WiFiBee-MT7681 (Arduino WiFi Wireless Programming)
SKU: TEL0107

Introduction

The WiFi Bee MT7681 is an Arduino WiFi XBee module based on the MT7681 serial Wi-Fi module. It is compatible with an XBee slot, supports Arduino wireless programming and is ideal for Internet of Things (IoT), home automation and robotics applications.
The WiFi Bee supports the IEEE802.11b/g/n standard protocols, with 2.4G Wi-Fi bands, 14 channels and WEP/WPA-PSK/WPA2-PSK/TKIP/AES encryption methods, compatible with many mainstream routers. It has STA and AP working modes, is free to switch, and supports TCP/IP communications. In addition the WiFi Bee also supports SmartLink intelligent connection. The unit is easy to use - just install the app (Android), and connect to your wireless router.

The WiFi Bee has same pin mappings as an XBee. It can be inserted in any device which has an existing XBee socket for full networking capability. For Arduino UNO and Arduino Mega2560 boards, the WiFi Bee can also be used as a wireless code upload module eliminating the need for a USB cable or physical connection. This is very convenient when debugging robots or mobile platforms. In addition the module breakouts include three I/O ports. Use it as a Wi-Fi remote control - each I/O supports software PWM outputs. You can configure the parameters using serial AT commands via the serial terminal.

Features

- Support IEEE802.11b/g/n Protocol
- Support WEP/ WPA-PSK/ WPA2-PSK/ TKIP/ AES Multiple Encryption Methods
- Support STA/ AP Operating Mode
- Support Arduino Wi-Fi Wireless Programming
- Support SmartLink intelligent Connection function
- Support OTA Firmware Upgrade

Specification

- Microcontrollers: MT7681
- Protocol Standard: IEEE802.11b / g / n
- Wi-Fi Radius: 180m (different environment, different transmission distance)
- Number of channels: 1-14
- Frequency range: 2.4-2.4835G
- Transmit power: 15-18dbm
- Encryption Method: WEP / WPA-PSK / WPA2-PSK / TKIP / AES
- Operating Mode: STA mode / AP mode (STA & AP mixed mode is not supported, only one device can be connected under AP mode)
- TCP 1000 port: TCP <-> UART data transparent port
- TCP 1001 port: AT command port
- UDP 3333 port: broadcast port
- Operating Voltage: 3.3V
- Recommended Input Voltage: 3.3V
- Power Consumption: 75 mA
- UART serial port: 1
- Baud rate: 9600, 14400, 19200, 38400, 57600, 115200bps (default)
- Dimension: 32 * 24 mm
- Weight: 3.5g

Board Overview
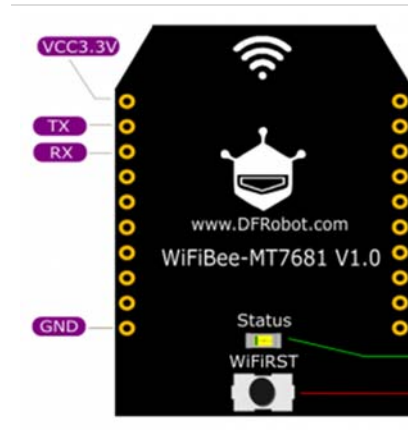
Label	Name	Description
VCC	VCC	3.3V only
IO0	GPIO 0	GIPO 0
IO1	GPIO 1	GIPO 1
IO2	GPIO 2	GIPO 2
GND	GND	GND
RESET	Arduino Reset	Reuse with GPIO 0, Arduino Reset Pin
WIFI Reset	WiFi Reset	Reset WiFi Only

WiFiBee Wireless Network Configuration

WiFiBee-MT7681 supports STA and AP mode. In the first, we'll demonstrate 3 kinds of connection methods under STA Mode:

1. SmartLink (Android APP)
2. Configure the wireless network parameters via AT command
3. Configure the wireless network parameters via Arduino UNO/Mega2560

LED Status	Mode
0.5/s	Boot Mode
1/s	Operational mode, not connect
2/s	Operational mode, connected

SmartLink

Download App DFRobot IoT Manager
https://github.com/DFRobot/WiFiBeeMT7681/raw/master/Android%20APP/IoTManager_V1.1.apk
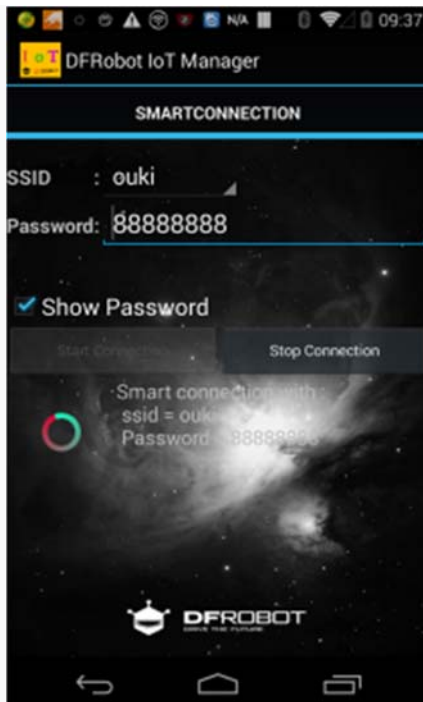
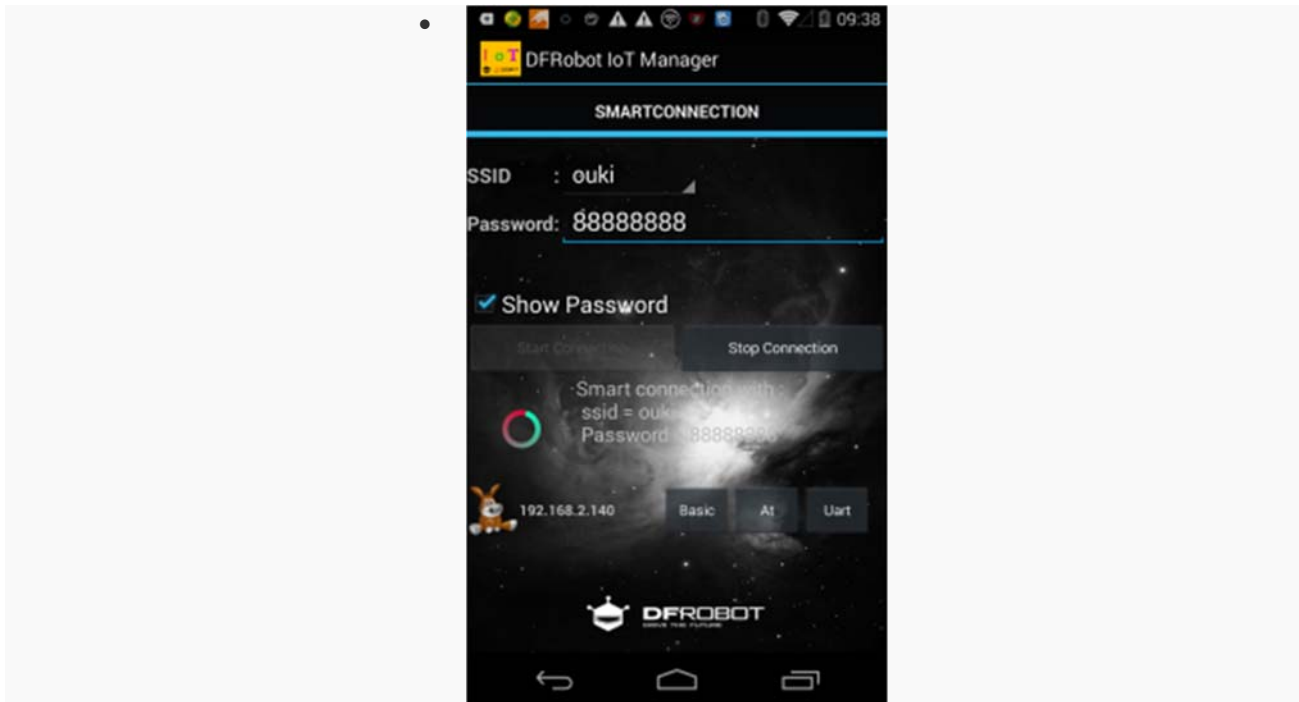- Connect to the WiFi network
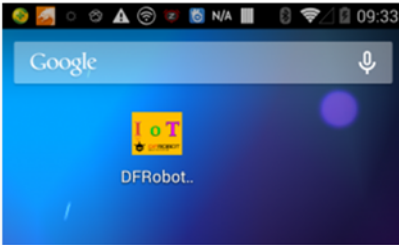
DFRobot IoT Manager

