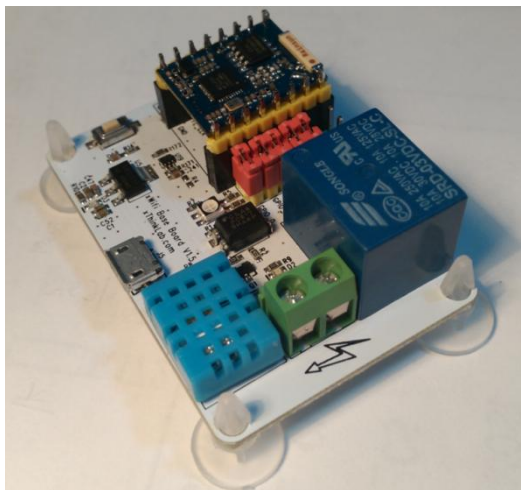


xWifi Simple Guide V1.0-MT7681



The purpose of this document is to give an user guide of xWifi kit.

1.Compiling firmware code.....	3
1.1 Based on MinGW Toolchain.....	3
1.2 Based on AndeSight Toolchain.....	6
2. Upgrade firmware via UART.....	8
3. Switch working mode.....	13
4.Application demo (Android)	14

1.Compiling firmware code

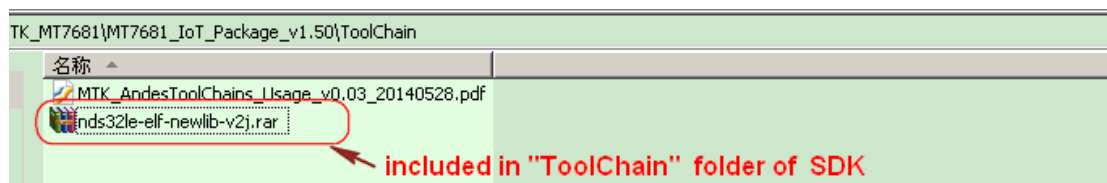
There are several ways to compile firmware source code ,we introduce two ways below based on Windows XP system .We can take either **MinGW** or **AndeSight** develop tools as the compiling environment.

There are some compile commands as follows:

```
make b=0 clean //clean object files
make b=0 //create recovery bin
make b=1 clean // clean object files
make b=1 //create sta bin
make b=2 clean // clean object files
make b=2 //create ap bin
```

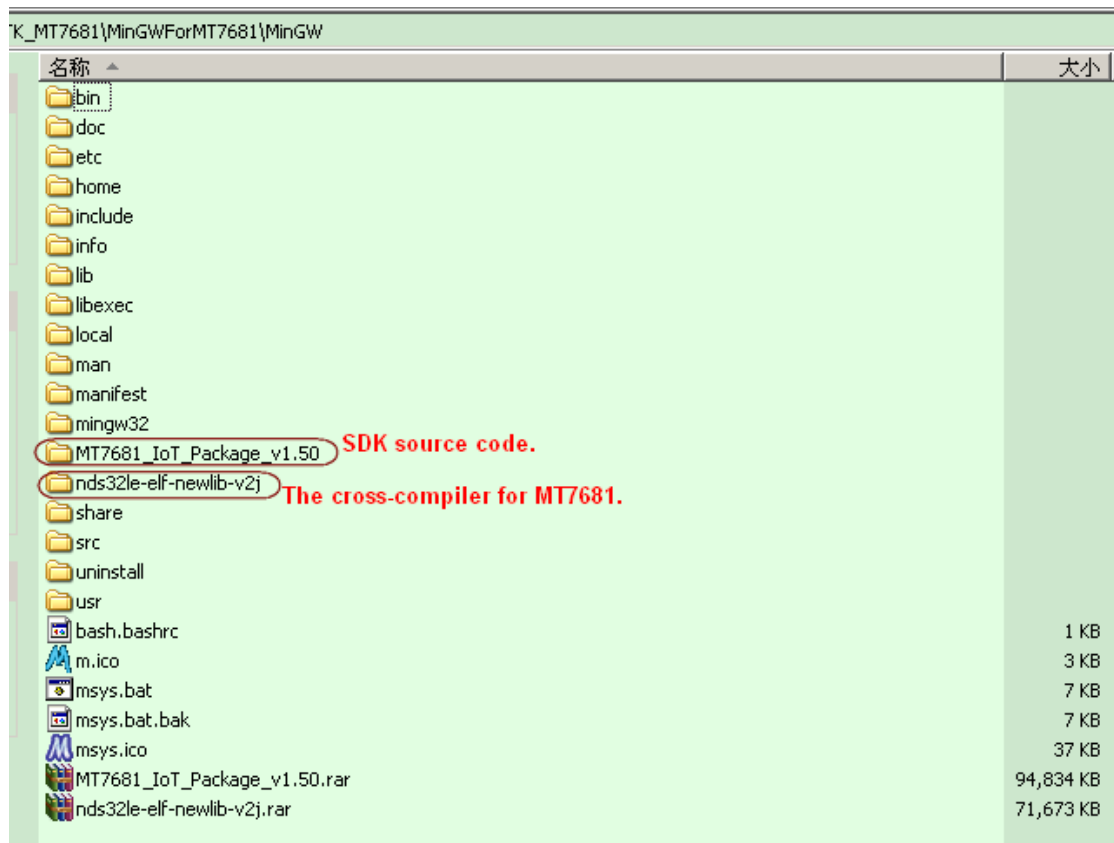
1.1 Based on MinGW Toolchain

Step 1:Find the special Toolchain for MT7681



Step 2:Access to link below and Download MinGW toolchain we have verified

Step 3:Copy Toolchain and SDK file into “MinGW” directory and extract it.



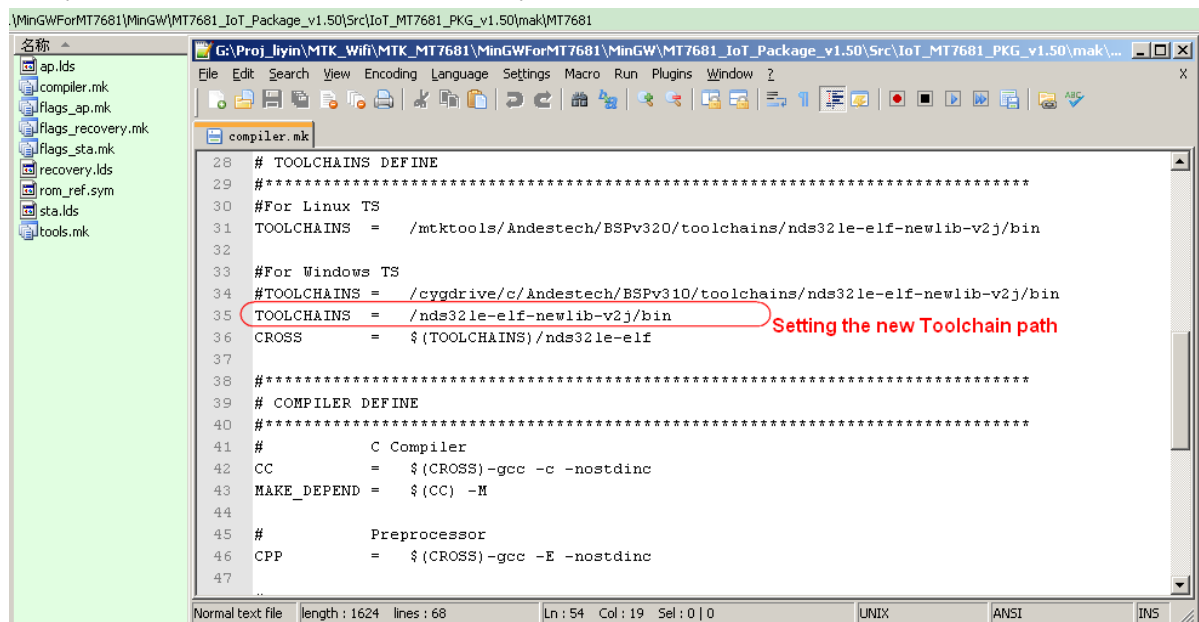
Step 4:

