Hello World : ESP32 Firmware – DelSquared – Medium

Notebook: First Notebook
Created: 02-08-2018 18:37
URL: https://medium.com/@DelSquared/hello-world-esp32-firmware-38b604c7a1f1

Hello World : ESP32 Firmware

We're currently aiming to make our contribution to the Tessel project. Tessel-Reach is a developing concept that we recently got introduced to.

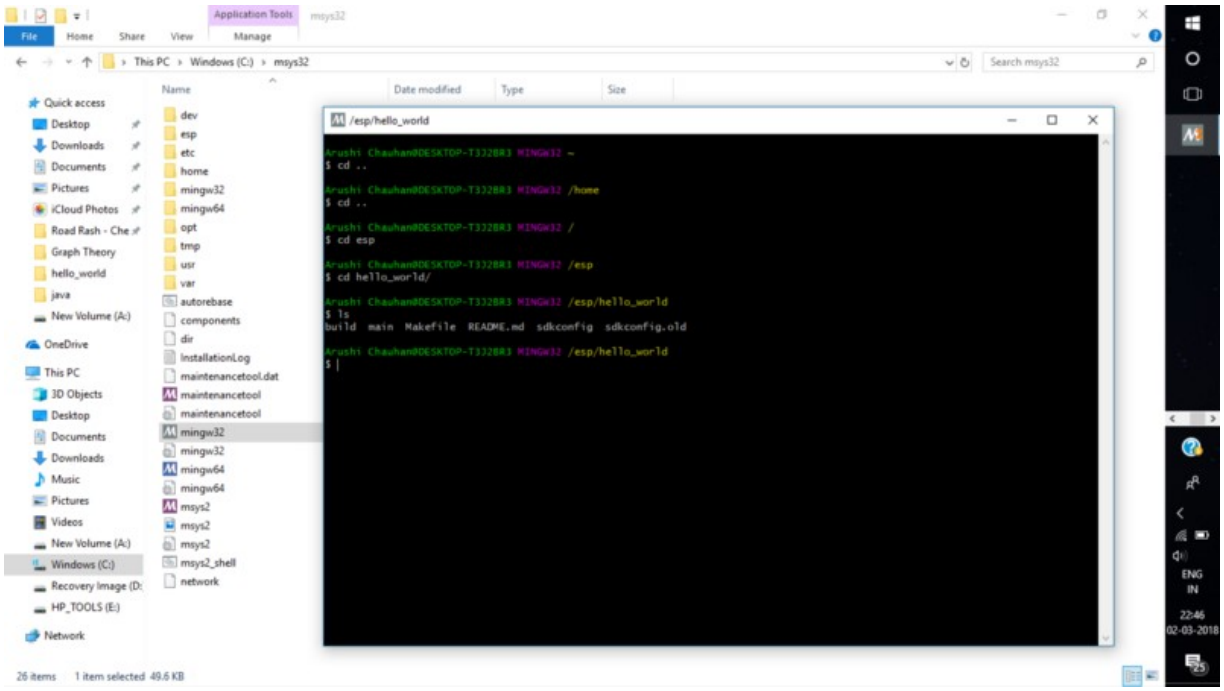
It aims at creating a star topology network with a Tessel 2 board as the hub and Reach modules as low power nodes, that are currently being configured on an ESP32 Thing.

The latest development on this project was to push a hello world program onto the firmware of ESP32. We began by achieving the same, while following the ESP-IDF tutorial, in order to familiarise ourselves with the platform we would be working on.

Connecting the ESP32 module to the laptop was just about plugging it into a USB port via a micro-USB cable. But that is not enough to run the module on Windows 10. We need to setup a toolchain so that we can 'talk' with the sensor i.e. push code to the sensor and make it do some exciting stuff :D

