WiFiMCU Tutorial

Basic

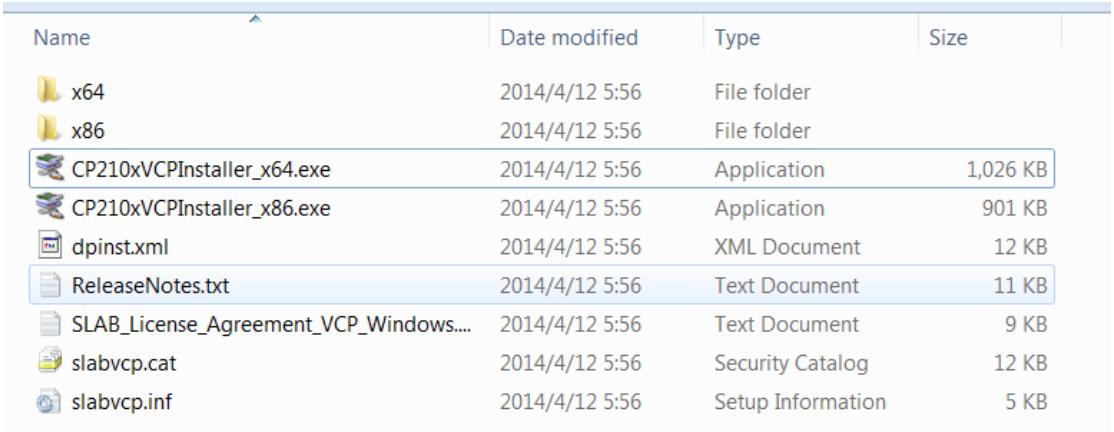
1, Install USB Driver

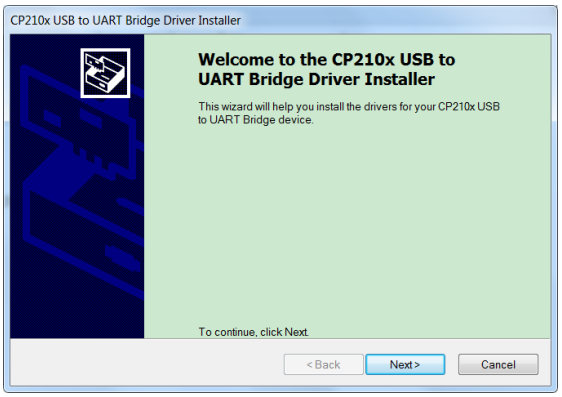
WiFiMCU uses CP2102 to converter USB data to UART TTL data. The USB to UART Bridge Virtual COM Port drivers are required for device operation. The latest driver can be found at:

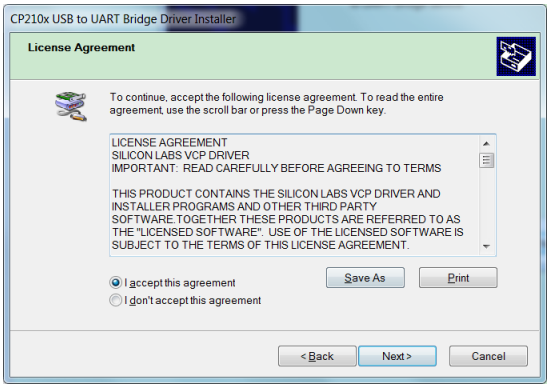
http://www.silabs.com/products/mcu/Pages/USBtoUARTBridgeVCPDrivers.aspx

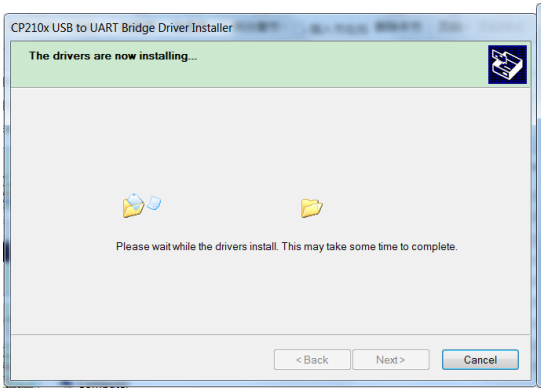
The following instructions are made in Windows 7 OS.

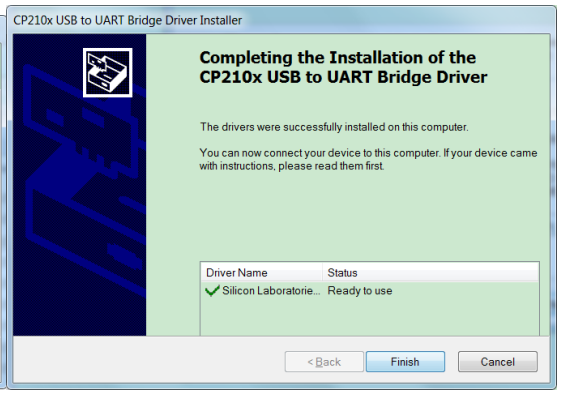
STEP 1 Install the USB Bridge VCP drivers. According to your Windows OS, choose "CP210xVCPInstaller_x86.exe" for 32 bit OS or "CP210xVCPInstaller_x64.exe" for 64 bit OS.







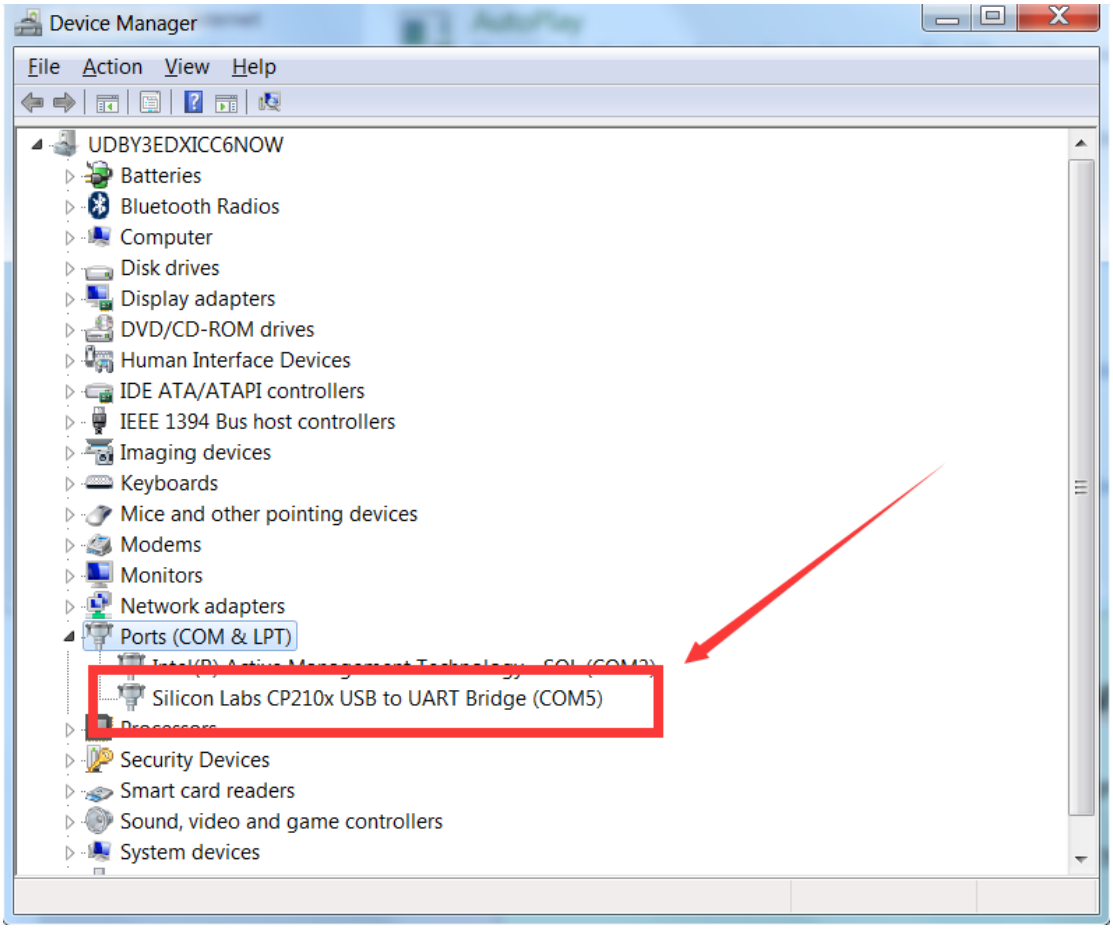




STEP 2 Check the Serial COM Port

Power up WiFiMCU with a Micro USB Cable. Go to “Start Memu”-> “Control Pannel”->”Device Manager”

If the Serial COM Port in “Ports(COM&LPT)” is shown as below, it means you have installed the driver successfully. Otherwise, change a Micro USB cable or a WiFiMCU to retry.



2, Quickly Start with WiFiMCU STUDIO

After install the USB to UART Bridge VCP driver. You can use WiFiMCU STUDIO to test WiFiMCU simplily and quickly. Here are the instructions to quickly start with WiFiMCU.