change the directory to

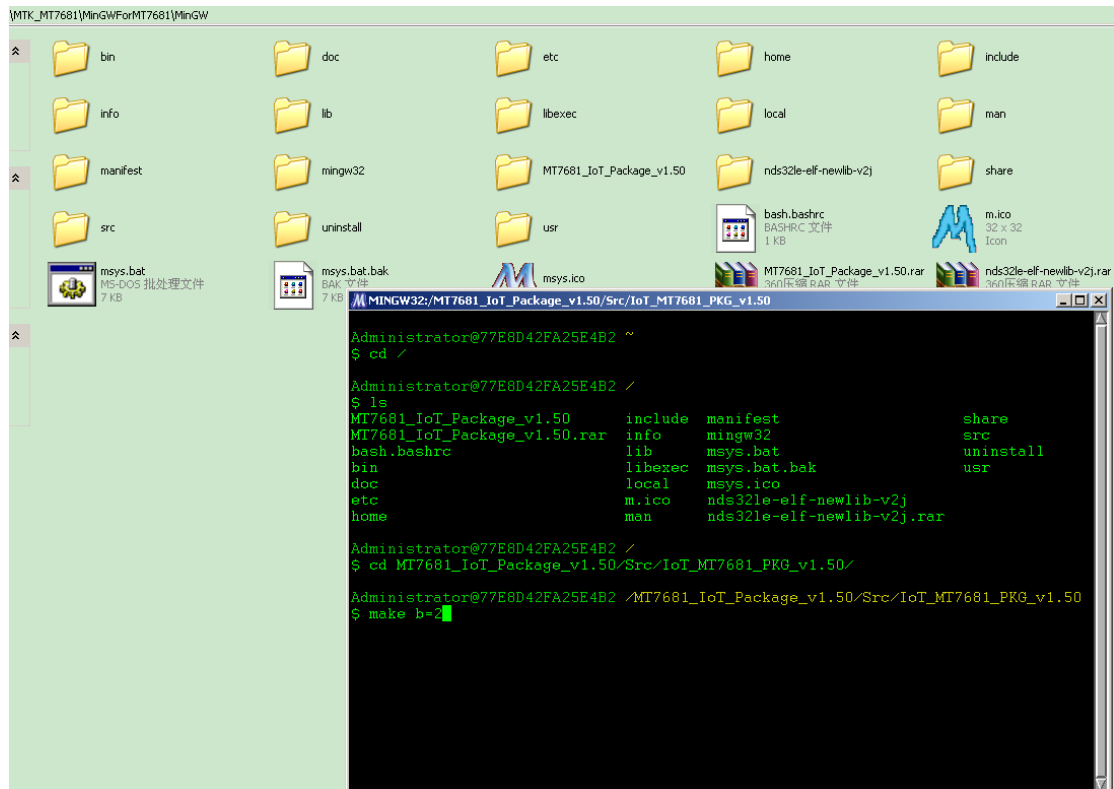
"\MinGW\MT7681_IoT_Package_v1.50\Src\IoT_MT7681_PKG_v1.50\mak\MT7681"

,open

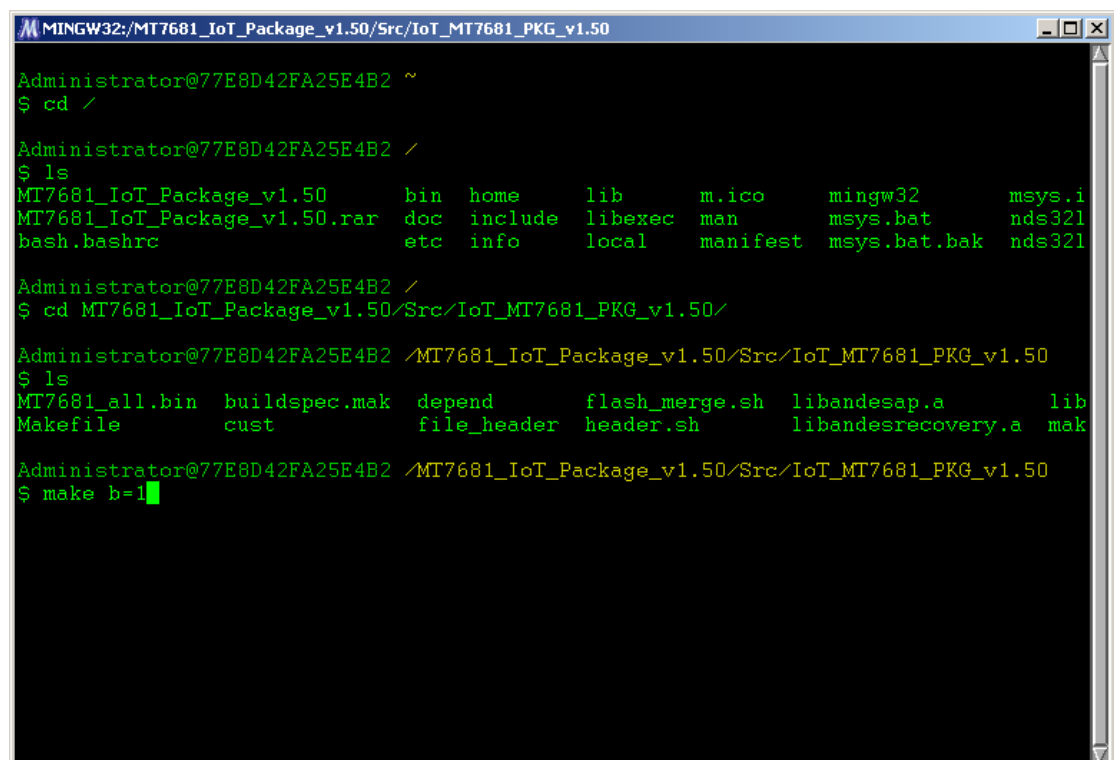
"compiler.mk" file and Set the new toolchain path like below .



Step 5:Run "msys.bat" file in "MinGW" directory:

