

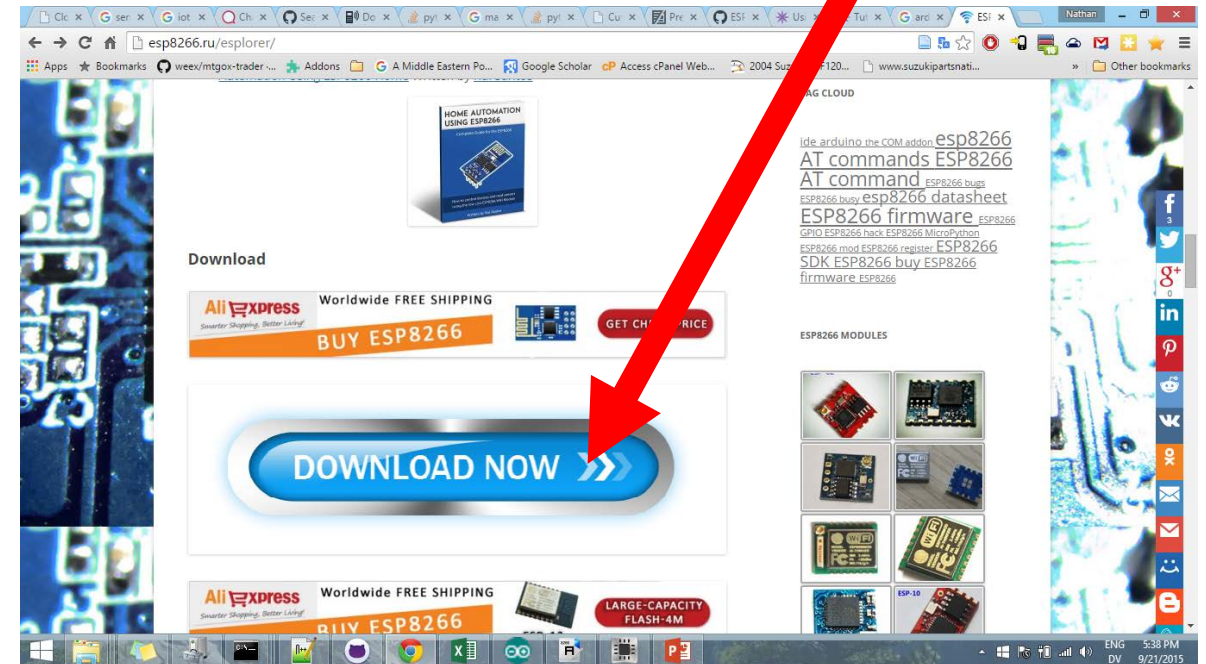
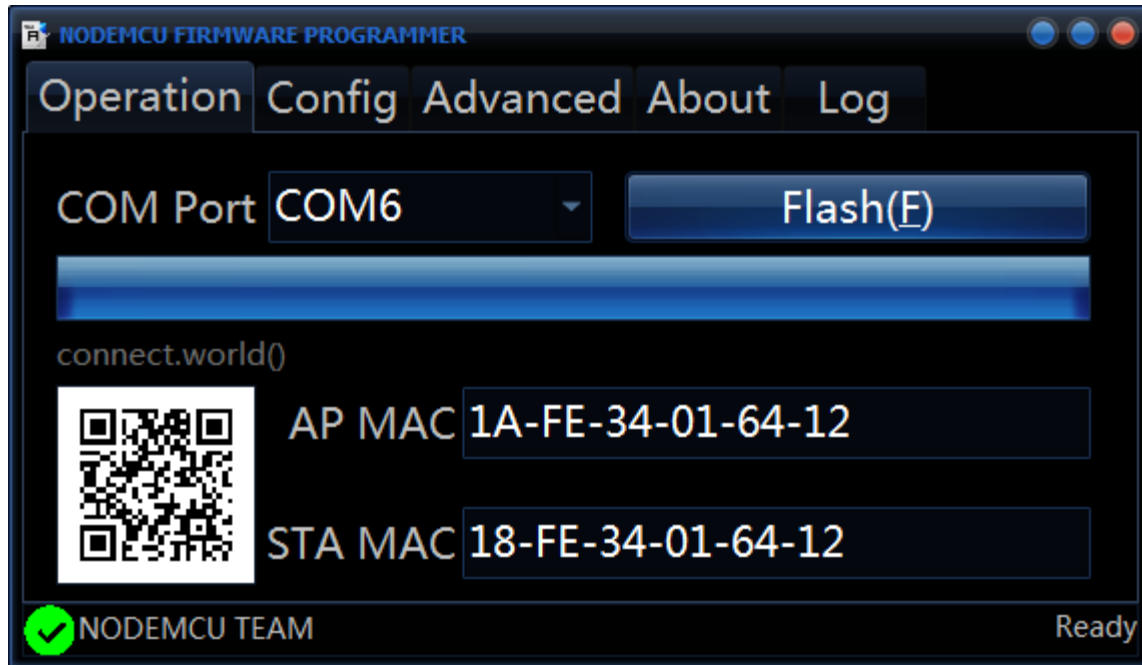
Tutorial for programming ESP8266 on windows, using the NodeMCU devkit

