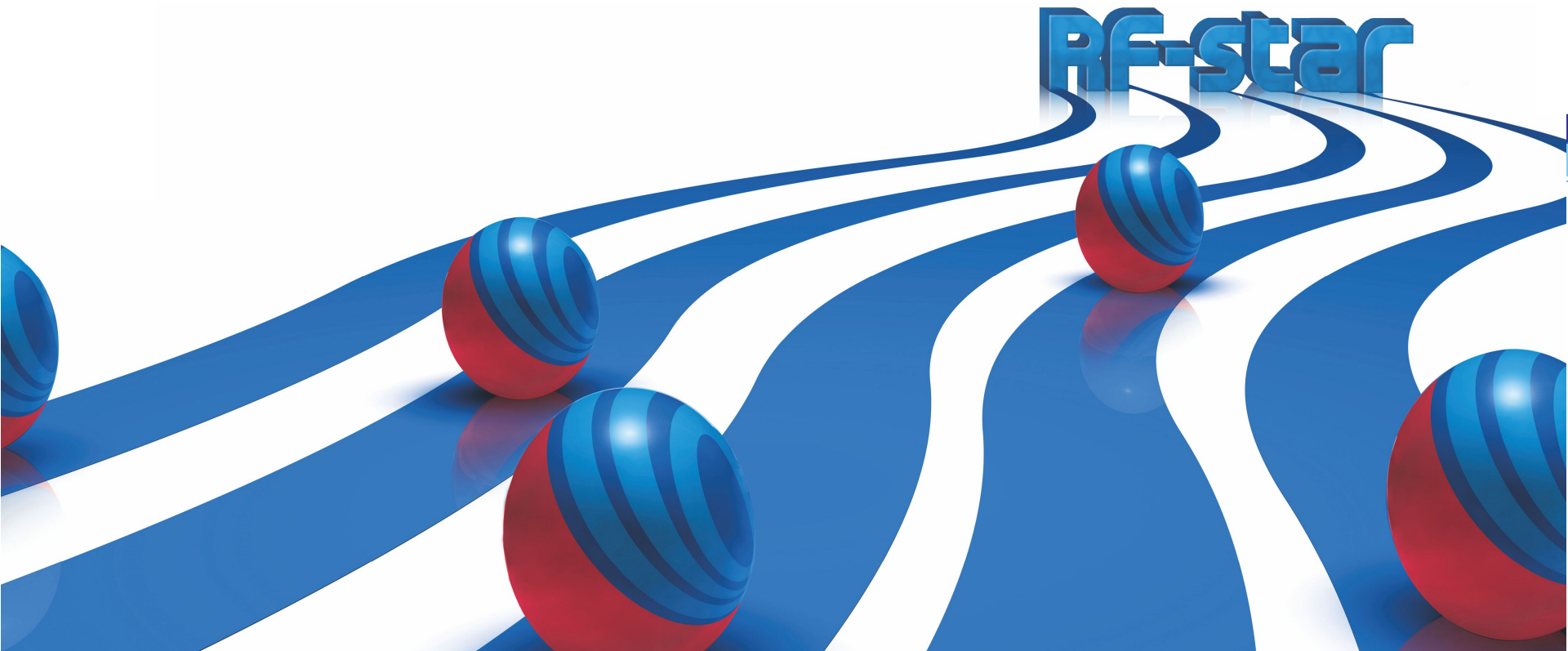
RF-star
信 驰 达

一步一步学习CC3200 与MQTT服务器通讯



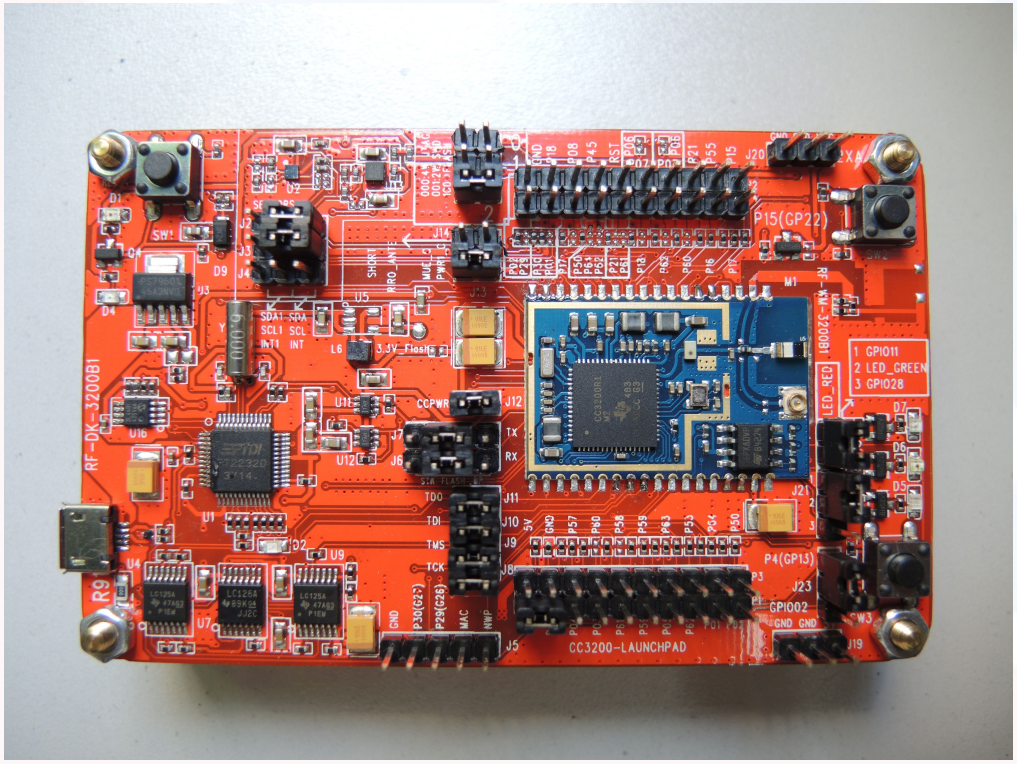
❖ 什么是MQTT？

1、MQTT（Message Queuing Telemetry Transport，消息队列遥测传输）是IBM开发的一个即时通讯协议，有可能成为物联网的重要组成部分。该协议支持所有平台，几乎可以把所有联网物品和外部连接起来，被用来当做传感器和致动器（比如通过Twitter让房屋联网）的通信协议。MQTT具有以下特点：

- 轻量级的 machine-to-machine 通信协议。
- publish/subscribe模式。
- 基于TCP/IP。
- 支持QoS。
- 适合于低带宽、不可靠连接、嵌入式设备、CPU内存资源紧张。
- 是一种比较不错的Android消息推送方案。
- FacebookMessenger采用了MQTT。
- MQTT有可能成为物联网的重要协议。

❖ 预先装备

2、 TI已经帮我们把MQTT的协议移植到了CC3200平台，请下载CC3200最新的SDK包，本篇以CC3200SDK_1.1.0版本、CCS6.0.1、cloudmqtt服务器、使用信驰达RF-DK-3200B1开发板讲解。