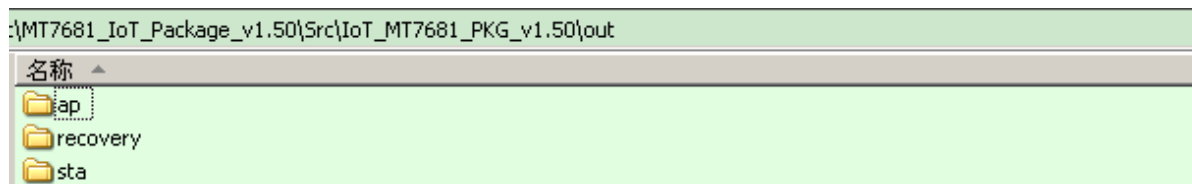


Step 6: Enter into “\MT7681_IoT_Package_v1.50\Src\IoT_MT7681_PKG_v1.50” directory ,compiling source code .



```
MINGW32:/MT7681_IoT_Package_v1.50/Src/IoT_MT7681_PKG_v1.50
Administrator@77E8D42FA25E4B2 /MT7681_IoT_Package_v1.50/Src/IoT_MT7681_PKG_v1.50
$ make b=1
echo Compiling cust/spi-flash_pub.c ...
Compiling cust/spi-flash_pub.c ...
echo Compiling cust/spi-flash_pub.c ... >> out/sta/build_sta.log
echo Compiling cust/tcpip/clock-arch.c ...
Compiling cust/tcpip/clock-arch.c ...
echo Compiling cust/tcpip/clock-arch.c ... >> out/sta/build_sta.log
echo Compiling cust/tcpip/dhccpc.c ...
Compiling cust/tcpip/dhccpc.c ...
echo Compiling cust/tcpip/dhccpc.c ... >> out/sta/build_sta.log
echo Compiling cust/tcpip/dhccpd.c ...
Compiling cust/tcpip/dhccpd.c ...
echo Compiling cust/tcpip/dhccpd.c ... >> out/sta/build_sta.log
echo Compiling cust/tcpip/iot_tcp_app.c ...
Compiling cust/tcpip/iot_tcp_app.c ...
echo Compiling cust/tcpip/iot_tcp_app.c ... >> out/sta/build_sta.log
echo Compiling cust/tcpip/iot_tcpip_interface.c ...
Compiling cust/tcpip/iot_tcpip_interface.c ...
echo Compiling cust/tcpip/iot_tcpip_interface.c ... >> out/sta/build_sta.log
echo Compiling cust/tcpip/iot_udp_app.c ...
Compiling cust/tcpip/iot_udp_app.c ...
echo Compiling cust/tcpip/iot_udp_app.c ... >> out/sta/build_sta.log
echo Compiling cust/tcpip/memb.c ...
Compiling cust/tcpip/memb.c ...
echo Compiling cust/tcpip/memb.c ... >> out/sta/build_sta.log
echo Compiling cust/tcpip/mt76xx_dev.c ...
Compiling cust/tcpip/mt76xx_dev.c ...
echo Compiling cust/tcpip/mt76xx_dev.c ... >> out/sta/build_sta.log
echo Compiling cust/tcpip/resolv.c ...
```

The directory of generated binary files is in



1.2 Based on AndeSight Toolchain

Step 1:

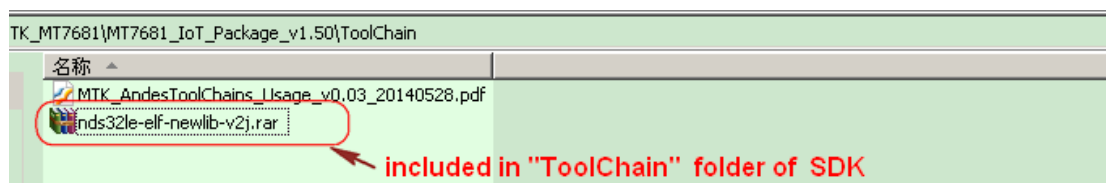
Access to the link below ,register and request a Evaluation Version of AndeSight

<http://www.andestech.com/en/download/andesight-download.htm>

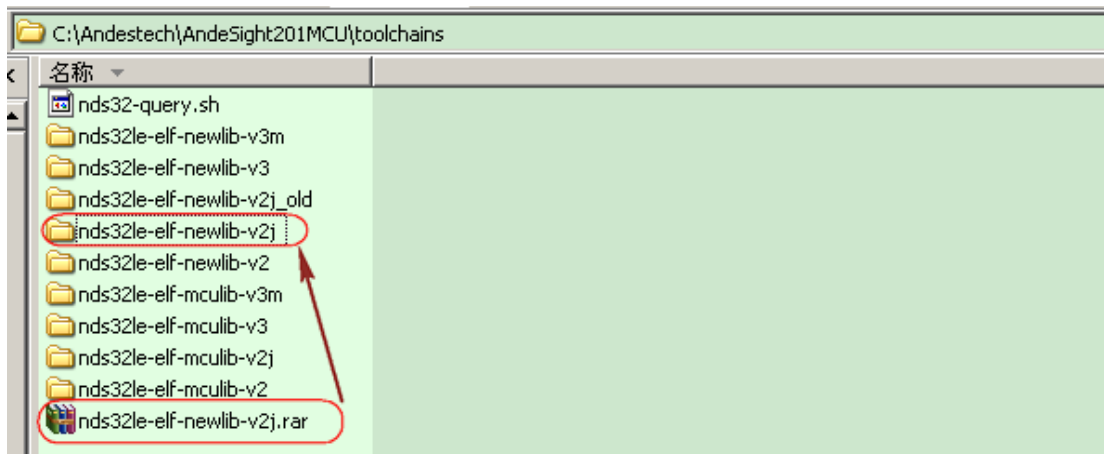
when you register successfully ,you will receive a Email with download information from Andes.Download it and install it .

Step 2:

copy Toolchain file into “C:\Andestech\AndeSight201MCU\toolchains”



Backup “nds32le-elf-newlib-v2j” folder into other name or delete it ,then uncompress “nds32le-elf-newlib-v2j.rar”.



Step 3:

Change the directory to

"C:\Andestech\AndeSight201MCU\toolchains\nds32le-elf-newlib-v2j", and edit "cygwin-andes.bat" like below .

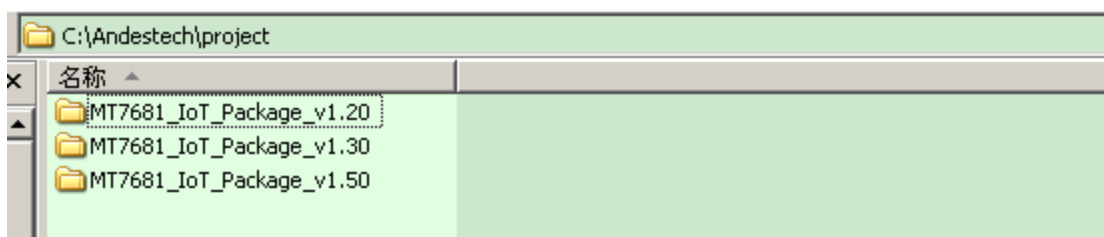
```

1  @echo off
2
3  @rem Batch file for Andes Technology Product to launch Cygwin
4
5  set TOP=C:\Andestech\AndeSight201MCU\cygwin
6  set TOP=%TOP:\/=%
7  set TMP=C:\Andestech\AndeSight201MCU\cygwin\cygwin_tmp
8  set PATH=C:\Andestech\AndeSight201MCU\cygwin\bin;%CD%\bin
9  set HOME=%CD%\bin
10
11 IF EXIST "C:\Andestech\AndeSight201MCU\cygwin\bin\bash.exe" set SHELL=/bin/bash
12
13 "C:\Andestech\AndeSight201MCU\cygwin\bin\bash.exe" --login -i
14
15 :END
16

```

Step 4:

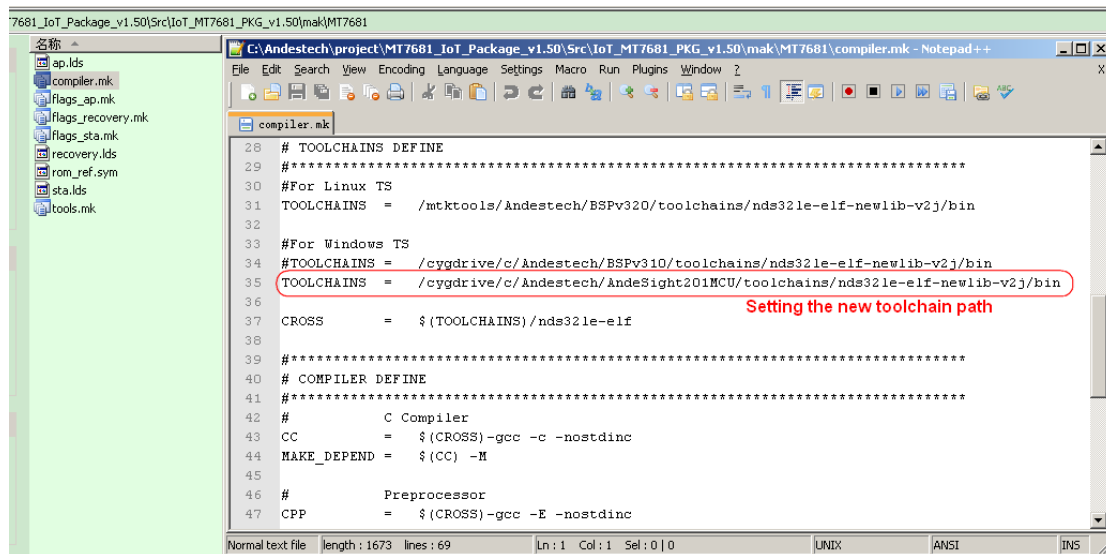
Extract SDK into a directory (such as C:\Andestech\project\)



