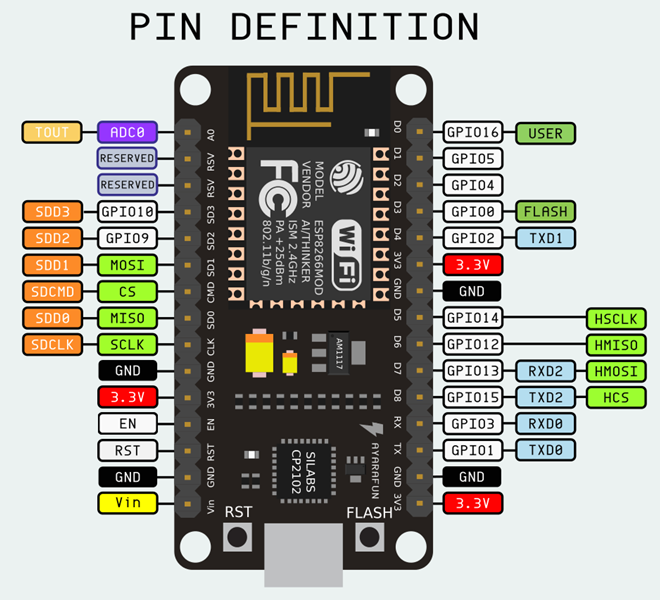
Terasys

This document describes about the DHT11 sensor interfacing to NodeMCU (ESP8266) board and would send the temperature and humidity data to Terasys IoT server.

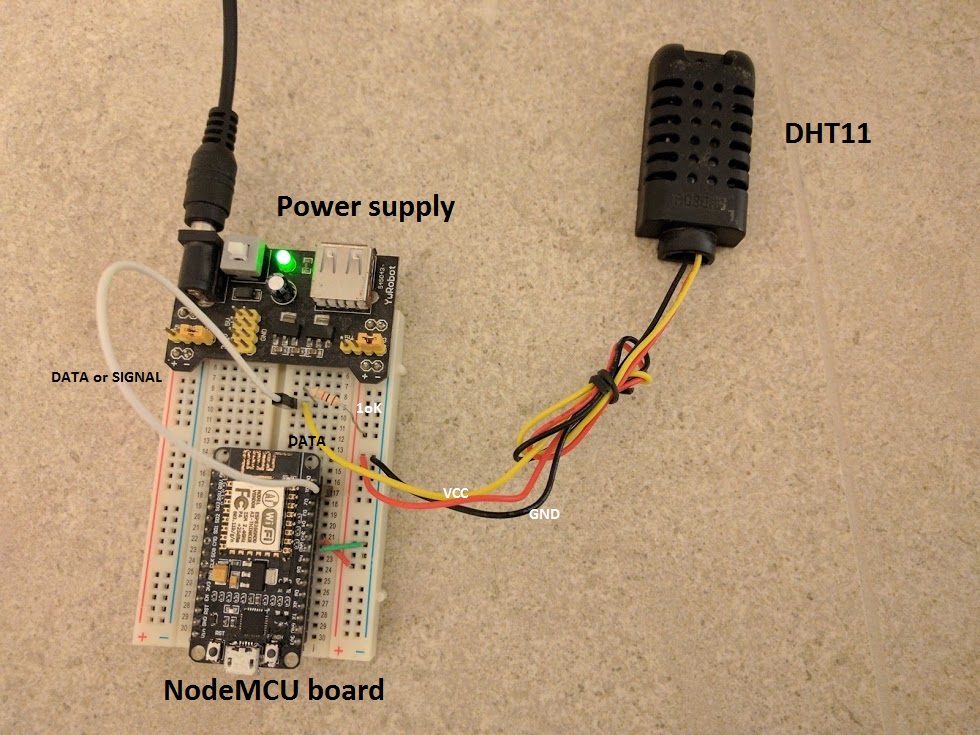
**Hardware connection:**

1. DHT11 sensor has three pins called GND,VCC and SIGNAL (DATA)
2. Connect the VCC pin to 3.3V supply, GND to GND and SIGNAL pin to D1 of NodeMCU (GPIO5)
3. RED and BLACK wires of the sensor are connected to +3.3V and GND, respectively (refer attached photo). The YELLOW wire (data) is connected to the pin 1 (D1, GPIO5) and pulled up to +3.3V with 1k - 10k resistor because the data line to the sensor is bidirectional (also used to get the accurate values)

**NodeMCU board:**



**DHT11 Sensor interface to NodeMCU (ESP8266) board:**



YELLOW color wire -> SIGNAL or DATA pin

RED color wire -> VCC (3.3V)

BLACK color wire -> GND (Ground or reference)

**Prerequisite Software:**

1. The following SWs to be installed on host PC (Linux or Windows or MAC)

i) Required JAVA (SE version 7 and above) installed.

ii) Later version of python

iii) CH340/CH341 "USB to serial" drivers are needed for NodeMCU board.

https://github.com/nodemcu/nodemcu-devkit/tree/master/Drivers

In Windows, a succesful connect will create a new serial port, viewable in "Device manager" under "Ports (COM & LPT)"".