1. Connect your phone to the target network, this is **ouki**

2. Open DFRobot IoT Manager, Select SSID **ouki**, Enter WiFi Password

3. Click '''Start Connection'''

- Wait for some seconds, when the LED flashes slowly (2/s), it means it has been connected to the WiFi. You can see the device IP address in the APP.

Configure the wireless network parameters via AT command

WiFi Bee also supports AT Command configuration, you can use the Xbee USB Adapter to connect WiFi Bee. If you have a DFRobot Leonardo with Xbee socket, you can achieve the same purpose with following debugging code. (Note: Since Uno has only one serial port, it doesn't support the same method, but you can check the next chapter.)

```
void setup() {  // put your setup code here, to run once:
    Serial.begin(115200);
    Serial1.begin(115200);
    while(!Serial);
}
void loop() {   // put your main code here, to run repeatedly:
```

```
    while(Serial.available())

        Serial1.write(Serial.read());

    while(Serial1.available())

        Serial.write(Serial1.read());

}
```
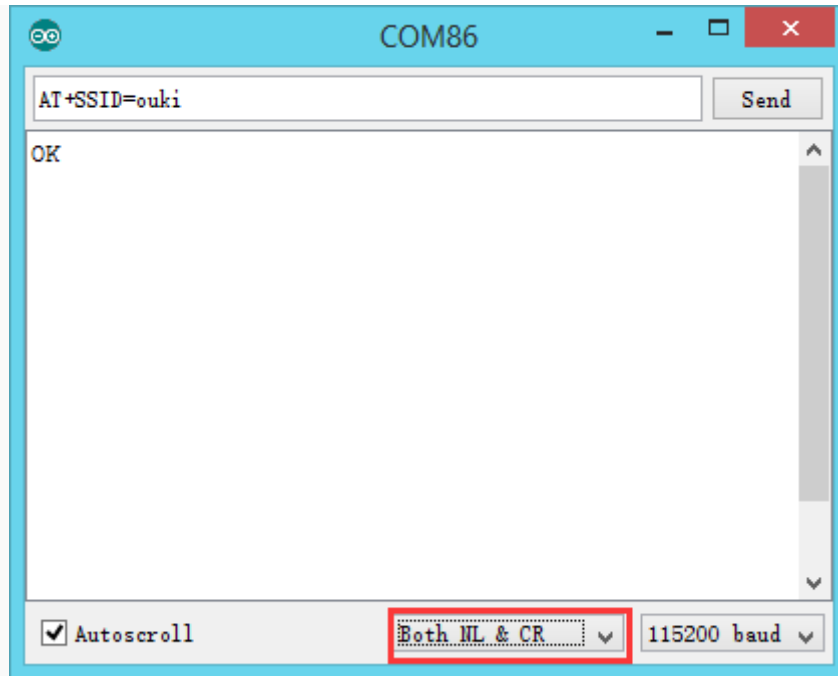
Plug WiFi Bee on the Adapter, open Arduino IDE serial monitor (Serial assistant), input "**+++**" to enter AT mode
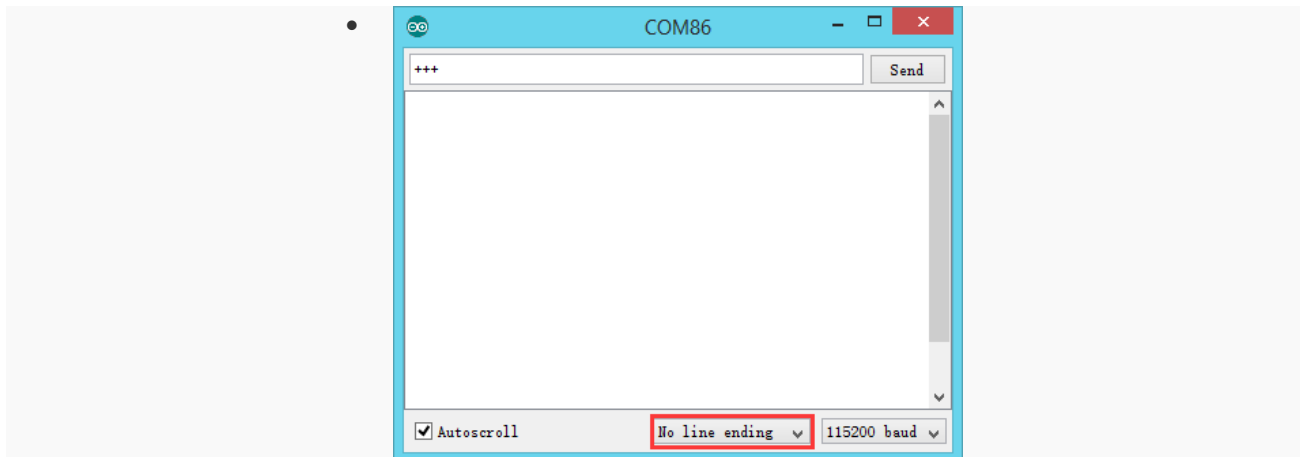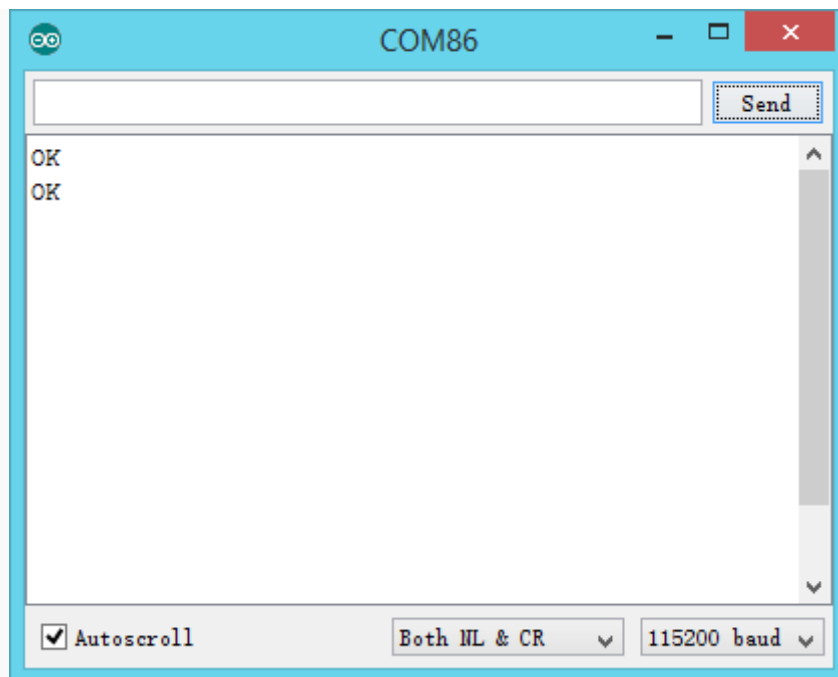

