Connecting a Raspberry Pi Zero W to an emonTx

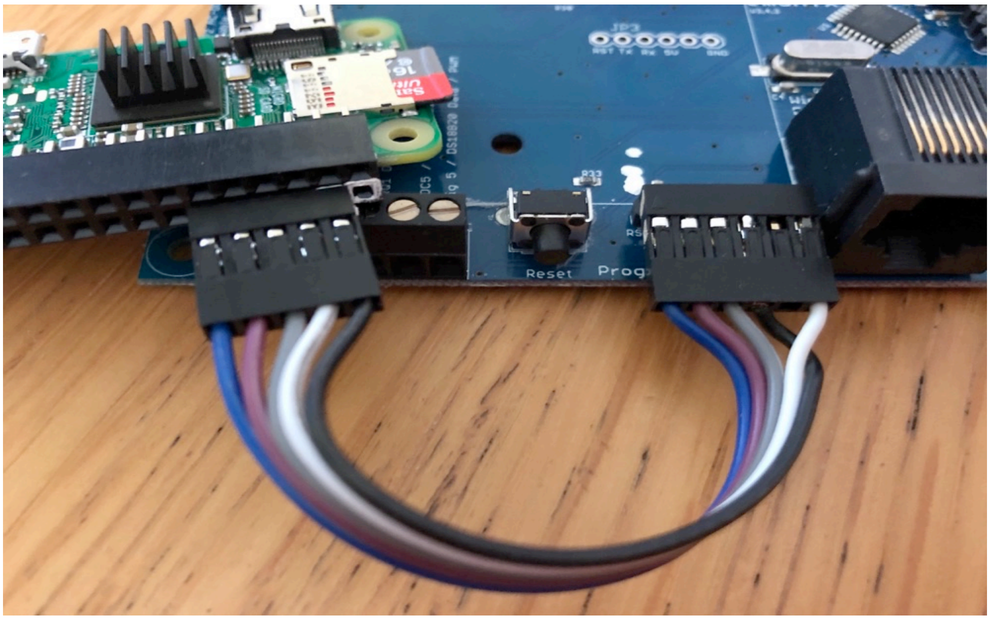
NB: Written in October 2020, and version numbers etc may have changed by the time you read this.

A brief "How To for Total Beginners" if you want to swap out an ESP8266 wifi chip attached to an emonTx in favour of using a Raspberry Pi Zero to transmit sensor data via wifi to your emonPi instead. I wanted to do this as I had had variable performance with the ESP8266 – both in terms of robustness of connectivity, and in wifi range. The following is based on my experiences, using macOS to programme the microSD card, but I am sure it is very similar to what you would do on a Windows PC.

I must thank the very kind (& patient) folk on the open energy monitor forums for their seemingly un-tiring help https://community.openenergymonitor.org ; it took me quite some time to get it all to work as I wanted, and this was the reason I thought that I would share the steps I took.