Step 5:

change the directory to

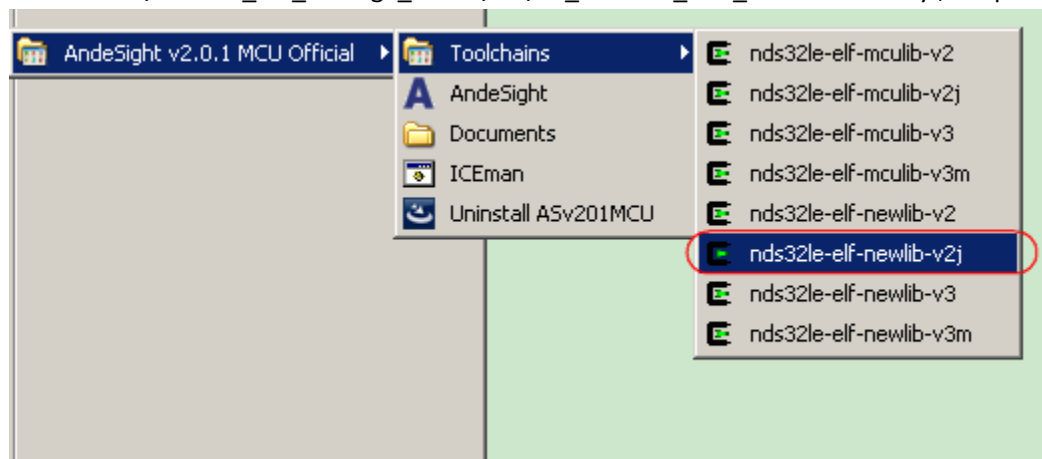
"C:\Andestech\project\MT7681_IoT_Package_v1.50\Src\IoT_MT7681_PKG_v1.50\mak\MT7681", open "compiler.mk" file and Set the new toolchain path like below .



Step 6:

Run "nds32le-elf-newlib-v2j" program,

Enter into "\"MT7681_IoT_Package_v1.50\Src\IoT_MT7681_PKG_v1.50" directory ,compile source code .




```
/cygdrive/c/Andestech/project/MT7681_IoT_Package_v1.50/Src/IoT_MT7681_PKG_v1.50
Your group is currently "mkpasswd". This indicates that
the /etc/passwd (and possibly /etc/group) files should be rebuilt.
See the man pages for mkpasswd and mkgroup then, for example, run
mkpasswd -l [-dl] > /etc/passwd
mkgroup -l [-dl] > /etc/group
Note that the -d switch is necessary for domain users.

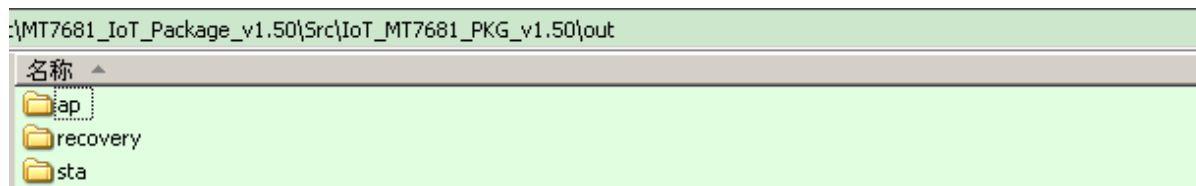
Administrator@77e8d42fa25e4b2 ~
$ pwd
/cygdrive/c/Andestech/AndeSight201MCU/toolchains/nds32le-elf-newlib-v2j/bin

Administrator@77e8d42fa25e4b2 ~
$ cd ../../../../

Administrator@77e8d42fa25e4b2 /cygdrive/c/Andestech
$ cd project/MT7681_IoT_Package_v1.50/Src/IoT_MT7681_PKG_v1.50/

Administrator@77e8d42fa25e4b2 /cygdrive/c/Andestech/project/MT7681_IoT_Package_v
1.50/Src/IoT_MT7681_PKG_v1.50
$ make b=1
```

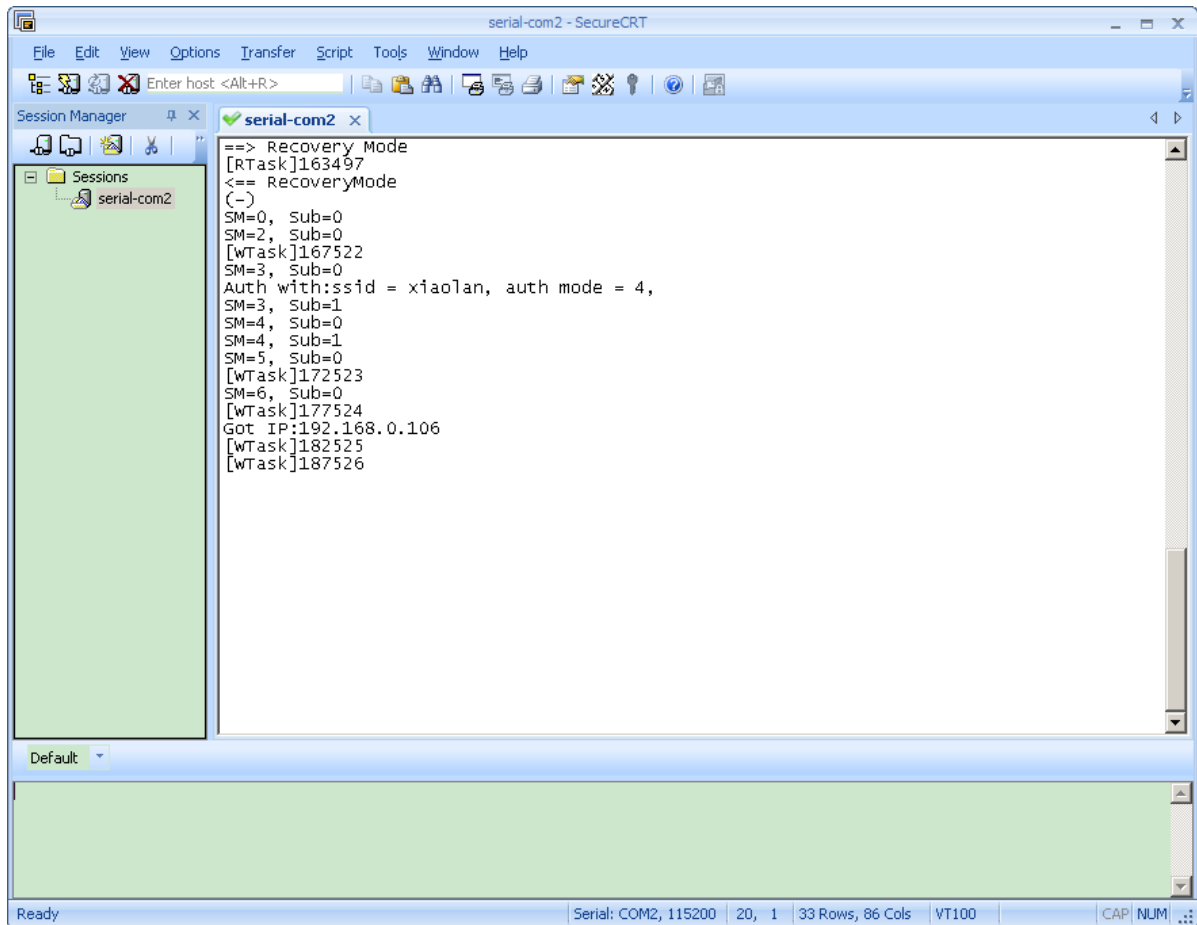
The directory of generated binary files is in



