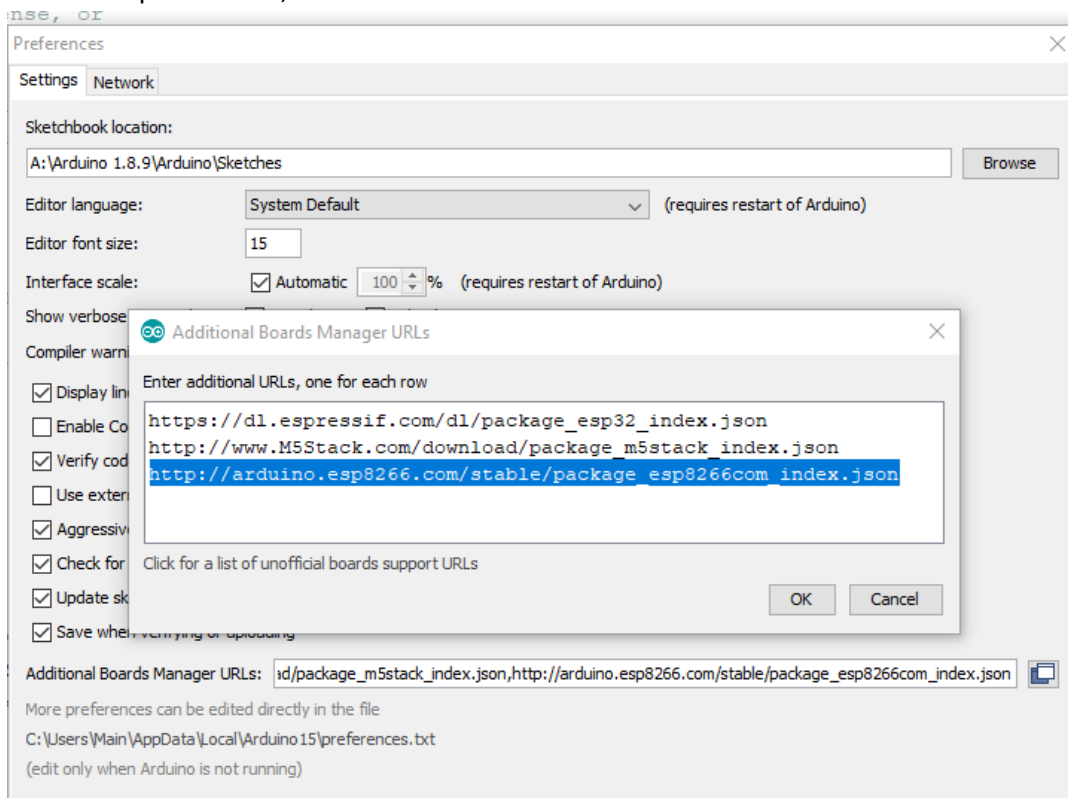


## Nicks method to set up an ESP8266 (ESP-01S 'S Series) WiFi gateway

The printer now serves its own web page onto my home network and allows me to control and monitor the printer from anywhere. BIG bonus is that I can connect and disconnect from the printer, during a print job, without having the control board reboot itself and ruin the print job mid way through. In past I tried connecting a computer and also a raspberry pi (running Octoprint) to my printer and there is a very high chance that - at some stage during a 10hour print – it loses serial connection and cause a board reboot.

Not bad for \$2

1. **Watch** Add WiFi To Your 3D Printer For \$5!!! - ESP8266 - LUC ESP3D - Chris's Basement FIRST  
<https://www.youtube.com/watch?v=pJGBRriNc9I>
2. **Read** <https://github.com/luc-github/ESP3D/wiki/Install-Instructions>
3. Make sure you can access <https://github.com/luc-github/ESP3D>
4. Install Arduino 1.8.5
5. In Arduino preferences, add a new URL



[http://arduino.esp8266.com/stable/package\\_esp8266com\\_index.json](http://arduino.esp8266.com/stable/package_esp8266com_index.json)

6. In Arduino install drivers for the WSP8266 board

**esp8266 by ESP8266 Community version 2.4.0 INSTALLED**

Boards included in this package:

Generic ESP8266 Module, Generic ESP8285 Module, ESPDuino (ESP-13 Module), Adafruit Feather HUZZAH ESP8266, Invent One, XinaBox CW01, ESPresso Lite 1.0, ESPresso Lite 2.0, Phoenix 1.0, Phoenix 2.0, NodeMCU 0.9 (ESP-12 Module), NodeMCU 1.0 (ESP-12E Module), Olimex MOD-WIFI-ESP8266(-DEV), SparkFun ESP8266 Thing, SparkFun ESP8266 Thing Dev, SweetPea ESP-210, LOLIN(WEMOS) D1 R2 & mini, LOLIN(WEMOS) D1 mini Pro, LOLIN(WEMOS) D1 mini Lite, WeMos D1 R1, ESPino (ESP-12 Module), ThaiEasyElec's ESPino, WifInfo, Arduino, 4D Systems gen4 IoD Range, Digistump Oak, Wifiduino, Amperka WiFi Slot, Seeed Wio Link, ESPectro Core.

