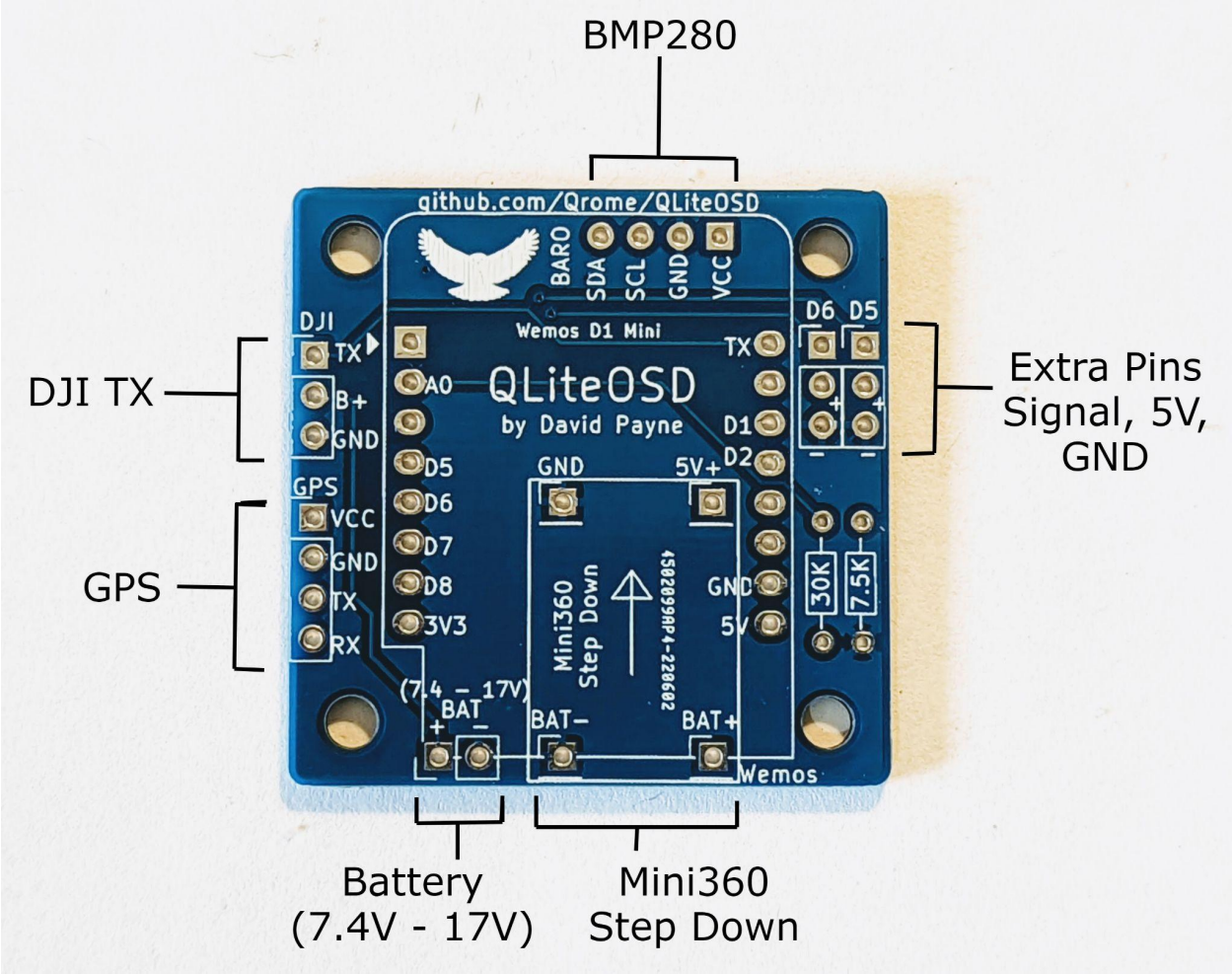
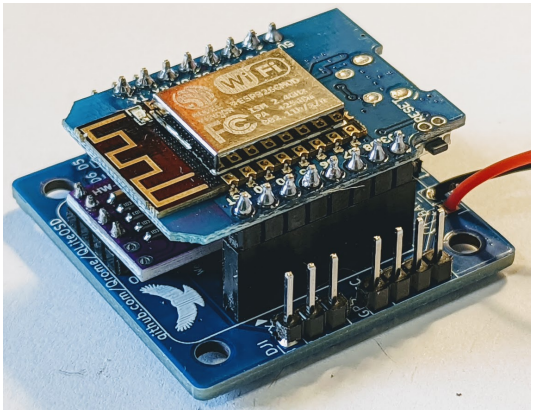
QLiteOSD Board v1.0 Assembly