**Downloading the LUA firmware into NodeMCU board:**

1. Download the LUA firmware (***nodemcu-master-13-modules-2017-02-11-20-55-17-float.bin***) from the following link.

https://github.com/gabod2000/NodeMCU-client/tree/master/firmware/nodemcu-master-13-modules-2017-02-11-20-55-17-float.bin

1. Download the following ***esptool*** which is used for flashing the LUA firmware into NodeMCU board.

https://github.com/espressif/esptool

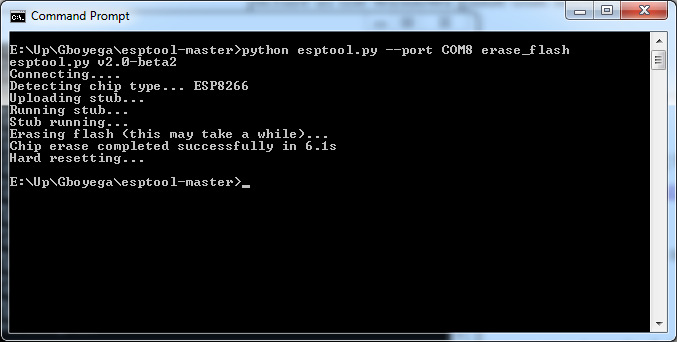
1. Before uploading the firmware, erase the flash content of NodeMCU using the following command.

***cd esptool-master***

***python esptool.py --port COM8 erase\_flash***

**Ex:**

***E:\esptool-master> python esptool.py --port COM8 erase\_flash***



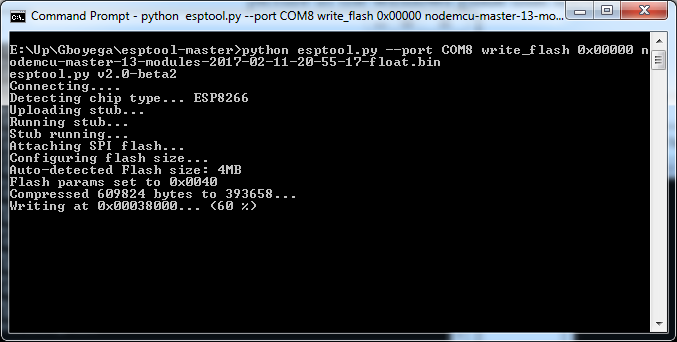
1. Program the downloaded LUA firmware into flash of NodeMCU using the following command.