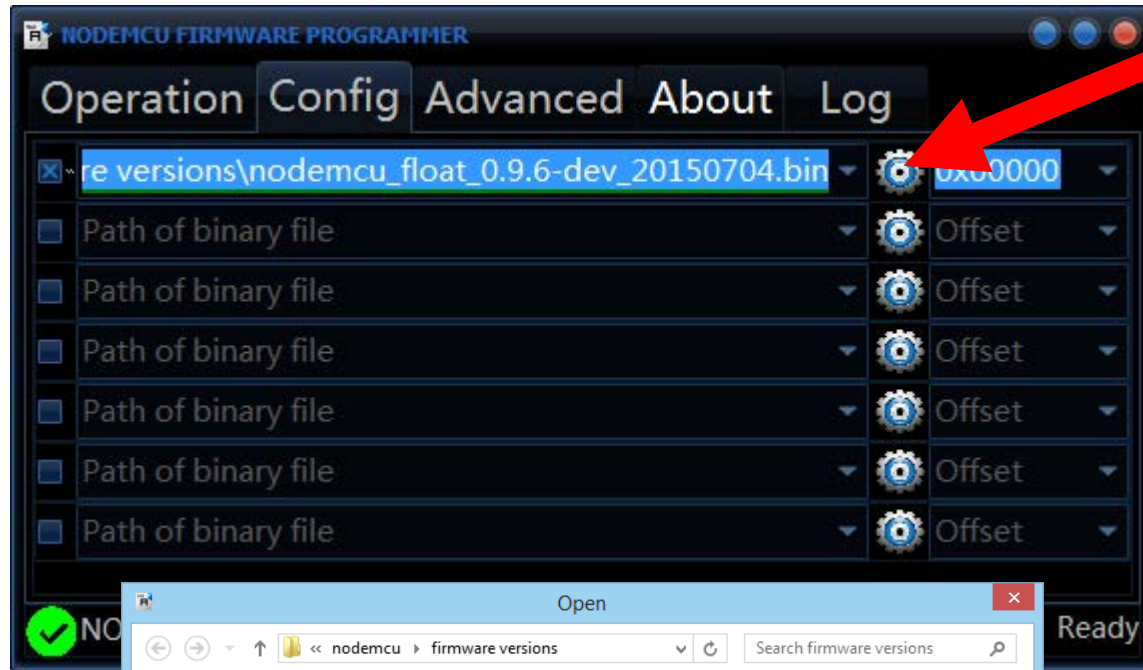
Get the nodemcu flasher from here:

<https://github.com/nodemcu/nodemcu-flasher>

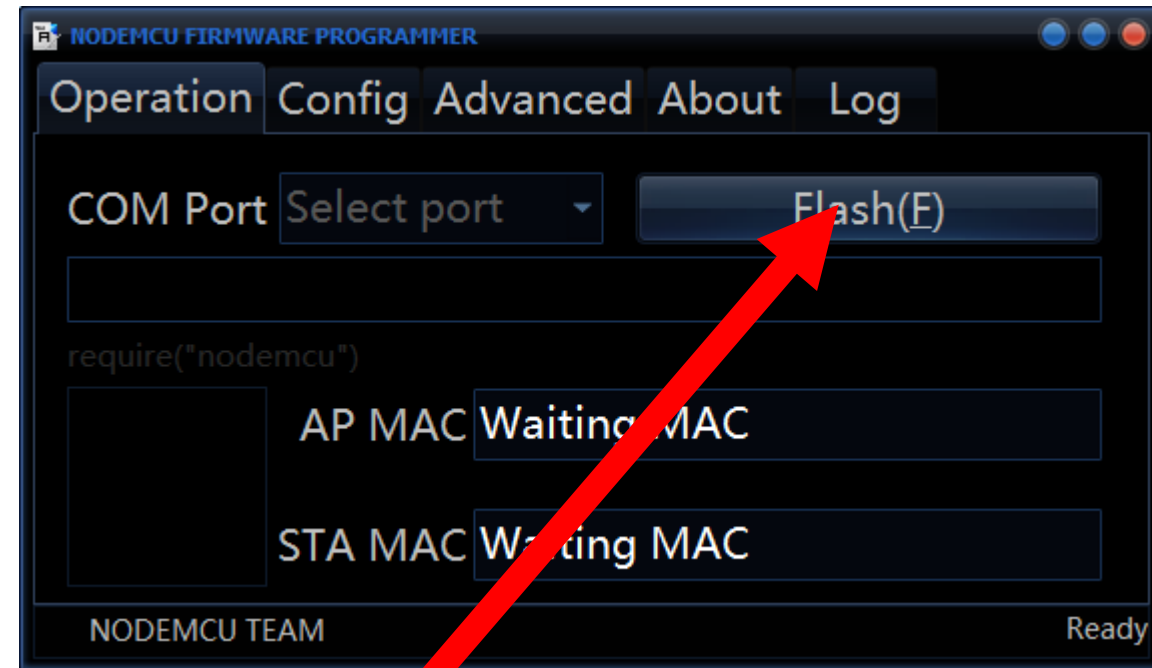
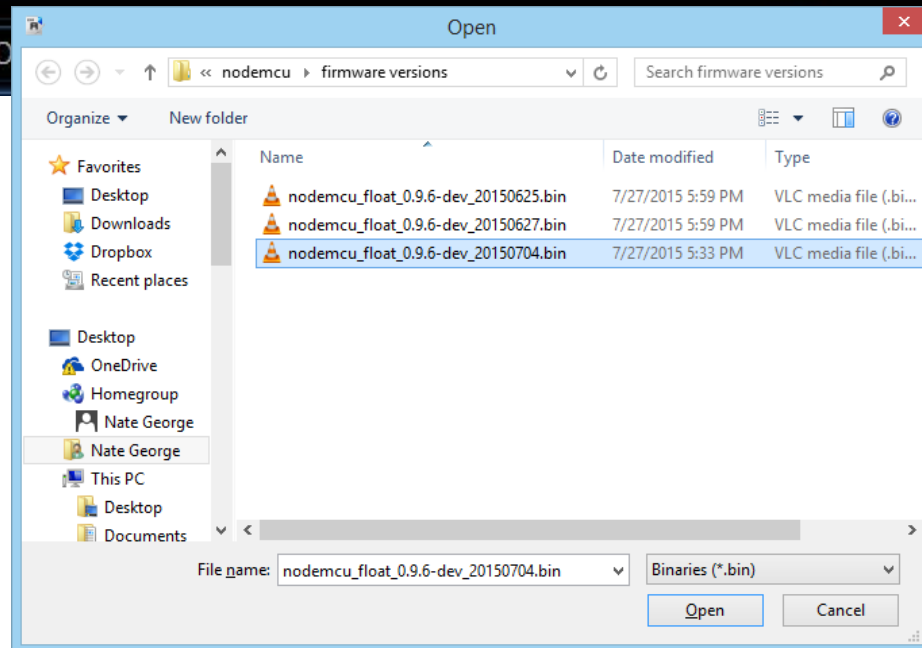
Get ESPlorer from here:

<http://esp8266.ru/esplorer/>

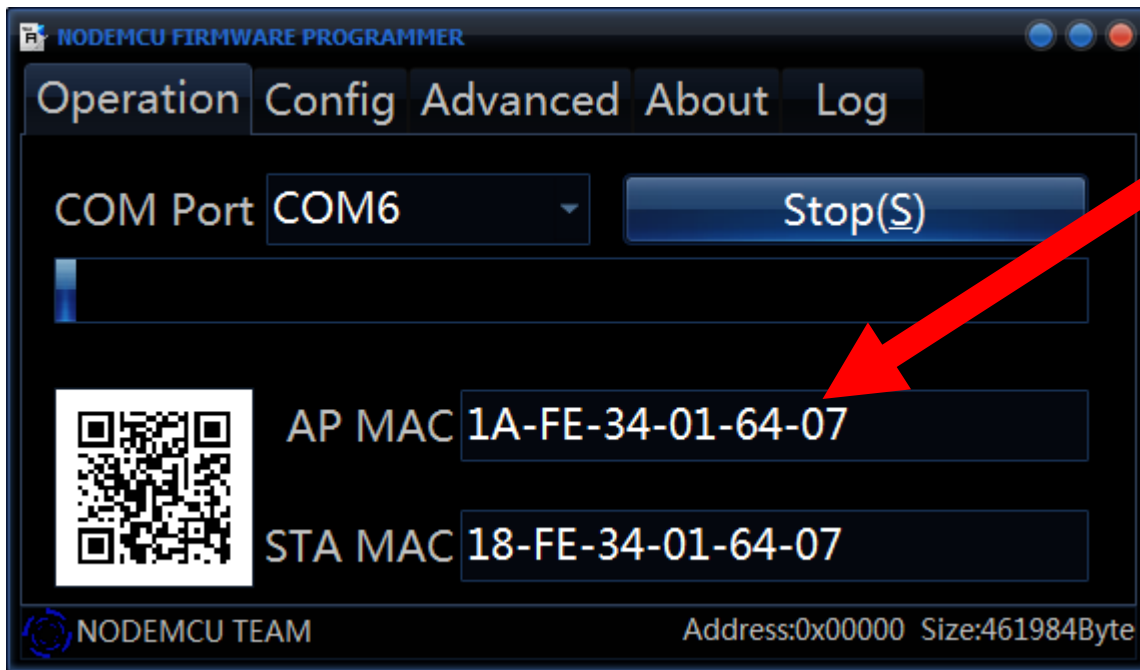




First, click on the gear under config, and choose your firmware file. Use the latest from <https://github.com/nodemcu/nodemcu-firmware/releases> which you can find by googling 'nodemcu releases'