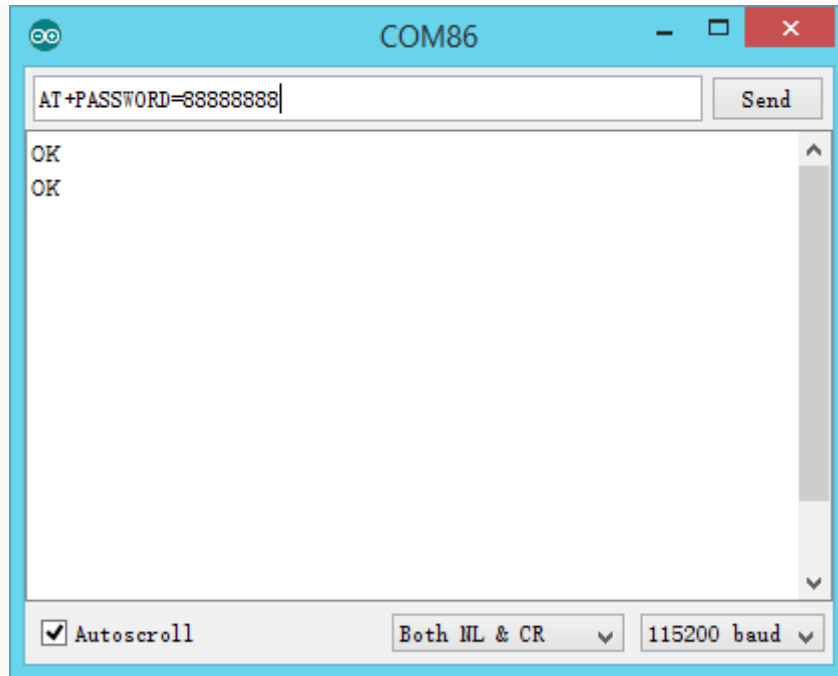1. Select **no line ending**, baudrate **115200bps, 8/n/1**, input "**+++**" to enter AT mode
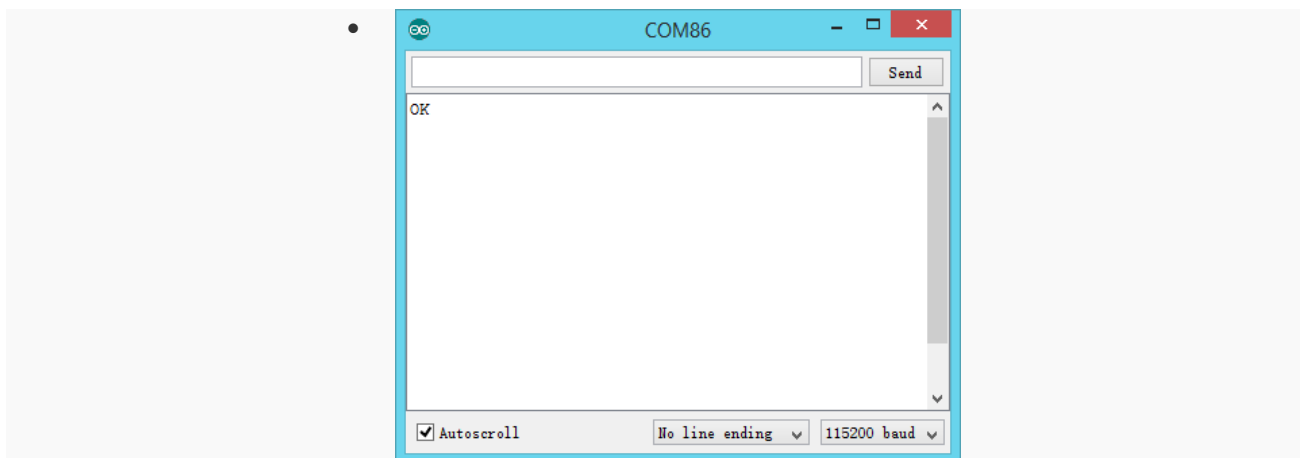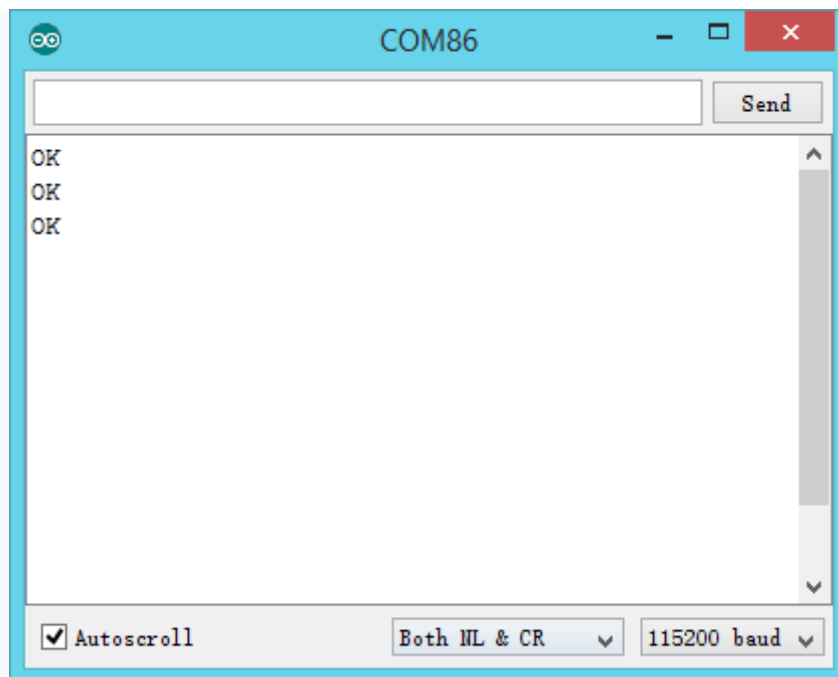