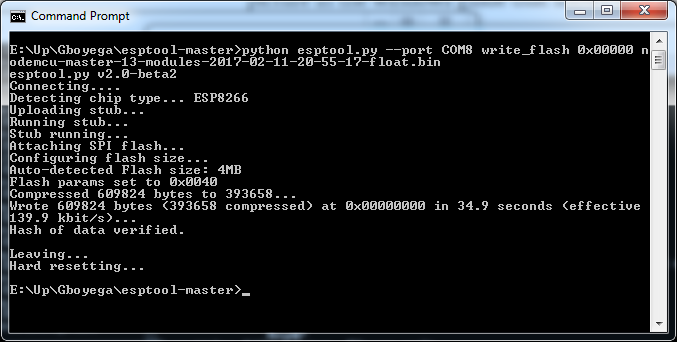
***cd esptool-master***

***python esptool.py --port COM8 write\_flash 0x00000 nodemcu-master-13-modules-2017-02-11-20-55-17-float.bin***

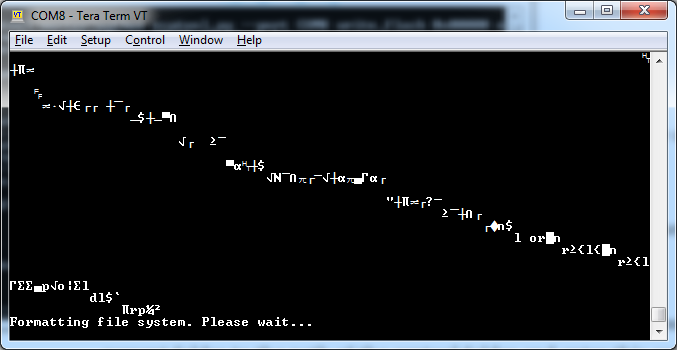
**Ex:**

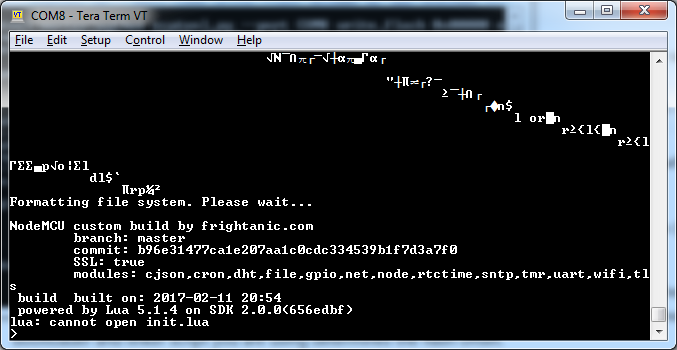
***E:\esptool-master> python esptool.py --port COM8 write\_flash 0x00000 nodemcu-master-13-modules-2017-02-11-20-55-17-float.bin***





1. Wait for while till it format the SPI flash of NodeMCU and boot LUA code.





1. Now NodeMCU is ready for uploading the ***Terasys*** LUA programs to communicate to IoT server.

**Downloading the Terasys's LUA programs into NodeMCU board:**

1. Download the Terasys's LUA code (\*.lua) from the following link.

https://github.com/gabod2000/NodeMCU-client/tree/master/src

1. Modify the ***config.lua*** file for your WiFi router name and its password settings and also add your location details (GPS longitude and latitude)
2. Using google maps, select your location, get the longitude & latitude values and use it in ***config.lua*** file.

**Ex:**

*--- LOCATION ---*

*lat = 13.0377*

*lon = 80.2277*

*--- WIFI CONFIGURATION ---*

*WIFI\_SSID = "mywifi\_router"*

*WIFI\_PASSWORD = "mywifi\_password"*

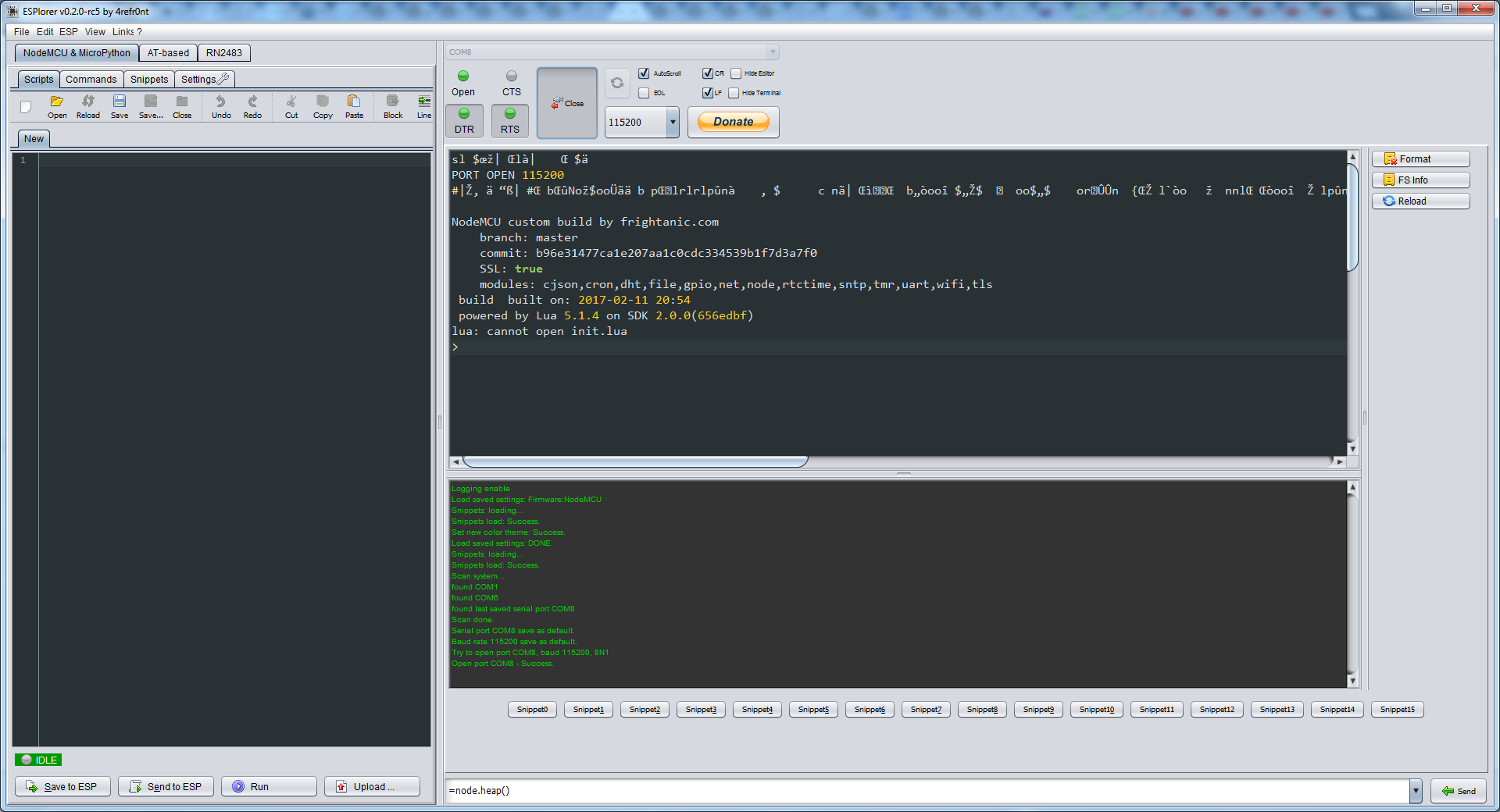
4. Download the ***esplorer*** from the following link.