❖ 操作流程

3、使用CCS导入MQTT工程，路径为：C:\ti\CC3200SDK_1.1.0\cc3200-sdk\example\mqtt_client；如图1。

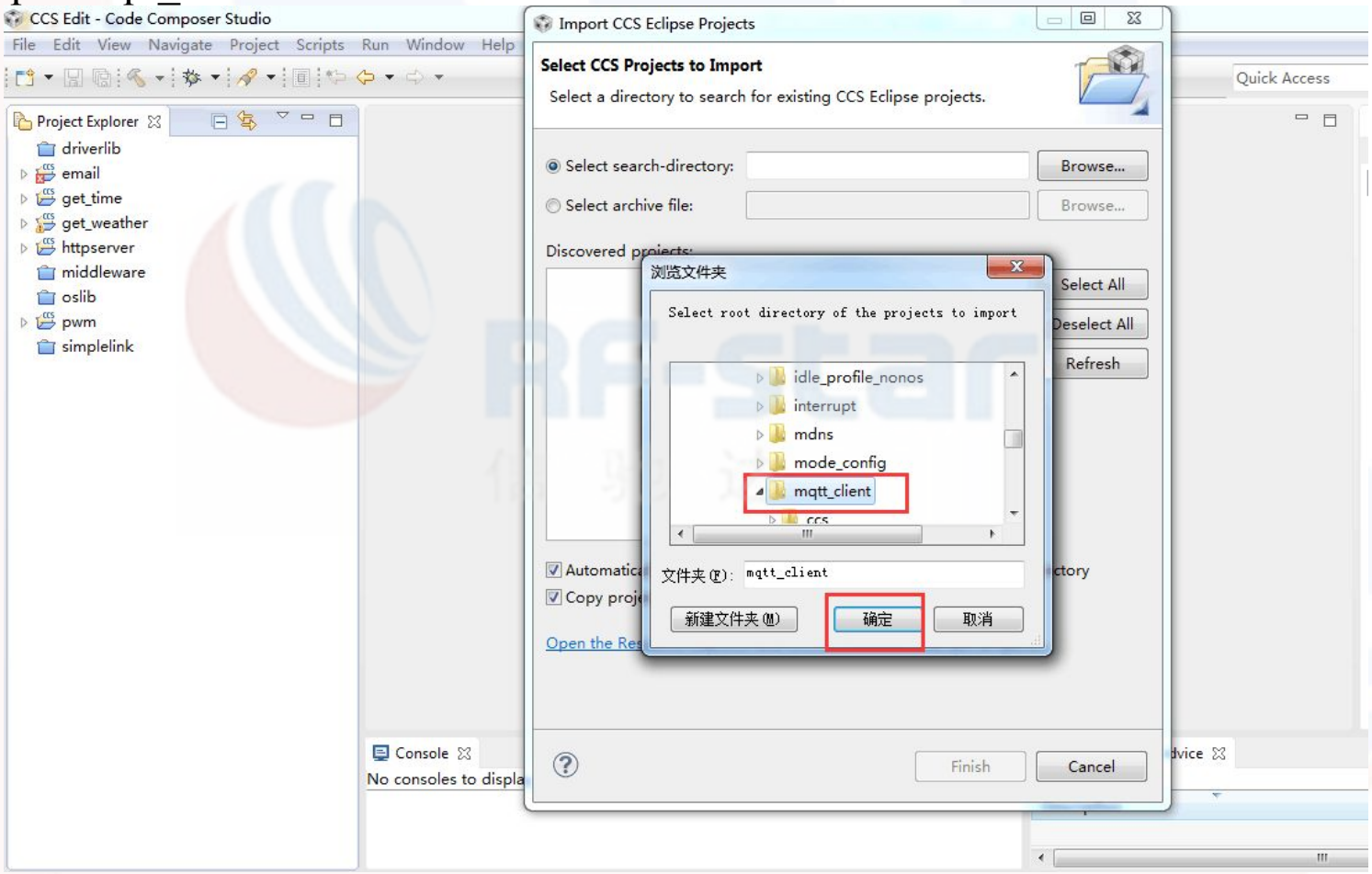


图 1

点击Finish完成导入，如图2。

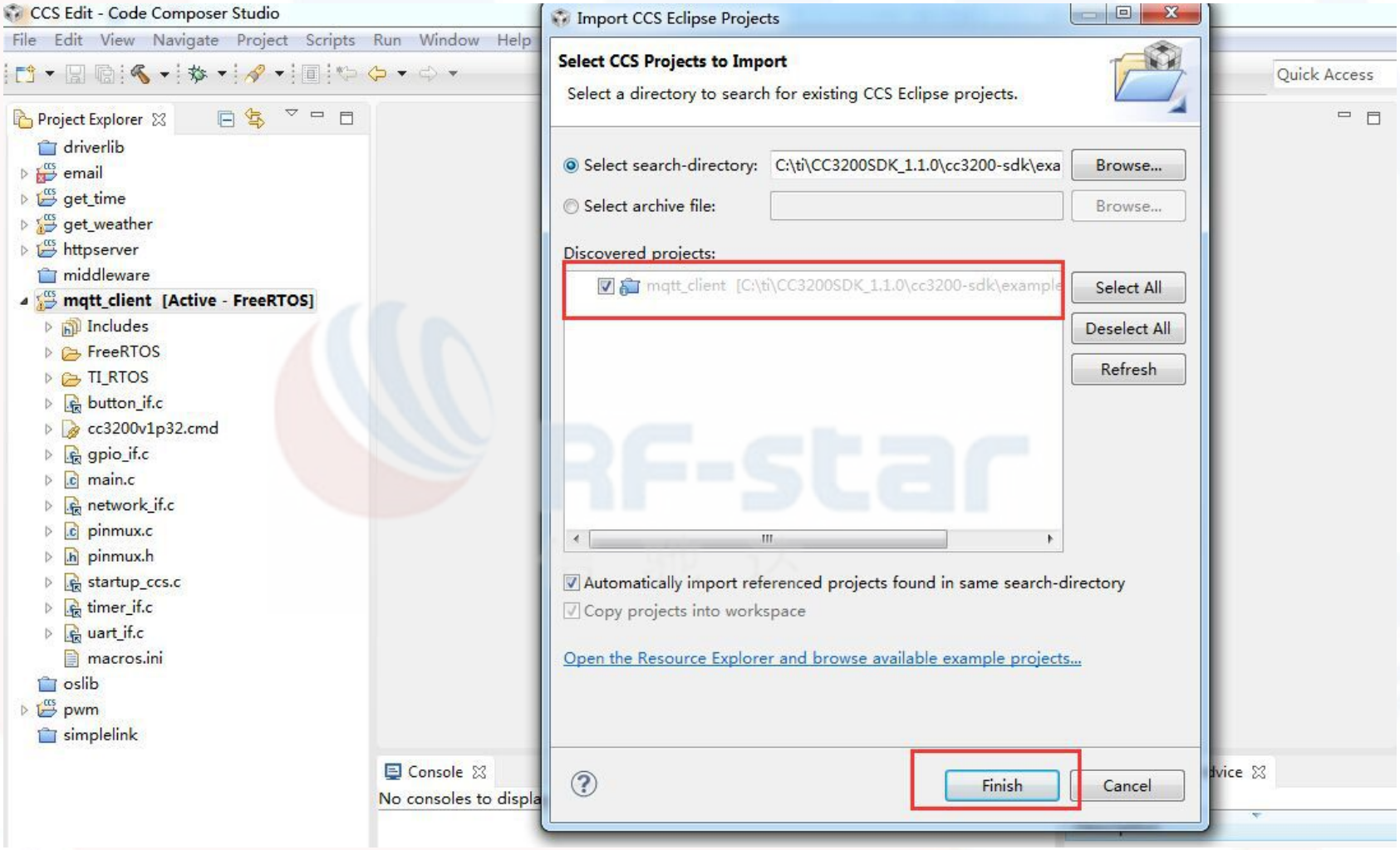


图 2

导入后如图3。

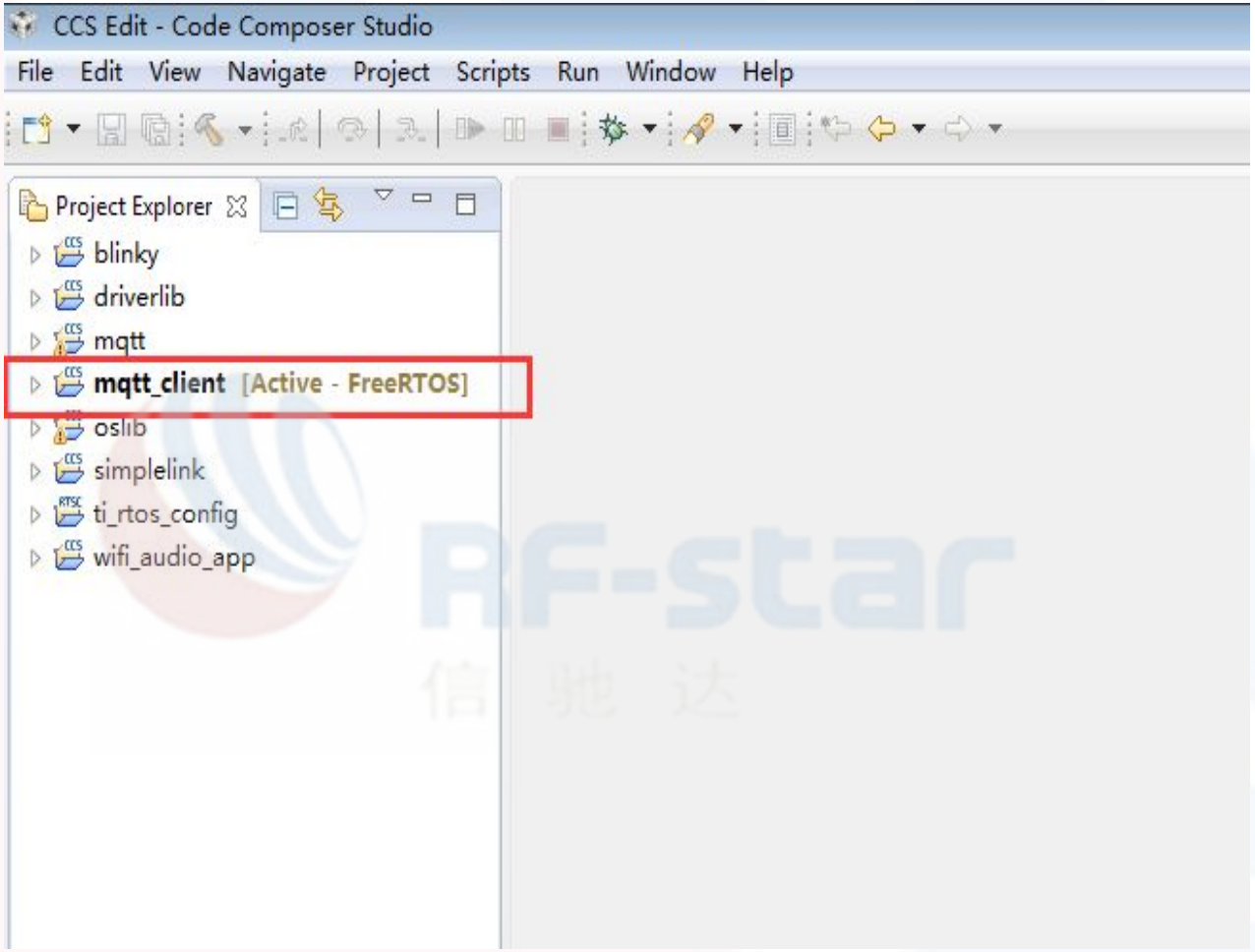

CCS Edit - Code Composer Studio
File Edit View Navigate Project Scripts Run Window Help
Project Explorer
blinky
driverlib
mqtt
mqtt_client [Active - FreeRTOS]
oslib
simplelink
ti_rtos_config
wifi_audio_app