Return **OK**

-

2. Select **Both NL & CR**, input **AT+SSID=Your WiFi SSID** (WiFi Name)
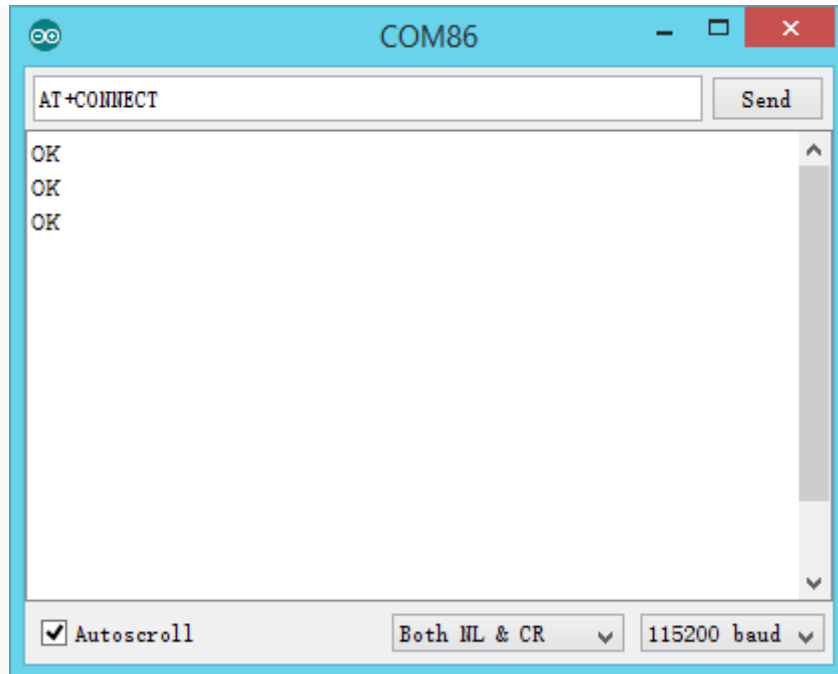
-

Return **OK**

-

3. Input **AT+PASSWORD=Your WiFI Password** (WiFi Password)

-
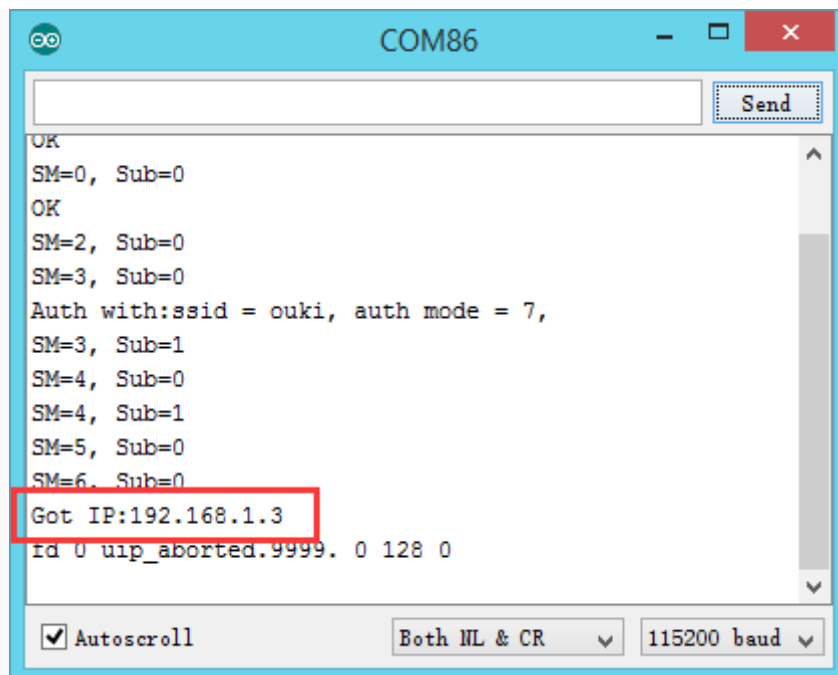

Return **OK**

4. Input **AT+CONNECT**

Return **OK**

Reboot WiFI Bee (Press WiFi Reset), when the led flashes slowly (2/s), it means it has setup the WiFi connection. The IP address is **192.168.1.3**
More the AT command please refer to the below section **WiFiBee AT Command**
https://www.dfrobot.com/wiki/index.php/WiFiBee-MT7681_(Arduino_WiFi_Wireless_Programming)_SKU:_TEL0107#AT_Command

Configure the wireless network parameters via Arduino UNO/Mega2560

Requirements

- **Hardware**
DFRduino UNO (or Mega2560) x 1
Gravity: IO Expansion Shield for Arduino V7.1   https://www.dfrobot.com/product-1009.html
WiFI Bee - MT7681

- **Software**
- Arduino IDE Click to Download Arduino IDE from Arduino® https://www.arduino.cc/en/Main/Software%7C

Sample Code

1. Plug WiFi Bee in the shield xbee socket
2. Turn the shield switch to "“PROG‘"”
3. Upload the following sketch
4. Turn the shield switch to "“RUN‘"“

```
void setup() {

  // put your setup code here, to run once:


}


const char ssid[]={

  "AT+SSID=ouki"};    // WiFi SSID

const char passwd[]={

  "AT+PASSWORD=88888888"}; // WiFi  password


void setup()

{

  Serial.begin(115200);


  delay(100);

  while(!Serial);

  Serial.print("+++");

  delay(1000);

  Serial.println(ssid);
```

```
    delay(100);

    Serial.println(passwd);

    delay(100);

    Serial.println("AT+REBOOT");

    delay(100);

}


void loop()

{

    while(Serial.available())

    {

        Serial.write(Serial.read());

    }

}
```

> **NOTE**: **RUN/PROG** is the serial switch, when you upload the sketch via USB, you have to turn the switch to "PROG" side. Once you want to use Xbee module, you need to turn it back to the "RUN" side. (Wireless Programming)

Result

WiFi Bee IP Address

AT Command

AT Command is generally applied to the terminal equipment and PC communication, In this section, we use AT command to configure network parameters.