2.0 Prepare

WiFiMCU STUDIO is an open source develop tool for WiFiMCU. The latest executable program can be downloaded at:

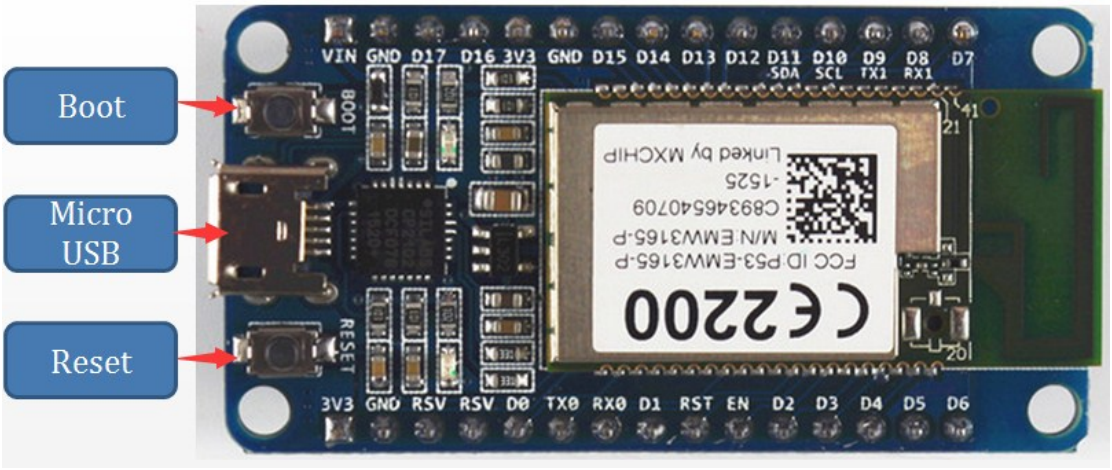
https://github.com/SmartArduino/WiFiMCU-STUDIO/tree/master/BIN

You can pull the source code from:

https://github.com/SmartArduino/WiFiMCU-STUDIO

2.1 Power Up

Power up the WiFiMCU with a Micro USB Cable.



Press “Reset” button to restart WiFiMCU.

Press “Boot” button while power up or press “Reset” button to enter into bootloader mode.

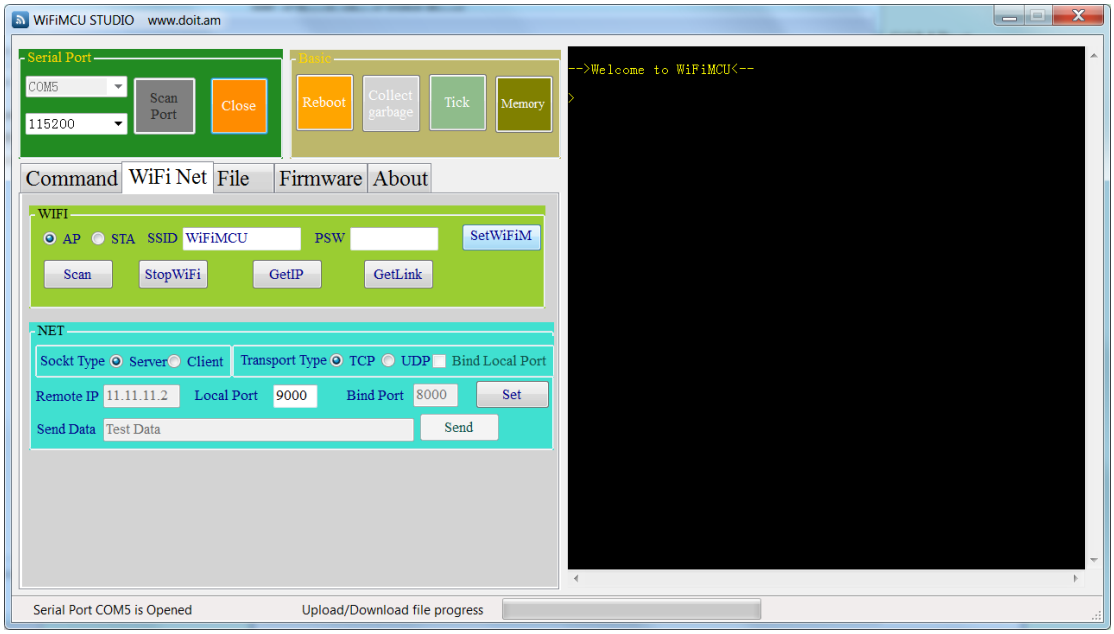
2.2 Check the COM Port

Go to “Start Memu”-> “Control Pannel”->”Device Manager” to check the Serial COM Port.

2.3 Run WiFiMCU STUDIO.

Click “Scan Port” button to scan the existing serial com ports in the computer.

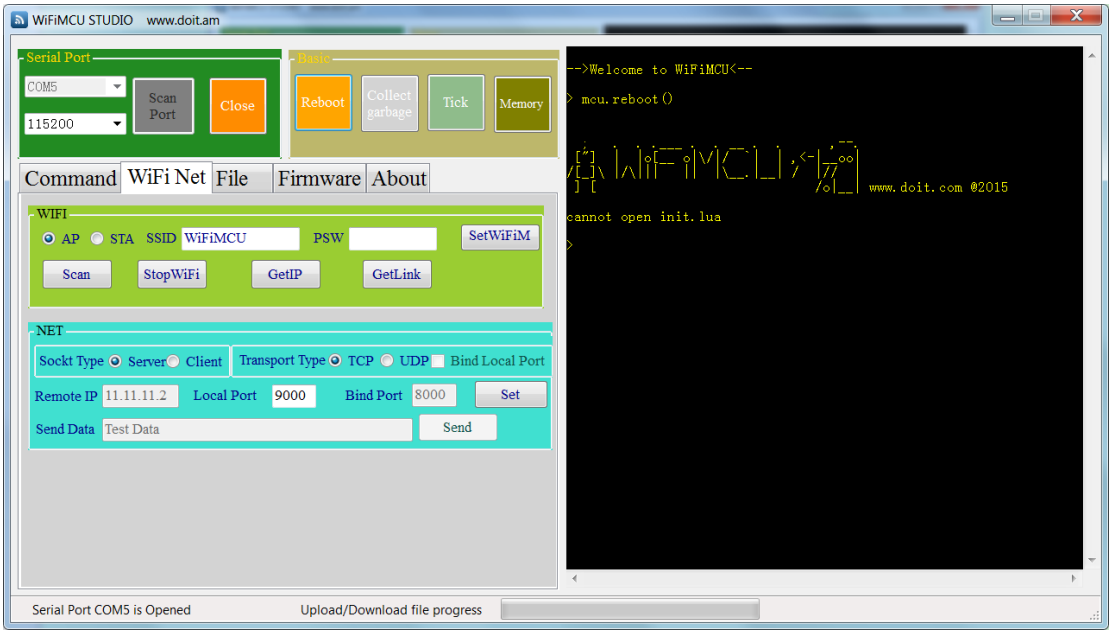
Make sure choose the right serial port of WiFiMCU, then click “Open” to open the serial com port. The default serial com parameters for WiFiMCU is 115299bps, 8 data bits, none check, one stop bit.



You can type a command in the right black textbox just like what your do in other serial tools such as “SecureCR”, “PuTTY”.

Type ”mcu.reboot()” in the command textbox and press enter, The commands are go into the Lua interpreter, and executed. The results will be shown in the textbox.

You will see like this:



You can also just click “Reboot” button to do the same thing.

Other common commands:

“collectgarbage()”, collect the garbage in Lua interpreter.

“=mcu.tick()”, get the current time tick of the MCU (ms) since startup.

“=mcu.mem()”, get the memory status.

All the commands for WiFiMCU can be found in the reference book:

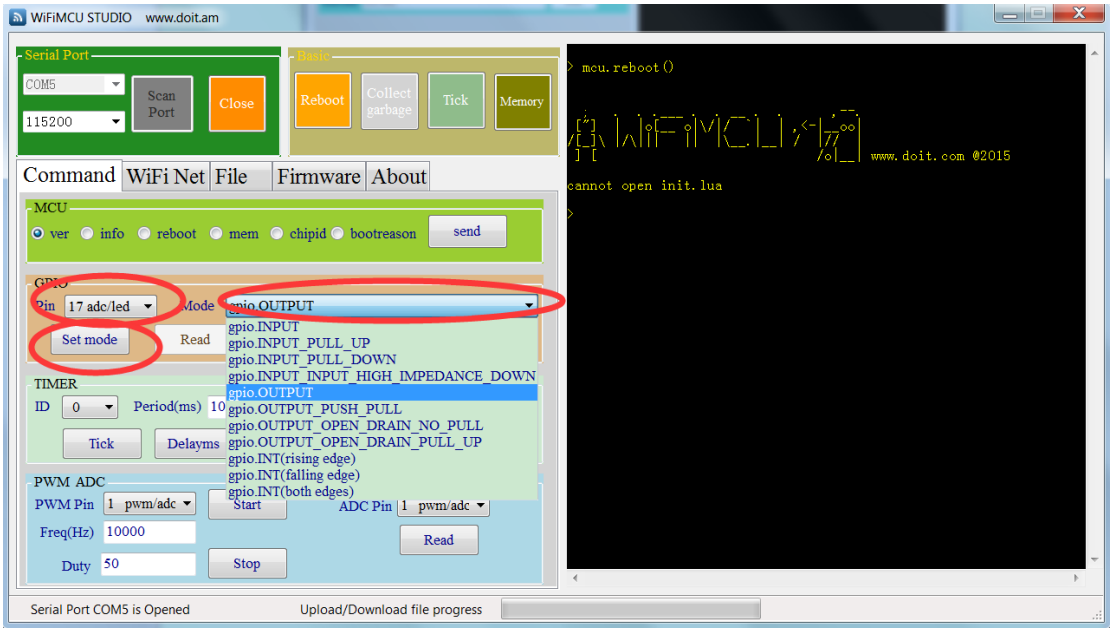
https://github.com/SmartArduino/WiFiMCU/tree/master/Document/