Introduction

The **QLiteOSD** is an Arduino based On Screen Display (OSD) for DJI FPV transmitter systems when you want all the OSD and none of the Flight Controller. This is a less expensive option for gaining full OSD for DJI FPV when you don't need, or want a Flight Controller.

Features

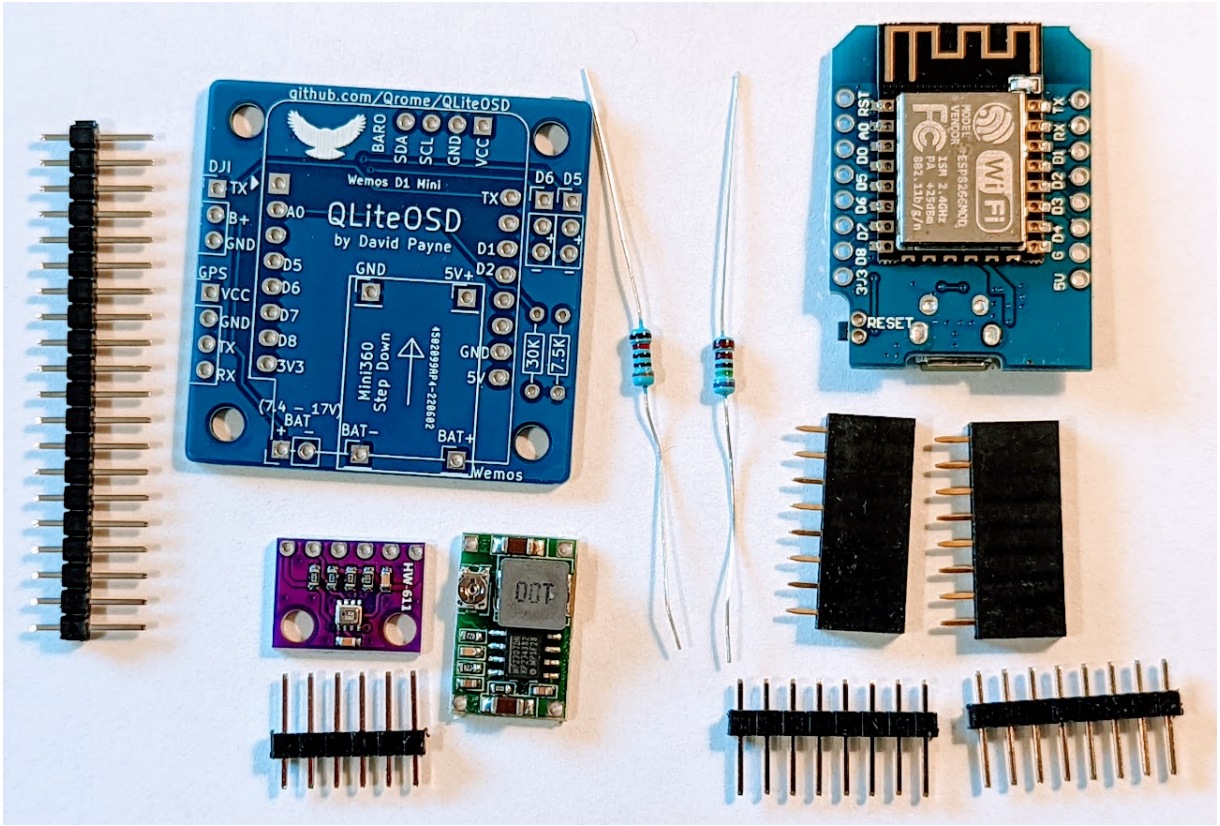


- Flight Pack Voltage (7.4 to 17V)
  - Per cell voltage
- Powers DJI Air Unit or Caddx Vista (B+ pin)
- Altitude
- GPS (optional and not included) Supports:
  - Latitude
  - Longitude