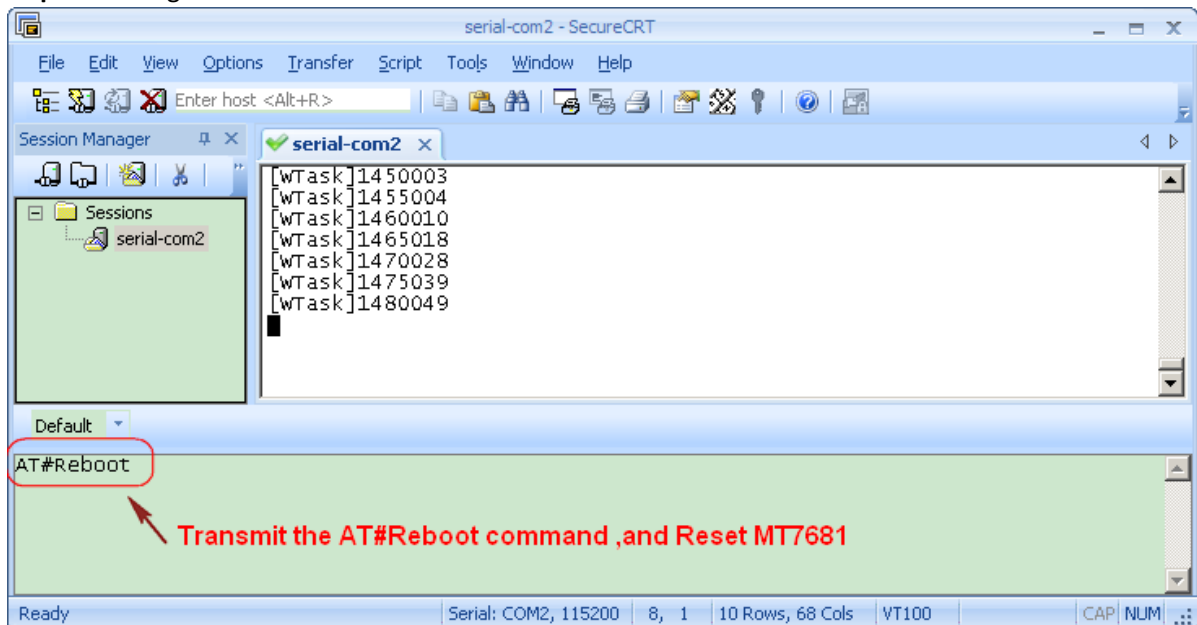
2.Upgrade firmware via UART

Step 1:Power on xWifi module .

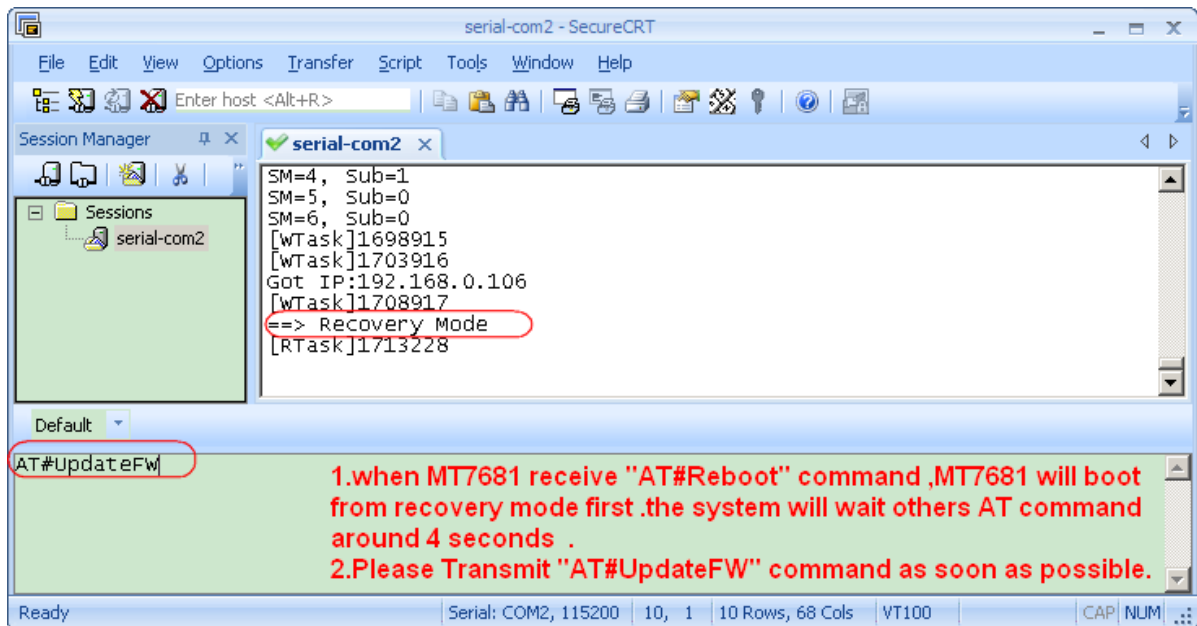
Step 2:Need a USB to UART Bridge to set up a connection between Computer and xWifi.



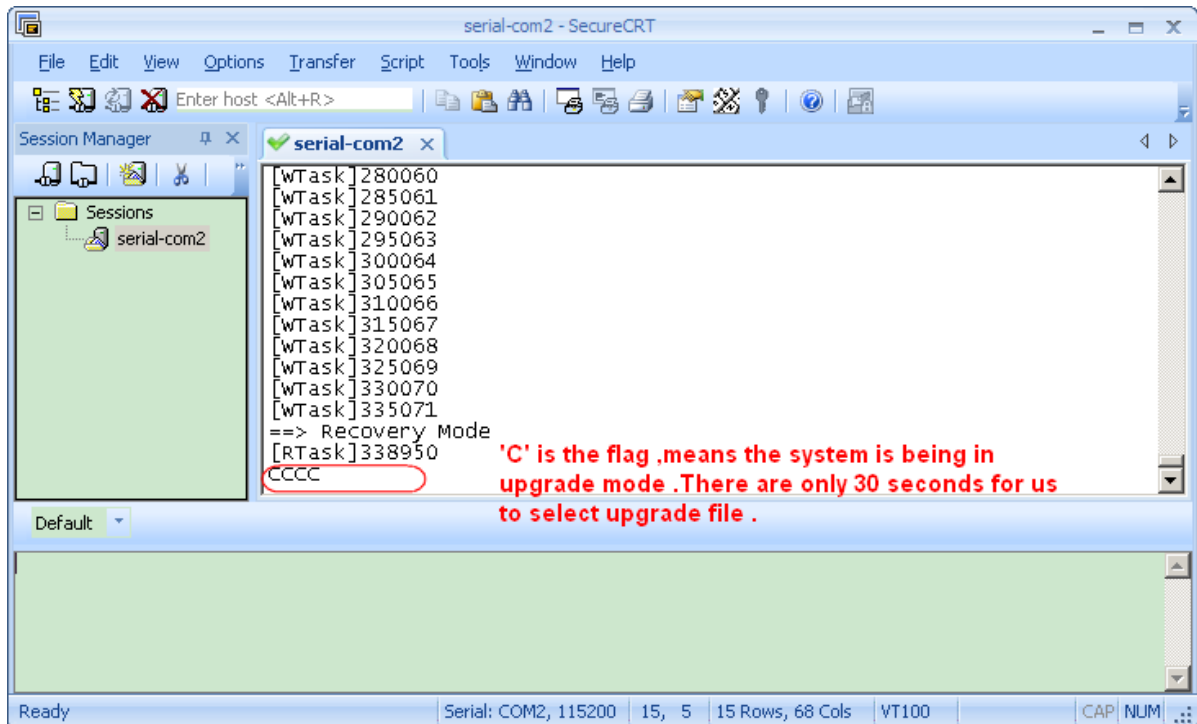
Step 3: Sending “AT#Reboot” command to reset MT7681