图 3

4．申请MQTT服务器账号与密码，在浏览器输入：http://www.cloudmqtt.com，点击Control Panel，如图4。

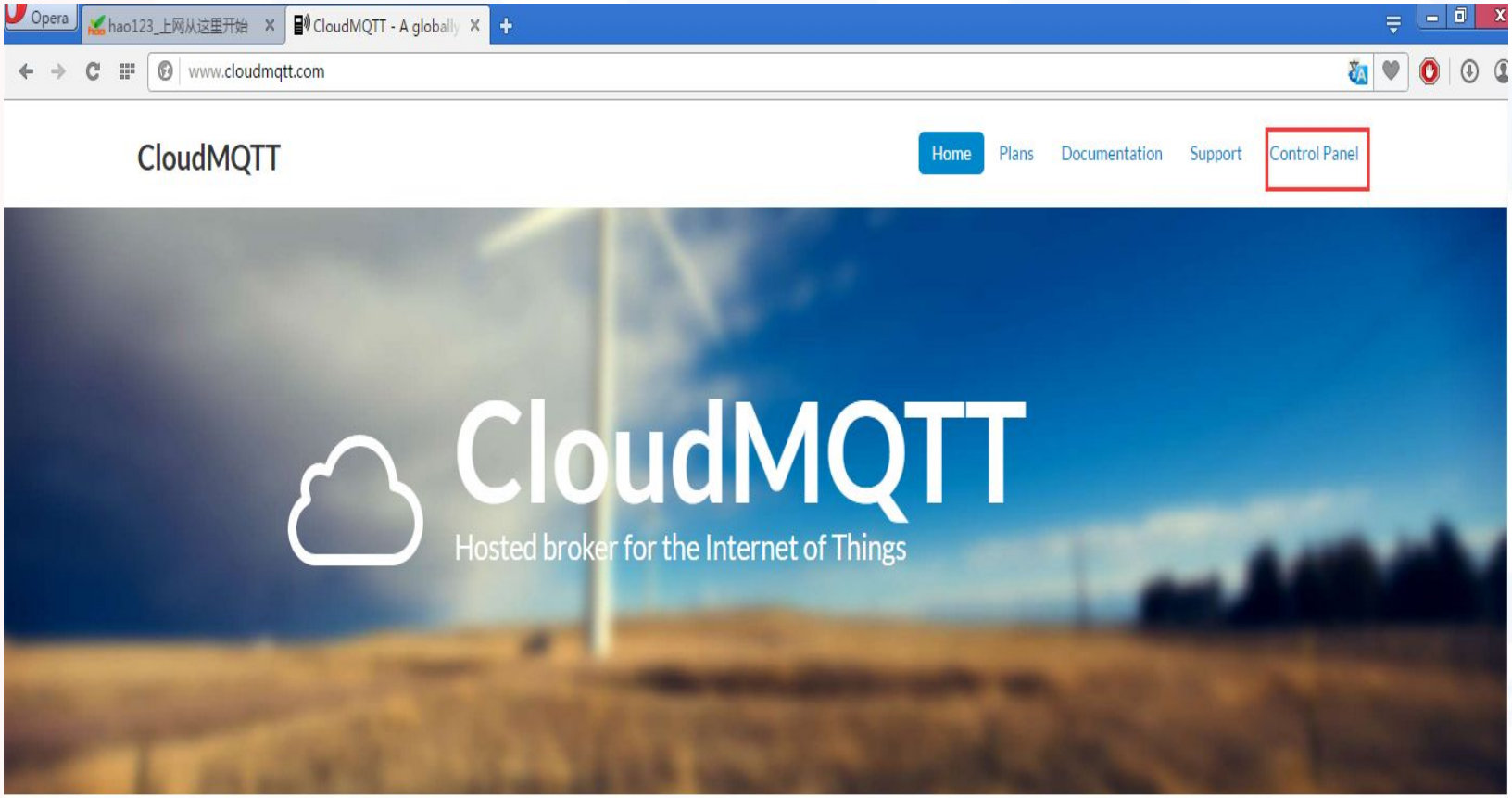


图 4

输入你的邮箱并点Sign up，我这里使用了qq邮箱演示，如图5。

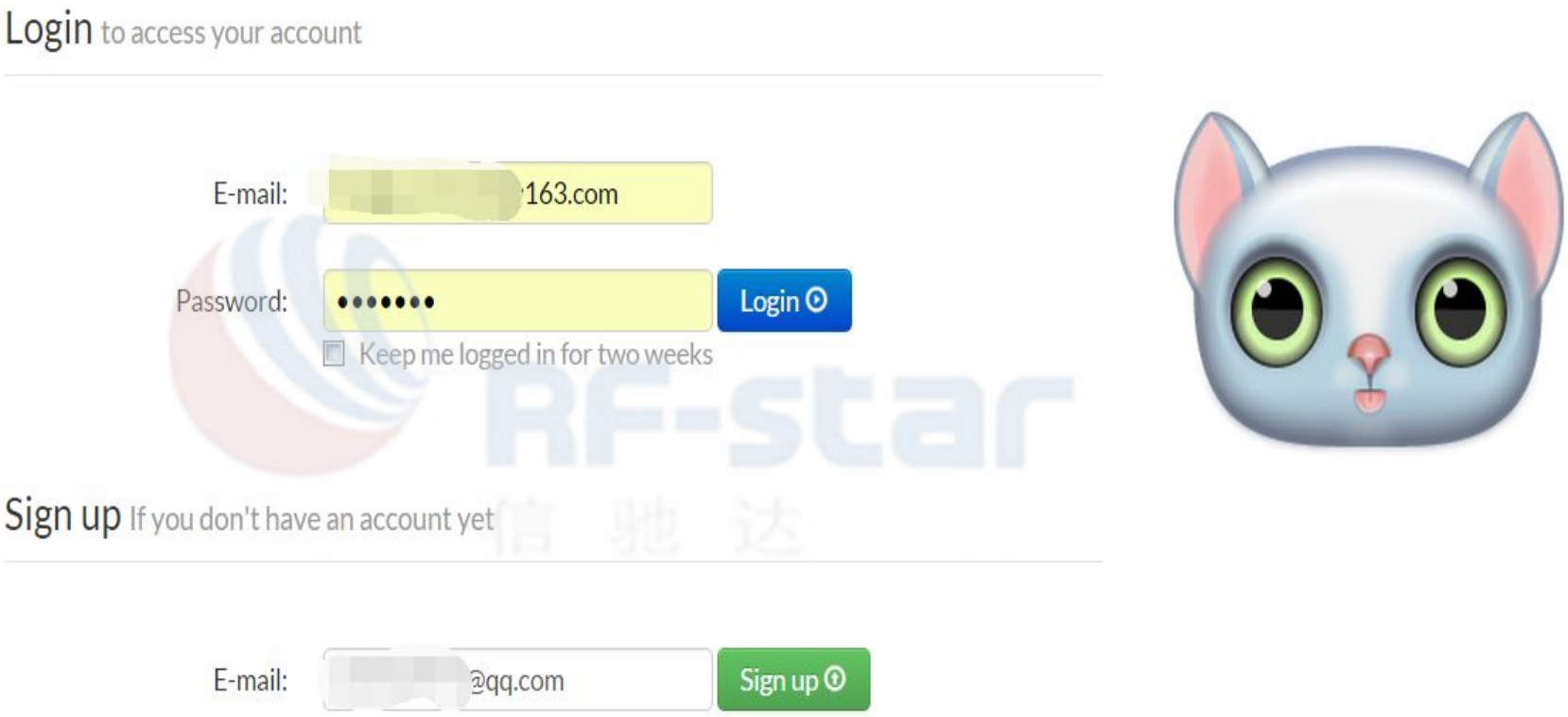


图 5

很快你将收到一封由cloudmqtt网站发过来的邮件，点击里面的链接，打开后如图6。