How to Use AT Command

AT Command	Description
+++	Enter AT Mode
AT+SSID <CR><LF>	AT+SSID=? Request WiFi SSID
	AT+SSID=SSIDName Set SSID Name
AT+PASSWORD <CR><LF>	AT+PASSWORD=? Request WiFi Password
	AT+PASSWORD=PASSWORD Set WiFi Password
AT+CONNECT <CR><LF>	Connect WiFi
AT+DISCONNECT <CR><LF>	Disconnect WiFi

Note: Except "+++", other commands need to add carriage returns <CR> and linefeed <LF> symbols, so in the previous section, you need to switch to "no line ending" when entering "+++", switch to "both CR & LF" when entering other commands.

More the AT command, please refer to the **WiFiBee AT Command List V1.1**
https://github.com/DFRobot/WiFiBee-MT7681/raw/master/AT%20Command%20Manual/EN/WiFiBee-MT7681%20AT%20Command%20V1.1.pdf

## WiFi Bee TCP Server Communication

WiFi Bee supports TCP & UDP service, in this section, we'll demonstrates the TCP <-->UART communication.
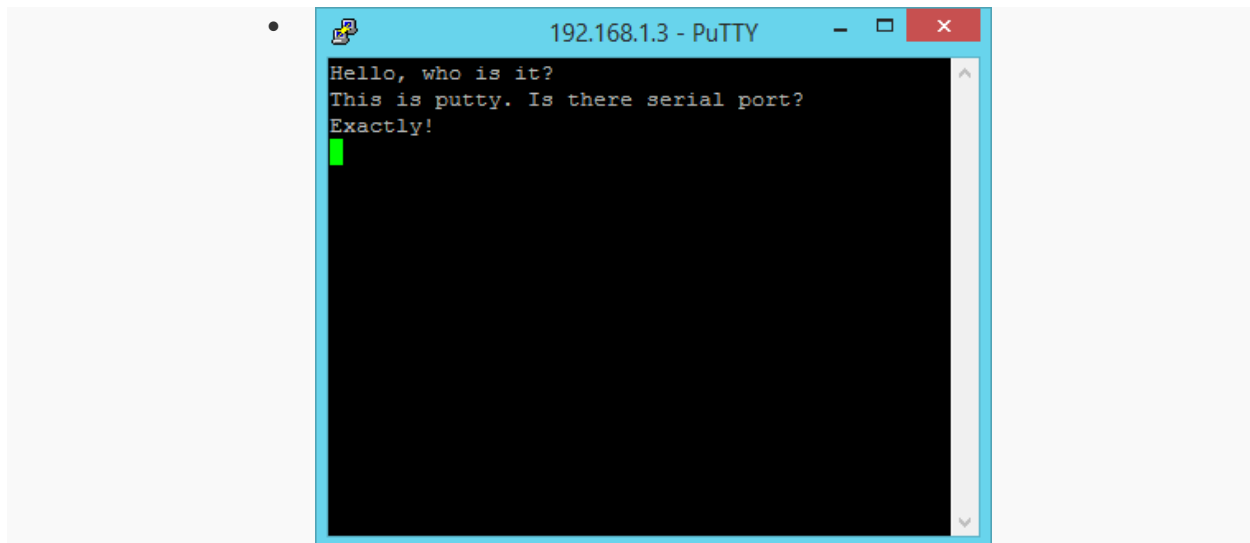
Requirements

- **Hardware**
  Xbee USB Adapter, or Leonardo with Xbee socket (Burned the debugging code) x1
  WifiBee-MT7681 x1
- **Software**
- Arduino IDE click to download Arduino IDE https://www.arduino.cc/en/Main/Software
- Net Assistant software, like putty http://www.putty.org/

Steps

1. Plug WiFiBee on the adapter
2. configure network parameter: "WiFiBee IP Address", "Port: 1000" (Port: 1000 is the WiFi Bee default server port. If you need it working under client mode, you need to configure its Remote Server Address, refer to AT Command List)
3. Click "Open" to open the terminal
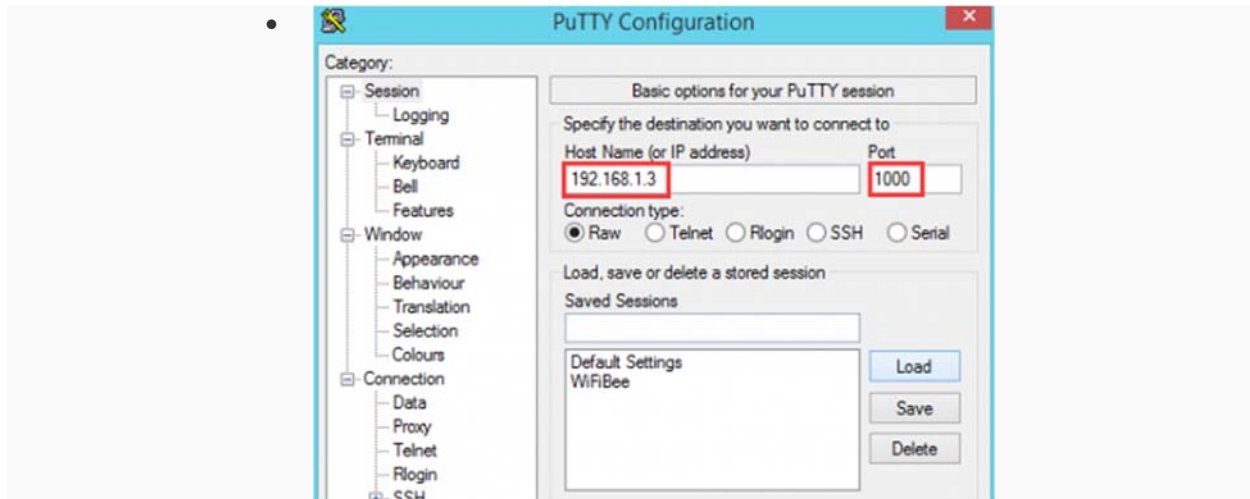
Configurate Putty Parameter