- Home Arrow
- Ground Speed
- Distance From Home
- GPS Logging of last 10 flights (v1.4)

Basic Kit

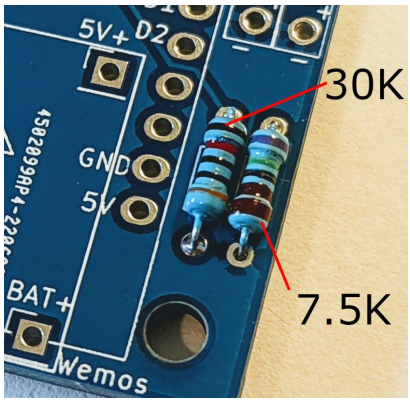
- **QLiteOSD** Main Board
- **Wemos D1 Mini** (ESP8266) Micro Controller
- 1 X **30K Resistor**
- 1 X **7.5K Resistor**
- **Mini360** Voltage Step Down to 5.5 to 6.0V
- **BMP280** (3v3 I2C) board - Barometer Sensor
- Header Pins

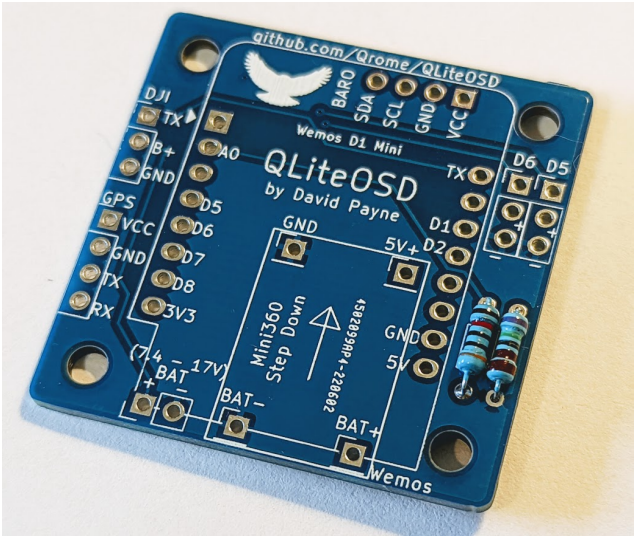


QLiteOSD Build Video: https://www.youtube.com/watch?v=reYGkunSCh4

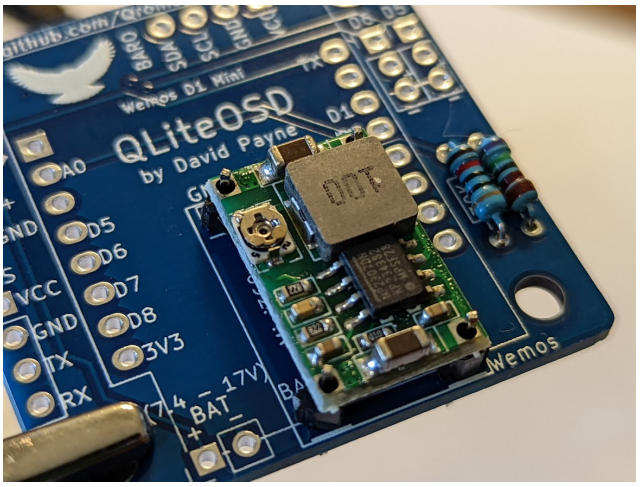
Assembly Steps