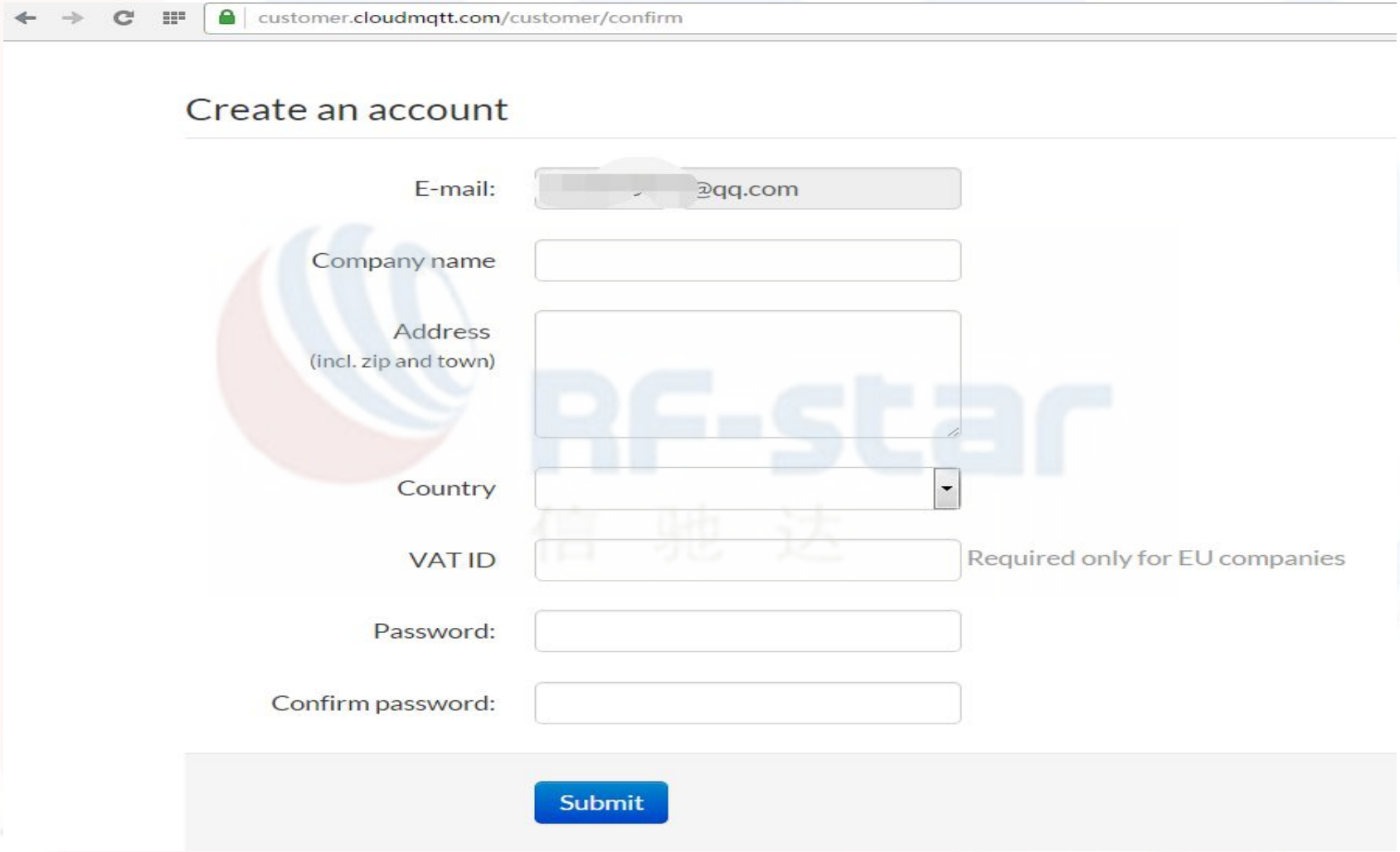


图 6

并填写资料，随便填一些资料，注意记住密码，提交后，再点击Create，如图7。

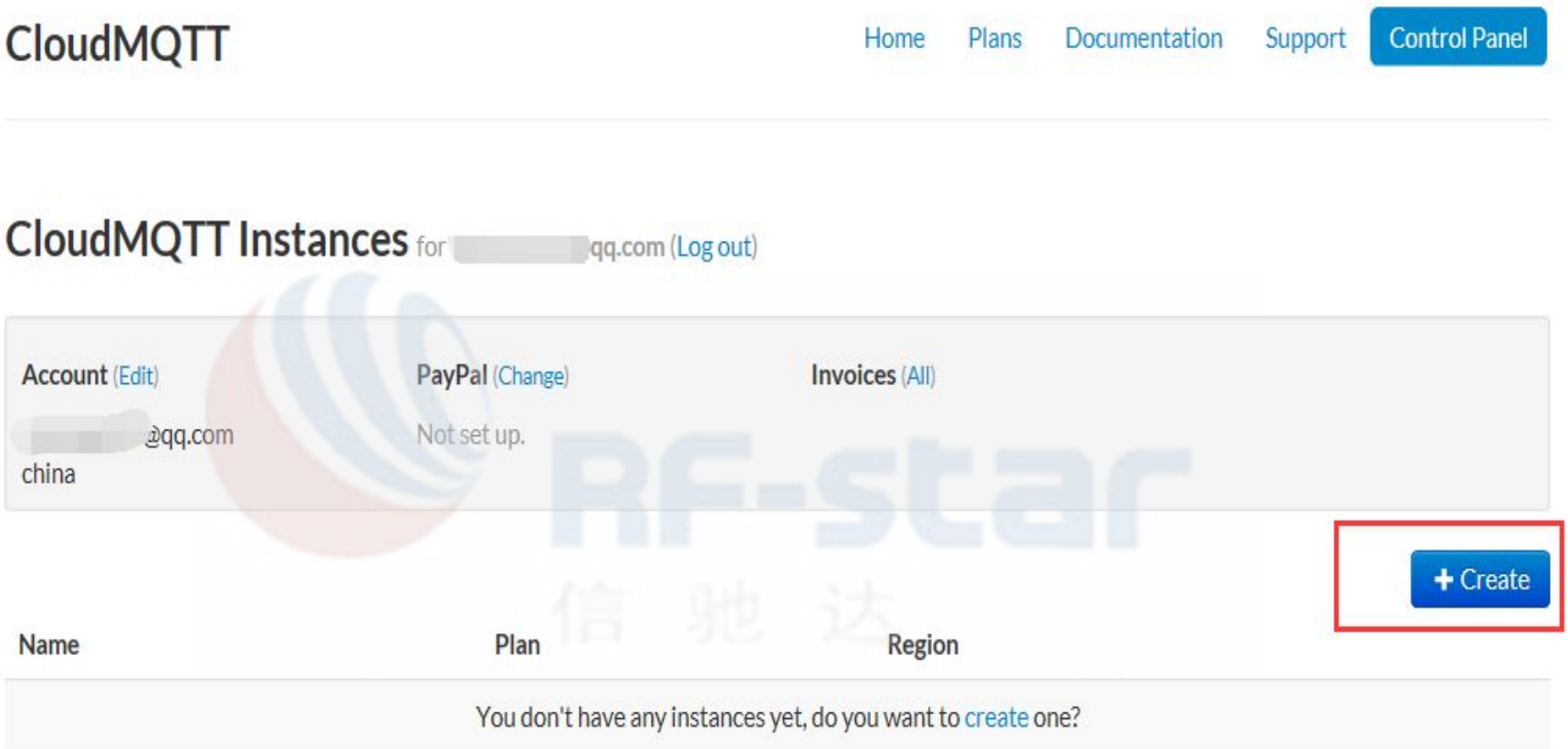


图 7

Name也是随便填，这里填tester1,再点击Create，如图8。

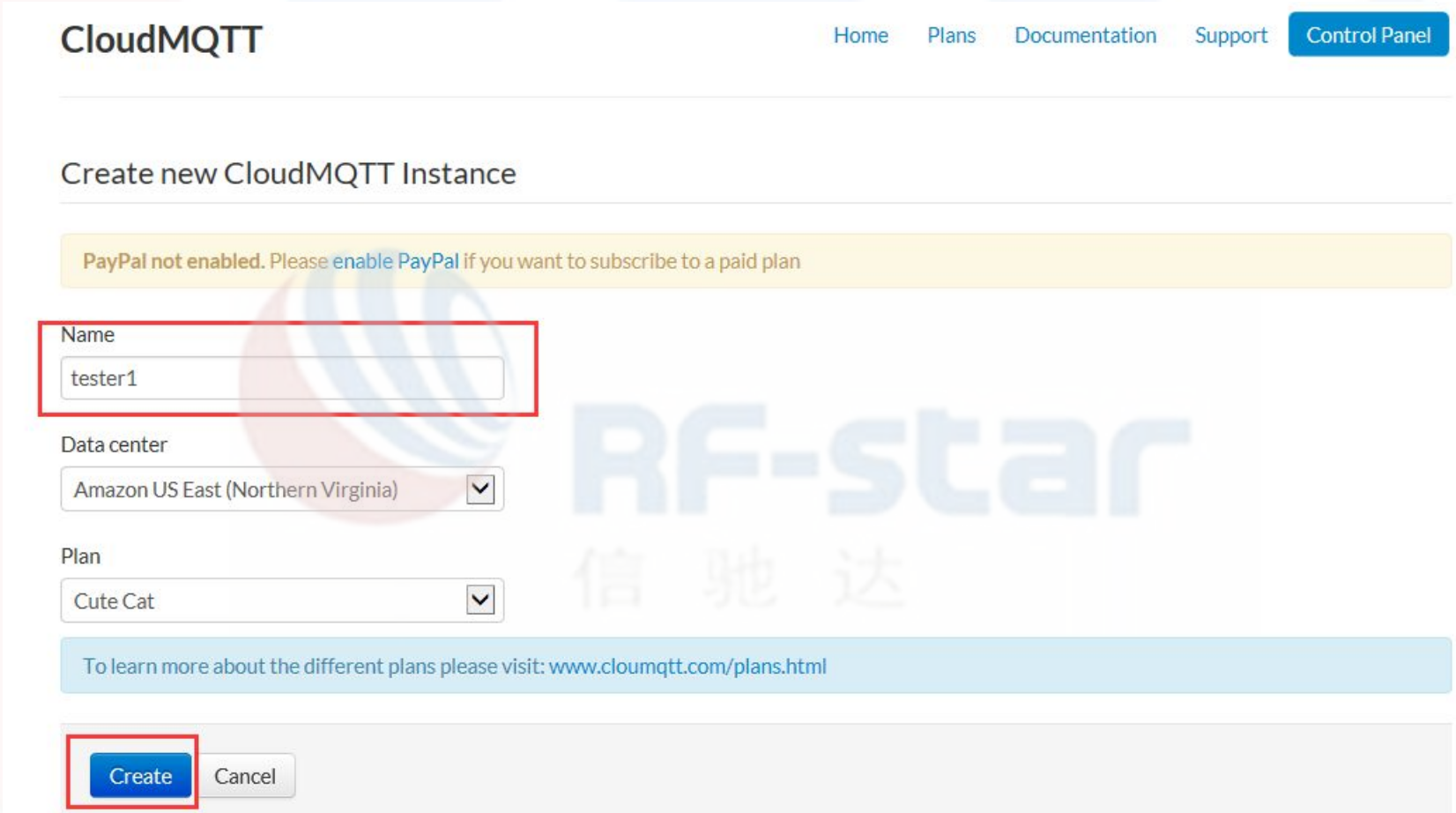


图 8

点击tester1的Details，如图9。