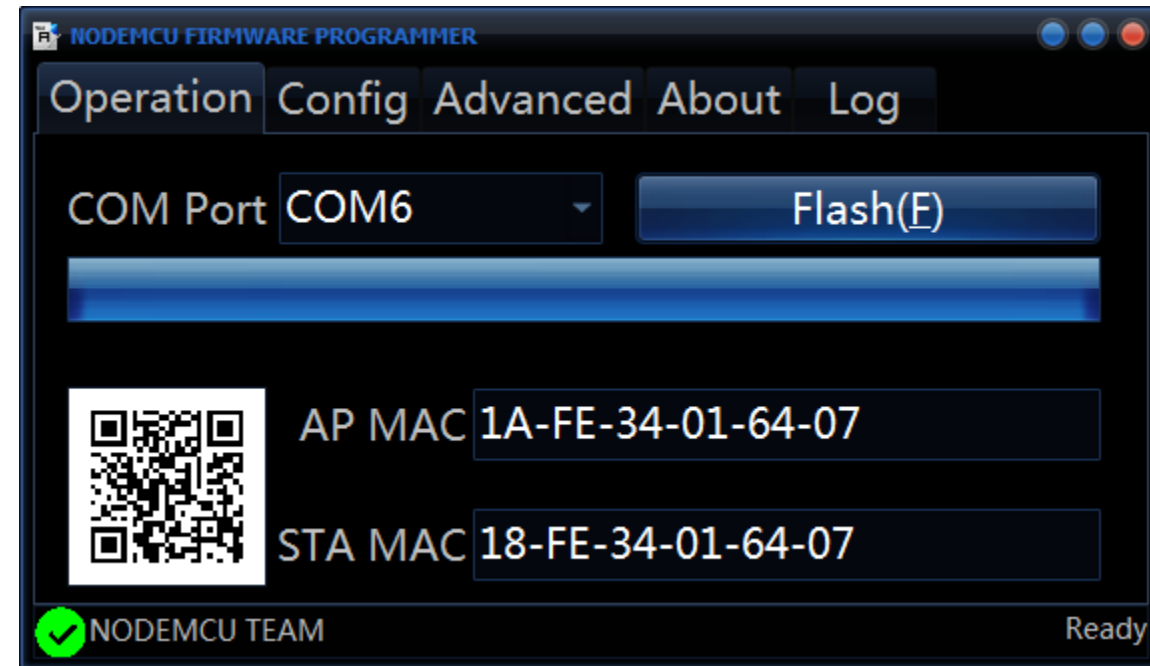
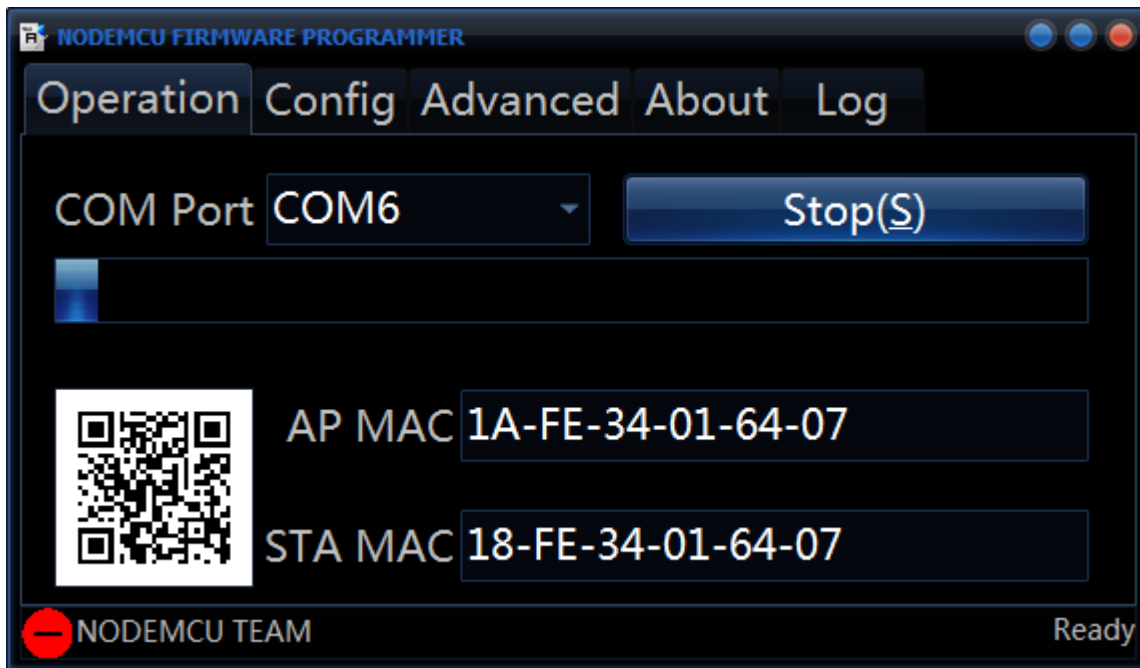


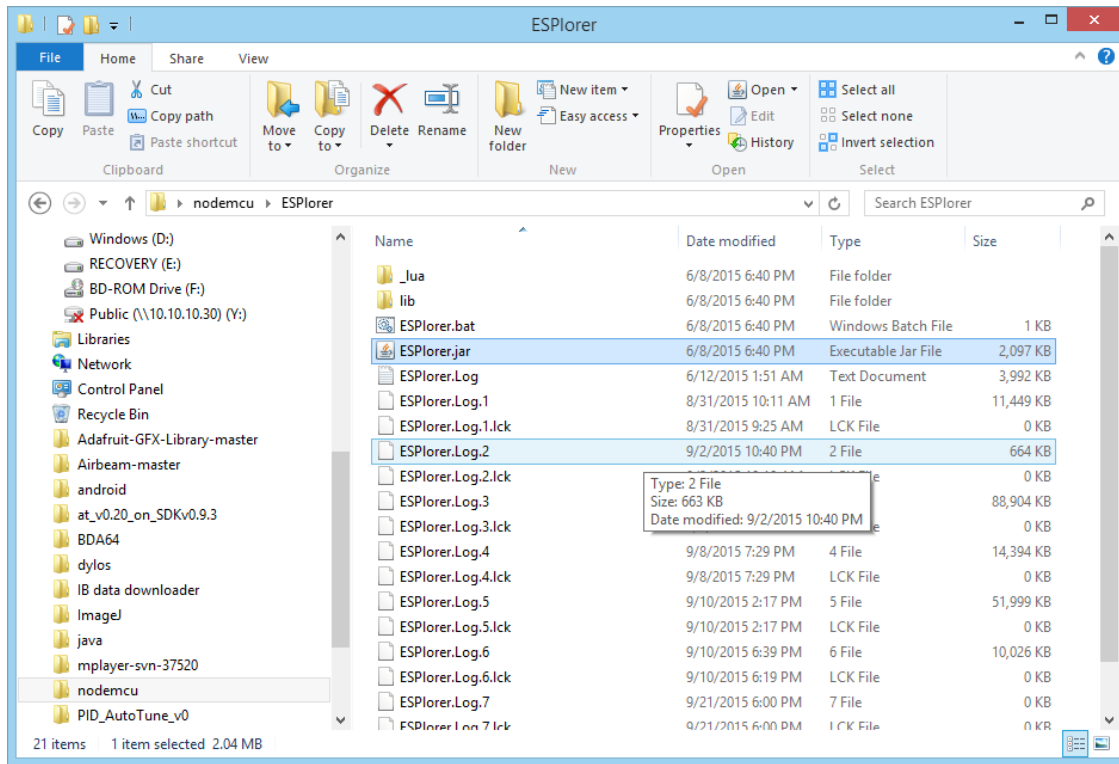
Next, click 'flash'