2.4 Toggle LED on WiFiMCU board

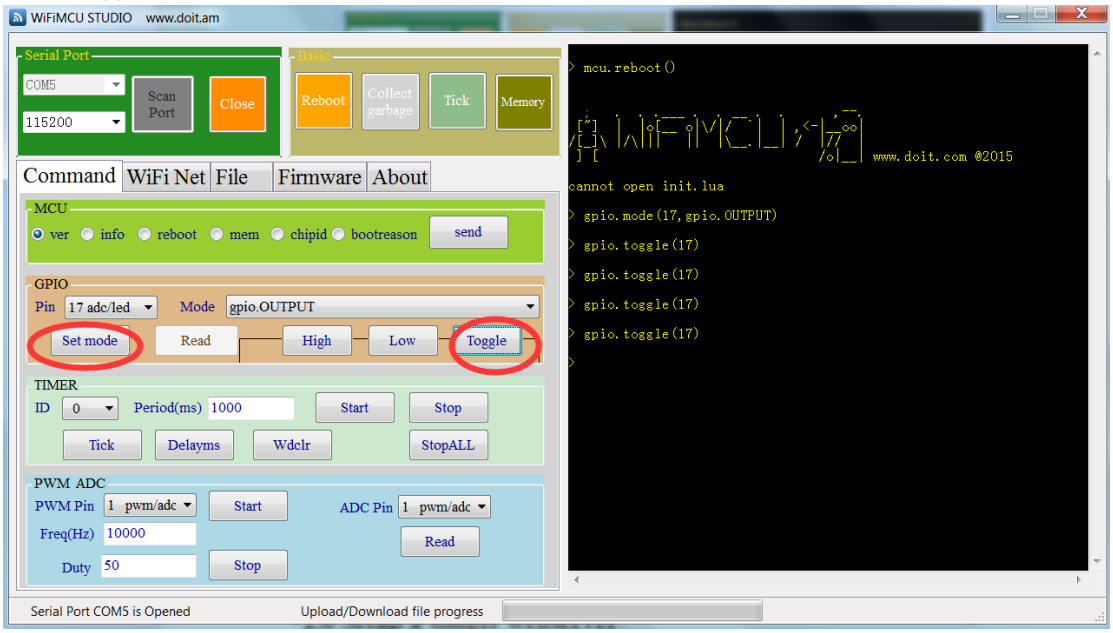
There is a LED connected to D17 on WiFiMCU board. You can toggle it very simply using WiFiMCU STUDIO.

STEP1, Switch to “Command” tabpage.

STEP2, Choose GPIO Pin “17 adc/led” in “GPIO” groupbox. Then choose “gpio.OUTPUT” in mode combox list.



STEP 3, click “Set mode”, command string “gpio.mode(17,gpio.OUTPUT)” will be sent to WiFiMCU and executed. The LED on WiFiMCU board will be lighted on or off after you click“Toggle” button.



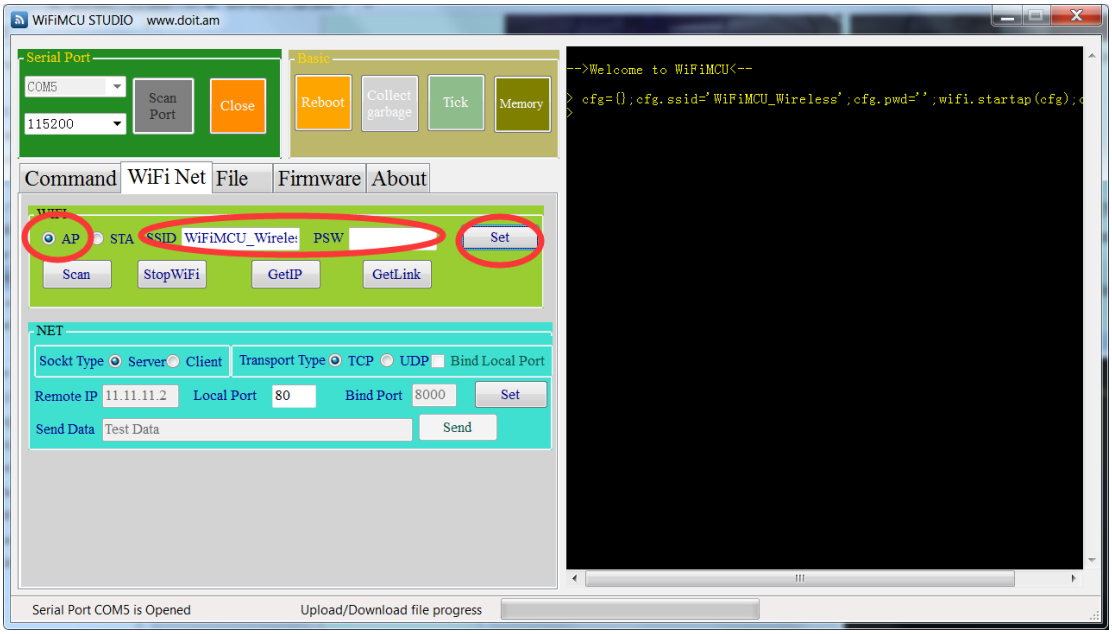
You can test the other gpio functions such as input/ interrup / pwm/ adc in the “Command” tabpage.

2.5 Start AP mode

The WiFi interface of WiFiMCU could work in either Access Point(AP), Station(STA), or AP+STA mode. The instructions below will setup a WiFi interface in AP mode for example.

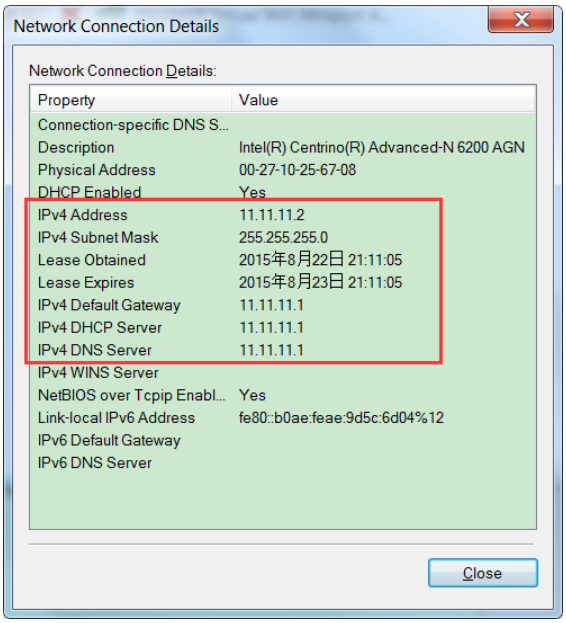
STEP 1, Switch to “WiFi Net” tabpage.

STEP 2, Choose “AP”, fill the SSID and PSW textbox. It will be “WiFiMCU_Wireless” for SSID and empty for PSW in default. Click “Set” to send the command string.



STEP 3, A WiFi interface with SSID: “WiFiMCU_Wireless” will be setup.

STEP 4, You can connected to the Open sercurity WiFi. The default IP for WiFiMCU is "11.11.11.1". The IP could be customized, check WiFiMCU Reference Book for more details.



2.6 Setup a simply webserver

WiFiMCU can be configured to support TCP/UDP Server and Client. It's very easy to setup sockets and connections. A simply webserver will be created by using Lua scripts in this section. We will use the WiFiMCU STUDIO to upload a Lua scripts file and run it.

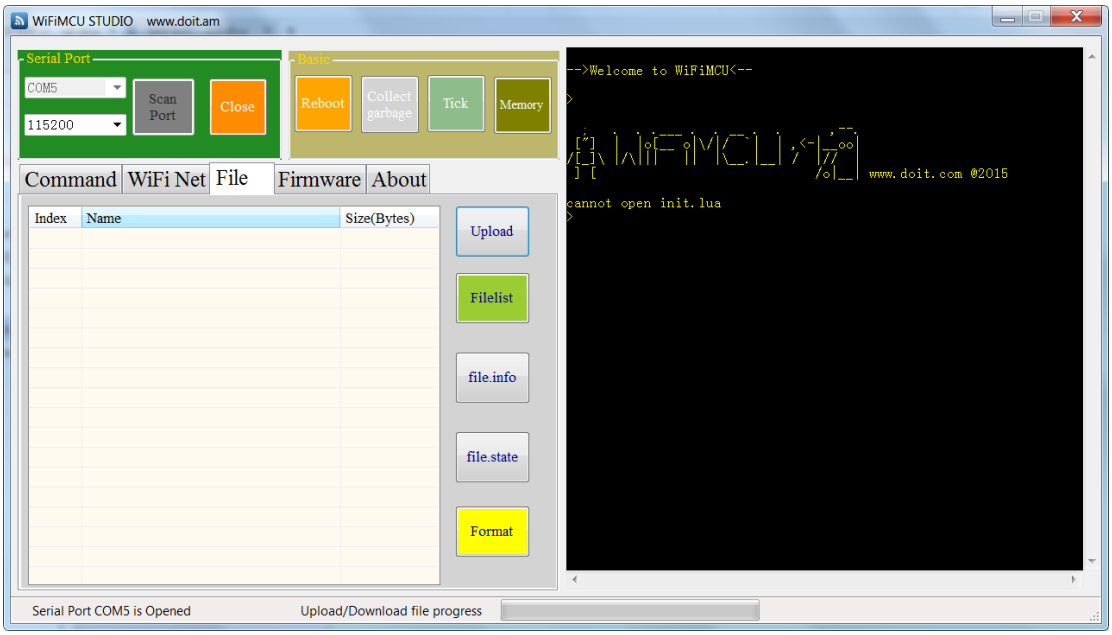
STEP 0, Make sure the WiFiMCU is running in AP mode or STA mode or AP+STA mode, if you are strange for this, follow the instructions in 2.5 section.

STEP 1, Save the Lua scripts below as “webserver.lua”.