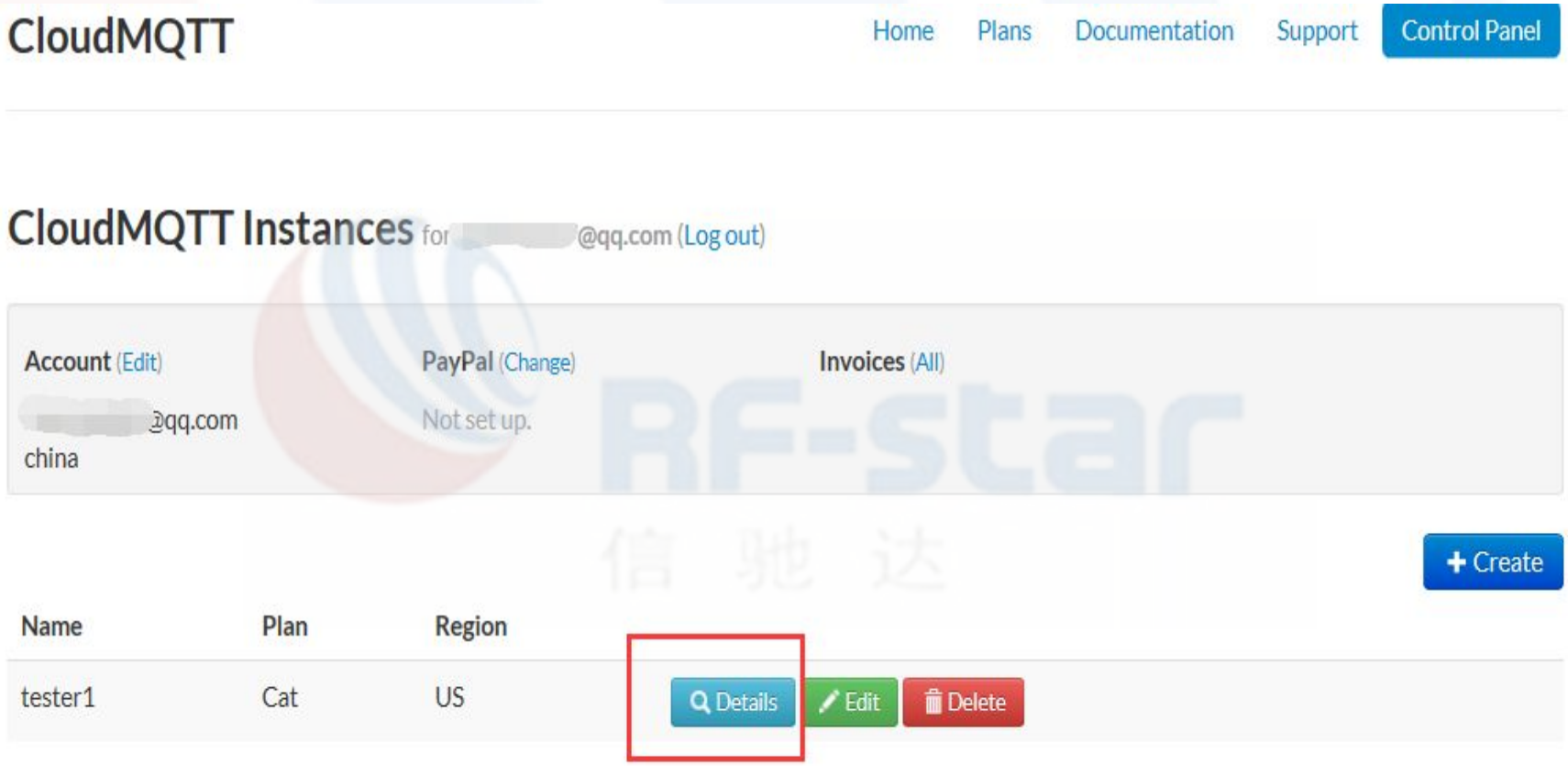


图 9

注意这几个参数，等会在程序里需要填写，如图10。

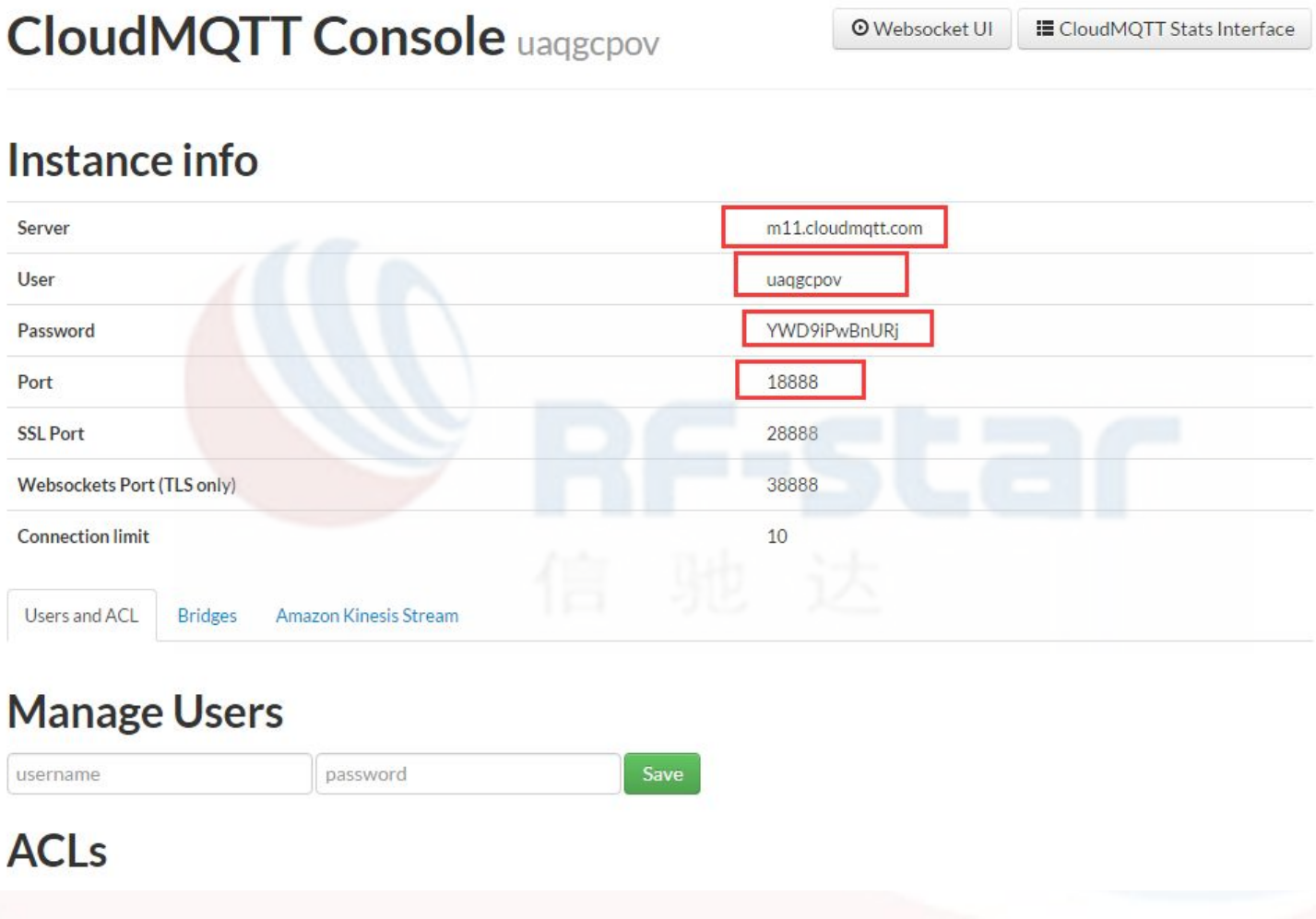


图 10

5．切回到CCS6中的程序，在mqtt_client中的main文件中，修改SERVER_ADDRESS、PORT_NUMBER为MQTT服务器tester1中，Instance info中的Server、Port，如图11。