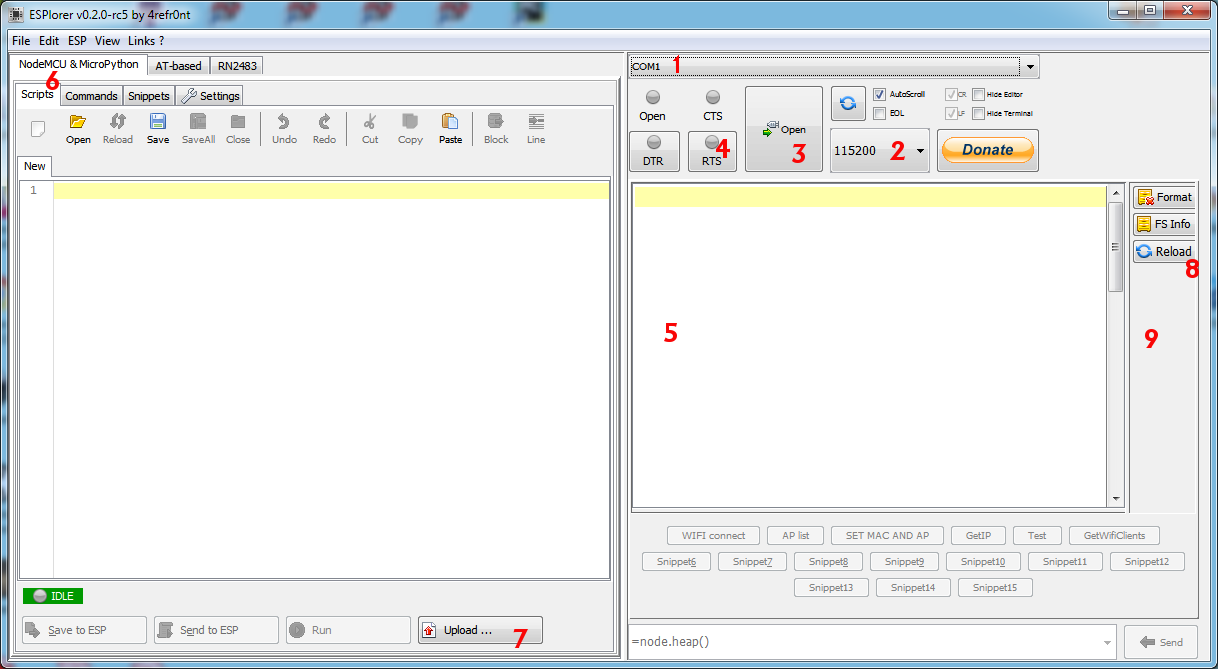
5. Run the ESPlorer.bat file.

6. For start, one can use ESPlorer at https://esp8266.ru/esplorer/. Install, run, in the top of the right panel choose the appropriate COM port, speed 115200 and press open the port.