7. Download the Github project to suit Arduino 1.8.5 and WSP8266 library 2.4.0

<https://github.com/luc-github/ESP3D/releases/tag/1.0>

Stable version (ESP8266 only):

Arduino ide 1.6.5 with stable 2.0.0 from ESP8266, please use <https://github.com/luc-github/ESP3D/releases/tag/v0.5.1>

Arduino ide 1.6.8 with stable 2.2.0 from ESP8266, please use <https://github.com/luc-github/ESP3D/releases/tag/v0.6.2>

Arduino ide 1.8.5 with stable 2.4.0 from ESP8266, please use <https://github.com/luc-github/ESP3D/releases/tag/1.0>

build passing

Arduino ide 1.8.5 with with 2.5.2 version of ESP8266 and git version of ESP32, please use <https://github.com/luc-github/ESP3D/releases/tag/2.0>

build passing

Development version for 2.1 & ESP-WEBUI (2.1 branch):

Arduino ide 1.8.9 with 2.5.2 version of ESP8266 or git version ESP32 for 100% support of ESP32 : build passing

8. And then put the unzipped project in the Arduino sketches folder

Clipboard	Organise	New	Open
↑ > This PC > Mirror Disks (A:) > Arduino 1.8.5 > Arduino > Sketches			
access	Name	Date modified	Type
top	ESP3D-1.0	23/08/2019 6:55 PM	File folder

9. Set up Arduino board as:

- Generic ESP8862 Module
- Flash size 1M (512K SPIFFS)
- Flash 40MHz
- Baud 115,200

The above suits the below (which I got by interrogating the ESP-01S by loading the program as described in <https://github.com/luc-github/ESP3D/wiki/Flash-Size>)

Open the Arduino IDE

Click File, Examples, ESP8266, CheckFlashConfig

Upload the sketch to the ESP8266

View the Serial Monitor (115200 baud)

(NB: If you dont get a 'Flash Chip configuration ok.' uploading will appear to work succesfully but the chip will crash on startup and never show an access point / serial output)

```
Flash real id:    00146085
Flash real size: 1048576 bytes

Flash ide  size: 1048576 bytes
Flash ide speed: 40000000 Hz
Flash ide mode:  DOUT
Flash Chip configuration ok.
```

10. Download the tool - this allows Arduino to upload the file system to the ESP-01S (as distinct from the program itself). The file system is used by the webserver – I think

<https://github.com/esp8266/arduino-esp8266fs-plugin/releases/download/0.4.0/ESP8266FS-0.4.0.zip>

11. Then extract the folder ESP8266FS and put it in the Arduino Tools folder

Clipboard	Organise	New	Open	Select
↑ > This PC > Mirror Disks (A:) > Arduino 1.8.5 > Arduino > tools				
access	Name	Date modified	Type	Size
top	ESP8266FS	23/08/2019 6:35 PM	File folder	
downloads	Mangler	29/11/2018 8:15 PM	File folder	
Documents	WiFi101	29/11/2018 8:15 PM	File folder	
References	howto.txt	3/10/2017 12:37 AM	Text Document	6 KB

12. Now check that you can see is the Arduino IDE, under Tools tab a new item 'ESP8826 Sketch Data Upload'. We will use this after sending the program to the chip