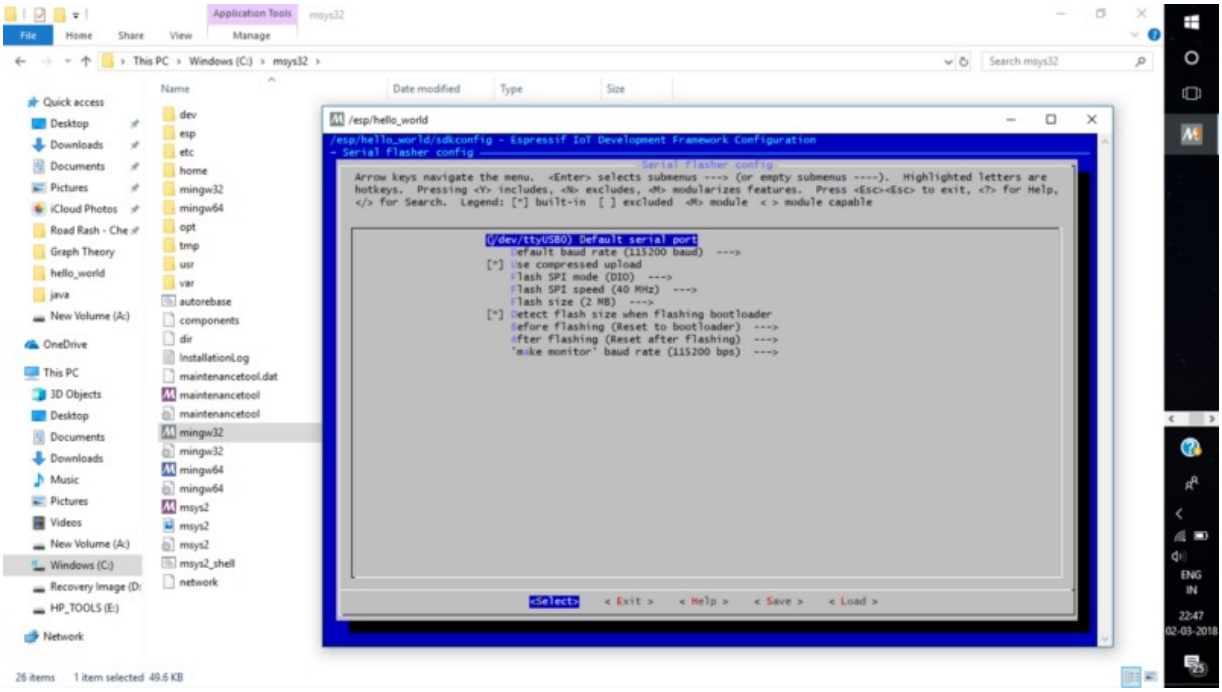
Here's how we went about it.

1. We kick-started the process by **setting up the toolchain** for the ESP32 module. This took about an hour — first downloading, then unzipping the compressed file. The toolchain provided an environment to interact with the ESP32 module on Windows 10 through minGW32 CLI. IDF_PATH was set as the system environment variable pointing to to the 'esp-idf' folder in esp.

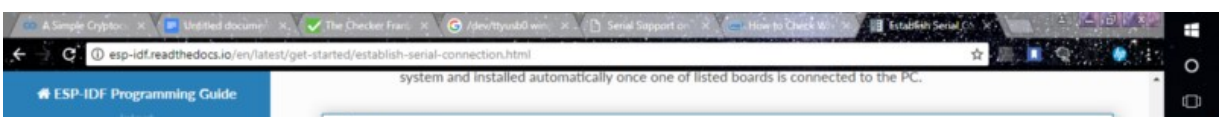


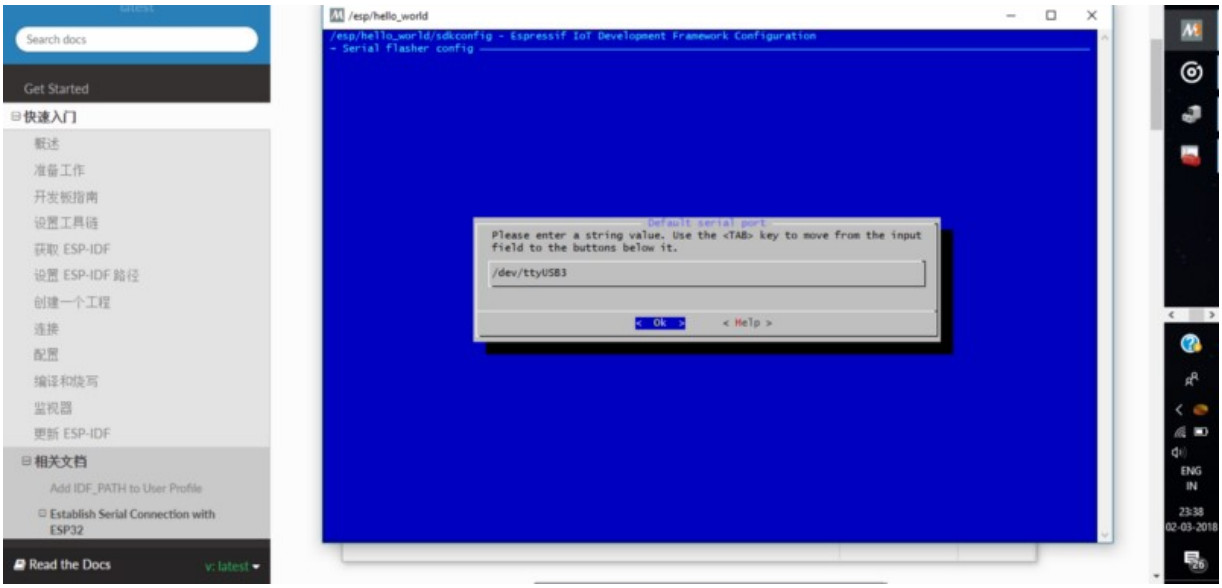
2. The hello world folder loaded into the msys32 folder. We had to the set IDF_PATH correctly. It was initially