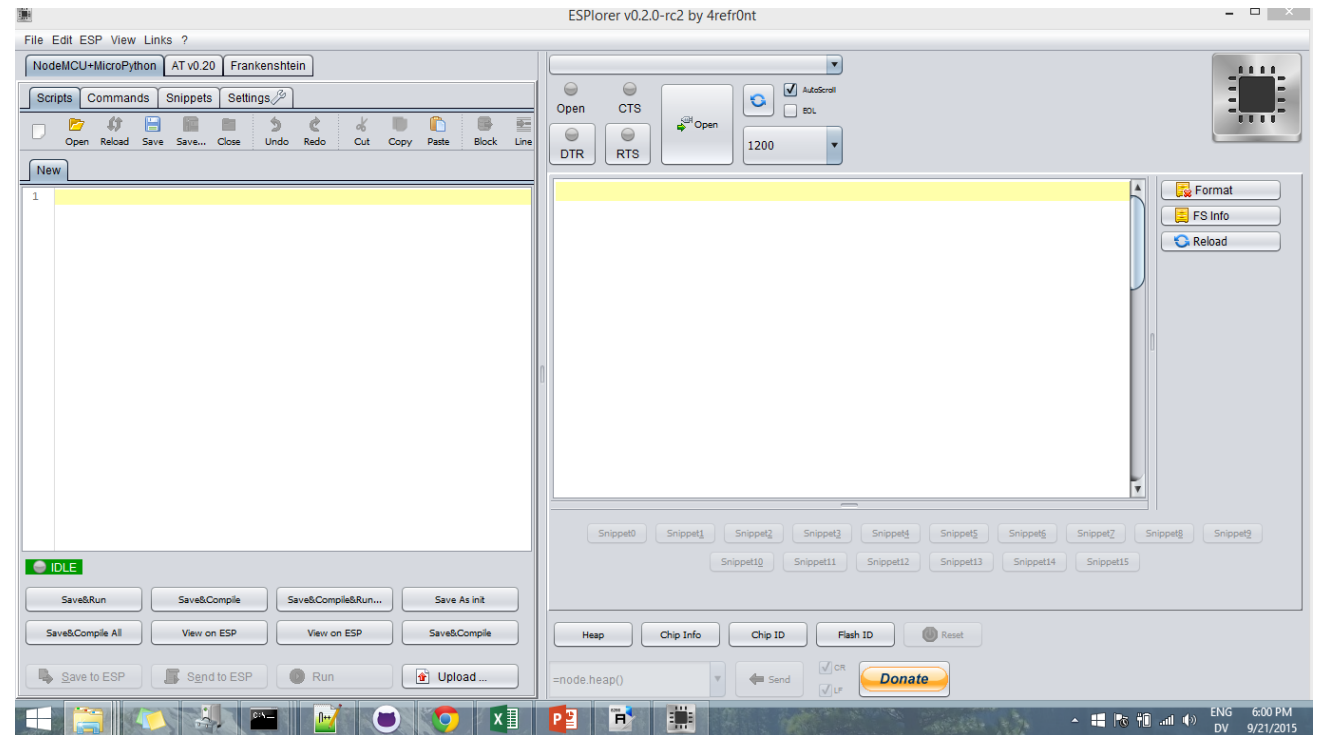
The MACs will be filled in along with the QR code. If something goes wrong, the red circle will appear in the bottom right, otherwise, once the status bar gets to full, and everything went alright, the status circle will change to green.

You should now reset the module.