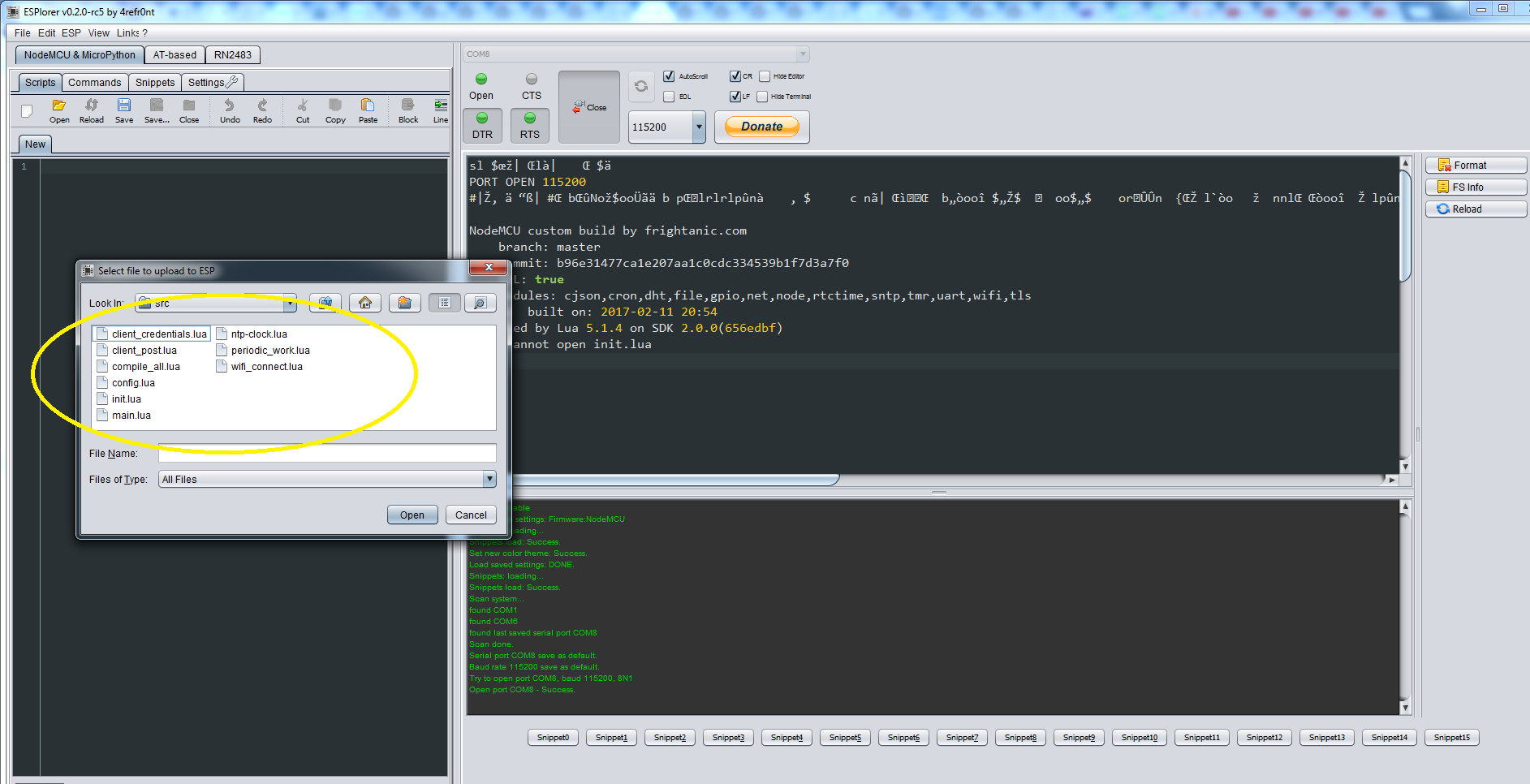
7. By switching RTS icon ON and OFF, the device resets and restarts.

