 0;
```

5. 编译调试，烧录运行

把本课程资料中的 wifi_AT 复制到 SDK 中的 src 目录下，然后进入 build 目录，运行以下命令编译。

```
cmake .. -DPROJ=wifi_AT -G "MinGW Makefiles"
```

```
make
```

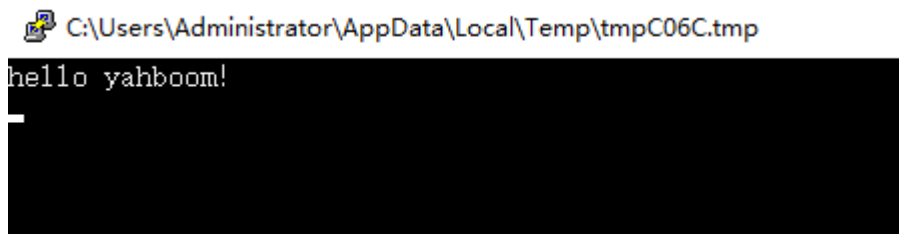
```
Scanning dependencies of target wifi_AT
[ 97%] Building C object CMakeFiles/wifi_AT.dir/src/wifi_AT/main.c.obj
[100%] Linking C executable wifi_AT
Generating .bin file ...
[100%] Built target wifi_AT
PS C:\K210\SDK\kendryte-standalone-sdk-develop\build> []
```

编译完成后，在 build 文件夹下会生成 wifi_AT.bin 文件。

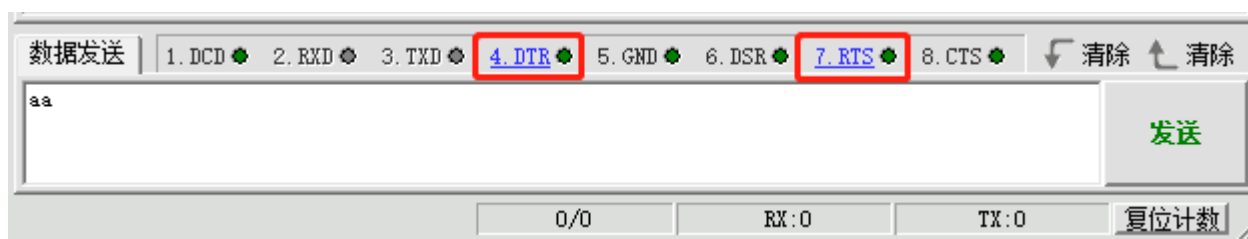
使用 type-C 数据线连接电脑与 K210 开发板，打开 kflash，选择对应的设备，再将程序固件烧录到 K210 开发板上。

五、实验现象

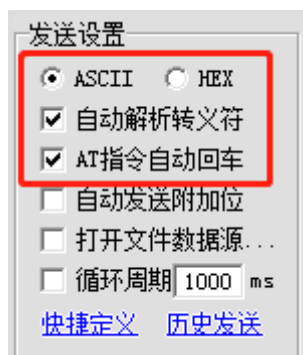
1. 烧录完成固件后，系统会弹出一个终端界面，如果没有弹出终端界面的可以打开串口助手显示调试内容。

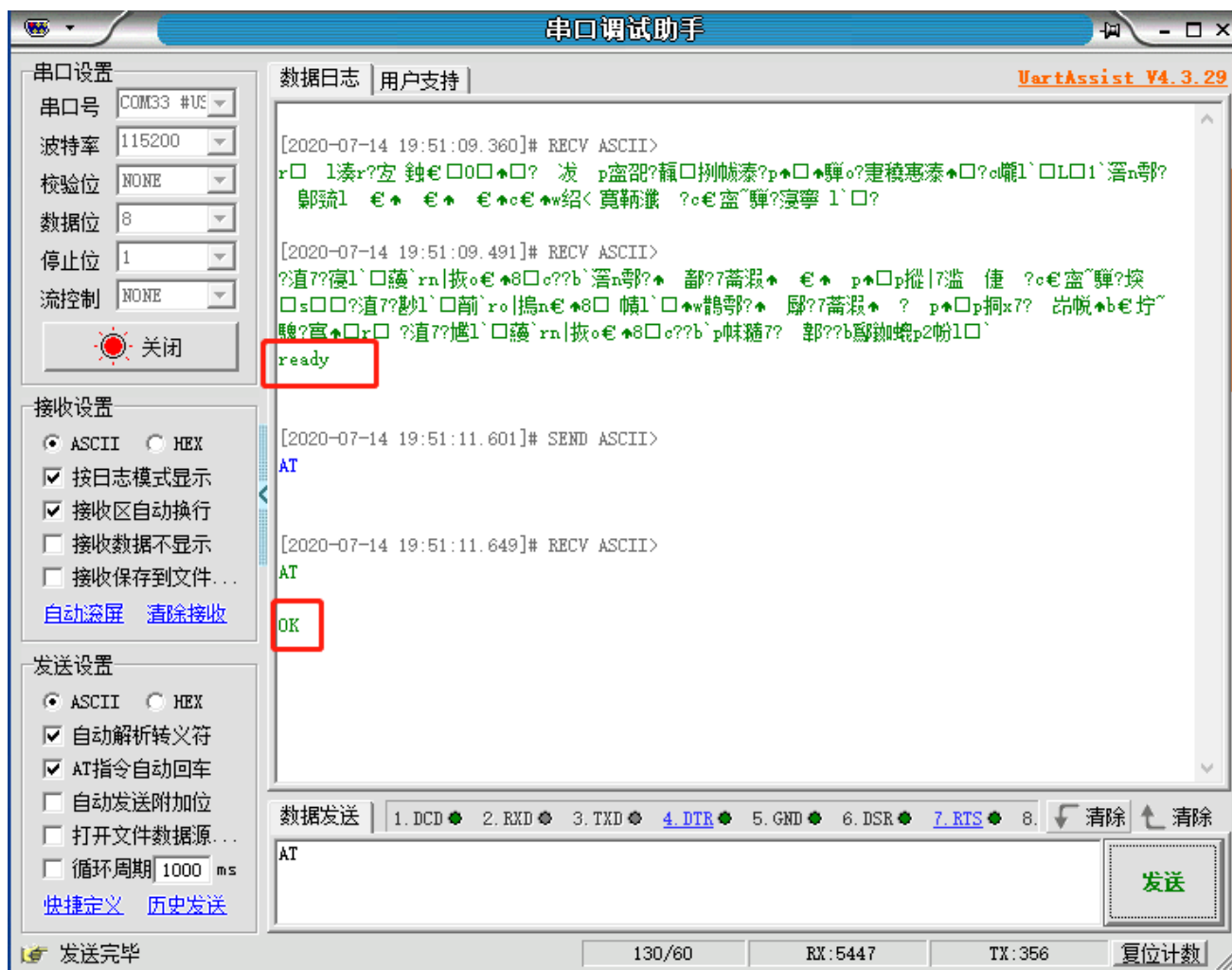


2. 打开电脑的串口助手，选择对应的 K210 开发板对应的串口号，波特率设置为 115200，然后点击打开串口助手。注意还需要设置一下串口助手的 DTR 和 RTS。在串口助手底部此时的 4. DTR 和 7. RTS 默认是红色的，点击 4. DTR 和 7. RTS，都设置为绿色，然后按一下 K210 开发板的复位键。



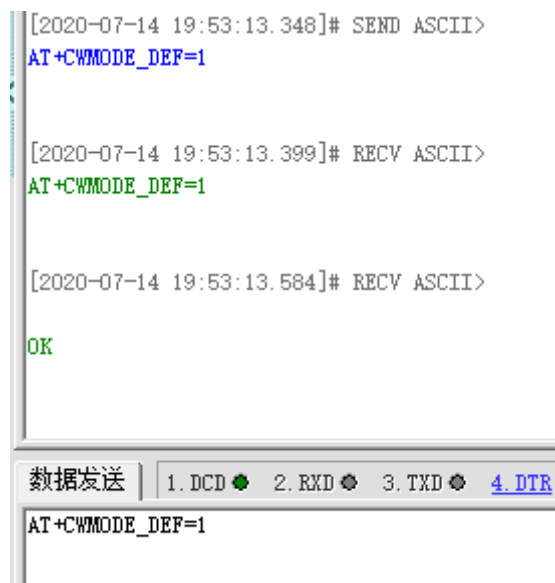
3. 从串口助手，可以接收到 hello yahboom! 的欢迎语。然后按一下 WiFi 模块的复位键，可以看到一大串乱码，这个不用管，只要看到 ready 字符则表示 WiFi 模块正常。然后在底部输入 AT 字符，然后点击发送，串口助手会显示 OK。如果没有接收到 OK，请确认发送设置中的参数为 ASCII，自动解析转义符，AT 指令自动回车。





4. 发送 AT 指令设置 WiFi 模块的模式为工作站。

AT+CWMODE_DEF=1



5. 发送 AT 指令设置连接的 WiFi 信号。

AT+CWJAP_DEF="WiFi 名称","密码"。

这里以 WiFi 名称为 Raspblock, 密码为 12345678 为例。WiFi 信号不可以有中文。接收到 WIFI CONNECTED 和 WIFI GOT IP 则表示连接成功。

AT+CWJAP_DEF="Raspblock","12345678"



6. 获取当前的 WiFi 模块 IP 地址

AT+CIFSR