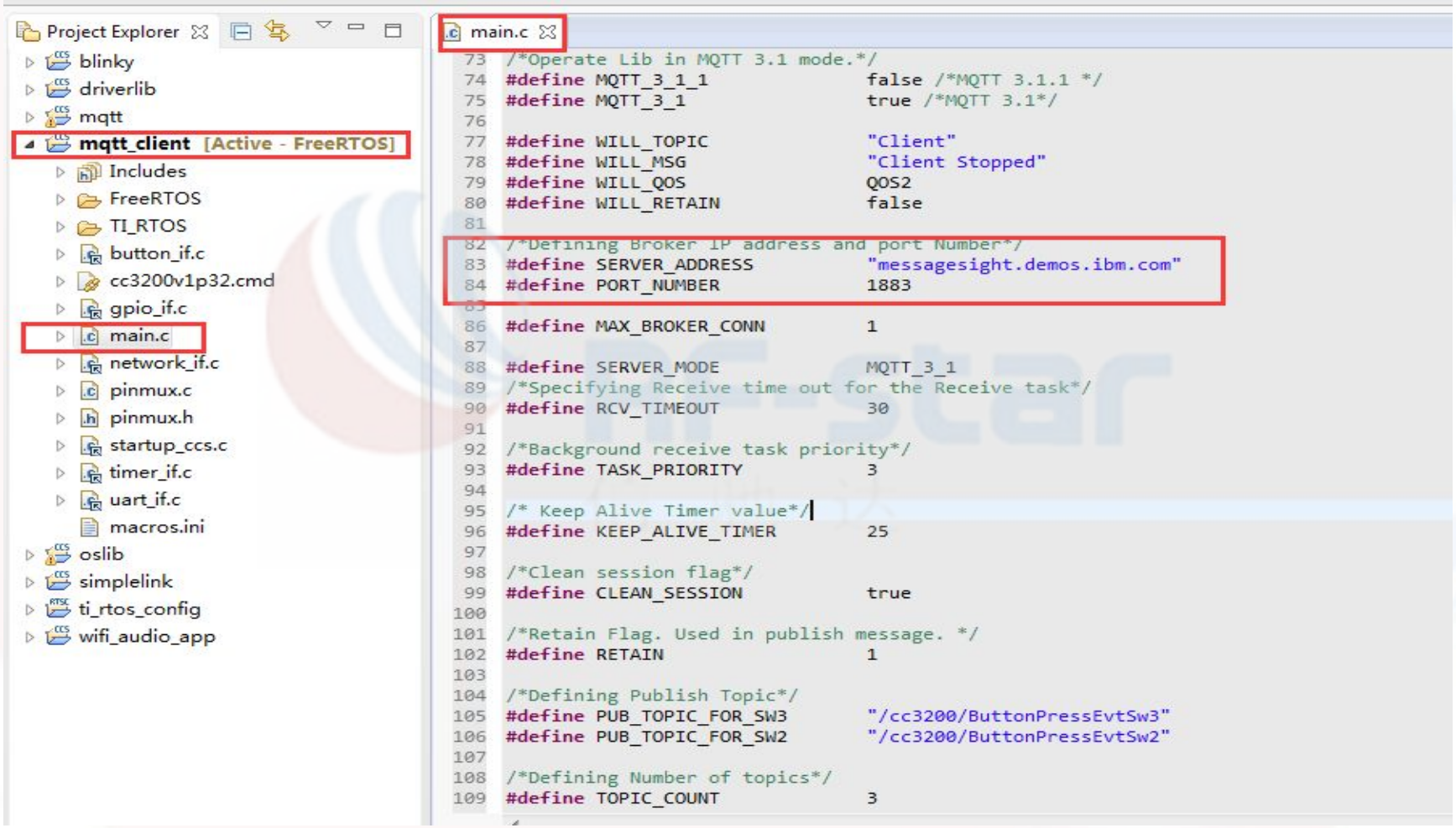


图 11

修改后效果如图12。

```
82  /*Defining Broker IP address and port Number*/
83  //#define SERVER_ADDRESS          "messagesight.demos.ibm.com"
84  #define SERVER_ADDRESS          "m11.cloudmqtt.com"
85  //#define PORT_NUMBER             1883
86  #define PORT_NUMBER             18888
87
88  #define MAX_BROKER_CONN         1
89
90  #define SERVER_MODE             MQTT_3_1
91  /*Specifying Receive time out for the Receive task*/
92  #define RCV_TIMEOUT             30
93
94  /*Background receive task priority*/
95  #define TASK_PRIORITY           3
96
97  /* Keep Alive Timer value*/
98  #define KEEP_ALIVE_TIMER        25
99
100 /*Clean session flag*/
101 #define CLEAN_SESSION           true
102
```

图 12

还是在main文件中，此两处需要修改，替换为MQTT服务器tester1中，Instance info中的User、Password，如图13。

main.c

```
182  SlSecParams_t SecurityParams = {0};
183
184  /*Message Queue*/
185  OsiMsgQ_t g_PBQueue;
186
187  /* connection configuration */
188  connect_config usr_connect_config[] =
189  {
190      {
191          {
192              {
193                  SL_MQTT_NETCONN_URL,
194                  SERVER_ADDRESS,
195                  PORT_NUMBER,
196                  0,
197                  0,
198                  0,
199                  NULL
200              },
201              SERVER_MODE,
202              true,
203          },
204          NULL,
205          "user1",
206          NULL,
207          NULL,
208          true,
209          KEEP_ALIVE_TIMER,
210          {Mqtt_Recv, sl_MqttEvt, sl_MqttDisconnect},
211          TOPIC_COUNT,
212          {TOPIC1, TOPIC2, TOPIC3},
213          {QOS2, QOS2, QOS2},
214          {WILL_TOPIC,WILL_MSG,WILL_QOS,WILL_RETAIN},
215          false
216      }
217  };
```

图 13

修改后效果如图14。

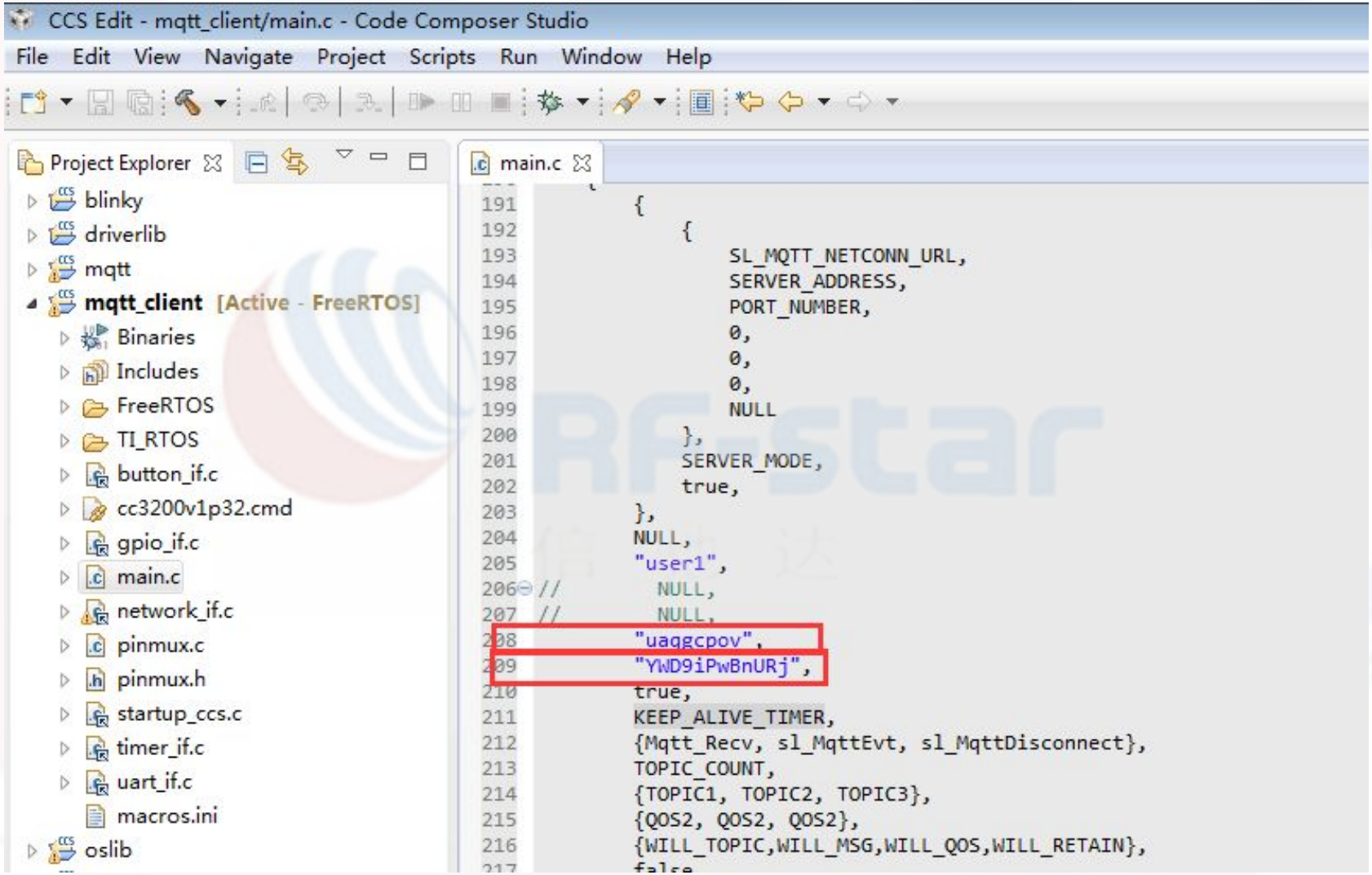


图 14

修改main文件中，打开common.h，需要修改里面的Wi-Fi热点的参数，如图15。

main.c

```
2  // Copyright (C) 2014 Texas Instruments Incorpo
13
16 // Application Name     -   MQTT Client
29
32 //! \addtogroup mqtt_client
36
37 // Standard includes
38 #include <stdlib.h>
39
40 // simplelink includes
41 #include "simplelink.h"
42
43 // driverlib includes
44 #include "hw_types.h"
45 #include "hw_ints.h"
46 #include "hw_memmap.h"
47 #include "interrupt.h"
48 #include "rom_map.h"
49 #include "prcm.h"
50 #include "uart.h"
51 #include "timer.h"
52
53 // common interface includes
54 #include "network_if.h"
55 #ifndef NOTERM
56 #include "uart_if.h"
57 #endif
58
59 #include "button_if.h"
60 #include "gpio_if.h"
61 #include "timer_if.h"
62 #include "common.h"
63 #include "utils.h"
64
65
66 #include "sl_mqtt_client.h"
67
```

图 15

修改三个参数，分别是SSID名，加密类型和密码，如图16。

main.c | common.h | network_if.h

```
51 extern "C"
52 {
53 #endif
54
55
56 //
57 // Values for below macros shall be modified as per access-point(AP) properties
58 // SimpleLink device will connect to following AP when application is executed
59 //
60 #define SSID_NAME           "Tenda_AP"     /* AP SSID */// SoftAP
61 #define SECURITY_TYPE       SL_SEC_TYPE_WPA/* Security type (OPEN or WEP or WPA*/
62 #define SECURITY_KEY        "12345678"              /* Password of the secured AP */
63 #define SSID_LEN_MAX        32
64 #define BSSID_LEN_MAX       6
65
66
67 #ifdef NOTERM
68 #define UART_PRINT(x,...)
69 #define DBG_PRINT(x,...)
70 #define ERR_PRINT(x)
71 #else
72 #define UART_PRINT Report
73 #define DBG_PRINT  Report
74 #define ERR_PRINT(x) Report("Error [%d] at line [%d] in function [%s]  \n\r",x,__LINE__,__FUNCTION__)
75 #endif
```

图 16

例程中，发布了两个按键的消息和订阅了三个LED的主题，如图17。

common.h | network_if.h | main.c

```
/* Keep Alive Timer value*/
#define KEEP_ALIVE_TIMER        25

/*Clean session flag*/
#define CLEAN_SESSION           true

/*Retain Flag. Used in publish message. */
#define RETAIN                  1

/*Defining Publish Topic*/
#define PUB_TOPIC_FOR_SW3       "/cc3200/ButtonPressEvtSw3"
#define PUB_TOPIC_FOR_SW2       "/cc3200/ButtonPressEvtSw2"

/*Defining Number of topics*/
#define TOPIC_COUNT             3

/*Defining Subscription Topic Values*/
#define TOPIC1                  "/cc3200/ToggleLEDCmdL1"
#define TOPIC2                  "/cc3200/ToggleLEDCmdL2"
#define TOPIC3                  "/cc3200/ToggleLEDCmdL3"

/*Defining QOS levels*/
#define QOS0                    0
#define QOS1                    1
#define QOS2                    2

/*Spawn task priority and OSI Stack Size*/
#define OSI_STACK_SIZE          2048
#define UART_PRINT              Report
```