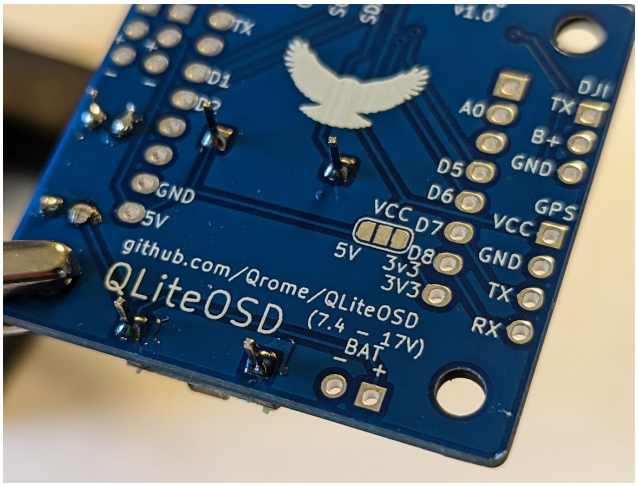
1. Add the **30K** and **7.5K Resistors** and solder the back. Trim the excess wire off the back.



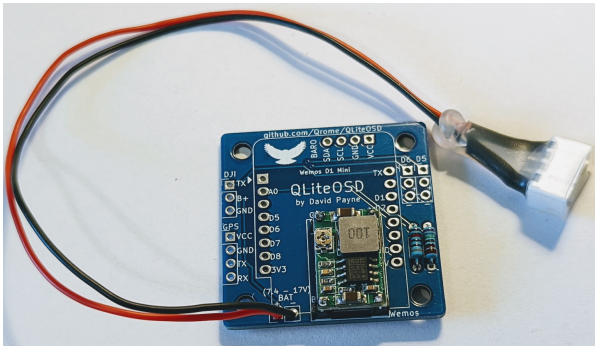


2. Add the **Mini360** Step Down board.
    a. Clip 4 individual header pins.
    b. Place them in the QLiteOSD Board long pin down through the hole.
    c. Place the **Mini360** onto the short ends of the pins.
    d. Solder the pins on the **Mini360**
    e. Solder the long pins on the back of the QLiteOSD board.
    f. Trim the long pins on the back.