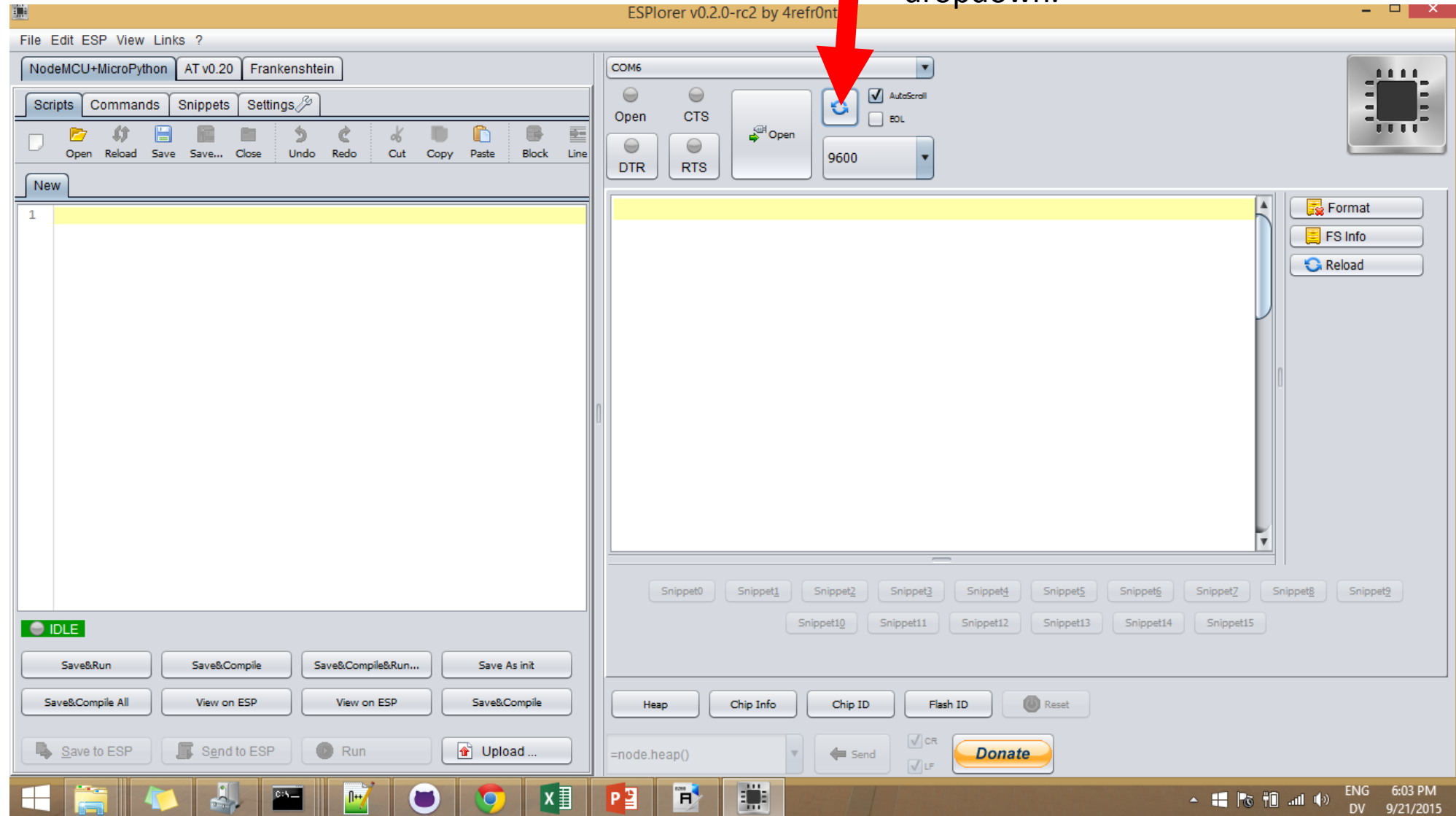




Open ESPlorer.jar



Click the refresh button, and it should detect your Esp8266 if it was flashed correctly and is plugged in. You may need to choose the COM port from the dropdown.



Click 'open', the response should look something like this.

ESPlorer v0.2.0-rc2 by 4refr0nt

File Edit ESP View Links ?

NodeMCU+MicroPython AT v0.20 Frankenshtein

Scripts Commands Snippets Settings

Open Reload Save Save... Close Undo Redo Cut Copy Paste Block Line

New

1

COM6

Open CTS DTR RTS Close AutoScroll BOL 9600

PORT OPEN 9600

Communication with MCU...

Got answer! AutoDetect firmware...

Can't autdetect firmware, because proper answer not received.

>

Format FS Info Reload

Snippet0 Snippet1 Snippet2 Snippet3 Snippet4 Snippet5 Snippet6 Snippet7 Snippet8 Snippet9 Snippet10 Snippet11 Snippet12 Snippet13 Snippet14 Snippet15