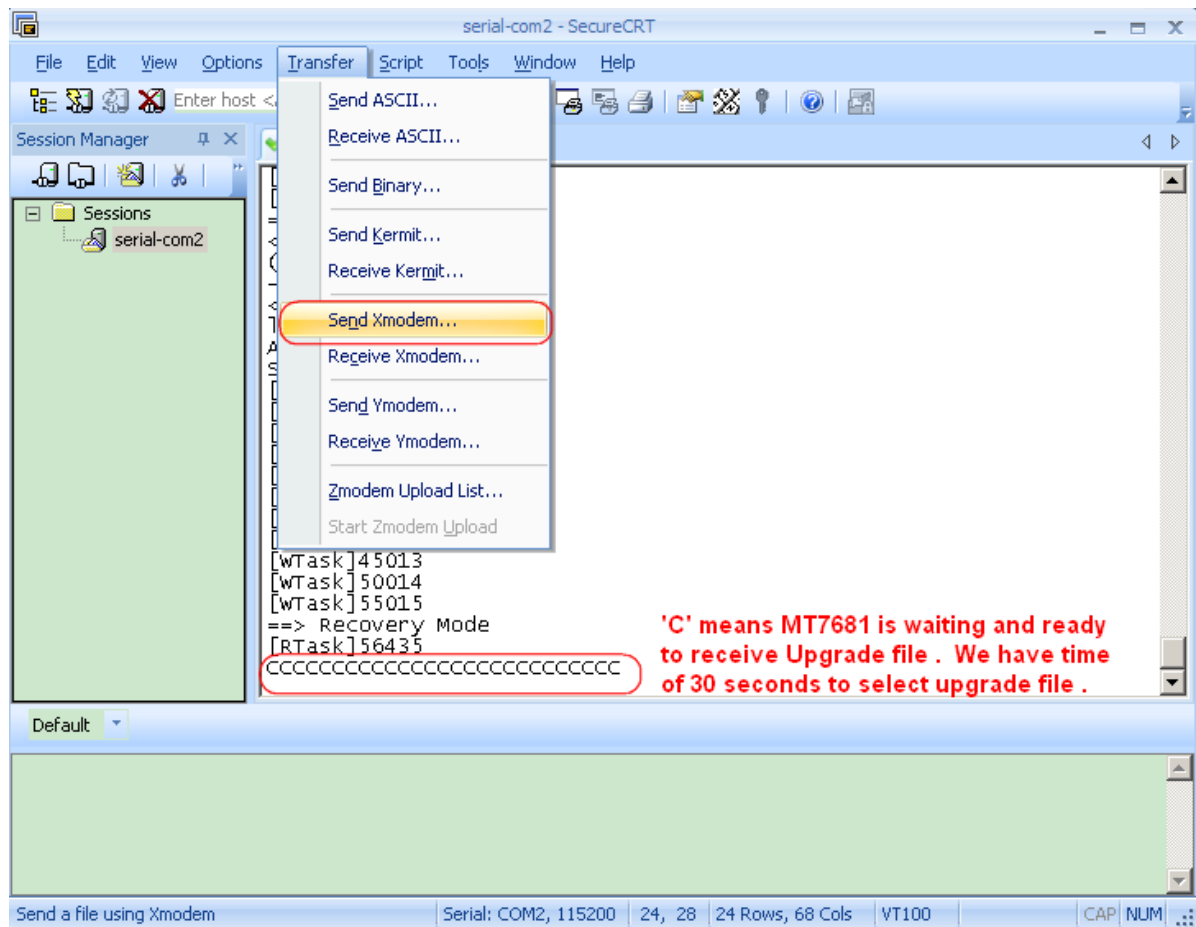


Step 4: Sending “AT#UpdateFW” command within 4 seconds



Step 5:Enter into upgrade mode



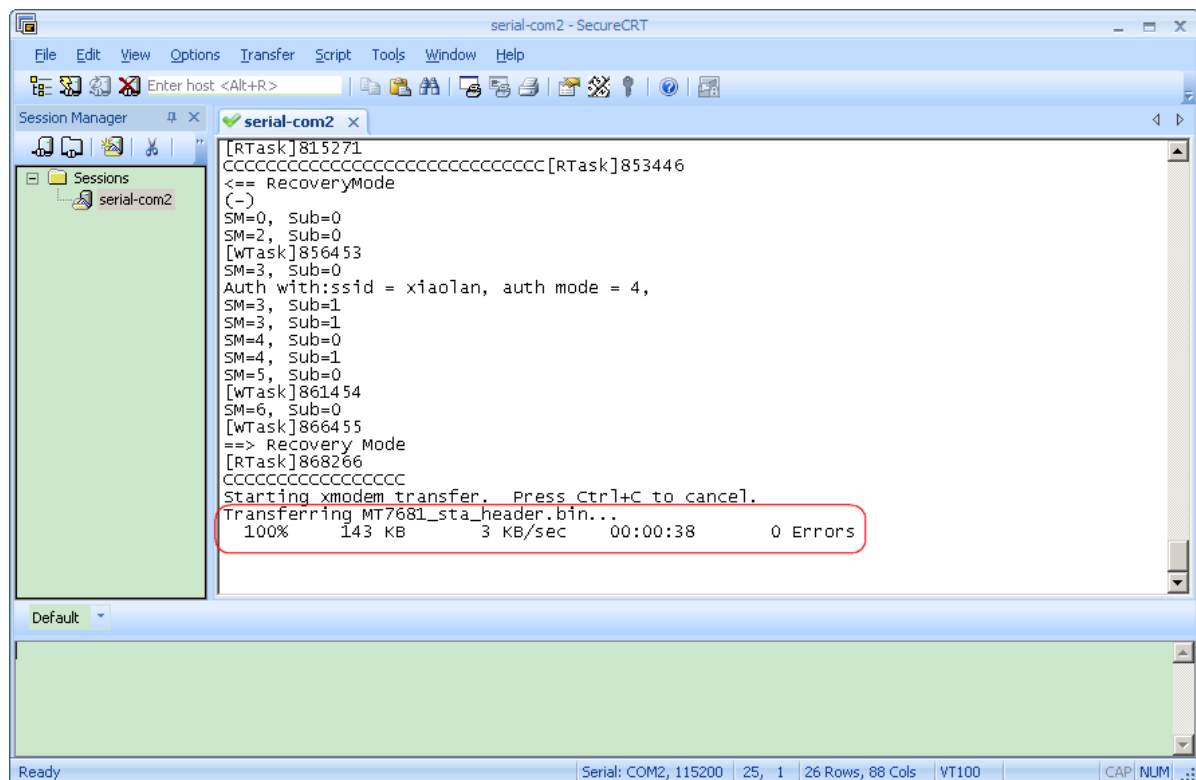
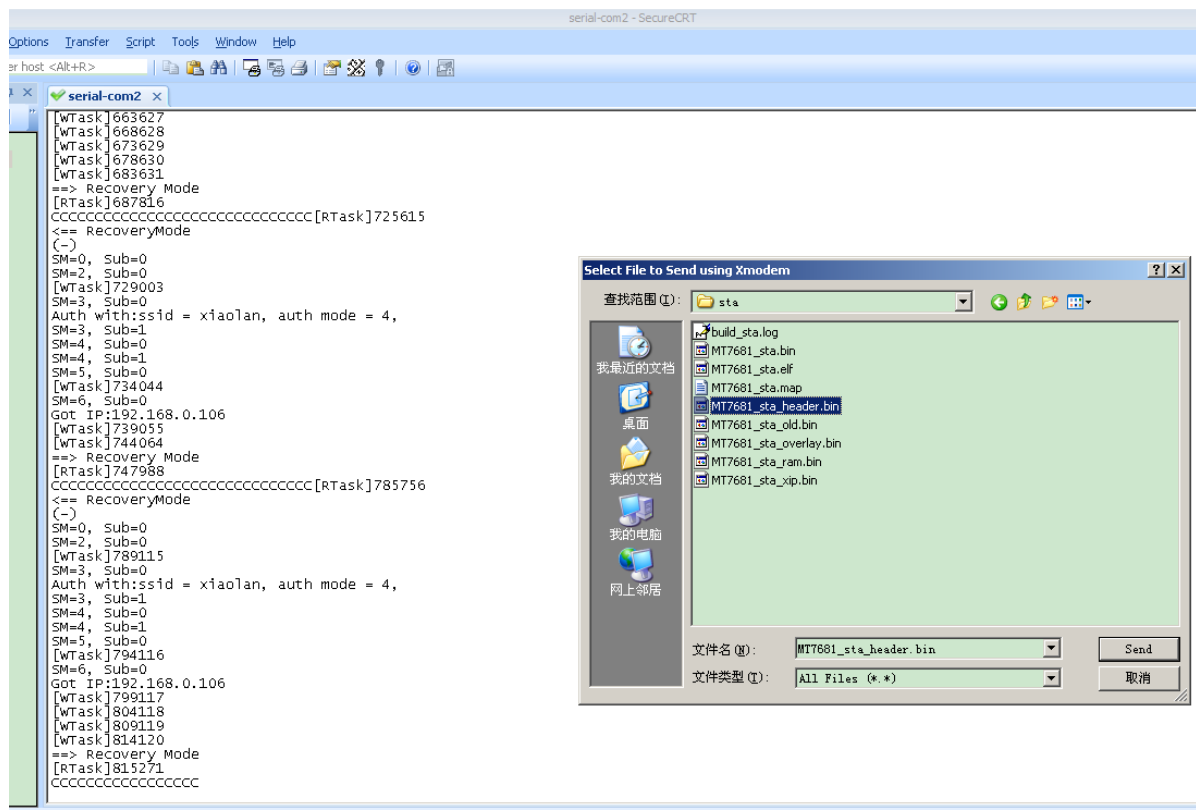