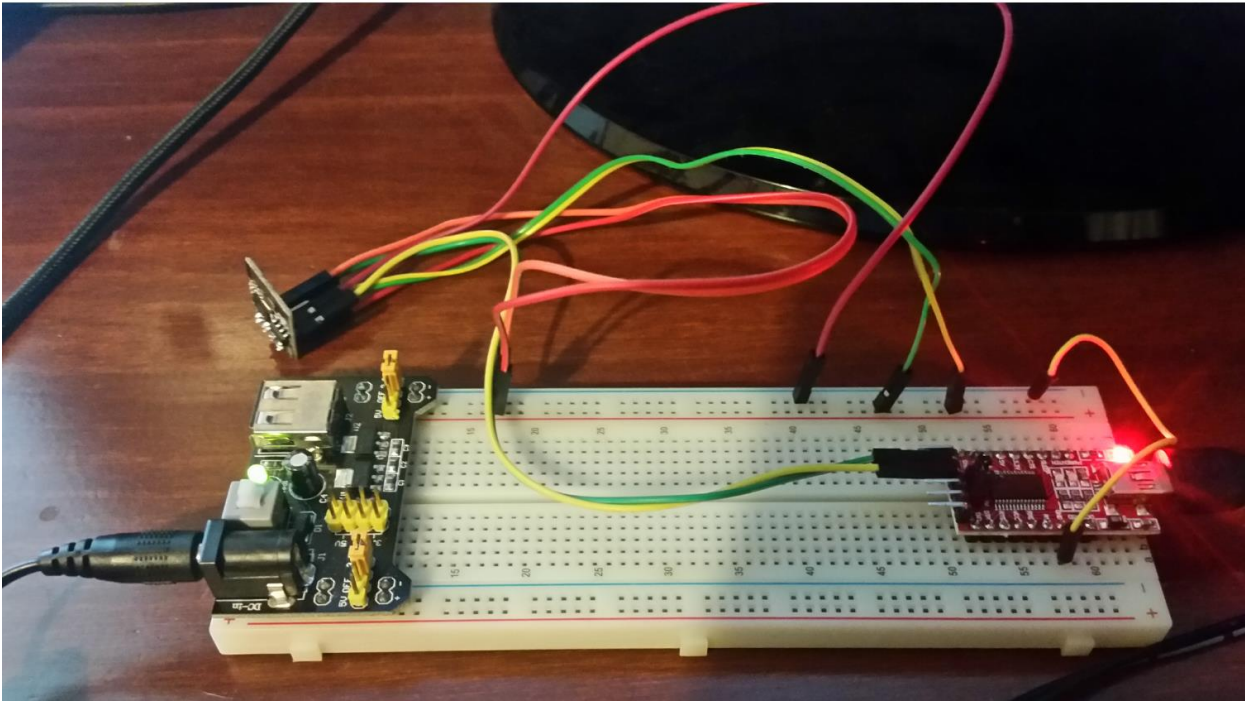
13. Set up a FTDI (USB to TTL) interface, making sure it is 3.3V not 5v type

14. Connect up the ESP-01S

- a. Connect FTDI Tx to ESP-01S Rx
- b. Connect FTDI Rx to ESP-01S Tx
- c. Connect FTDI and ESP-01S grounds together and to an external 3.3V supply ground
- d. Connect FTDI and ESP-01S 3.3V input pins to the external 3.3V supply +ve side
- e. Connect ESP-01S IO0 pin to ground
- f. Connect ESP-01S RESET pin to 3.3V
- g. Connect ESP-01S ENABLE pin to 3.3V

15. Connect FTDI USB port to the PC

16. Turn on 3.3V supply



17. Check you can compile the project (ie open file esp3d.ino) in Arduino 1.8.5

Done compiling.

```
Archiving built core (caching) in: C:\Users\Main\AppData\Local\Temp\arduino_cache_146015\core\core_esp8266_esp8266_generic_Cpu
Sketch uses 372795 bytes (74%) of program storage space. Maximum is 499696 bytes.
Global variables use 39468 bytes (48%) of dynamic memory, leaving 42452 bytes for local variables. Maximum is 81920 bytes.
```

18. Load the program to the ESP-01S (check you have set IO0 pin to ground, and the enable and reset pins to 3.3V so this will work). If successful you should see

Done uploading.

```
Archiving built core (caching) in: C:\Users\Main\AppData\Local\Temp\arduino_cache_146015\core
Sketch uses 372795 bytes (74%) of program storage space. Maximum is 499696 bytes.
Global variables use 39468 bytes (48%) of dynamic memory, leaving 42452 bytes for local varia
Uploading 376944 bytes from C:\Users\Main\AppData\Local\Temp\arduino_build_155641/esp3d.ino.b
..... [ 21% ]
..... [ 43% ]
..... [ 65% ]
..... [ 86% ]
..... [ 100% ]
```