发布的消息

订阅的主题

图 17

把程序保存，编译，并下载CC3200的开发板上。

6．把串口调试工具打开，可以查看程序运行打印出来的信息，如图18。

图 18

切换到浏览器MQTT服务器界面上，点击“Websocket UI”，如图19。

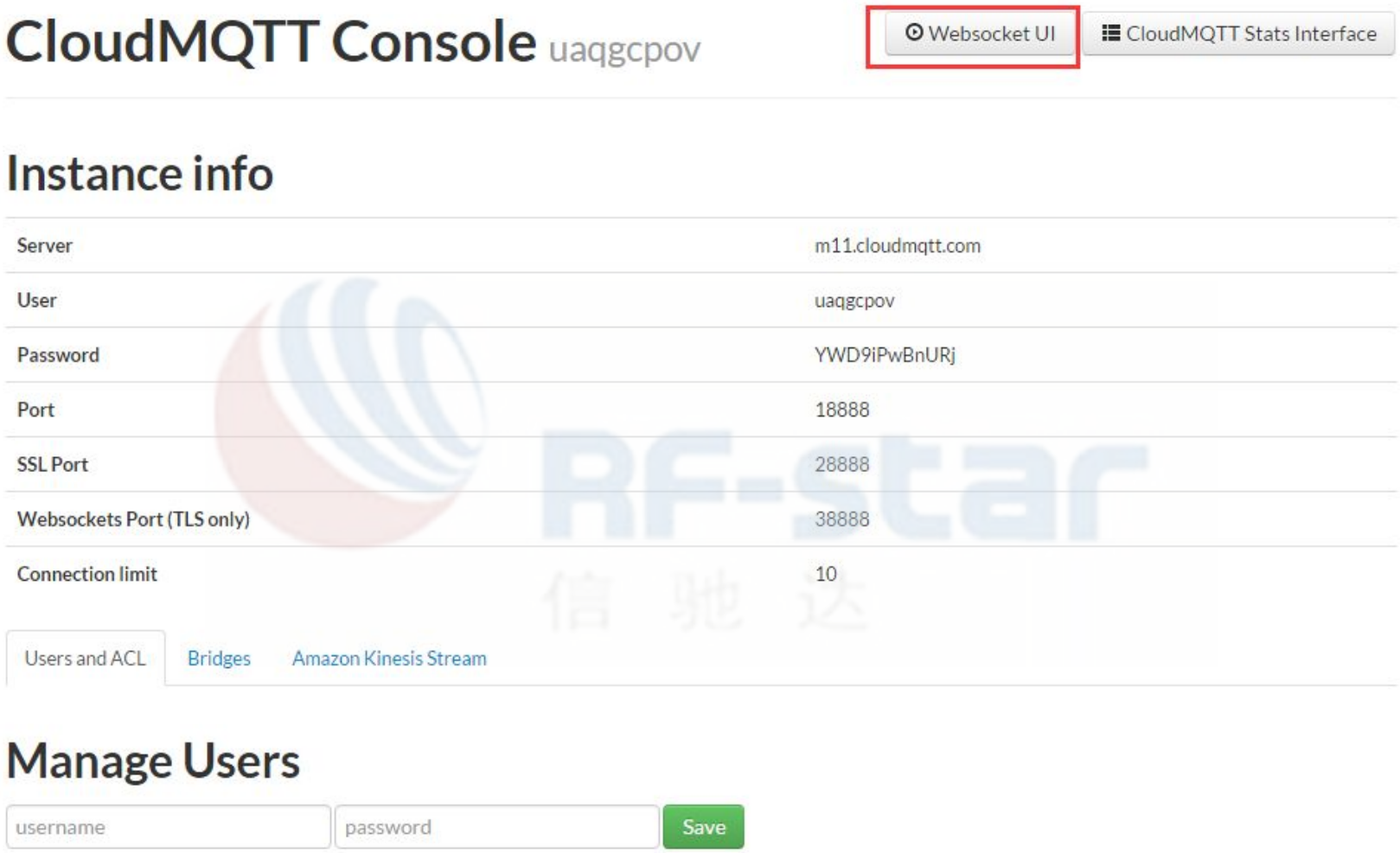


图 19

打开后界面如图20。

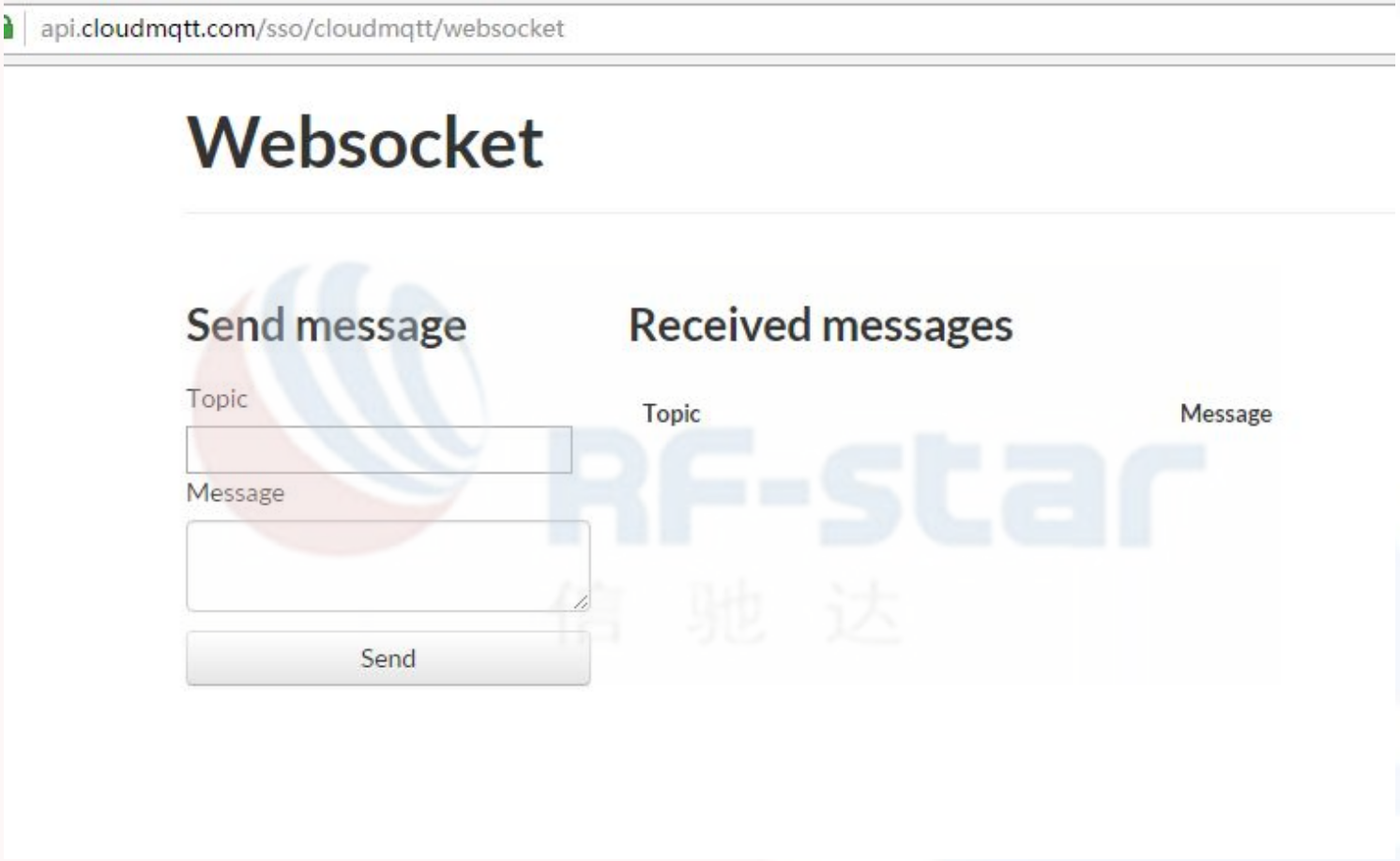


图 20

按下开发板SW2按键，从浏览器可以看到服务器上有数据上传，如图21。

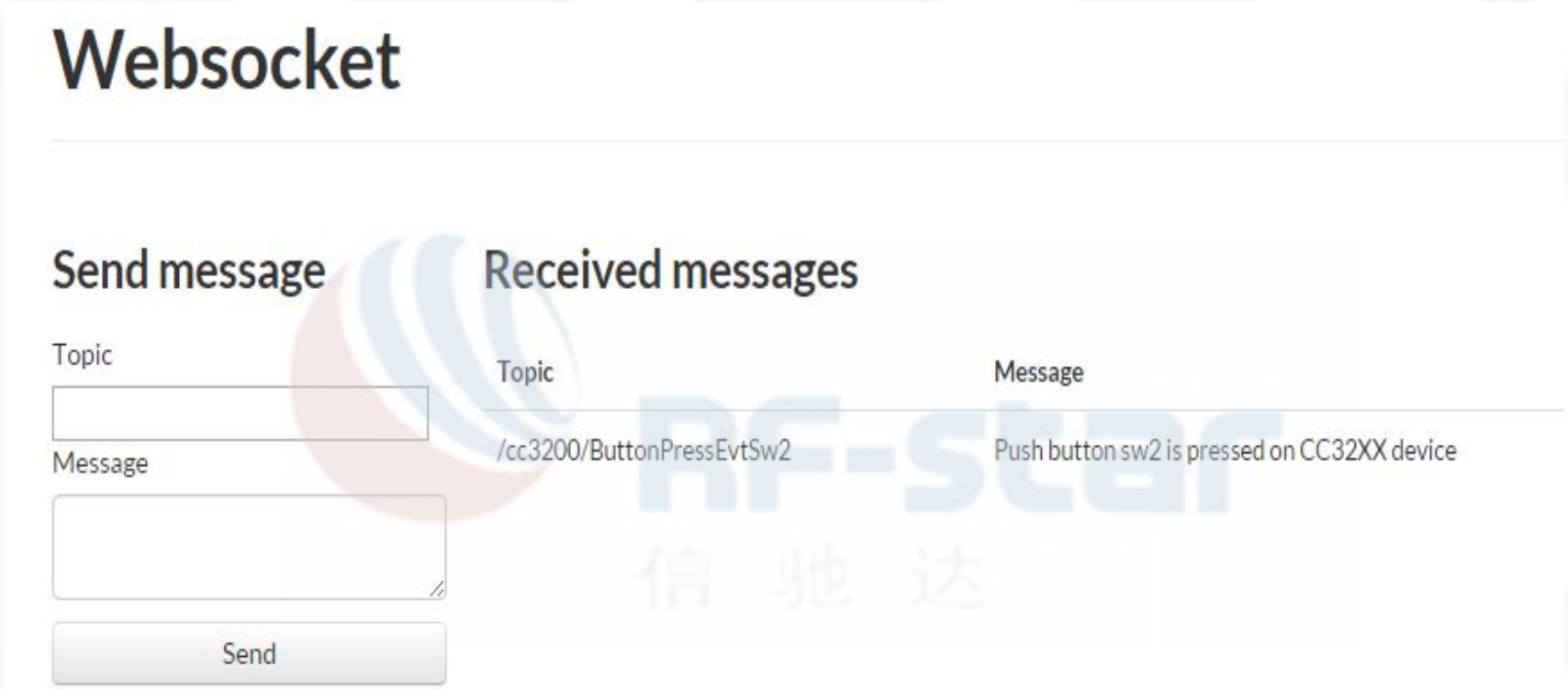


图 21

同时串口工具也打印有CC3200向服务器发布的消息，如图22，SW3按键的功能也是相似的。