Heap Chip Info Chip ID Flash ID Reset

=node.heap() Send CR LF Donate

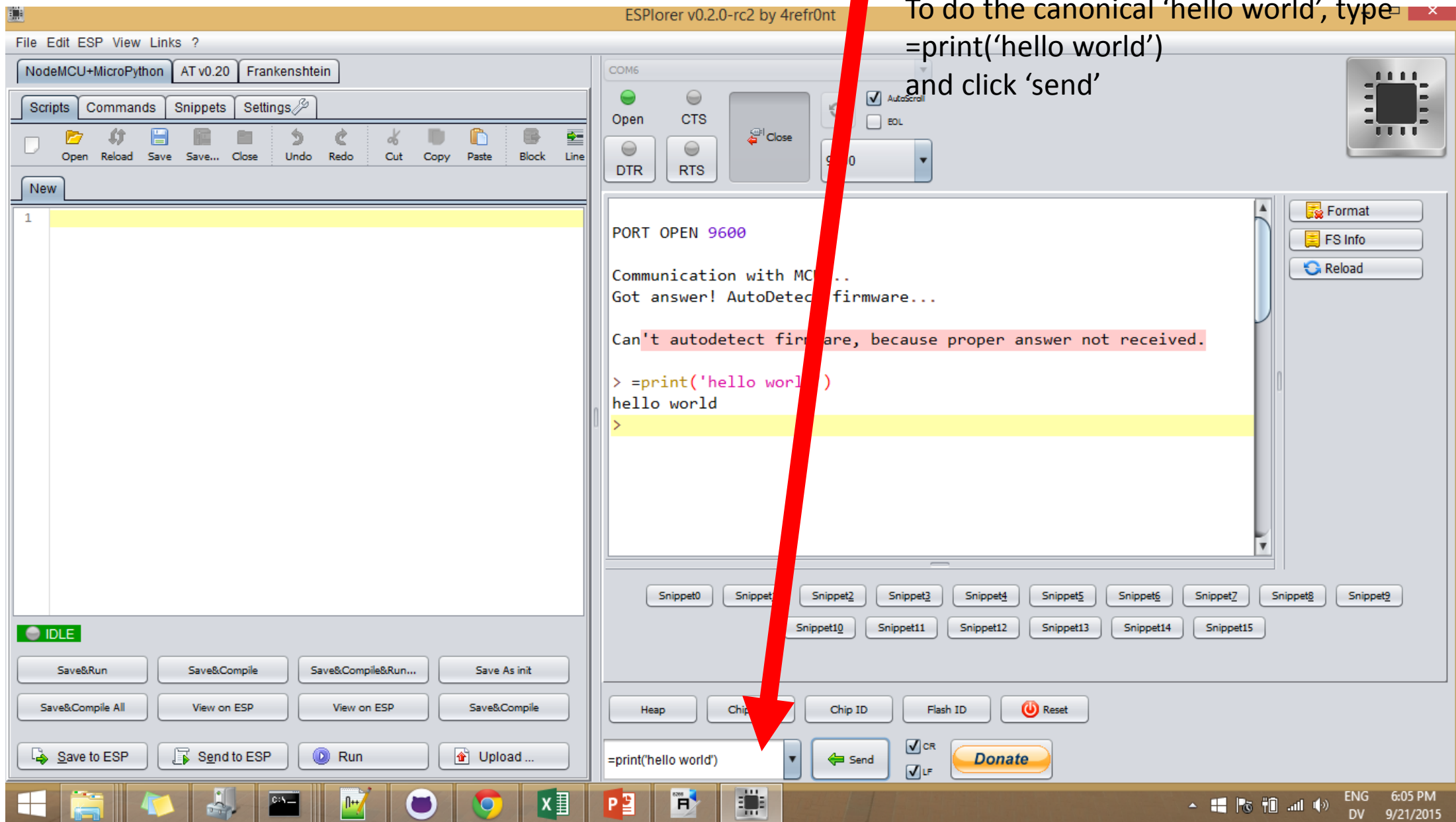
IDLE

Save&Run Save&Compile Save&Compile&Run... Save As init

Save&Compile All View on ESP View on ESP Save&Compile

Save to ESP Send to ESP Run Upload ...

ENG 6:04 PM 9/21/2015



You can enter commands down here, then click 'send'
To do the canonical 'hello world', type
`=print('hello world')`
and click 'send'

Click 'reload' to list the files on the device

ESPlorer v0.2.0-rc2 by 4refr0nt

File Edit ESP View Links ?

NodeMCU+MicroPython AT v0.20 Frankenshtein

Scripts Commands Snippets Settings

Open Reload Save Save... Close Undo Redo Cut Copy Paste Block Line

New

1

COM6

Open CTS Close AutoScroll EOL

DTR RTS 9600

```
> =print('hello world')
hello world
>

-----
sendToTS.lc      : 3224 bytes
sendToTS.lua     : 4751 bytes
-----
Total file(s)    : 2
Total size       : 7975 bytes

> r,u,t=file.fsinfo() print("Total : "..t.." bytes\r\nUsed : "..u.." bytes\r\nRemain: 3380970 bytes")
Total : 3396281 bytes
Used : 15311 bytes
Remain: 3380970 bytes
>
```

Format Info Reload

sendToTS.lc sendToTS.lua

Snippet0 Snippet1 Snippet2 Snippet3 Snippet4 Snippet5 Snippet6 Snippet7 Snippet8 Snippet9

Snippet10 Snippet11 Snippet12 Snippet13 Snippet14 Snippet15

Heap Chip Info Chip ID Flash ID Reset

Save&Run Save&Compile Save&Compile&Run... Save As init

Save&Compile All View on ESP View on ESP Save&Compile

Save to ESP Send to ESP Run Upload ...

=print('hello world')

Send CR LF Donate

ENG 6:07 PM 9/21/2015

Click 'heap' or send '=node.heap()' to get the heap size (free memory).

The screenshot shows the ESPlorer v0.2.0-rc2 IDE interface. The top menu bar includes File, Edit, ESP, View, and Links. Below the menu is a toolbar with icons for Open, Reload, Save, Save..., Close, Undo, Redo, Cut, Copy, Paste, Block, and Line. The main editor area on the left is titled 'New' and contains a single line of code: `> =print('hello world')`. The right panel displays the output of the code, showing the results of the `print` statement and the output of the `file.fsinfo()` function. The output includes the size of the files `sendToTS.lc` and `sendToTS.lua`, the total file size, and the total, used, and remaining memory. The bottom right panel contains buttons for 'Heap', 'Chip Info', 'Chip ID', 'Flash ID', and 'Reset'. A red arrow points from the text above to the 'Heap' button. The bottom status bar shows the system tray with icons for Windows, taskbar, and system clock.

File Edit ESP View Links ?