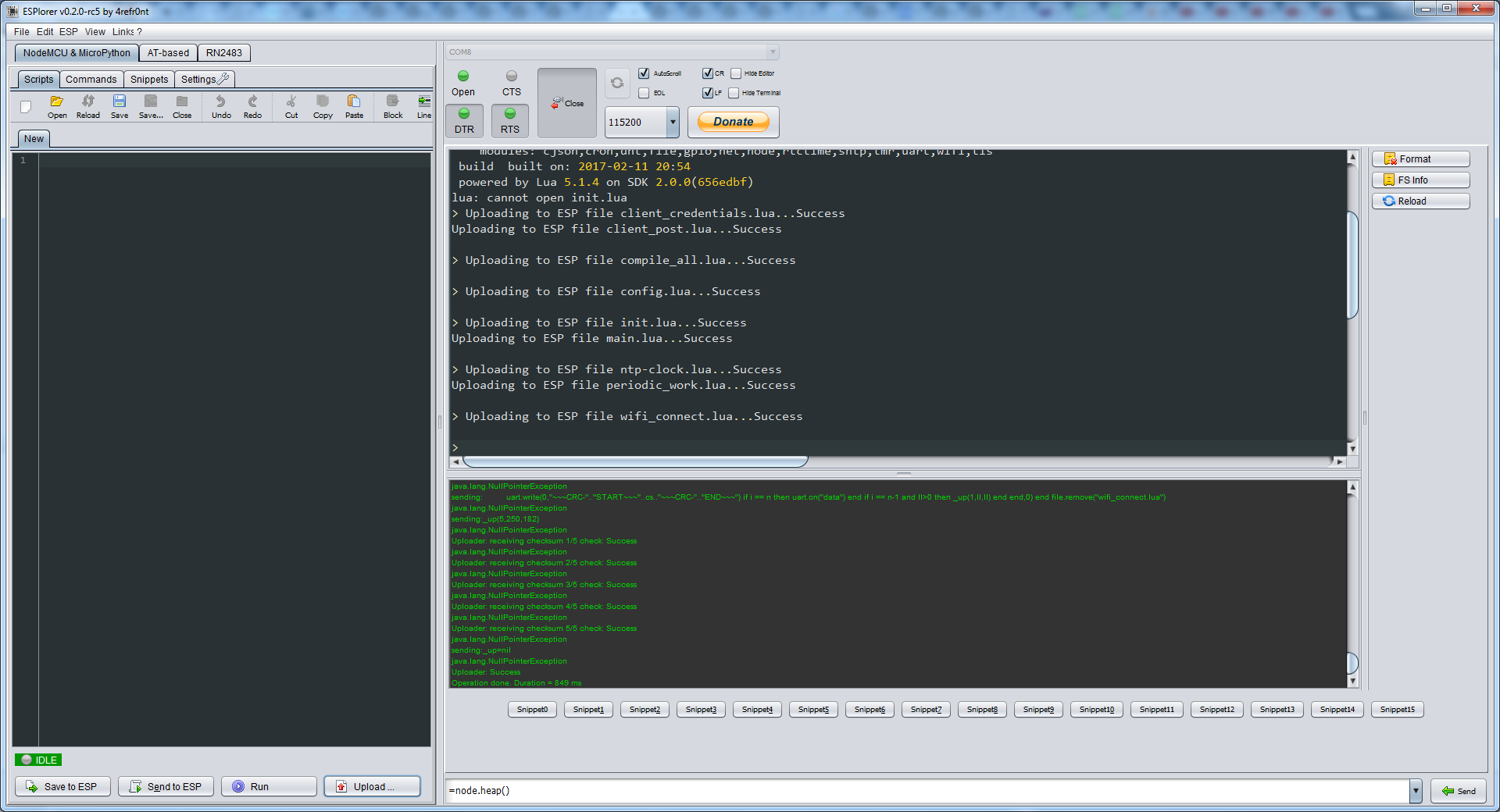
8. The Lua prompt appears. Message about missing init.lua is normal.