```
[2020-07-14 20:10:39.473]# SEND ASCII>
AT+CIFSR

[2020-07-14 20:10:39.538]# RECV ASCII>
AT+CIFSR
+CIFSR:STAIP,'192.168.3.100'
+CIFSR:STAMAC,"84:0d:8e:72:a2:7a"

OK
```

数据发送 | 1. DCD ● 2. RXD ● 3. TXD ● 4. DTR ●

AT+CIFSR

7. ping 一下网络试试通不通，有数值则表示网络正常。

```
[2020-07-14 20:21:31.198]# SEND ASCII>
AT+PING="www.baidu.com"

[2020-07-14 20:21:31.285]# RECV ASCII>
AT+PING="www.baidu.com"
+19

[2020-07-14 20:21:32.229]# RECV ASCII>
OK
```

数据发送 | 1. DCD ● 2. RXD ● 3. TXD ● 4. DTR ●

AT+PING="www.baidu.com"

六、实验总结

1. WiFi 模块与 K210 通过串口连接的方式通讯。
2. K210 在此过程只起到中间搬运数据的功能，不会处理数据。
3. 其实 WiFi 模块如果连接其他串口芯片，也是可以通过 AT 指令操作的。
4. 以下连接的 WiFi 信号，下次 WiFi 模块重启后，开机会自动连接这个 WiFi 信号。