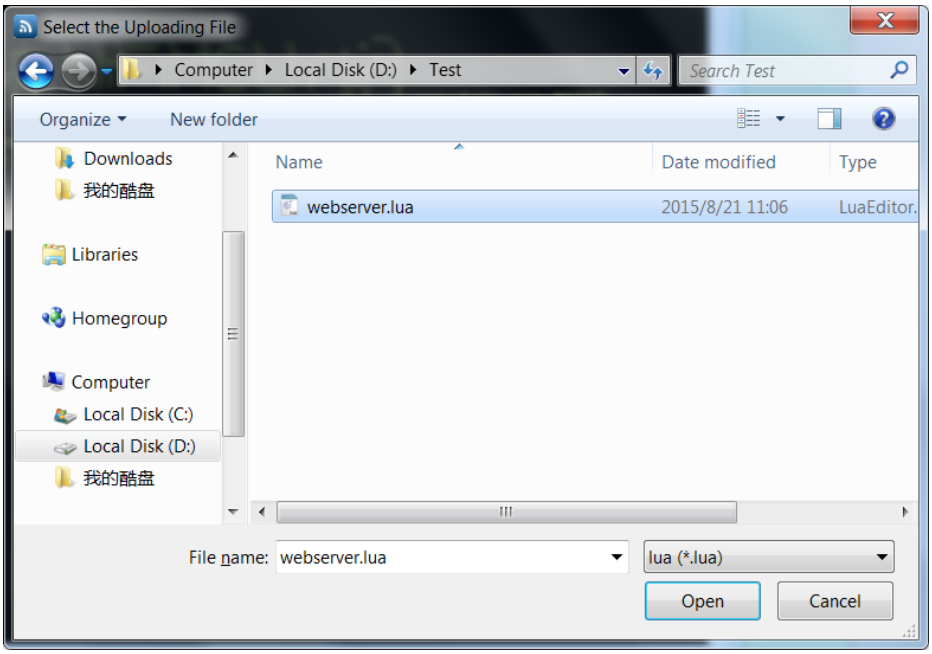
```
1. skt = net.new(net.TCP,net.SERVER)
2. net.on(skt,"accept",function(clt,ip,port)
3. print("accept ip:"..ip.." port:"..port.." clt:"..clt)
4. net.send(clt,[[HTTP/1.1 200 OK
5. Server: WiFiMCU
6. Content-Type:text/html
7. Content-Length: 28
8. Connection: close
9.
10.
11. <h1>Welcome to WiFiMCU!</h1>]])
12. end)
13. net.start(skt,80)
```

webserver.lua

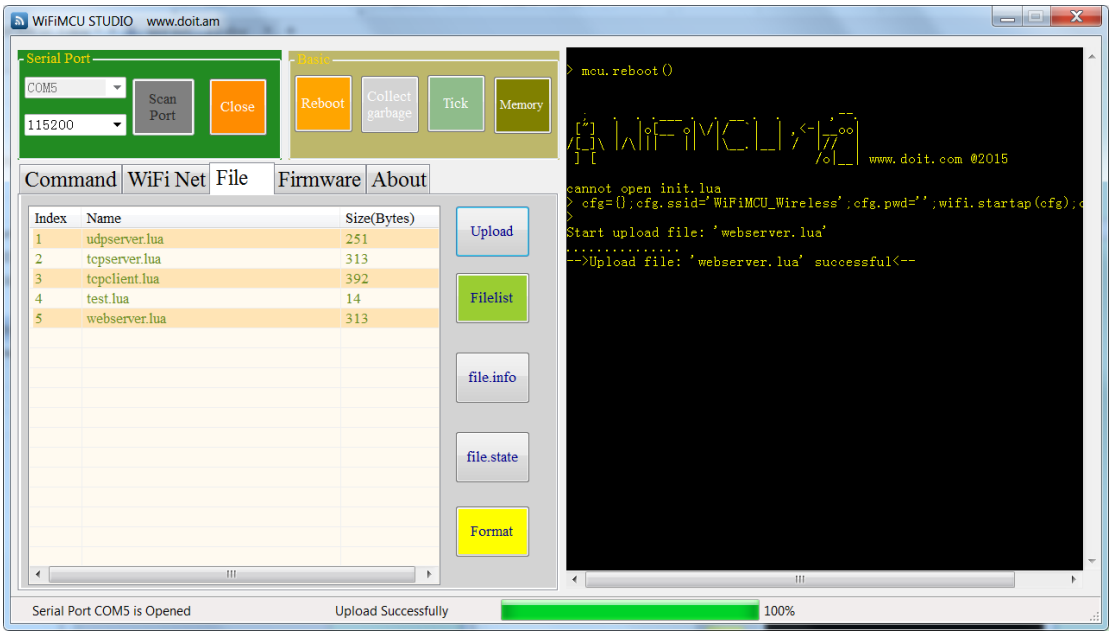
STEP 2, Switch to “File” tabpage.



STEP 3, Click “Upload” button to choose “webserver.lua”.

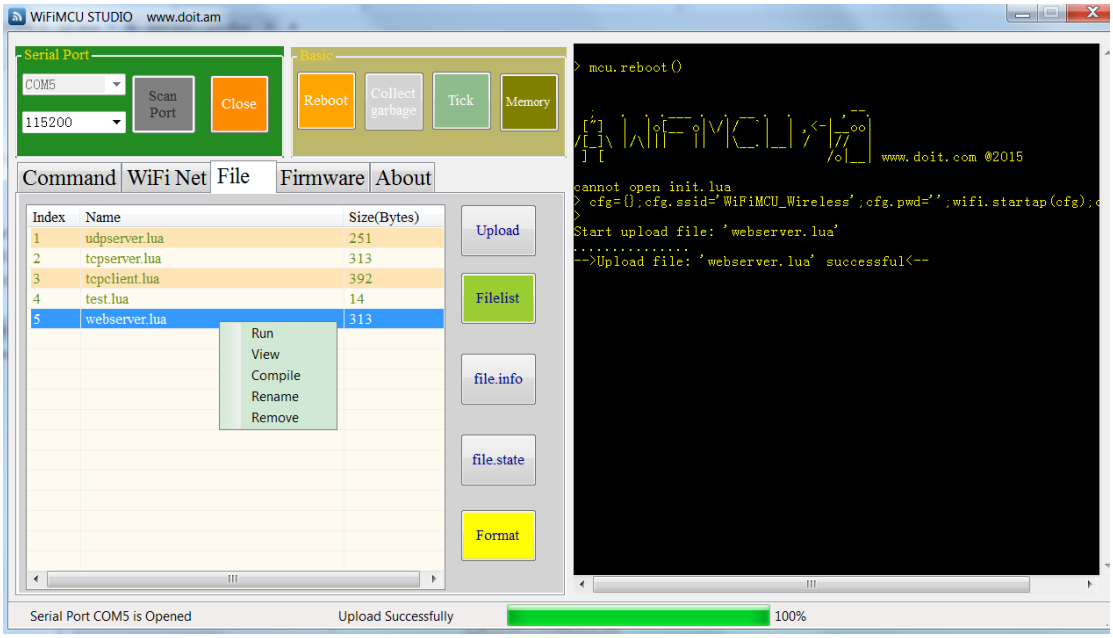


STEP 4, The uploading procedure will be started automatically. If uploading successfully, the files stored in WiFiMCU will be listed in the listbox.



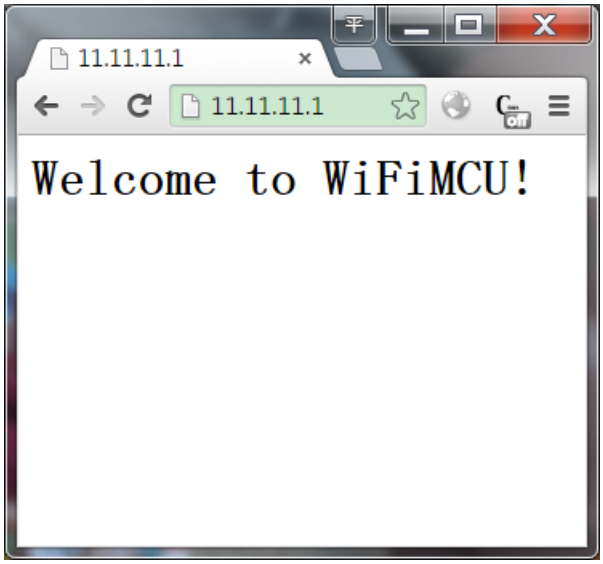
STEP 5, Choose “webserver.lua” in the listbox. Right Click the mouse, a submenu list will be shown. Click “Run” to run the script. Command string “dofile('webserver.lua')” will be sent to Lua interpreter.

You can test the other operations for the selected file freely.



STEP 6, Use a PC or a smart Phone connect to the AP that WiFiMCU made. For example, such as “WiFiMCU_Wireless” we made in section 2.5.

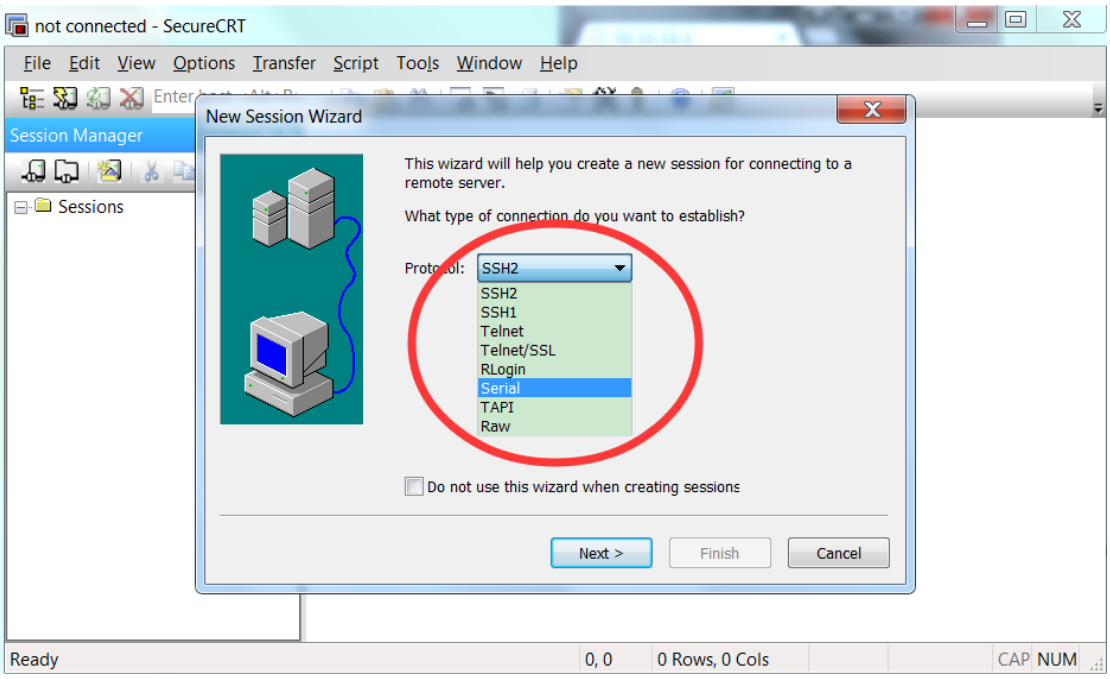
STEP 7, Open a browser, and type ”11.11.11.1” in the address field. You will get:

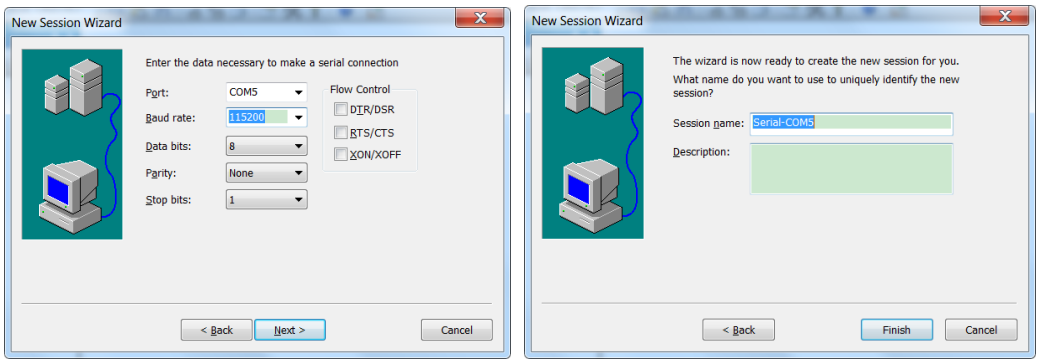


3, Use SecureCRT (Optional)

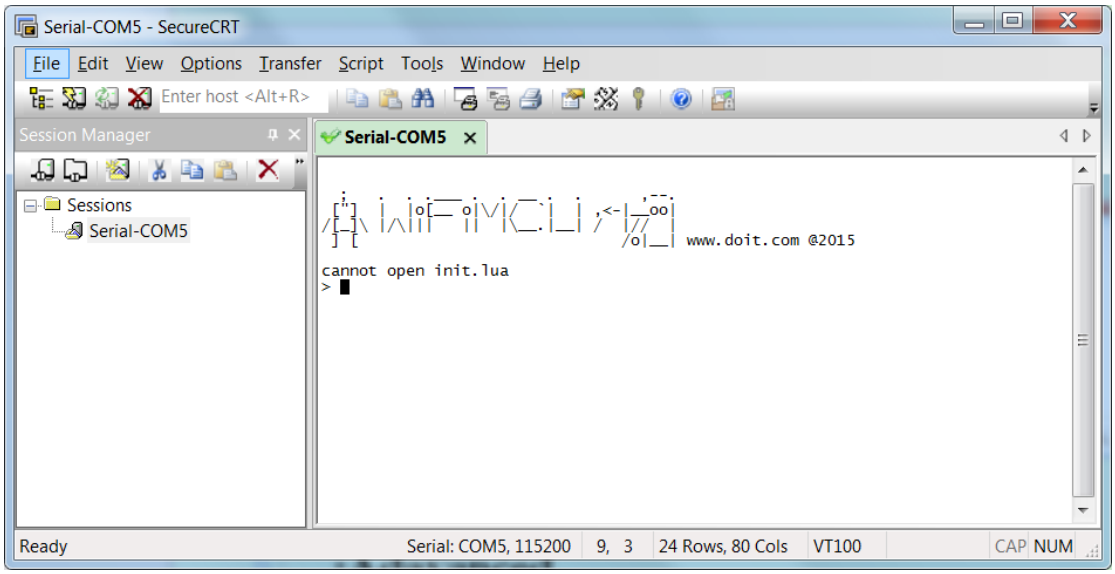
You can use any serial port tools to interact with WiFiMCU. Here is a simply instruction to use SecureCRT for example.

STEP 1, Setup a new session. Set the parameters as: 115200, n, 8, 1.

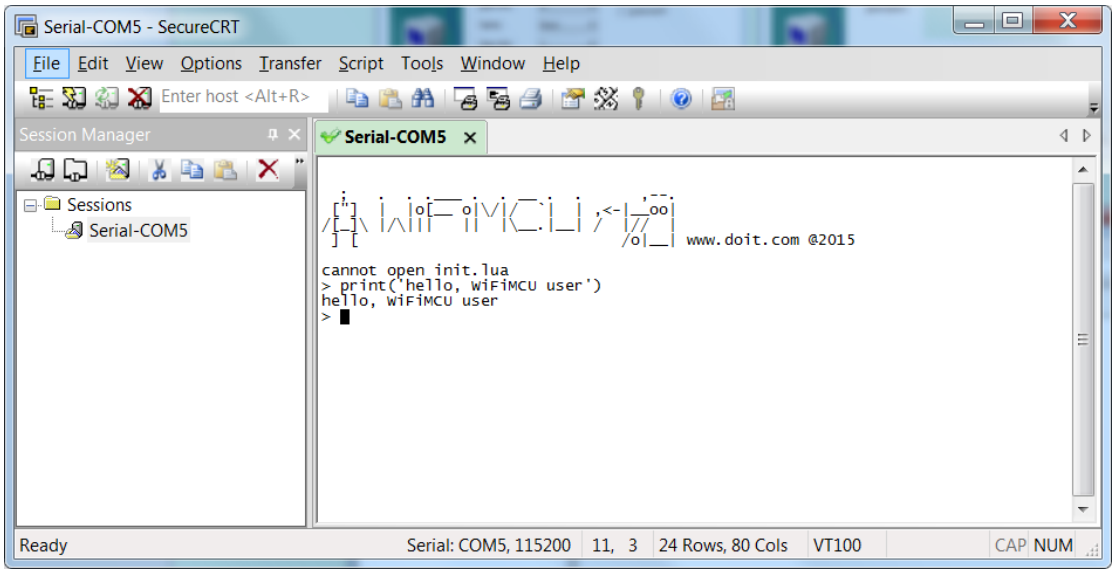




STEP 2, Connect WiFiMCU with a Micro USB, then open the session.



STEP 3, Type commands in the interface. Enjoy it.



Adavanced

1 Flash LED -use TIMER module

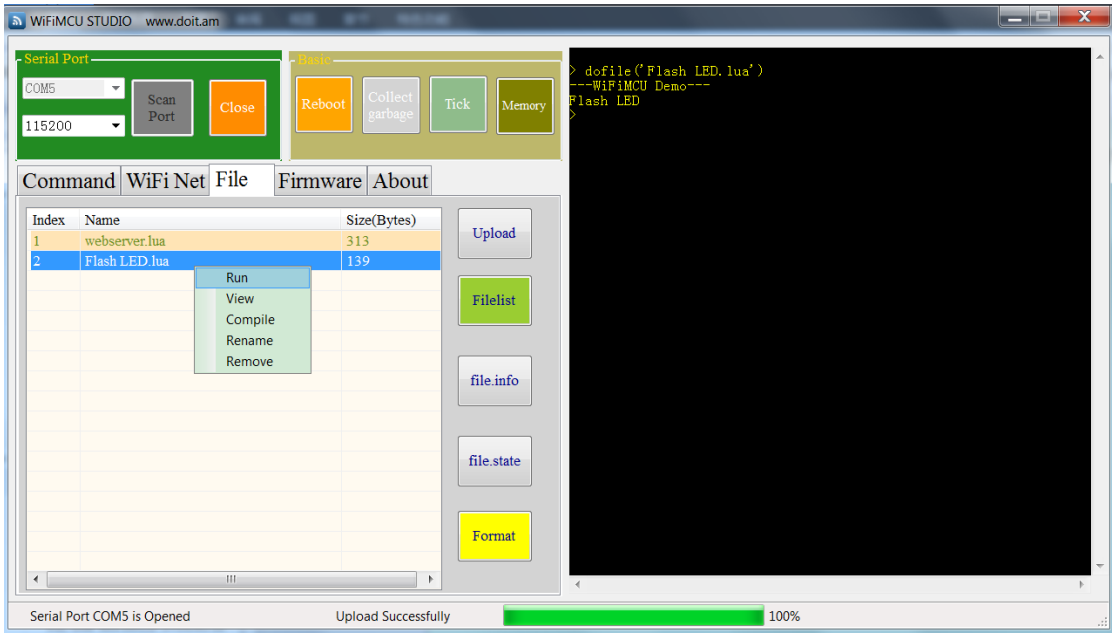
A LED is flashed every 500ms in this example. The LED on WiFiMCU is used. Timer0 is set to toggle gpio17 in the Lua scripts.

STEP 1, Save the Lua scripts below as “Flash_LED.lua”.

```
1. print("---WiFiMCU Demo---")
2. print("Flash LED")
3. pin = 17
4. gpio.mode(pin,gpio.OUTPUT)
5. tmr.start(0,500,function()
6.   gpio.toggle(pin)
7. end)
```

Flash LED.lua

STEP2, Upload the scripts via WiFiMCU STUDIO, Run this script.



STEP 3, The LED on WiFiMCU board will be flashing every 500ms.

2 Breathing LED -use PWM module

A breathing LED could be easily setup by the PWM function on the WiFiMCU board.

STEP 1, Find a LED with dupont lines. Connect the anode to D1(or whatever pin that supports PWM function) while the cathnode to a GND pin on WiFiMCU board.

STEP 2, Save the Lua scripts below as “Breathing_LED.lua”.

```
print("---WiFiMCU Demo---")
print("Breathing LED")
pin = 1
freq =10000
duty =0
dir = 1
tmr.start(1,50,function()
  if dir ==1 then
        duty = duty + 5
      if duty>100 then
            duty =100
            dir = 0
      end
else
      duty = duty - 5
      if duty< 0 then
            duty = 0
            dir = 1
      end
 end
 pwm.start(pin,freq,duty)
end)
```

STEP 3, Upload the scripts via WiFiMCU STUDIO, Run this script.