Step 6: Selecting the upgrade file .

The generated file include:

MT7681_sta_header.bin, MT7681_ap_header.bin, MT7681_recovery_header.bin.

Firmware marked with "****_header" which can be used as upgrading files via UART interface.



3.Switch working mode

3.1 Read BootIndex value of 0x18001 via AT#FLASH command:

AT#FLASH -r98305

[0x18001]=[0x00] //Boot as STA mode,

if [0x18001]=[0x01] //Boot as AP mode

AT#FLASH -s98305 -v1 //Switching working mode into AP mode .

AT#FLASH -s98305 -v0 //Switching working mode into STA mode .

3.2 Example:

Switching to AP Mode is simple,just modify the value of Flash Offset: 0x18001 to 1.

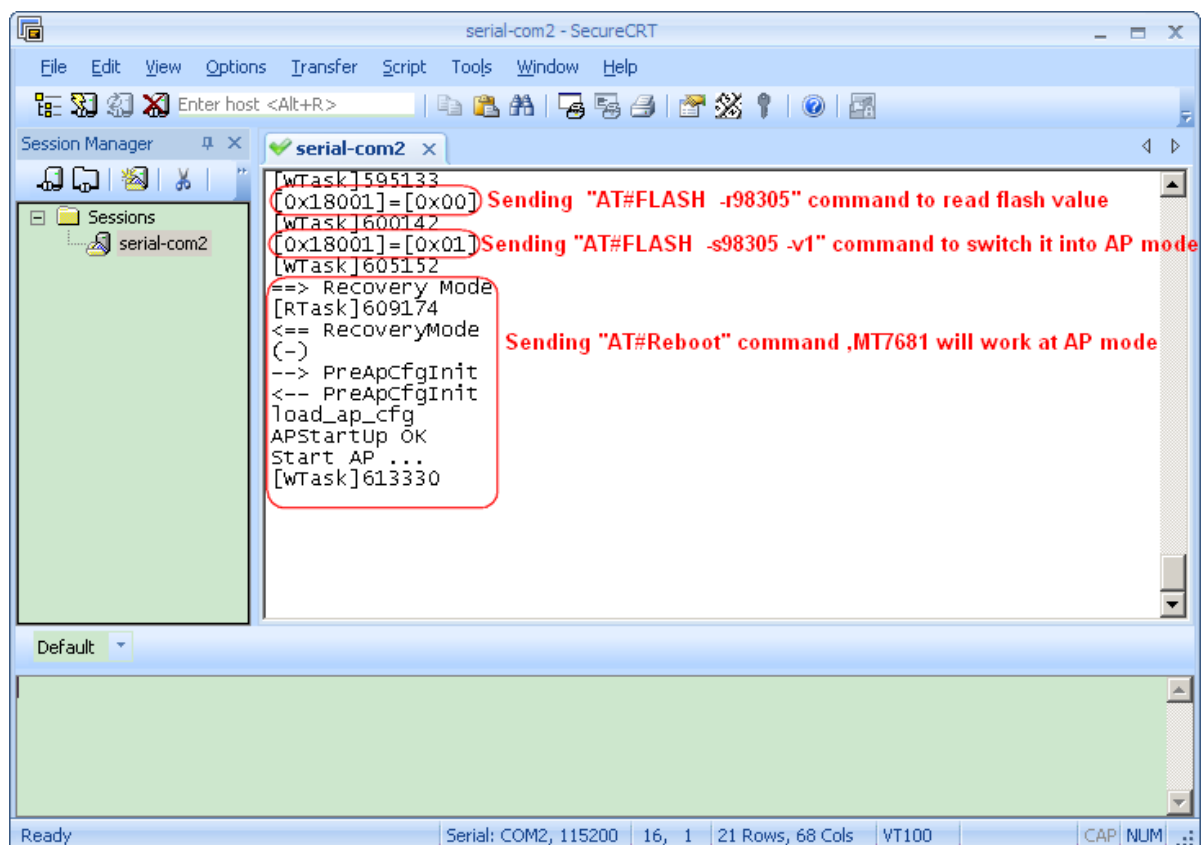
Step1:

Modify BootIndex value of 0x18001 via AT#FLASH command to 1:

AT#FLASH -s98305 -v1

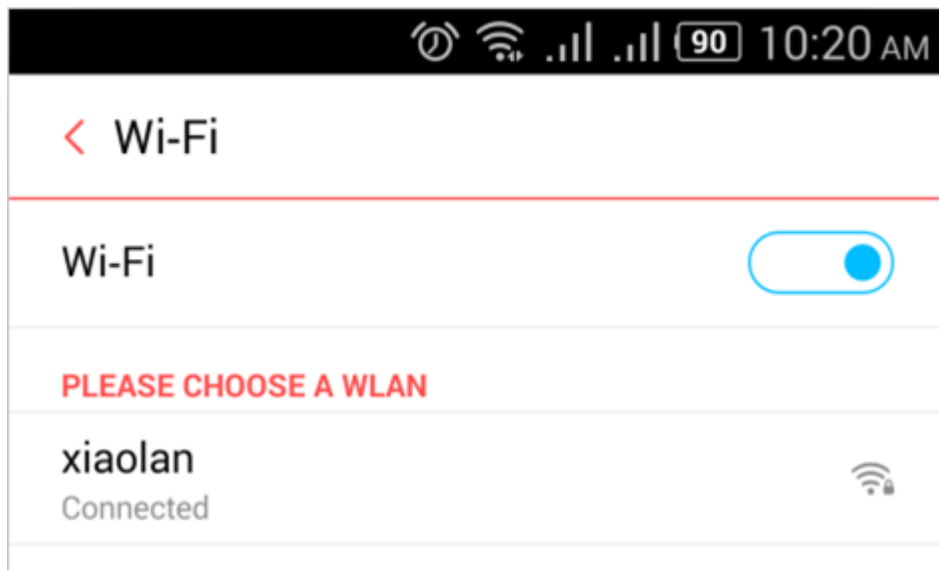
Step2:

Power on MT7681 again, it will boot in AP mode,

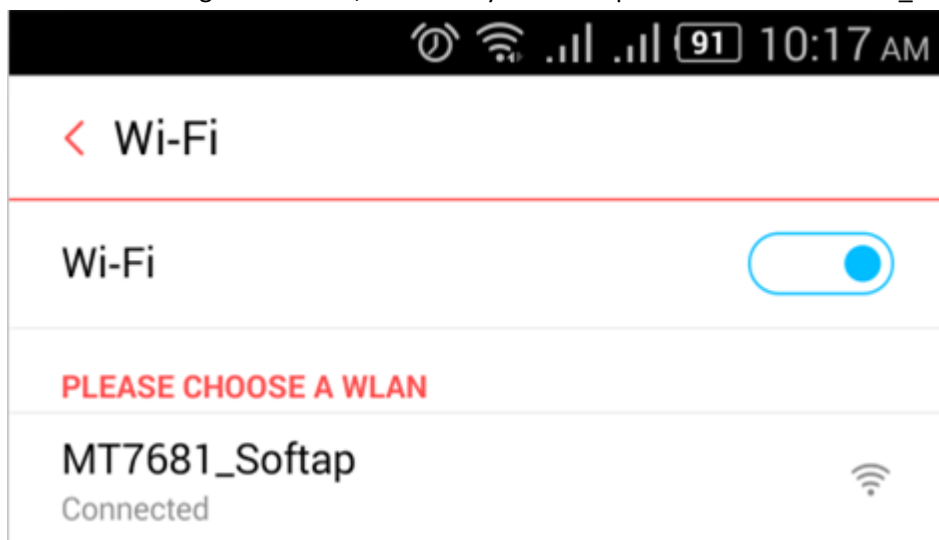


4.Application demo (Android)

4.1 If MT7681 is being in STA mode ,make sure your smart phone connect your wifi router (example : 'xiaolan' is my router's SSID).

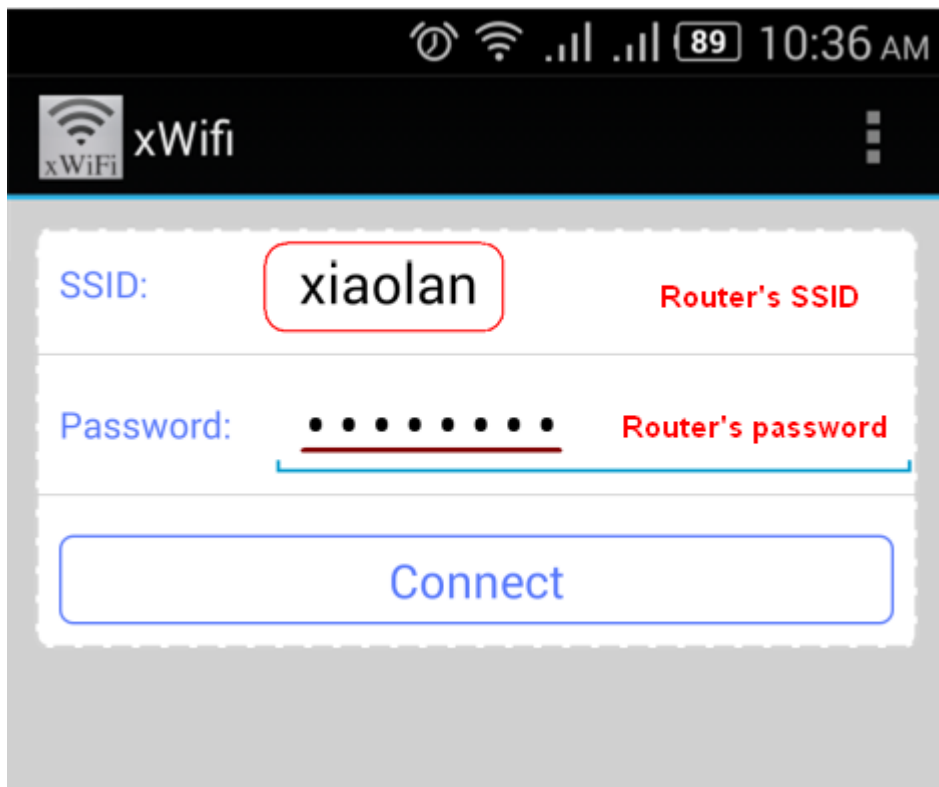


If MT7681 is being in AP mode ,make sure your smart phone connect “MT7681_Softap”.

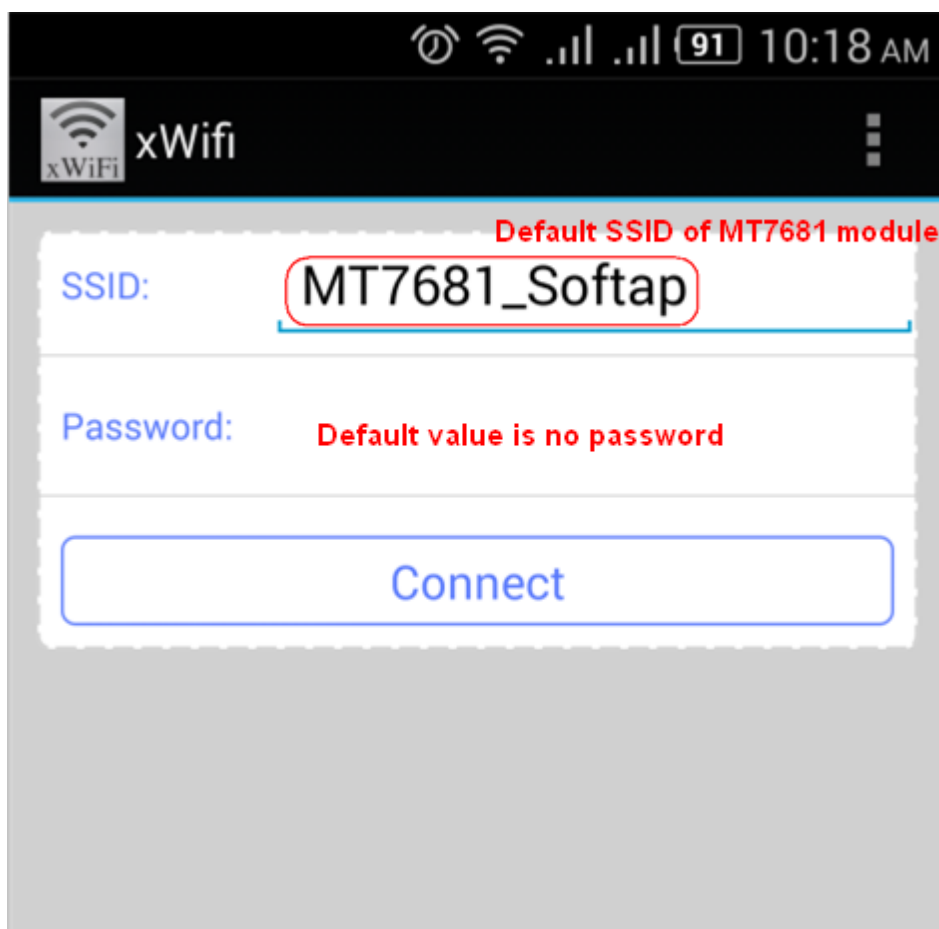


4.2 Install ‘xWifi_v1.0.apk’ (you can find it in APK directory of SDK)and open the app ,the screen below will show you first .

If MT7681 is being in STA mode:



If MT7681 is being in AP mode:



4.3 we have completed the follow functions:

a.Click 'Color Picker' ,the RGB-LED on base board will show the color you picked

b.Click the Light slide button ,The relay will be close or open with blue indicator on or off on base board .

c.Click 'Temp&Hum' ,trigger a measurement of Temperature and Humidity ,the result will show you immediately .