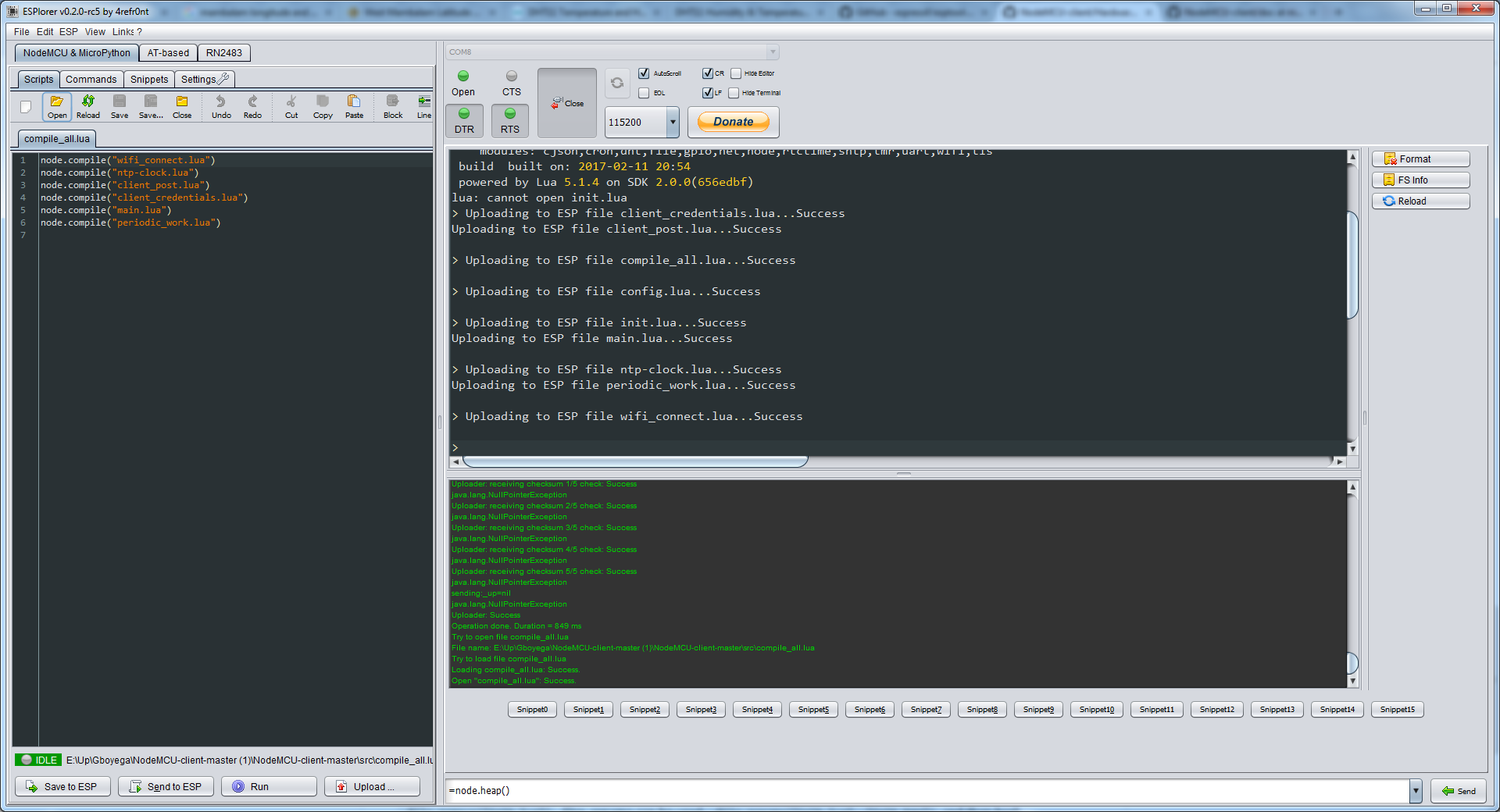


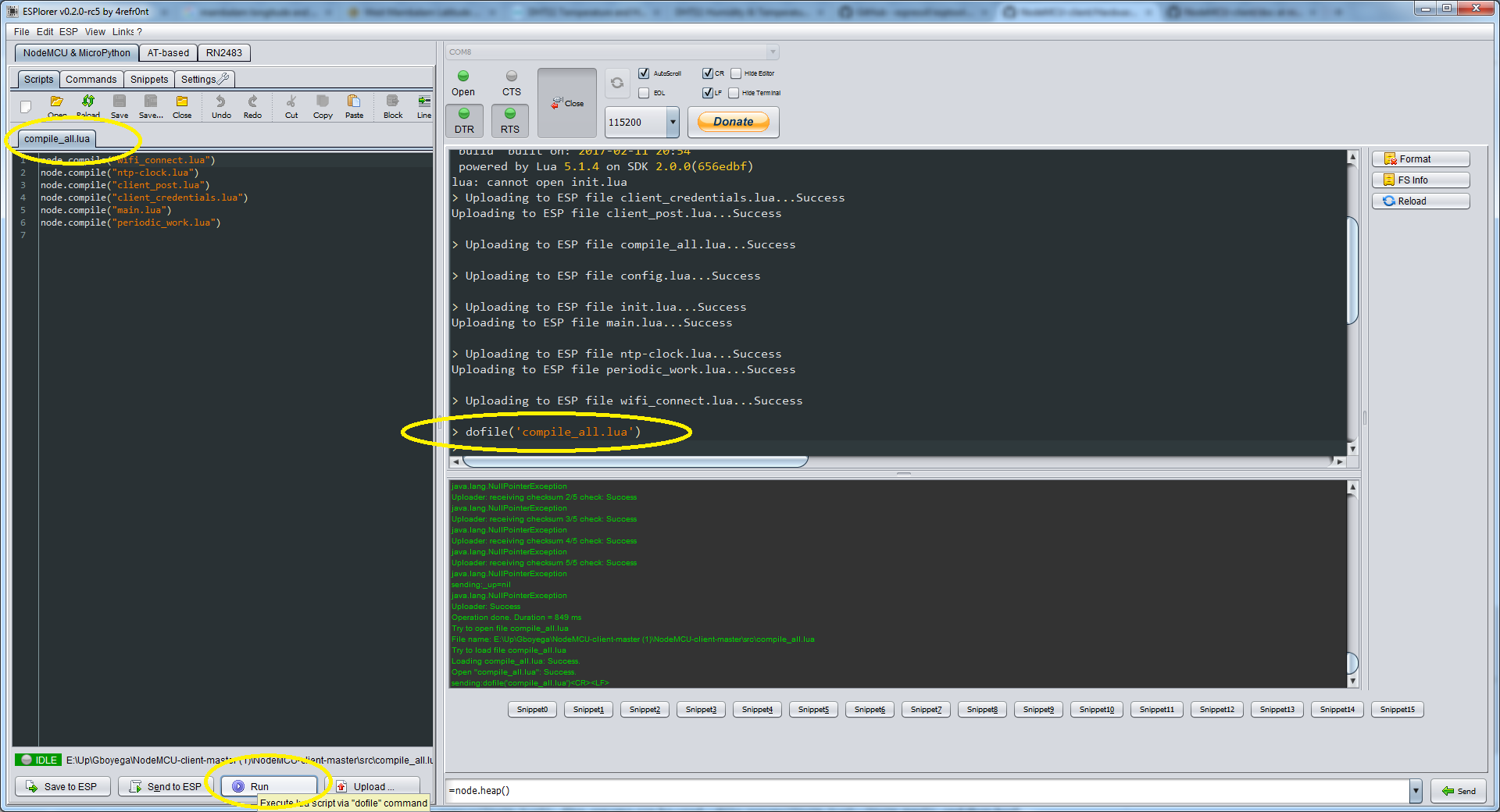
9. In the left panel, choose tabs "NodeMCU & MicroPython", "Scripts", at the bottom right click a button "Upload", select all lua files and upload them at once. On the right panel, the names of uploaded files should appear (otherwise press Reload button).