set to the esp folder but it worked only with the **esp-idf folder**. The path given on the website did not work.

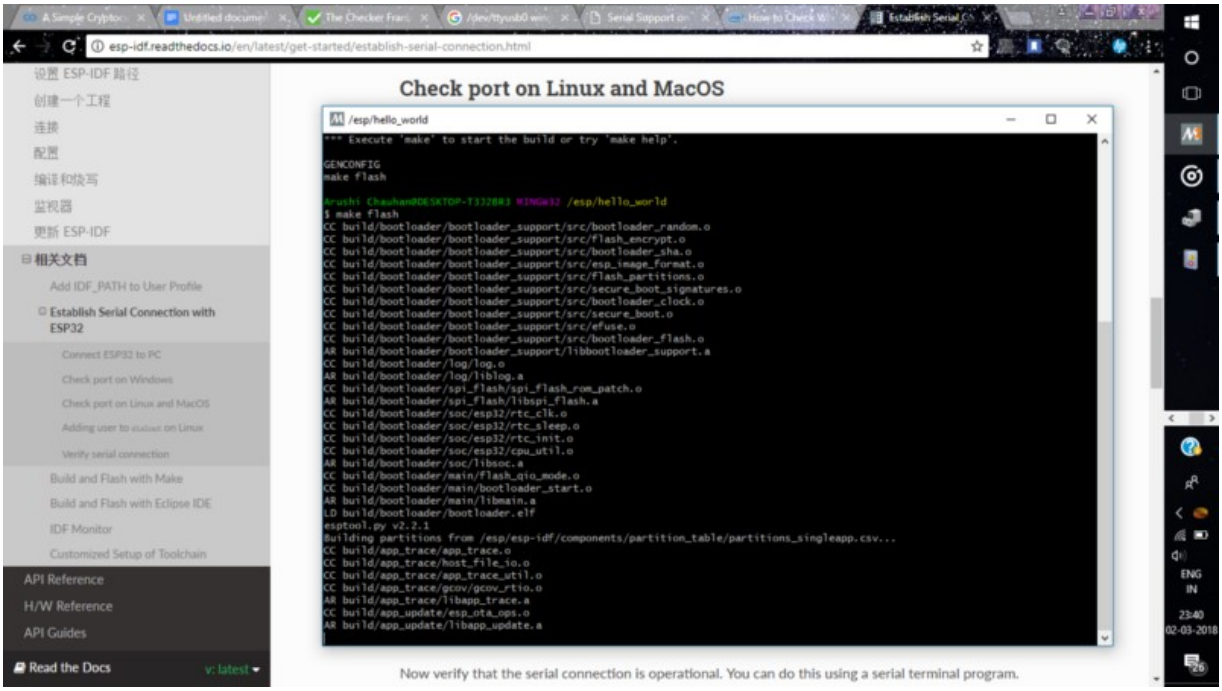


3. We then gave the **make menuconfig** command and selected the serial flash option. Following that, we selected the default serial port, which did not work. We had to do some hit and trial to figure out the correct port. We tried /dev/ttyUSB0, /dev/ttyUSB1,.....,/dev/ttyUSB3, /dev/ttyS2 /dev/tty. The device manager in Windows 10 showed the connected port to be COM3. We googled around and it finally worked at COM3 (/dev/ wasn't required).