Putty

-
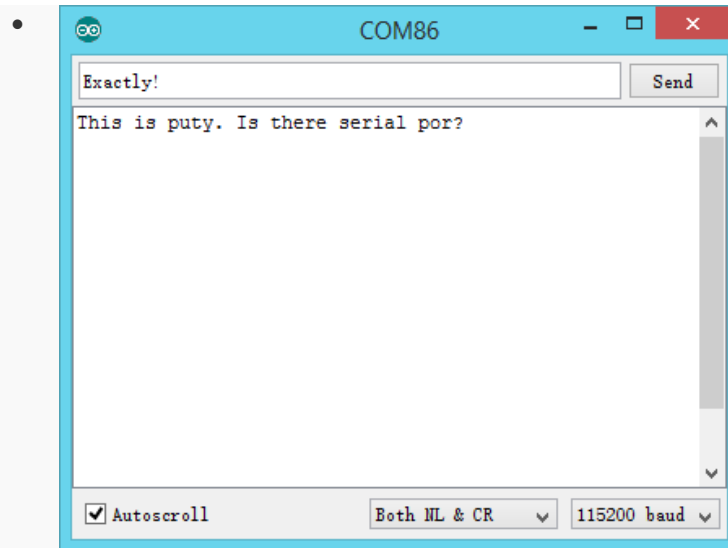
  Serial

Arduino Wireless Programming

WiFIBee supports WiFi Wireless Programming function, just connect WiFi bee to network which you PC in. For now, it supports Arduino UNO, Mega2560, DFRduino M0.
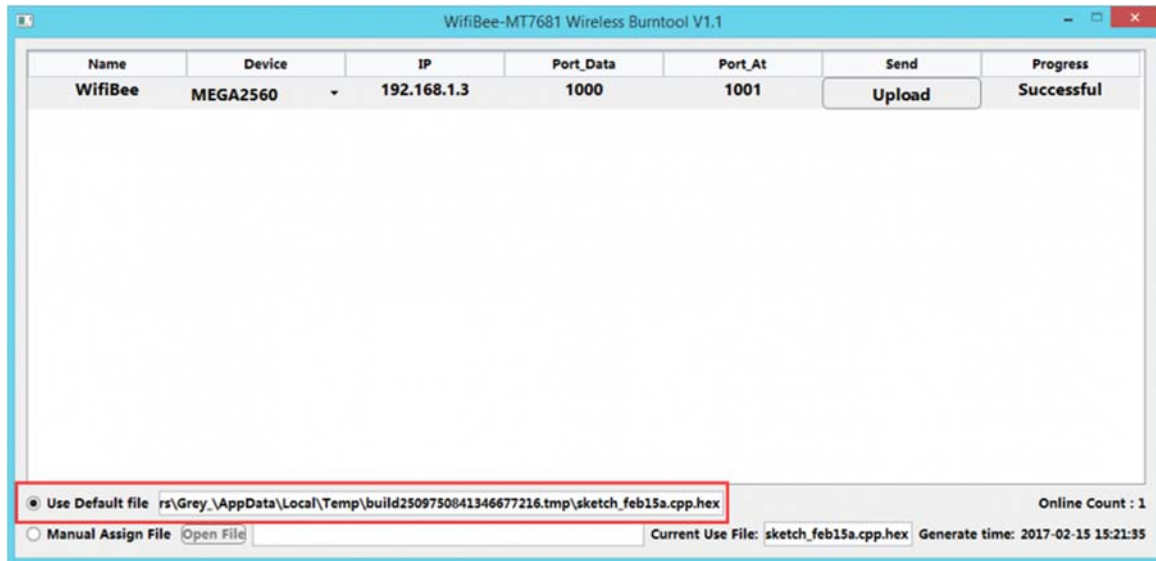
Requirement

- **Hardware**
  UNO or Mega2560 x1
  WifBee-MT7681 x1
  IO expansion shield V7.1 or mega sensor shield V2.4 x1
- **Software**
  Arduino IDE Click to download Arduino IDE https://www.arduino.cc/en/Main/Software

Steps

- Plug WiFiBee on the shield xbee sockets. (Only COM0 supports Wireless programming)
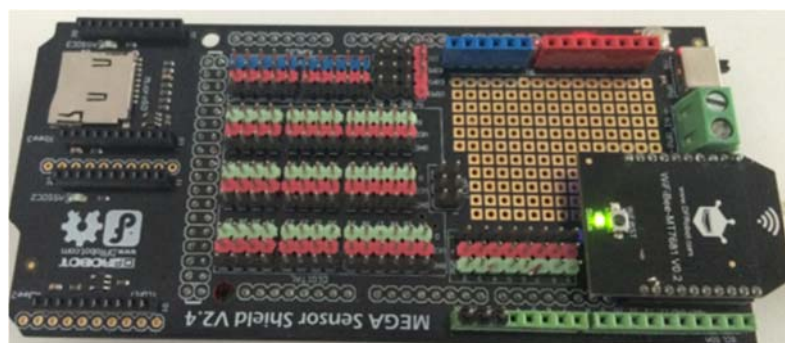
- Download the Wireless Programming Tool **WiFiBee-MT7681 Wireless BurnTool**(Windows)
  - https://github.com/DFRobot/WiFiBee-MT7681/raw/master/Wireless%20Programming%20Tool/WifiBeeMT7681%20Wireless%20Burntool%20V1.1.exe