3 Socket programming –use Net module

Please refer to github:
https://github.com/SmartArduino/WiFiMCU/tree/master/Document/demos/5%20net

4 WiFi to Serial transparent transmission

Please refer to github:
https://github.com/SmartArduino/WiFiMCU/blob/master/Document/demos/9%20uart/2%20uart_2_wifi_trans.lua

5 Update Firmware

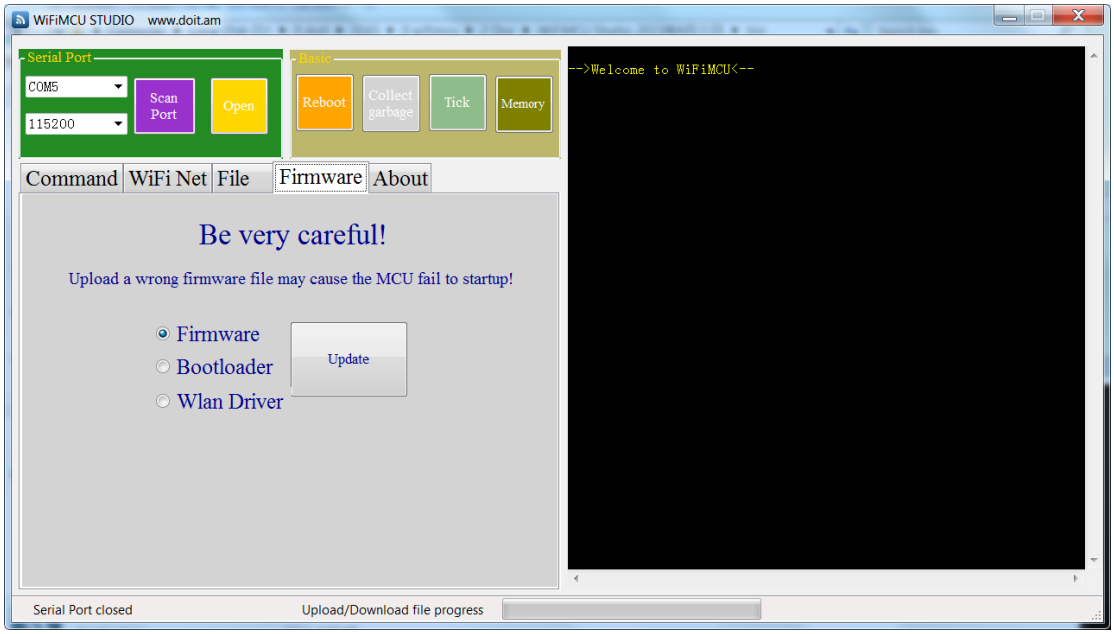
The firmware, bootloader, or the WLAN driver for WiFiMCU can be updated. There are two ways: Using a serial port with Y modem protocols, Using SWD programmer.

5.1, Get the latest firmware

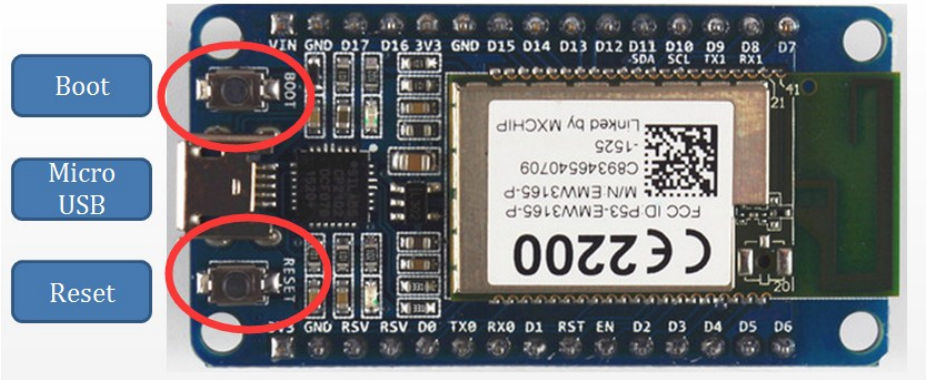
You can get the latest firmware, bootloader, or WLAN driver at:
https://github.com/SmartArduino/WiFiMCU/releases
Otherwise, You can download the source code and recompile your own firmware:
https://github.com/SmartArduino/WiFiMCU
The IDE of source code is IAR Embedded Workbench V7.20

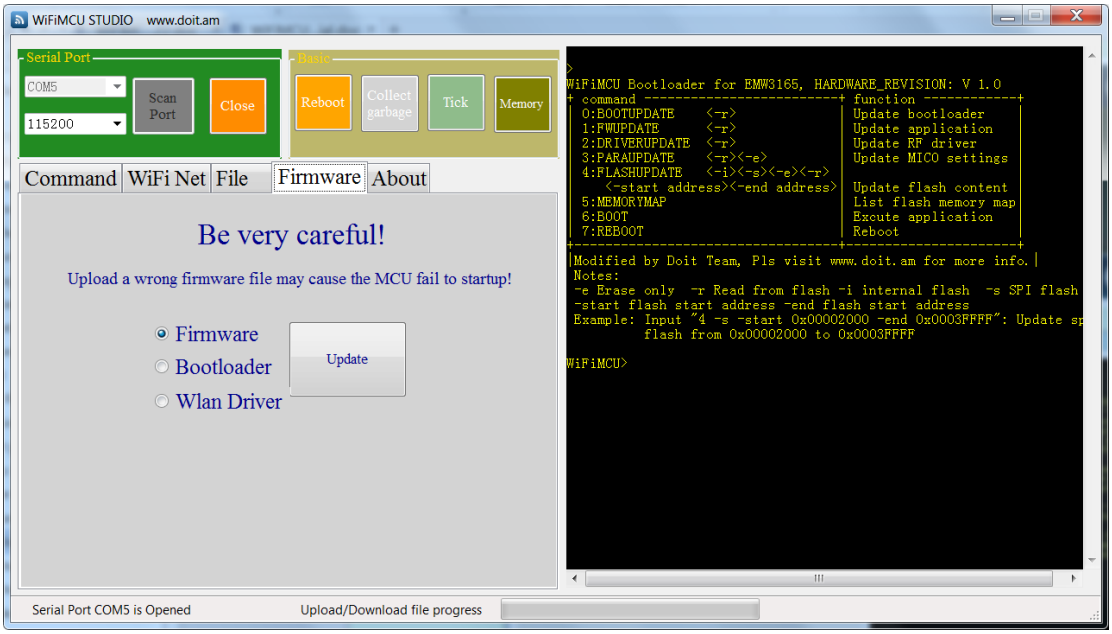
5.2, Use WiFiMCU STUDIO to update firmware

STEP 1, Open WiFiMCU STUDIO and connect to WiFiMCU.
STEP 2, Switch to “Firmware” tabpage.