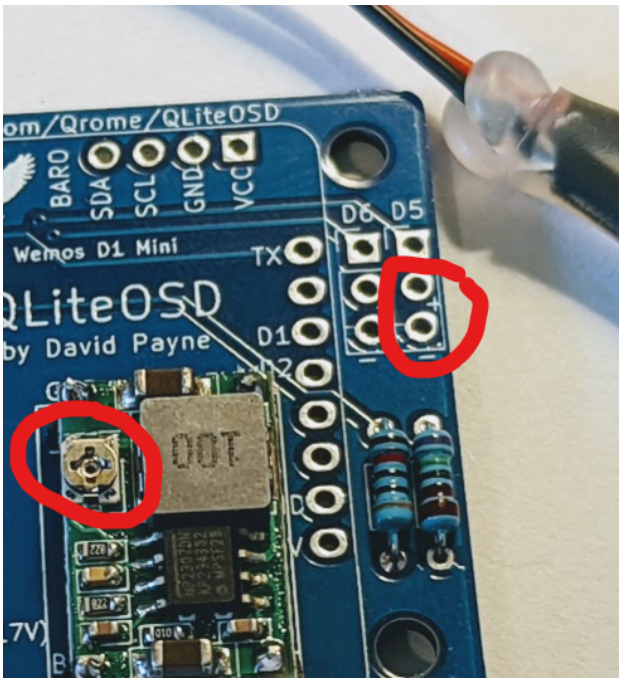


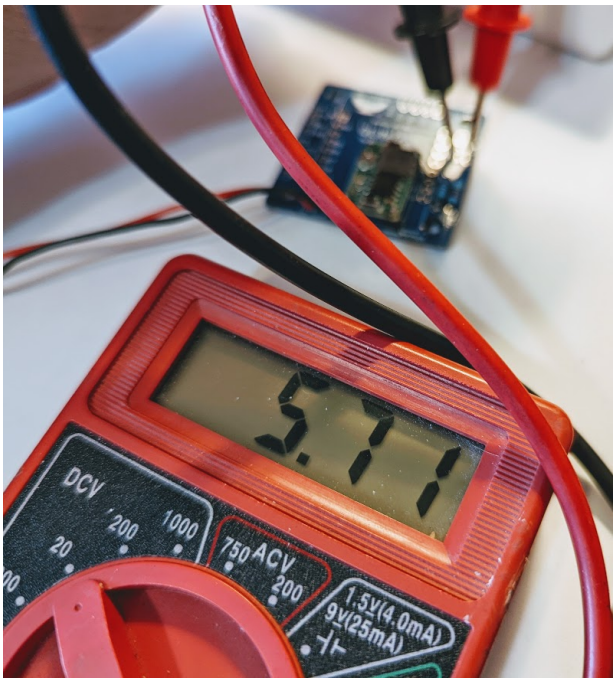


3. Solder your flight battery pack connector (of your choice). Note: this can be **7.4V to 17V** and will be the battery that the voltage will be read from. This battery will also power your DJI FPV transmitter unit.

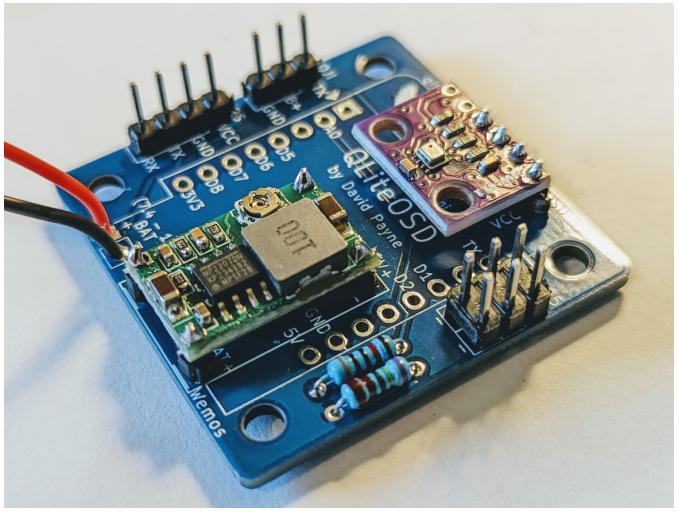


4. Adjust the voltage out from the **Mini360**. This should be set in the range of **5.5V to 6.0V** – it is not recommended to go much over 6.0V. If you have it set to this range the default voltage reading will be quite accurate.
    a. Connect your flight battery
    b. Adjust the dial potentiometer with a small flat screwdriver by turning it all the way to the right then back left as you test the voltage on the 5V and GND pins on the QLiteOSD board (see pins circled below).
    c. Output Voltage should be somewhere in the **5.5V to 6.0V** range.