WiFiBee-MT7681 Wireless BurnTool

- Open the tool, the software will search the device in the network automatically. select "Use Default File" cache files. it will be the last compiled HEX file. And you can load the HEX manually, "Manual Assign File".
-
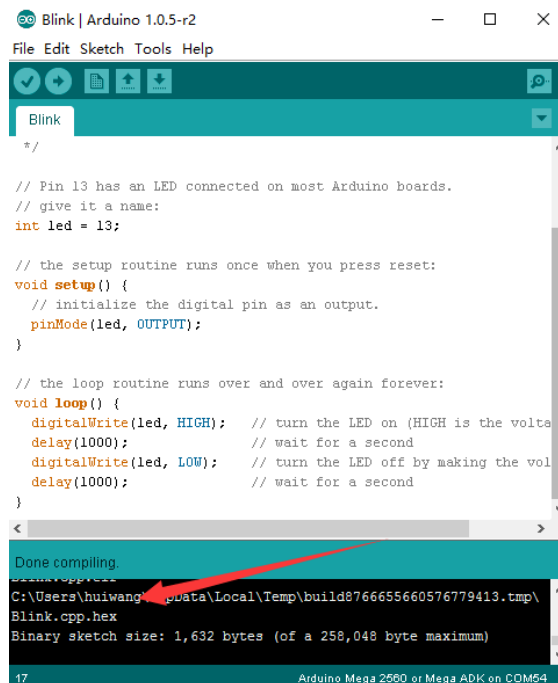
- Open arduino IDE, open a sample code, such as "Blink".
- Select the board, UNO or Mega2560, click Update.
- It shows percentage during uploading
- Note: It requires Baudrate 115200 bps for the wireless programming function
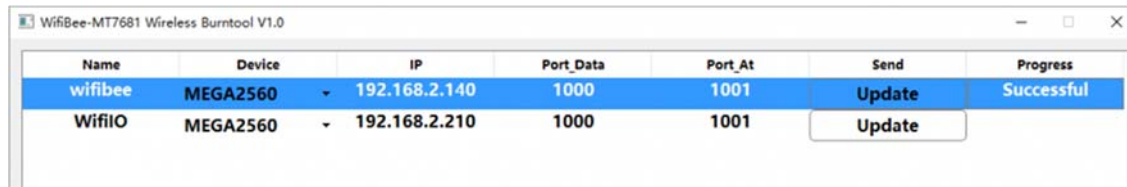
Result

Watch the board D13 indicator LED, it should be Blinking now.

WiFiBee AP Mode

WiFi Bee can work under STA and AP mode. When it works under AP mode, the device can connect to WiFi Bee directly. But please note, AP mode only support one device simultaneously. And the IP address under AP mode has been fixed to 192.168.81.1

- AP Command under AT Mode

1) Mode Switch: STA->AP

| AT Command	Description
AT+ROLE=ROLE_AP	Return OK
AT+REBOOT	Reboot WiFiBee and enter AP Mode

2) Mode Switch: AP->STA:

AT Command	Description
AT+ROLE=ROLE_STA	Return OK
AT+REBOOT	Reboot WiFiBee and enter STA Mode

3) Set and Query SSID and Password under AP Mode

AT Command	Description
AT+SOFTAPCONF=?	Return WiFiBee SSID and Password
AT+SOFTAPCONF=jansion1,a12345678	Set AP SSID and Password "jansion1" "a12345678".

- AP mode also support APP control and Wireless Programming, just work as STA mode.
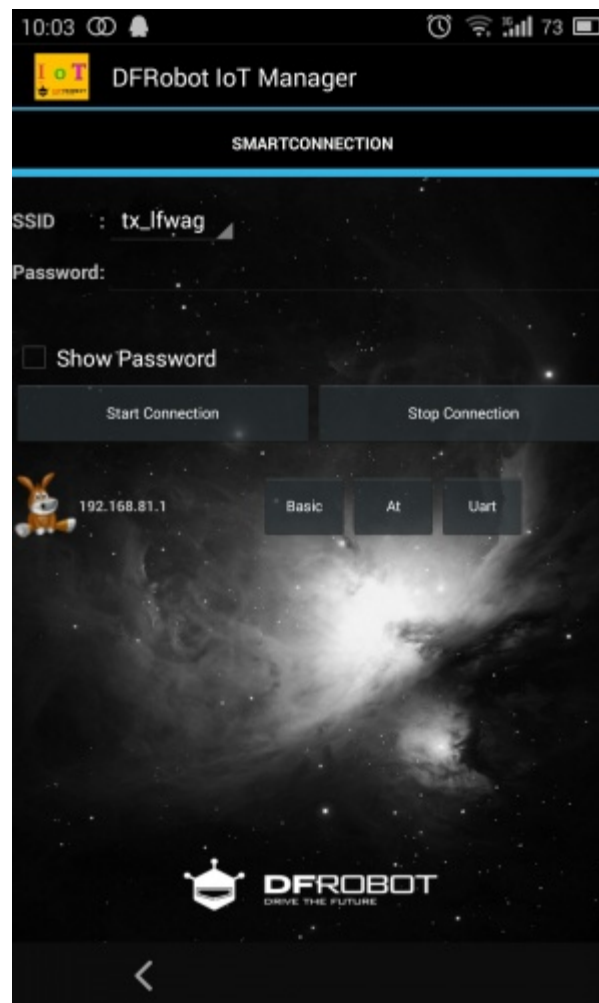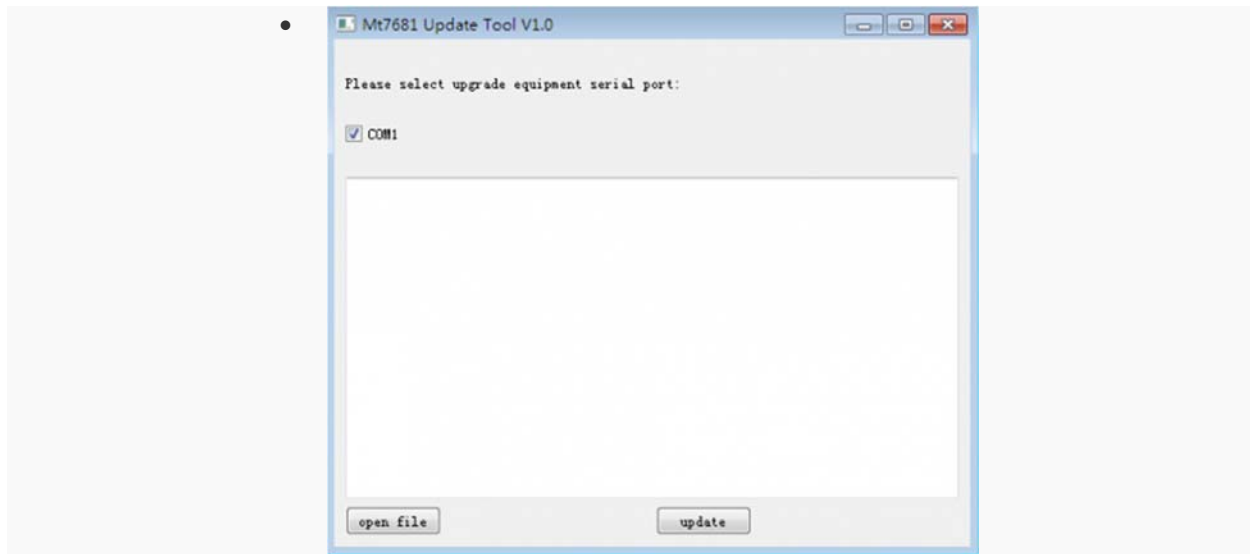