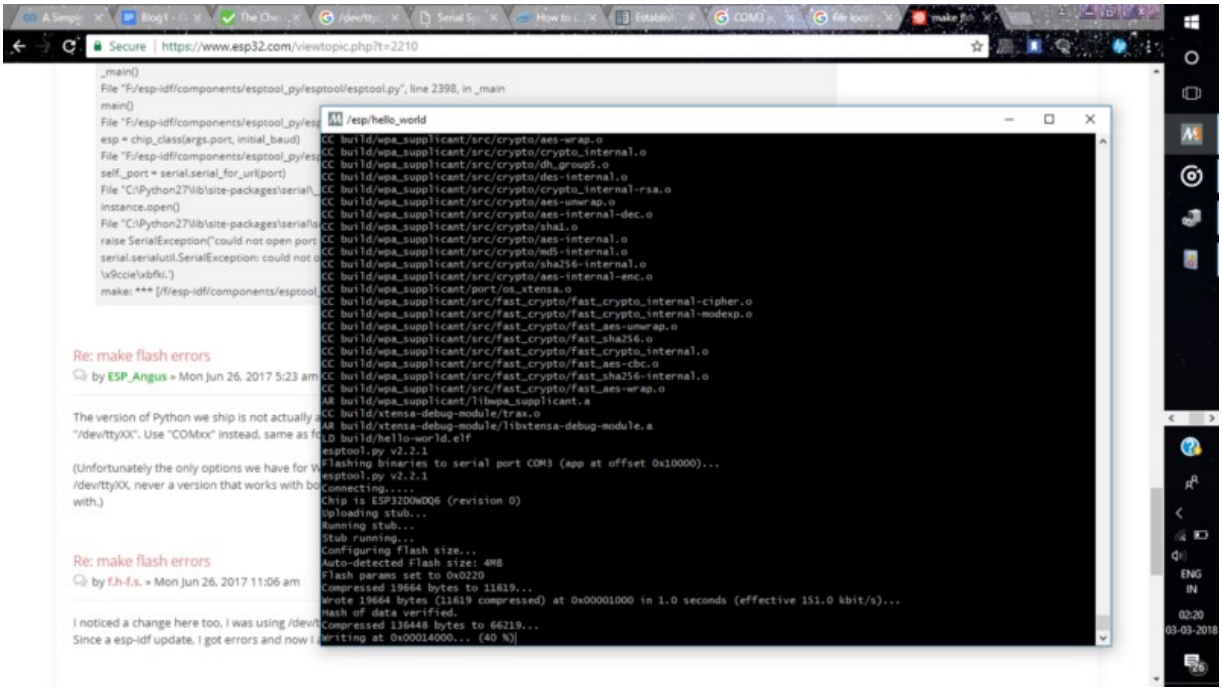




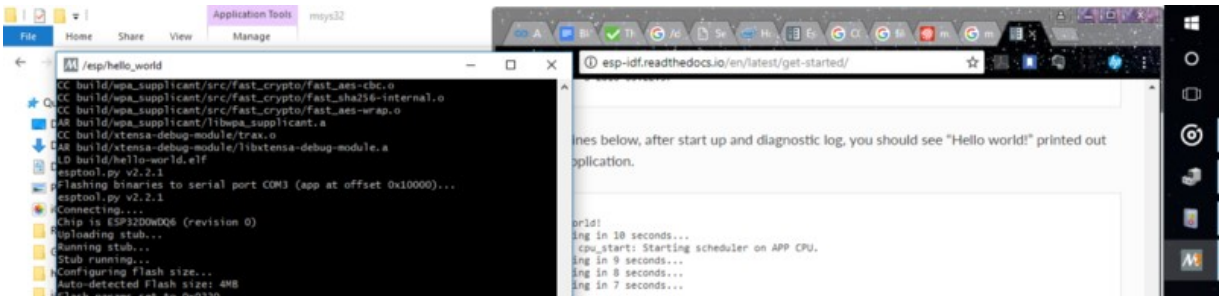
4. The next command we gave was **make flash**. We then waited for the process to complete. The **sdkconfig file was generated.**