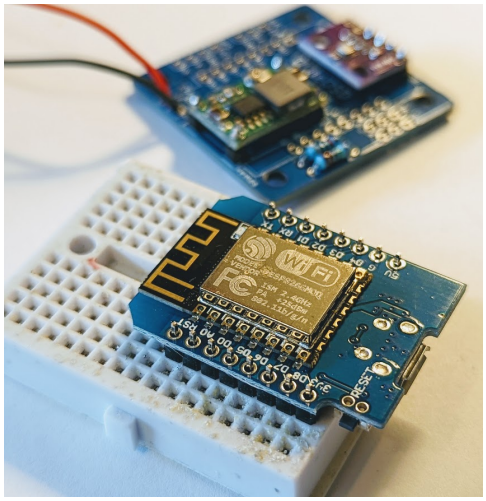


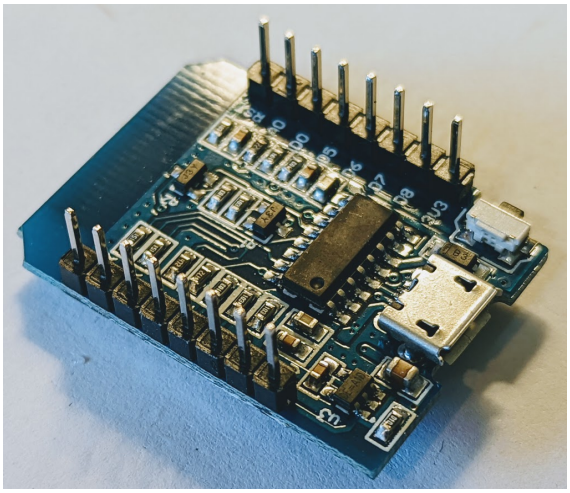


5. Solder on the **BMP280** (3v3) altitude sensor board using 4 pins. VCC, GND, SCL, and SDA found on the QLiteBoard marked by the word BARO. This board uses the I2C pins. Make sure the pins on the **BMP280** are aligned perfectly with the QLiteOSD. Note: VCC is defaulted to running 3.3V from the Wemos D1 Mini. (Solder bridge not required).
    a. Place the 4 pins long ends down through the QLightOSD
    b. Place the **BMP280** over the short ends ensuring VCC pins line up between the boards. Solder the 4 pins on **BMP280** and back of the main board.
    c. Trim the 4 long pins on the back of the main board used to connect the **BMP280**.
    d. Solder the rest of the header pins to the boards unless you plan to solder wires directly.

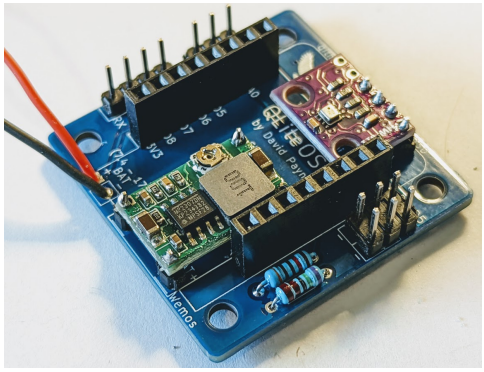


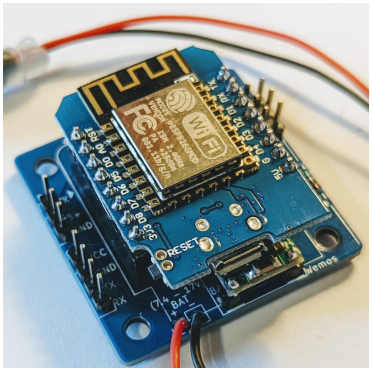
6. Solder the male header pins onto the **Wemos D1 Mini** (ESP8266) board.
    a. Put the header pins into a breadboard to help ensure they are correctly spaced and lined up straight.
    b. Solder pins.

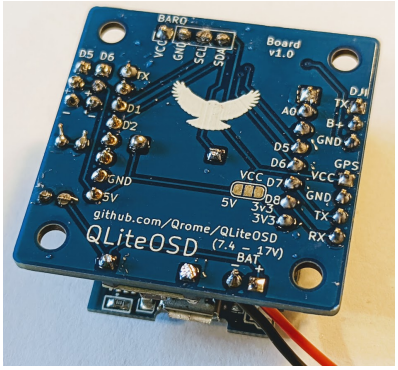




7. Pace the female header pins for the **Wemos D1 Mini** into the QLiteOSD board and hold them in place by putting the **Wemos D1 Mini** into the socket. Solder the female header pins on the bottom of the QLiteOSD board.