NodeMCU+MicroPython AT v0.20 Frankenshtein

Scripts Commands Snippets Settings

Open Reload Save Save... Close Undo Redo Cut Copy Paste Block Line

New

1

`> =print('hello world')`
hello world
`>`

sendToTS.lc : 3224 bytes
sendToTS.lua : 4751 bytes

Total file(s) : 2
Total size : 7975 bytes

`> r,u,t=file.fsinfo() print("Total : "..t.." bytes\r\nUsed : "..u..
Total : 3396281 bytes
Used : 15311 bytes
Remain: 3380970 bytes
>`

Snippet0 Snippet1 Snippet2 Snippet3 Snippet4 Snippet5 Snippet6 Snippet7 Snippet8 Snippet9
Snippet10 Snippet11 Snippet12 Snippet13 Snippet14 Snippet15

Idle

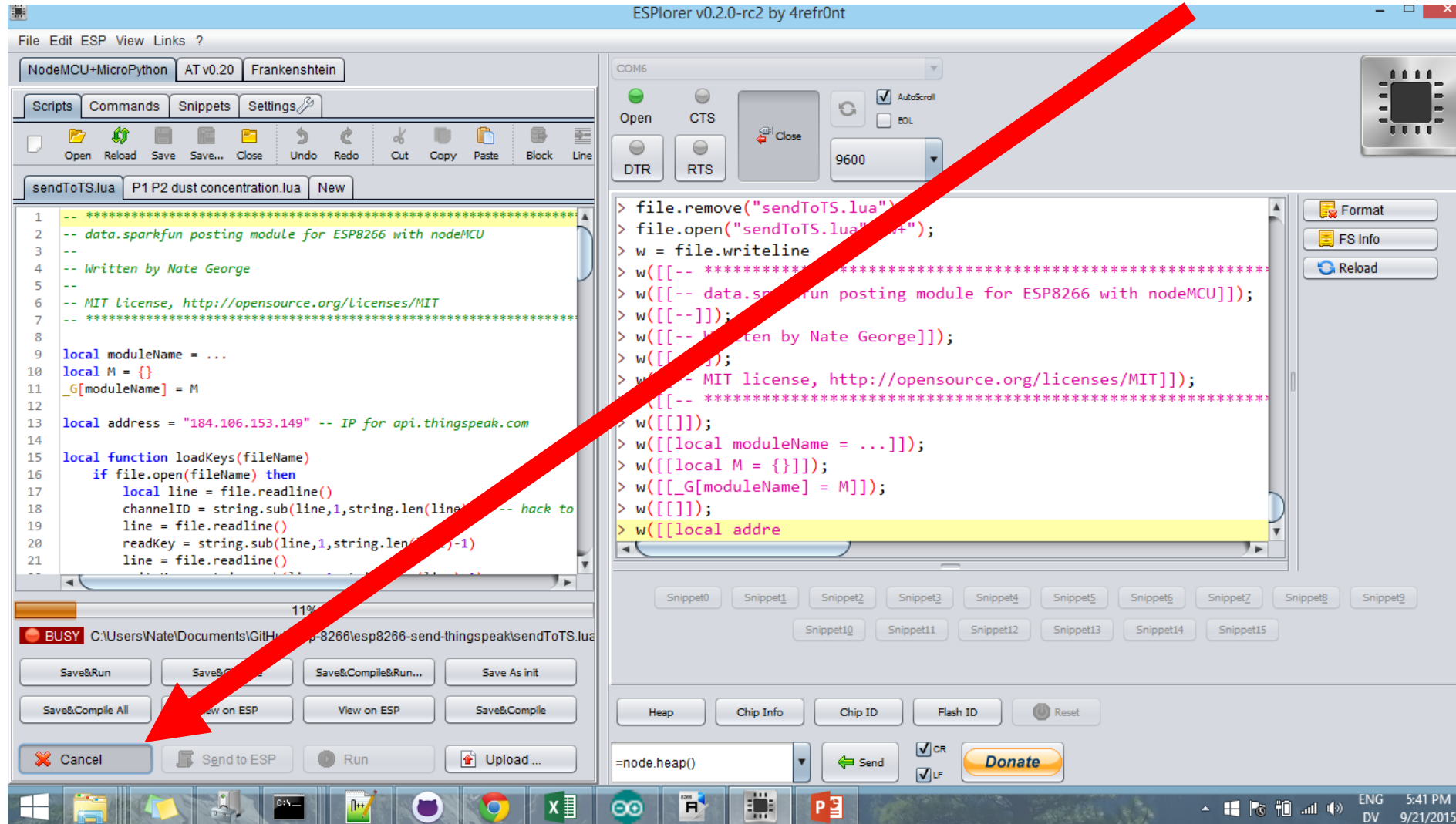
Save&Run Save&Compile Save&Compile&Run... Save As init
Save&Compile All View on ESP View on ESP Save&Compile
Save to ESP Send to ESP Run Upload ...

Heap Chip Info Chip ID Flash ID Reset

`=print('hello world')` Send CR LF Donate

ENG 6:07 PM
DV 9/21/2015

Click 'save to ESP' to write your file to the device. Clicking 'saveandcompile' will compile it, and save a lot of memory at runtime.



The programming is done in the Lua language, which is c-like. I made a few chunks of code for using NodeMCU:

Wifi network chooser:

<https://github.com/wordsforthewise/ESP-8266-network-connect>

Makes a server at 192.168.4.1 and asks for wifi credentials, then logs into the network.

Send data to sparkfun:

<https://github.com/wordsforthewise/esp8266-send-to-sparkfun>

Send data to thingspeak:

<https://github.com/wordsforthewise/esp8266-send-thingspeak>