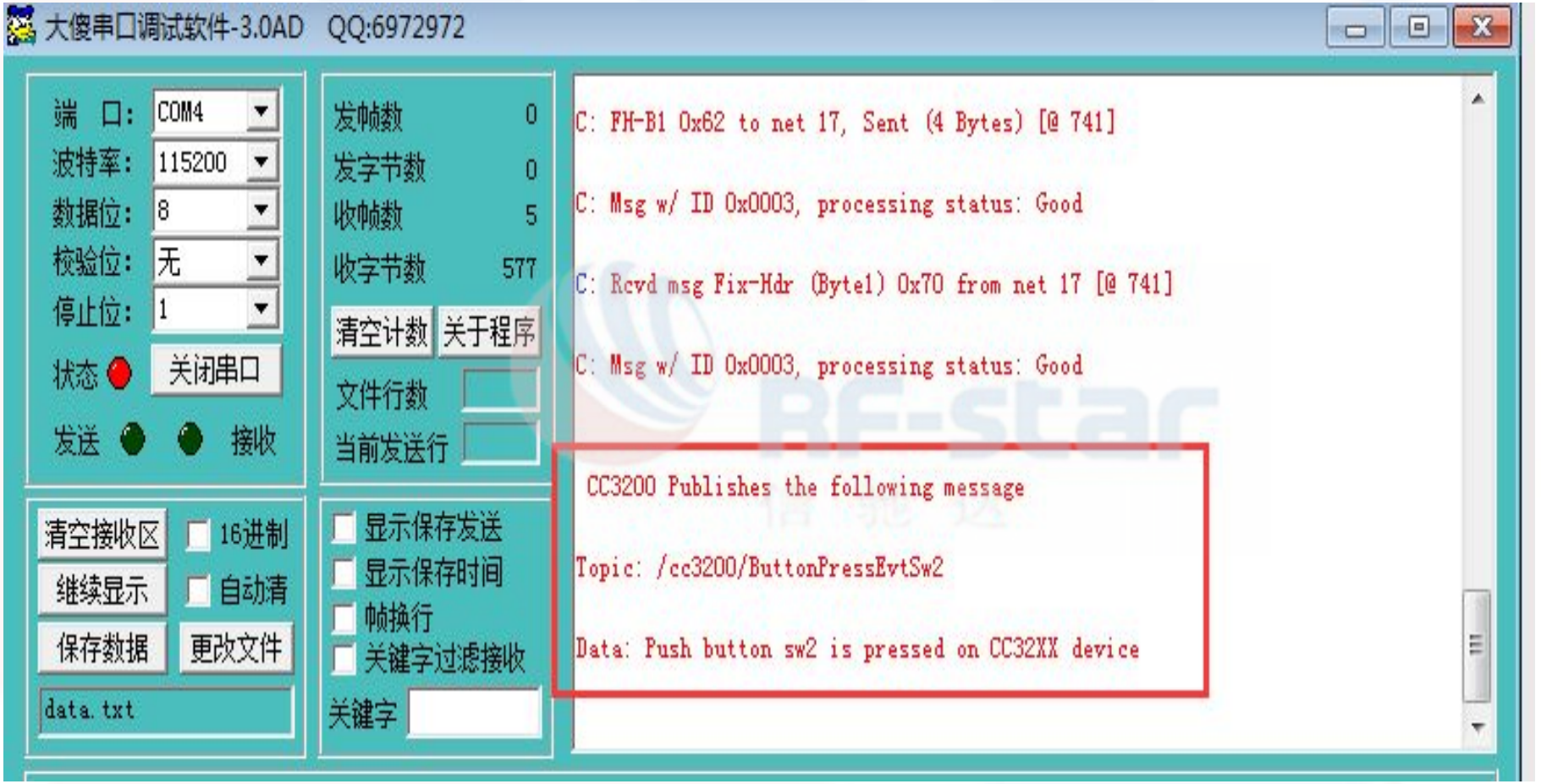


图 22

从MQTT服务器下发信息给CC3200，由于在程序中接收到消息，只比较主题的信息，没对里面的数据处理，所以发任意字符即可，如图23，Topic为“/cc3200/ToggleLEDCmdL1”，数据为“abcd”。

图 23

可以看到开发板的红灯点亮，如图24，再发一次，红灯熄灭，同理，发送/cc3200/ToggleLEDCmdL2、/cc3200/ToggleLEDCmdL3，可以使另外两个灯状态翻转。

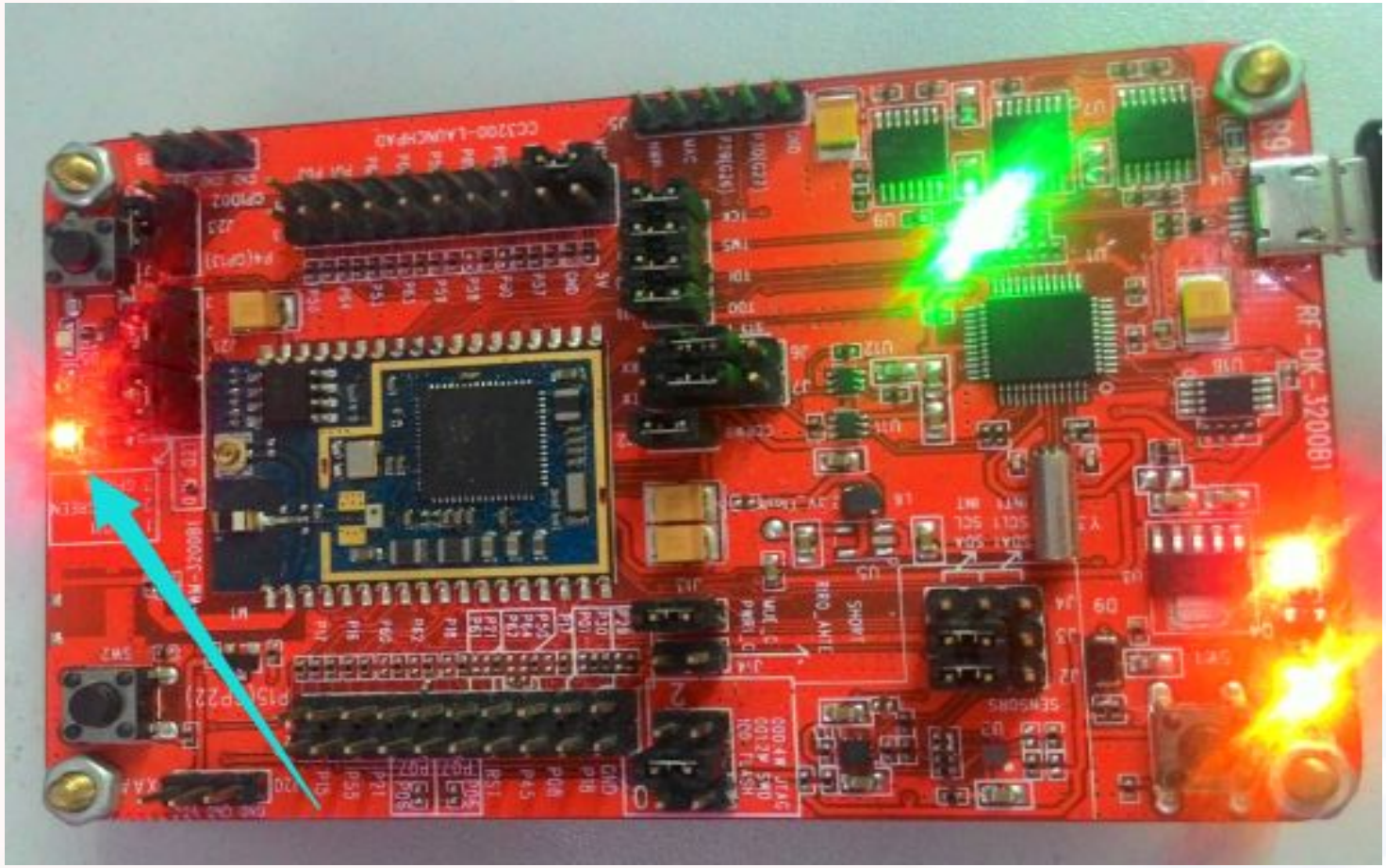


图 24

RF-star
信 驰 达

THANKS

謝謝觀賞