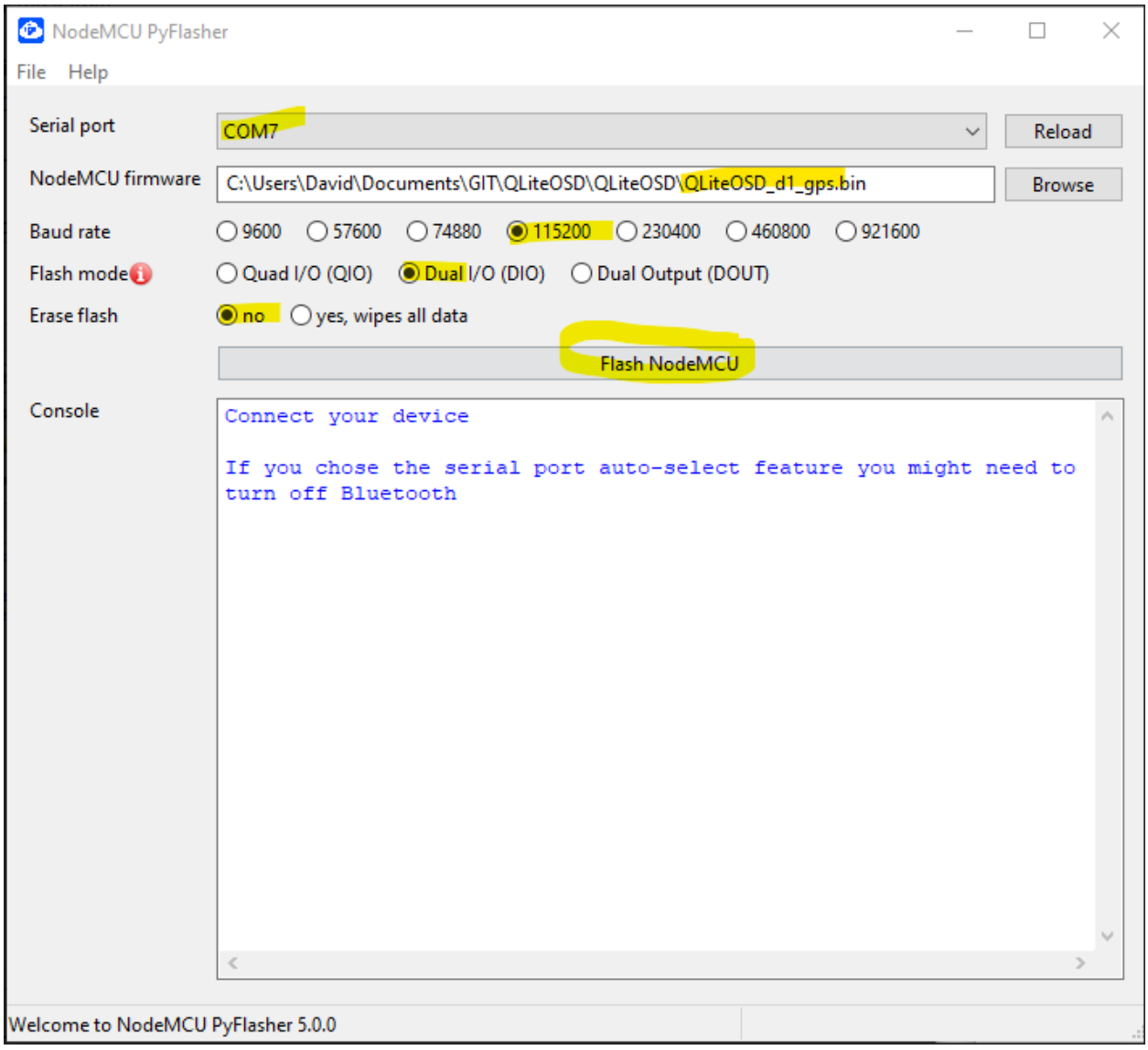


Assembly complete. Now you need to load the firmware to the **Wemos D1 Mini** (ESP8266). This can be done by loading a pre-compiled binary file or by downloading the QLiteOSD source code and compiling and loading it to the board.