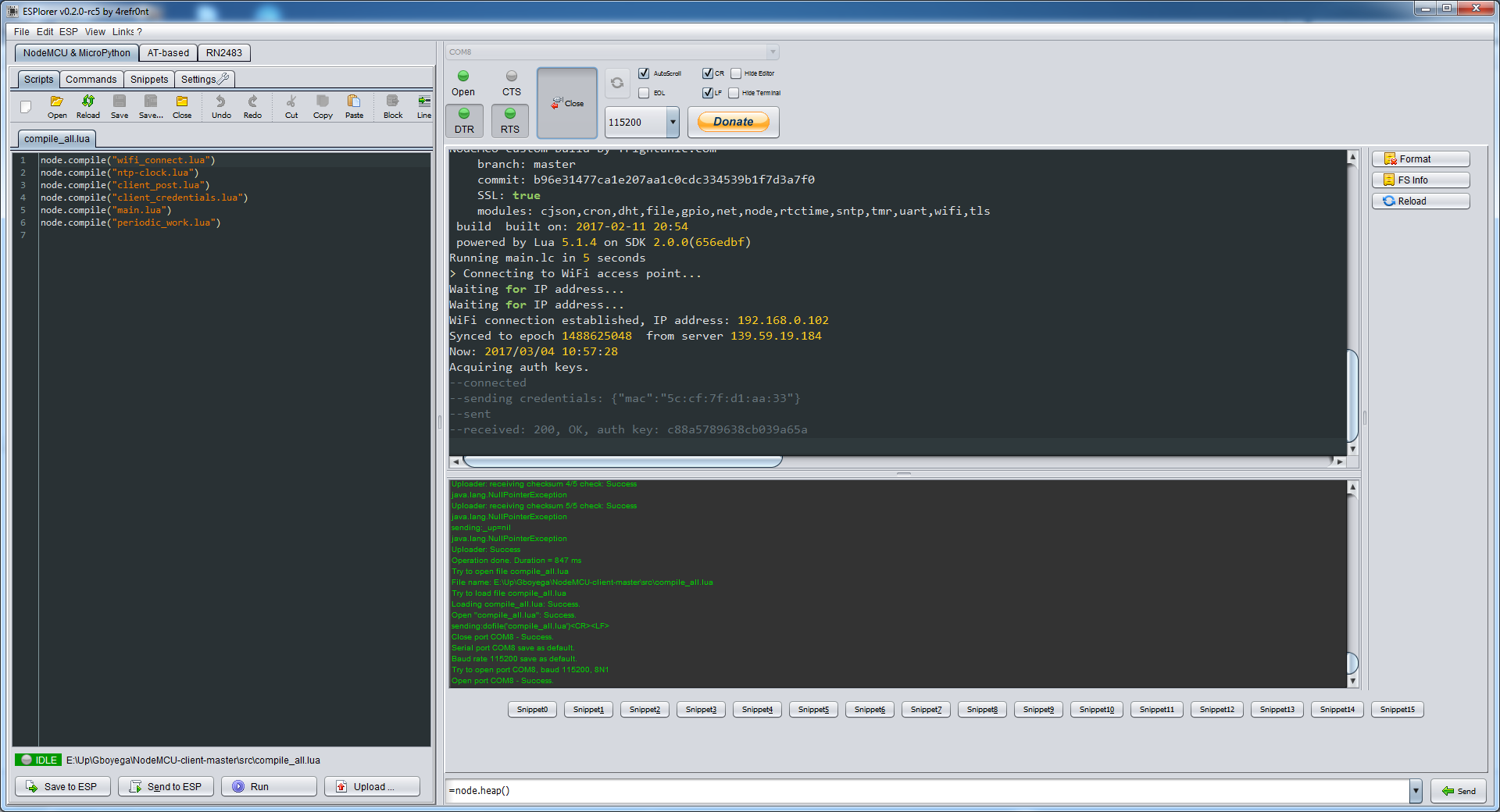




10. Click on compile\_all.lua file to run it. It will compile the code to reduce the memory footprint.







Now the DHT11 sensor data (temperature and humidity) sending to the Terasys IoT server.