19. Reboot board and the load the filesystem (see 12 above). If successful you should see

```
[SPIFFS] data      : A:\Arduino 1.8.5\Arduino\Sketches\ESP3D-1.0\esp3d\data
[SPIFFS] size      : 512
[SPIFFS] page      : 256
[SPIFFS] block     : 8192
/404.htm

/favicon.ico

/index.html.gz

/macrocfg.json

[SPIFFS] upload    : C:\Users\Main\AppData\Local\Temp\arduino_build_155641\esp3d.spiffs.bin
[SPIFFS] address   : 0x7B000
[SPIFFS] reset     : ck
[SPIFFS] port      : COM31
[SPIFFS] speed     : 115200

Uploading 524288 bytes from C:\Users\Main\AppData\Local\Temp\arduino_build_155641\esp3d.spiffs.bin to flash at 0x0007B000

..... [ 15% ]
..... [ 31% ]
..... [ 46% ]
..... [ 62% ]
..... [ 78% ]
..... [ 93% ]
..... [ 100% ]
```

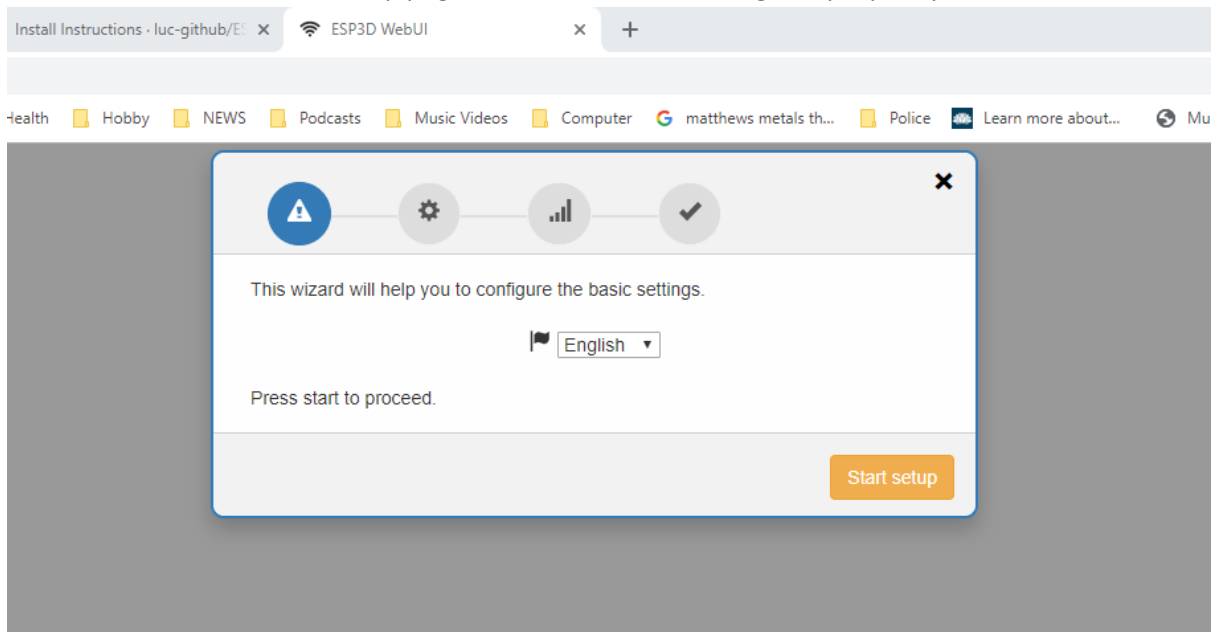
20. Now connect ESP-01S IO0 pin to 3.3V to get out of programming mode and reboot the ESP-01S

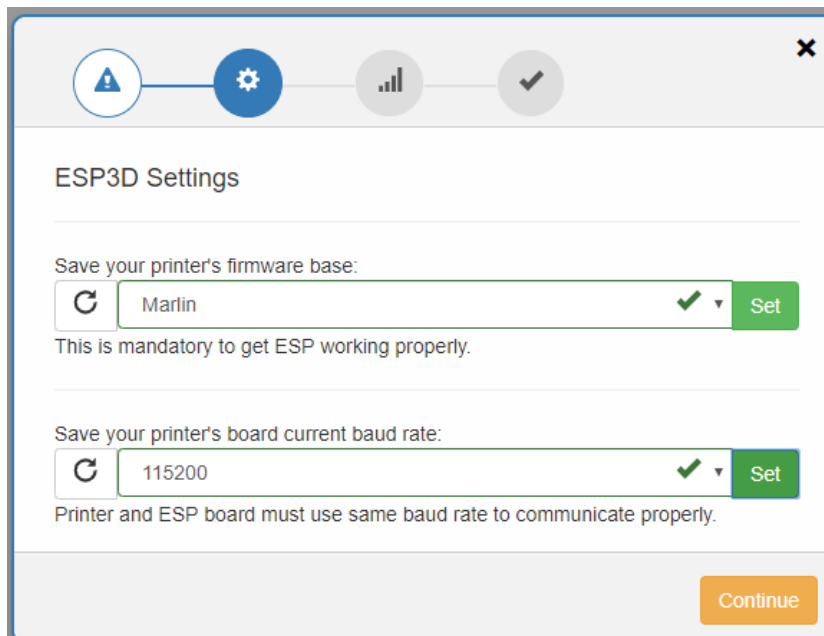
21. If you look the the Arduino serial monitor you should see M117 192.168.0.1