Loading Firmware from Binary File

1. Load the CH340 USB drivers so that the Wemos D1 mini will show up on your system.

   a. Download CH340 drivers from SparkFun: https://learn.sparkfun.com/tutorials/how-to-install-ch340-drivers/all#drivers-if-you-need-them
   b. Install by following the prompts.
2. Download and install the NodeMCU Pyflasher tool from GitHub: https://github.com/marcelstoer/nodemcu-pyflasher/releases/tag/v5.0.0
   a. Extract to a folder on your computer where you can run it from.
3. Plug in the USB port to the Wemos D1 Mini – the board should show up on Windows as a COM port with a number next to it. (i.e. COM3)
4. Run the NodeMCU-Pyflasher tool.



   a. Select the Serial Port (the new com port created by the device being plugged in).
   b. Browse and select the QLiteOSD firmware compiled binary file.
   c. Baud rate 115200
   d. Dual I/O

    e. Erase flash - no
    f. Press the “Flash NodeMCU” button.
    g. Wait for it to load the firmware 100%
5. Unplug and plug it back in again. If the QLiteOSD blue LED is flashing steady, then it is ready to be used.

Note: It is not advised to leave the DJI VTx on for any extended period of time. If you have it sitting for more than a couple of minutes, you should put a fan on the DJI VTx to help keep it cool.

*This is a kit. Basic knowledge of electronics is required and some skill for soldering and working with battery powered systems is required. THE COMPANY SHALL NOT, IN ANY CIRCUMSTANCES, BE LIABLE FOR SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES, FOR ANY REASON WHATSOEVER.

FAQ

Will the QLiteOSD send the armed command to the DJI Video Transmitter?

- Yes, the arm signal is sent when the OSD detects a change of +1.5 meters (about 5 feet) above the altitude the OSD was powered on with. This is nice as it will tell the DJI system to start recording and bump the transmit power up if you are using auto power on arming. This does not require GPS.

Is GPS required to use the QLiteOSD?

- No, the GPS is not required. If you choose not to use the GPS, then you may want to consider using the NOGPS binary or compile the source with the defined USE_GPS commented out. This will remove the items on the OSD that are GPS related.

Can I move the locations of the OSD elements around in the goggles?

- Yes, it would require editing the OSD_positions_config.h and referencing the mapping image to define the locations. You can enable or disable elements by the number assigned to it. See the notes in that file. After making changes to the config file you would then compile and load the binary to your OSD.

Can I have RSSI or LQ in the QLiteOSD?

- This is currently not supported. Often this can be setup with audible alerts in your own radio. The QLiteOSD does not currently require any connection from the aircraft receiver.

Can I adjust the reading of the pack voltage?

- Yes. The best way to get an accurate reading of the pack voltage is to adjust the output voltage of the Mini360 step down to as close to 6V as you can. When the Wemos D1

mini board is powered with 6V, the internal system 5V is very close to 5V and this is used in the calculation.

How do I enable the QLiteOSD GPS logging feature?

- When using the GPS and the default compiled binary, it is enabled by default and every flight should log to the internal memory on the Wemos D1 Mini. To disable logging, compile the source code with #define LOG_GPS commented out in the OSD_positions_config.h file. Logging starts when the altitude changes + 1.5 meter above the powered on altitude and there is a GPS lock. Note: Starting in v1.4 the device will only log the last 10 flights.

How do I download the GPX (GPS Log files) from the QLiteOSD?

- Put the QLiteOSD device into WiFi Mode, join the Access Point and download the files.
    - When the OSD is on, jumper the Wemos D1 Mini D3 pin with GND for 3 seconds. The light will turn solid indicating it is now in WiFi Mode and is visible as an Access Point.
    - QLiteOSD_xxxxx – where the xxxxxx is the unique serial number on the chip will show. Join your phone or computer to the Access Point using the default password 12345678.
    - After connected to the divide over wifi, pull open a web browser to the following address: http://192.168.4.1 – this will pull up the logged files.
- There are several free GPX Log file viewers including GeoTracker for Android. Google Earth will also view them.
- GPS logging in QLiteOSD v1.4 has been updated to not fill the memory and only saves the last 10 flights. It is possible for the GPS units in the OSD to freeze or get sluggish when the device memory is filled. Update to v1.4 if running an older version.