MT7681 Firmware Update
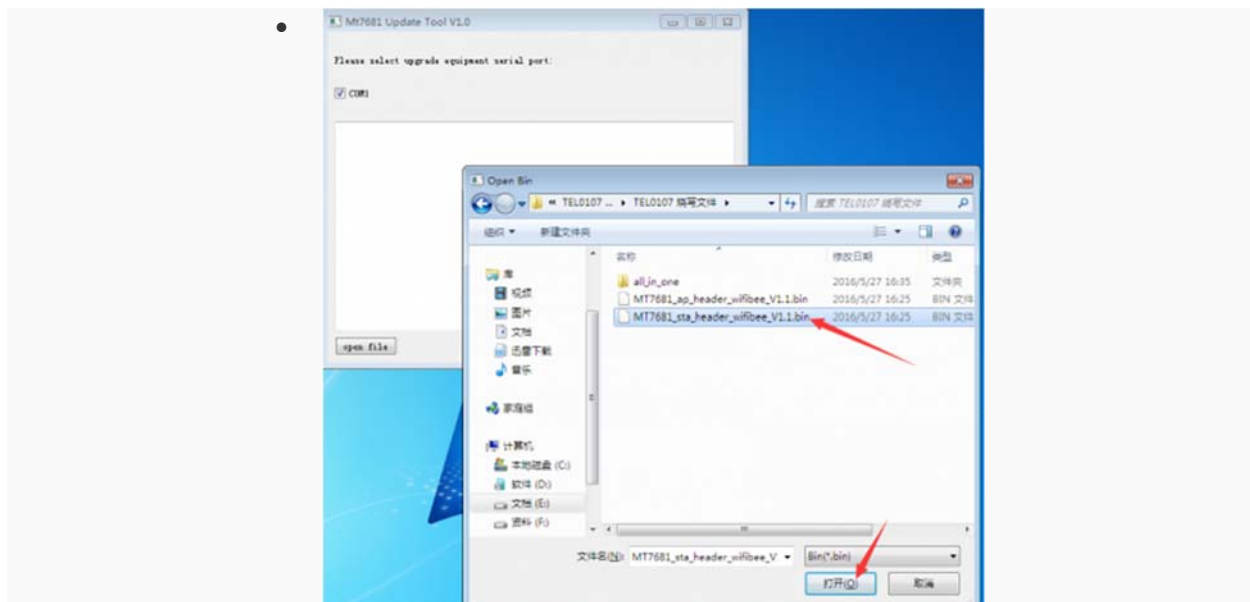
The users can update the firmware by themselves.

- **Hardware**
  WiFiBee x1
  USB Xbee Adapter x1
  Micro USB cable x1
- **Software**

  Arduino IDE Click to download Arduino IDE https://www.arduino.cc/en/Main/Software
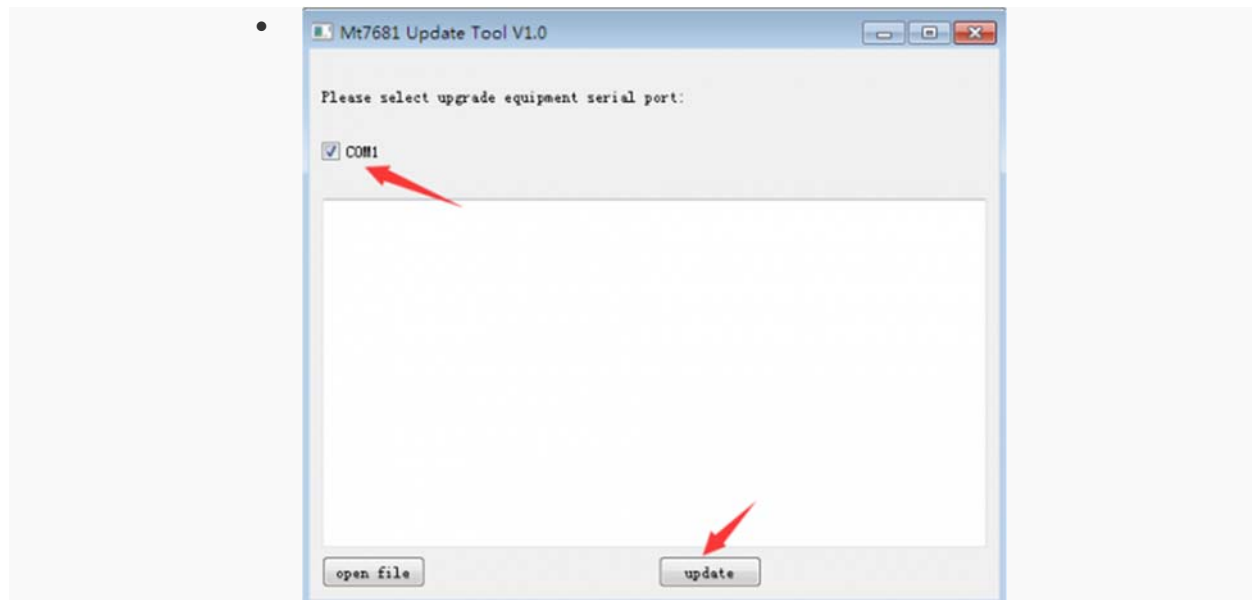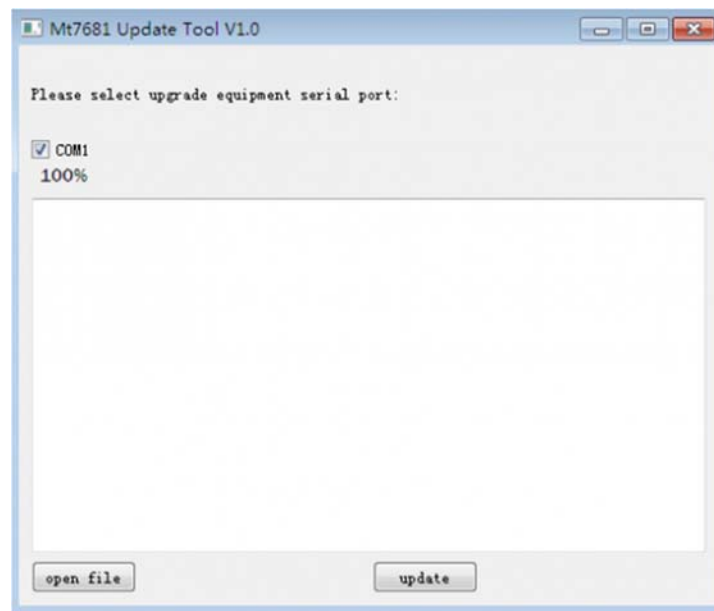
Steps

- Plug WiFi Bee on the adapter

1. Open MT7681 Update Tool

2. Select the newest firmware

3. Select COM port and click "Update"

4. Success