22. Unplug USB and FTDI (no longer needed) and on the PC check for a new WiFi network SSID ESP3D

23. Log into this with password 12345678

24. You should then see the UI setup page. Go ahead and run though steps (pretty obvious)





ESP3D Settings

Save your printer's firmware base:

☒

This is mandatory to get ESP working properly.

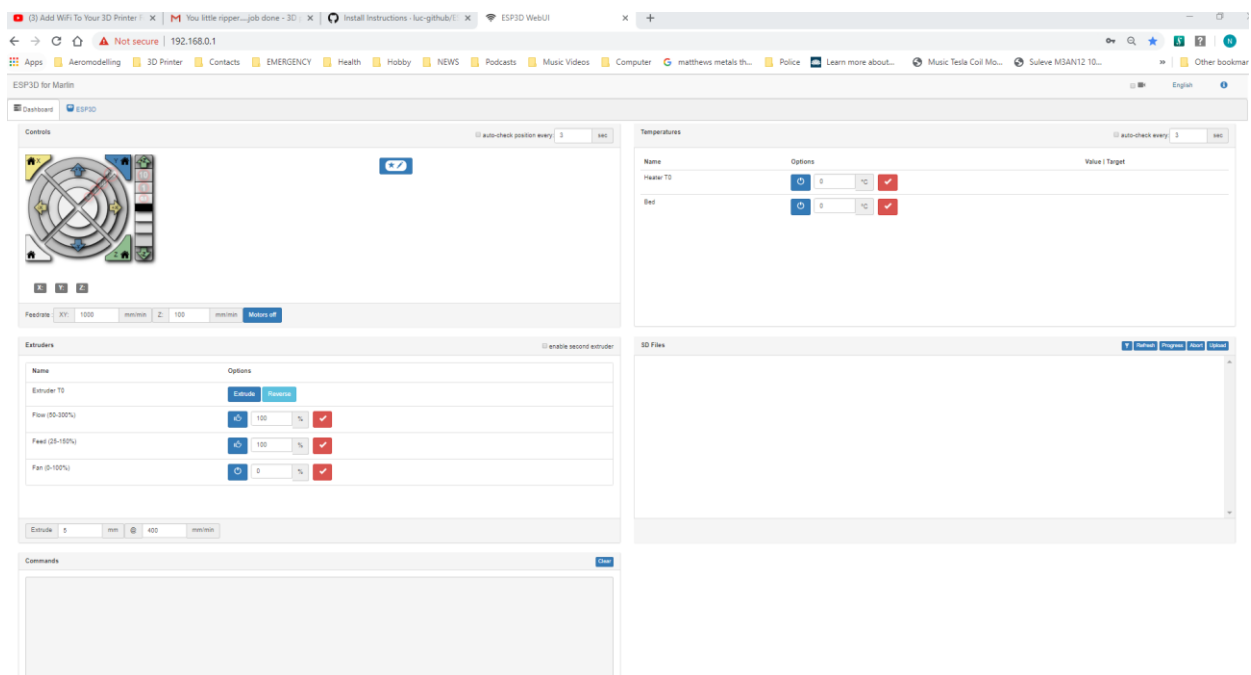
Save your printer's board current baud rate:

☒

Printer and ESP board must use same baud rate to communicate properly.

Etc etc

25. Should now get access to the UI. Pretty intuitive to use and configure



26. Last step – depower and unplug jumper wires. Plug the ESP-01S into the BTT 32bit board and power up. Gives you WiFi access without risk of having the board do a reboot ½ way thorough a print – the way it does if you connect a serial host in the middle of a print job.

References:

<https://www.youtube.com/watch?v=pJGBRriNc9I>

<https://github.com/luc-github/ESP3D>

<https://github.com/luc-github/ESP3D/wiki>

<https://github.com/luc-github/ESP3D/wiki/Install-Instructions>

<https://github.com/esp8266/Arduino/blob/master/doc/filesystem.rst#uploading-files-to-file-system>