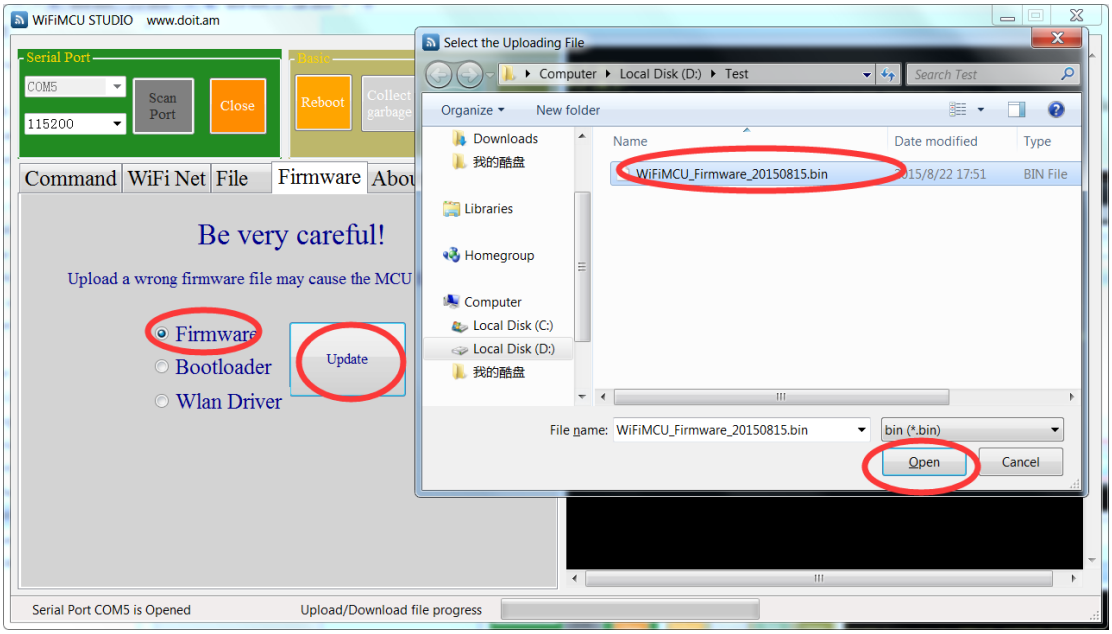


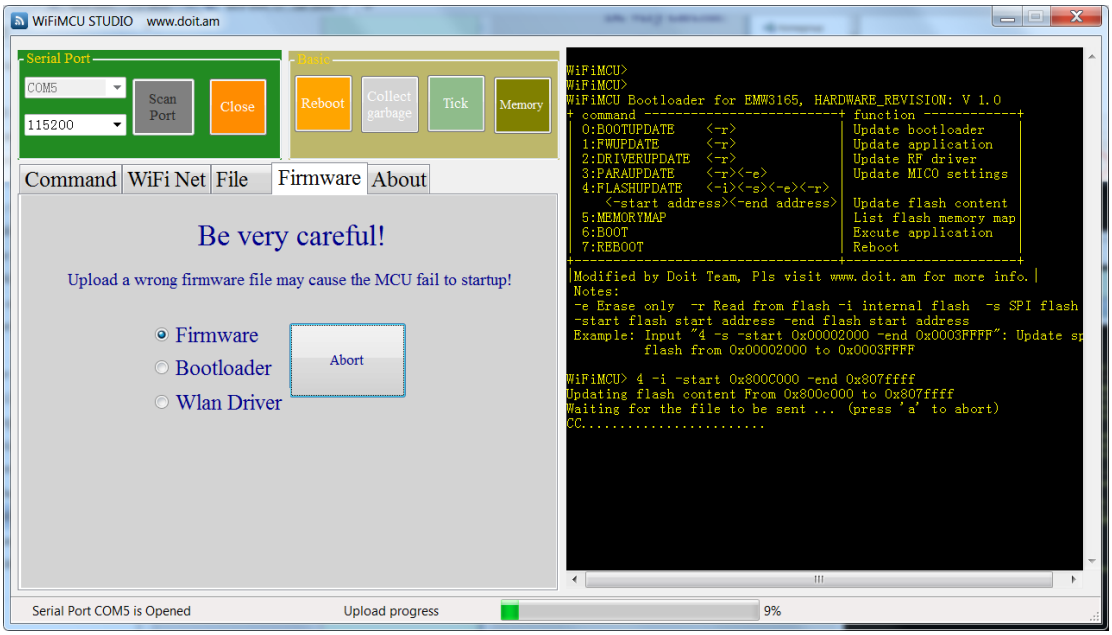
STEP 3, On the WiFiMCU board, Press"Boot" button while press "Reset" button in order to enter into bootloader mode.



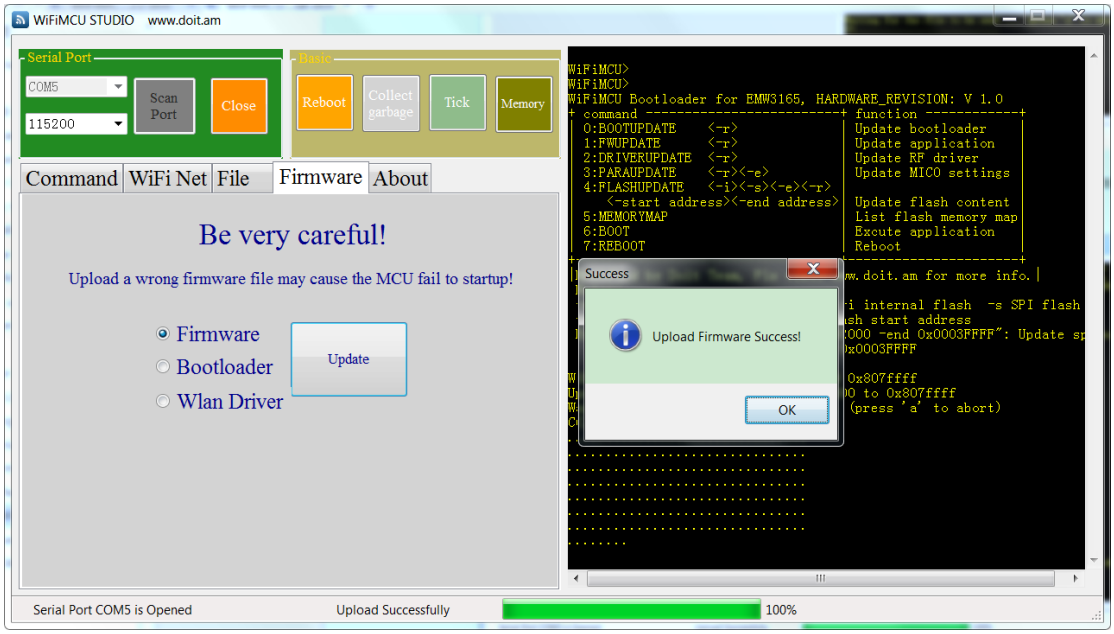


STEP 4, Choose upload type :“Firmware”, “Bootloader”, “Wlan Driver”, and click “Update” to choose a binary file. The update procedure will started automatically.

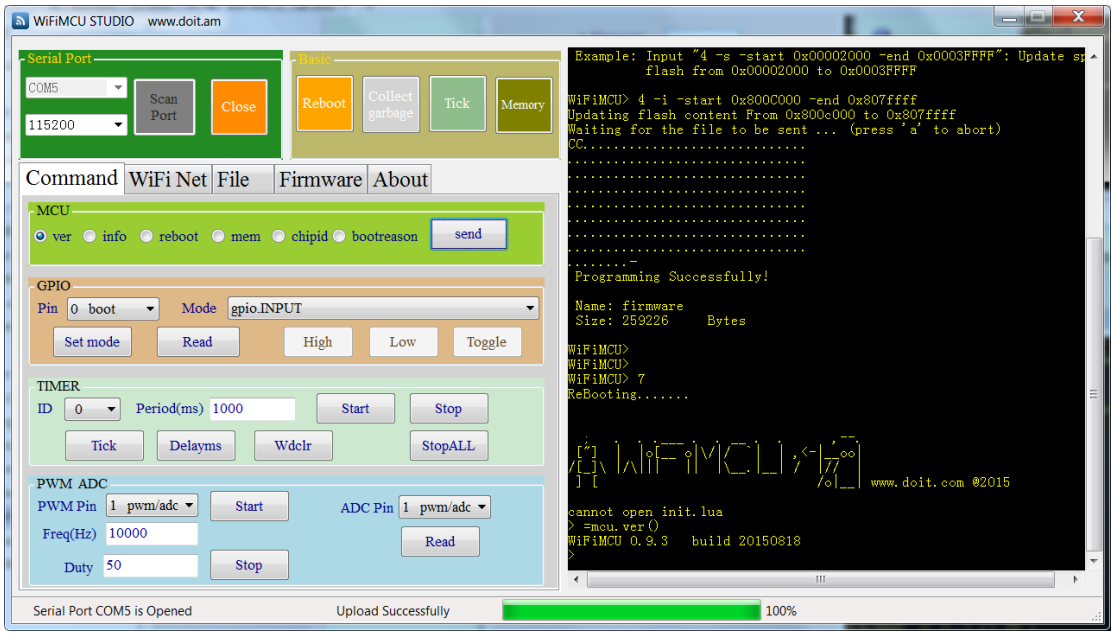




STEP 5, Wait for finishing. You can “Abort” the update. Be careful, if updating is failed or Abort by user, the WiFiMCU may fail to startup.



STEP 6, Click “Reboot” button in WiFiMCU STUDIO to activate new firmware. Type command string ”=mcu.ver()” to check the firmware version.

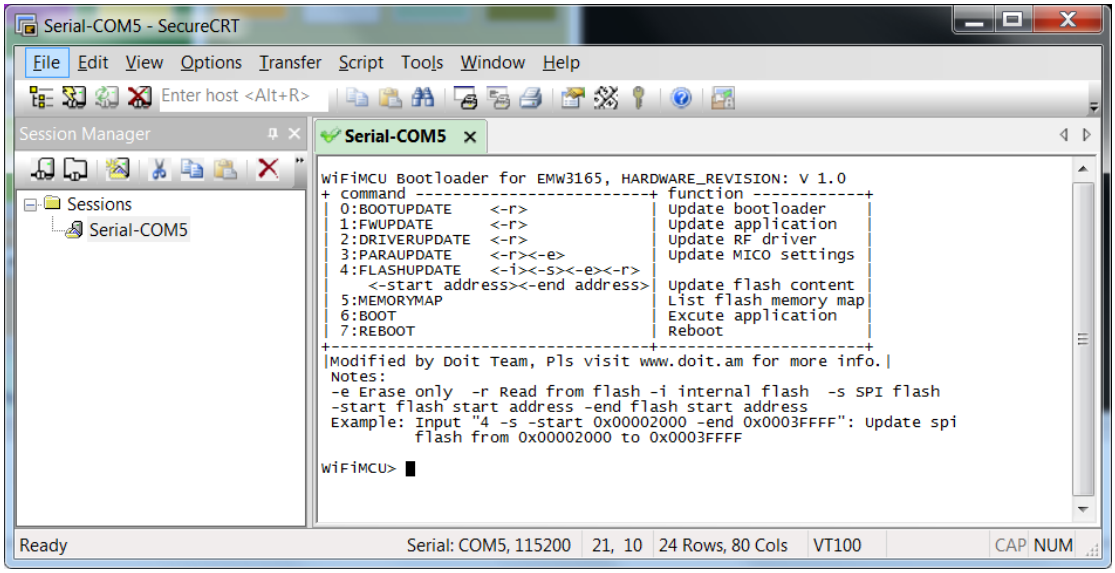


5.3, Use SecureCRT to update firmware

Beside WiFiMCU STUDIO, the firmware, bootloader and WLAN driver can be updated using a serial terminal tool which supports Y modem transmission protocol. The updated method with SecureCRT is taken as an example in this section.

STEP 1, Setup a serial interface connection with WiFiMCU just as what we have done at “Basic”->”3, UseSecureCRT(Optional)”.

STEP 2, On the WiFiMCU board, Press”Boot” button while press “Reset” button in order to enter into bootloader mode. The same operations as STEP 2 in 5.2 section.



STEP 3, Use command string to update the firmware, bootloader, or WLAN driver follow the instructions.