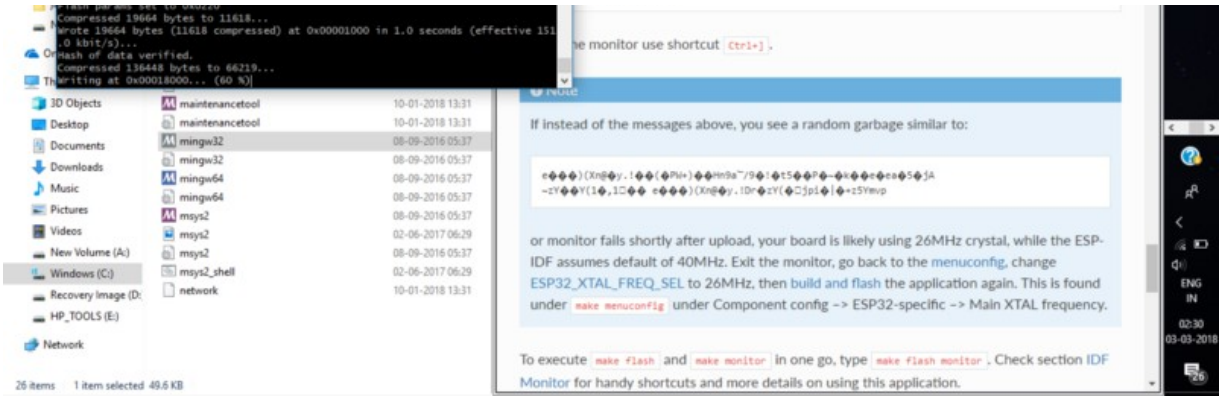


5. Upon its completion, this is what we got.

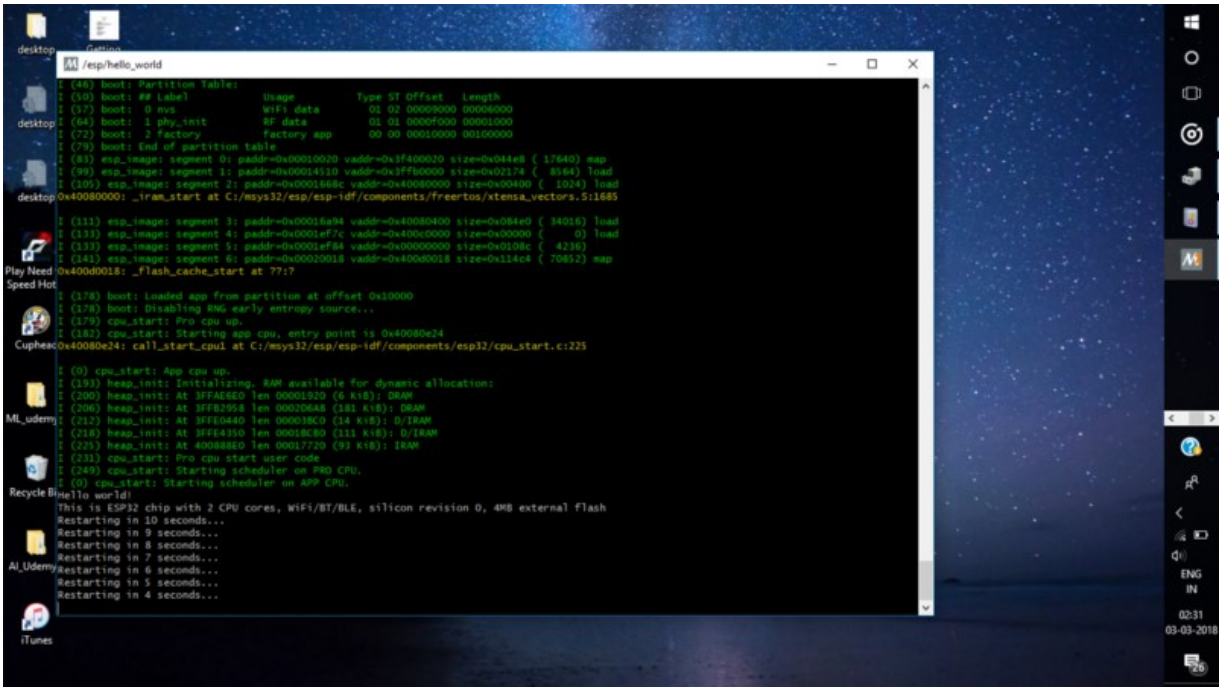


6. We then followed with the command **make monitor**. After doing so, the screen showed some garbled characters — so on referring to the guide again, we set the **XTAL frequency** to 26 Hz instead of 40 Hz in the component configuration menu under the serial flash option (**make menuconfig**).





7. The process was finally complete and we got **'hello world'** up and running on our screen!



Till we meet again,

Cheers!