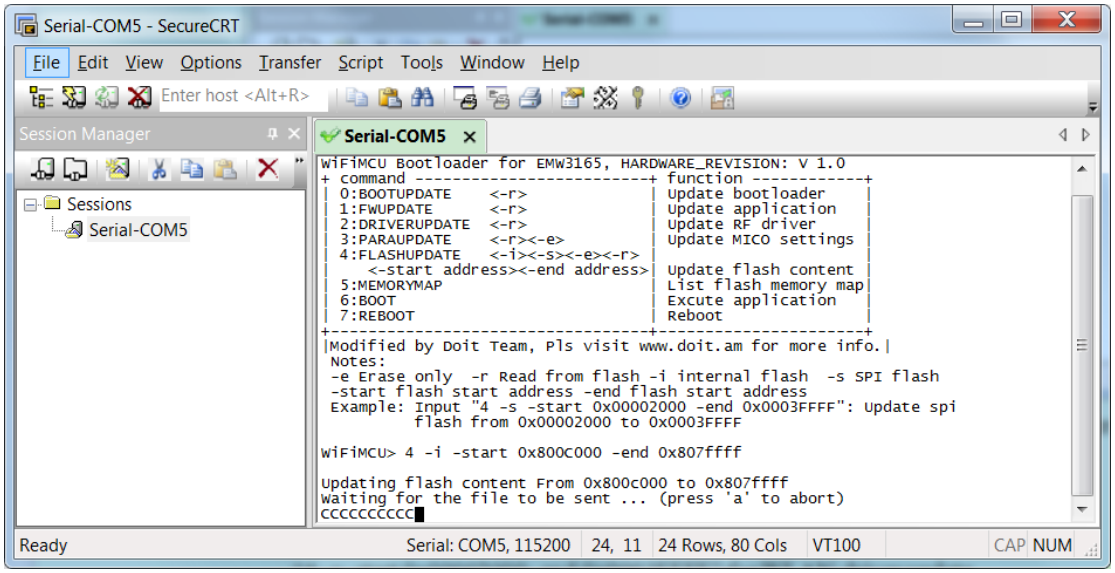
“4 -i -start 0x800C000 -end 0x807ffff” for firmware update.

“4 -i -start 0x8000000 -end 0x8007fff” for bootloader update.

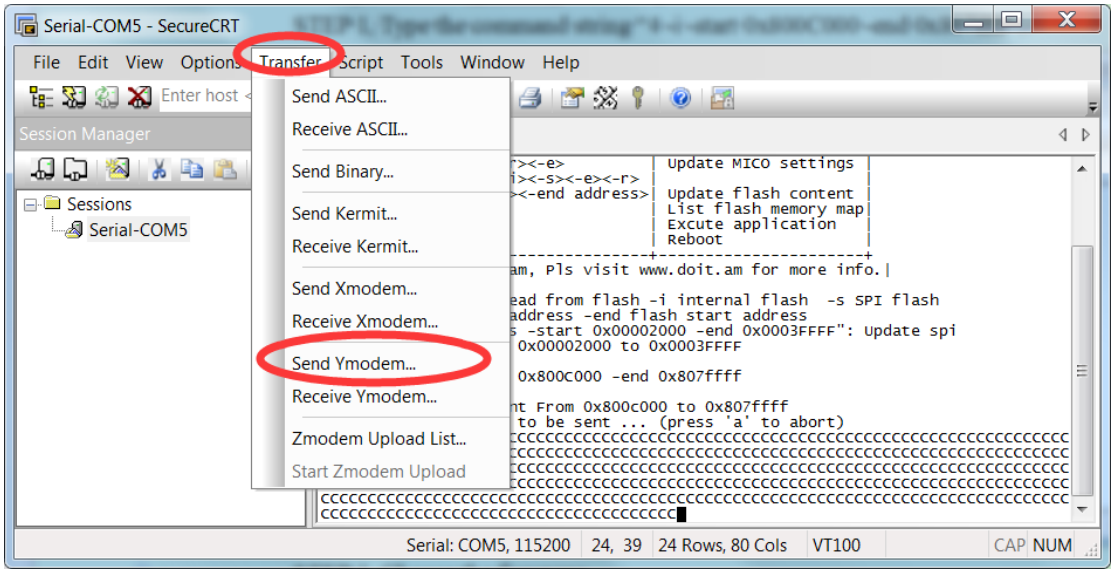
“4 -s -start 0x00002000 -end 0x0003FFFF” for WLAN driver update.

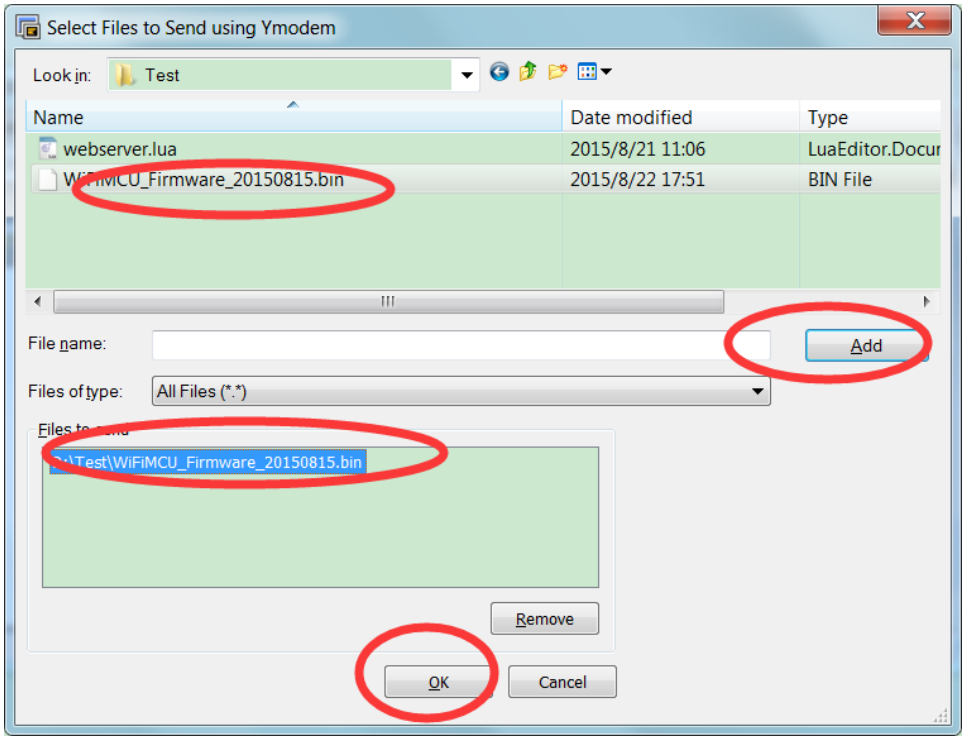
Taking firmware updatding for example:

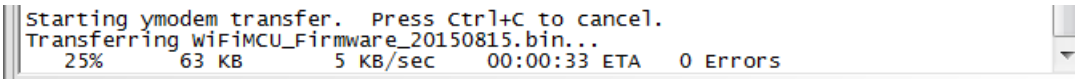
STEP 4, Type the command string “4 -i -start 0x800C000 -end 0x807ffff”.



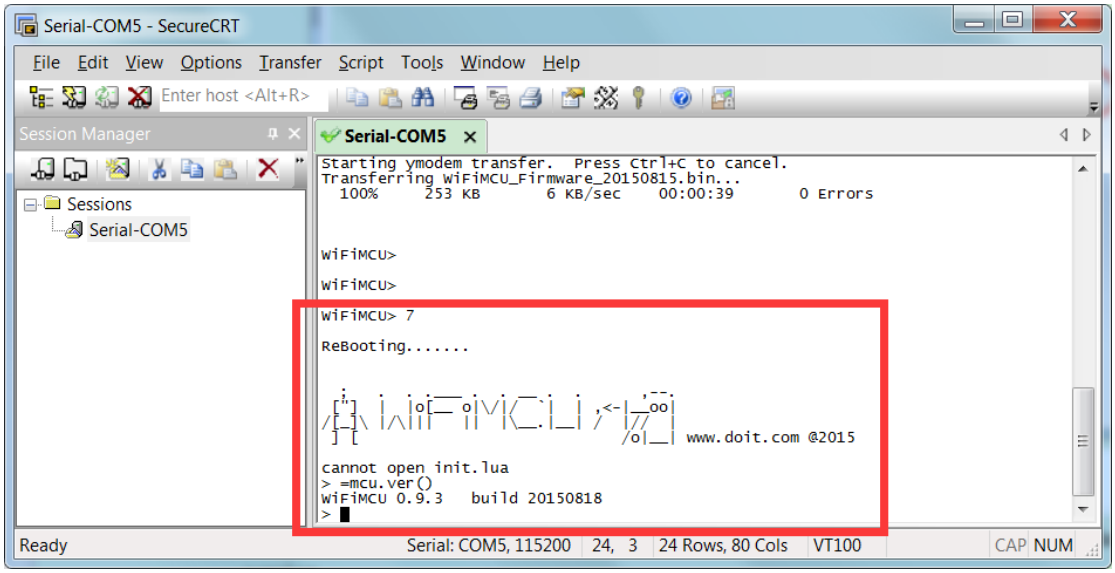
STEP 5, Choose the firmware, Waiting for finishing.







STEP 6, Reboot WiFiMCU, type command string ”=mcu.ver()” to check the firmware version.



5.4, Use SWD to update firmware

The SWD interface for WiFiMCU is shown below.The tutorial for SWD programmer can be found easily. You can compile the source code by using IAR and download the firmware, or just use SEGGER J-Flash to program the flash.

Helpful Links

- CP210x USB Bridge VCP driver:
  http://www.silabs.com/products/mcu/Pages/USBtoUARTBridgeVCPDrivers.aspx
- WiFiMCU binary firmware and source code:
  https://github.com/SmartArduino/WiFiMCU/releases
- WiFiMCU STUDIO:
  https://github.com/SmartArduino/WiFiMCU-STUDIO
- eLua :
  https://github.com/elua/elua

- NodeMCU
  https://github.com/nodemcu/nodemcu-firmware
- Lua 5.1.4 manual:
  http://www.lua.org/manual/5.1/
- Discussion:
  www.emw3165.com
  http://bbs.smartarduino.com
  http://bbs.doit.am
- Buy:
  http://www.smartarduino.com/view.php?id=94744

  http://www.aliexpress.com/item/WiFiMCU-Wireless-WiFi-Development-Board-Using-Lua-From-EMW3165/32440839773.html