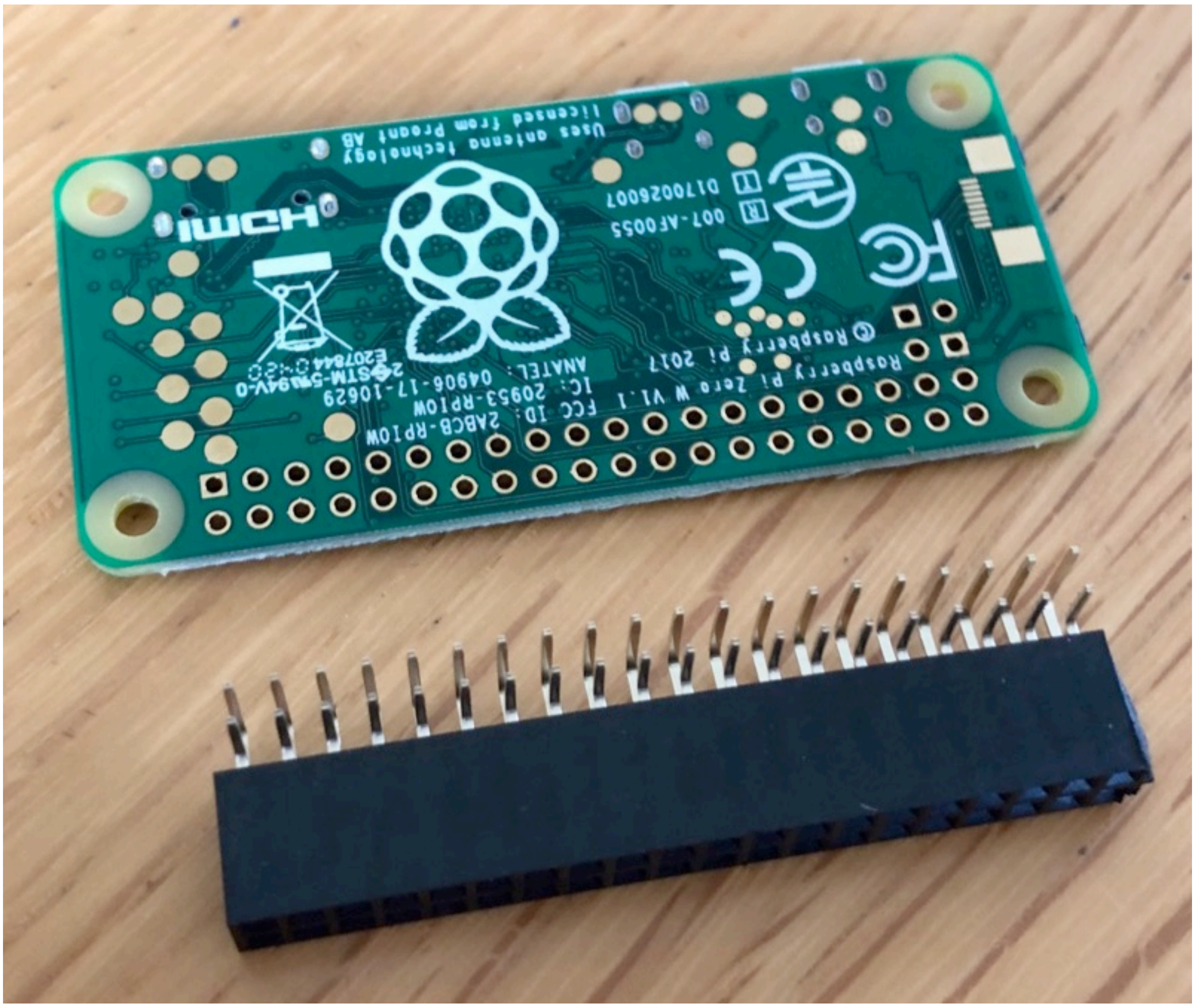
Requirements

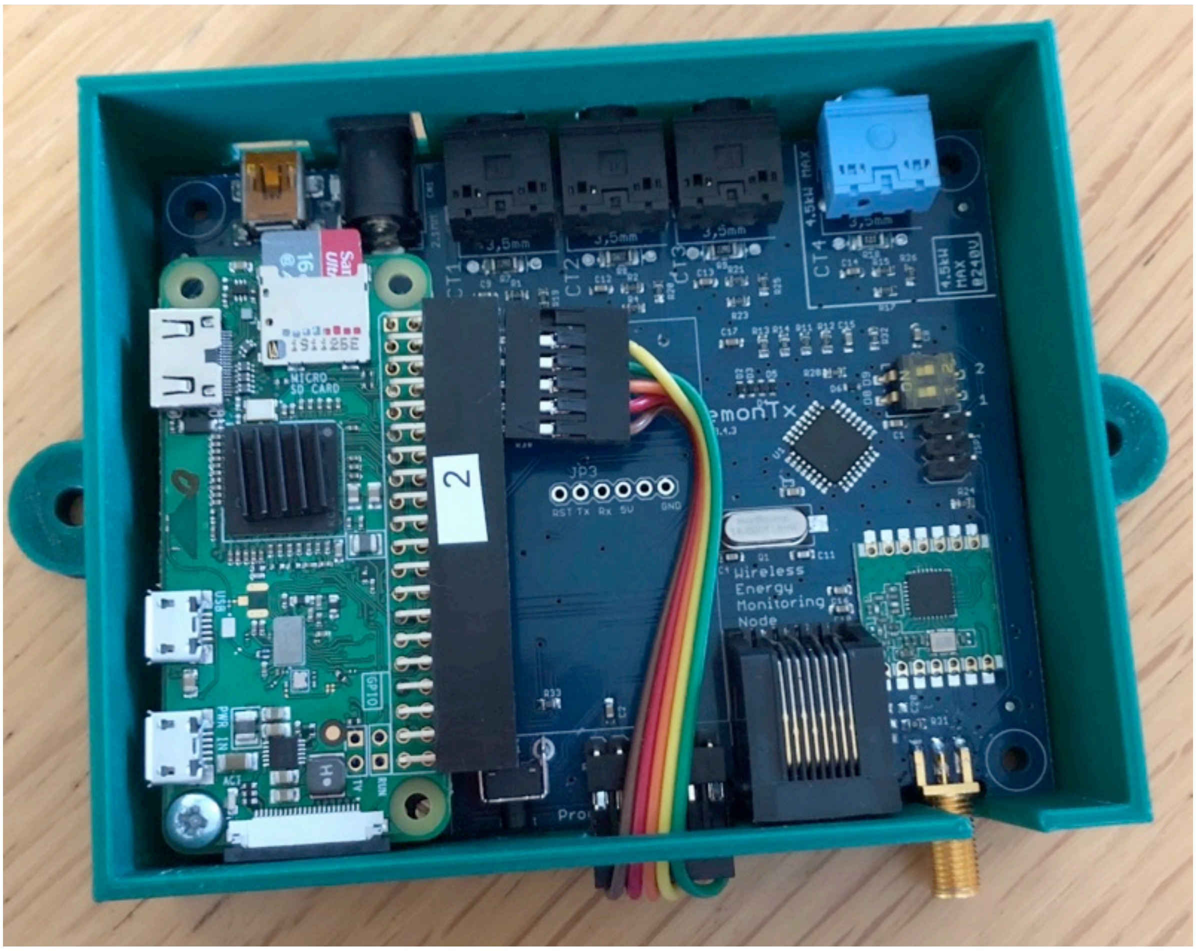
- A wifi enabled Raspberry Pi Zero W (either with a header (WH) or without (W)). Either https://shop.pimoroni.com/products/raspberry-pi-zero-wh-with-pre-soldered-header or https://shop.pimoroni.com/products/raspberry-pi-zero-w . No power supply is needed, but if you are going to use the original metal emonTx case then a case for the Rasperry Pi Zero is probably sensible. NB: The benefit of the Pi Zero without the pre-soldered header pins is that it gives you more flexibility with case sizes etc; but it does mean you have to be OK with doing a bit of soldering…
- An emonTx https://shop.openenergymonitor.com/emontx-energy-monitor-transmitter/
- A 5 core cable to connect the emonTx to the Raspberry Pi Zero W (see photo for an example). If you don't already have any, then a kit like this will give you many options. https://smile.amazon.co.uk/gp/product/B07VC383KW/ref=ppx_yo_dt_b_asin_title_o05_s01?ie=UTF8&psc=1



Optional extras

- *If no headers on the Pi Zero W, solder on a right angled header* [*https://thepihut.com/products/0-1-2x20-pin-strip-right-angle-female-header*](https://thepihut.com/products/0-1-2x20-pin-strip-right-angle-female-header)
- *A case – I 3D printed this one from Thingiverse; the only change I needed to do was to increase the emonTx programming pin slot height by 5mm to allow the 5 core cable to go back in to the case.* [*https://www.thingiverse.com/thing:3908393*](https://www.thingiverse.com/thing:3908393)
- *Some stand off pins to raise the Pi Zero off from the emonTx within the case. If you don't already have any then a kit like this will give you many options.* [*https://smile.amazon.co.uk/gp/product/B084NXPTZ6/ref=ppx_yo_dt_b_asin_title_o01_s00?ie=UTF8&psc=1*](https://smile.amazon.co.uk/gp/product/B084NXPTZ6/ref=ppx_yo_dt_b_asin_title_o01_s00?ie=UTF8&psc=1)





Steps

- Download the latest version of emonSD; at the time of writing, it was **emonSD-24Jul20.img** from https://github.com/openenergymonitor/emonpi/wiki/emonSD-pre-built-SD-card-Download-&-Change-Log
- Burn this image to a microSD card (I used a 16GB card) using Balena Etcher https://www.balena.io/etcher/
- When done, remove card from computer and re-insert it so that it will show in Finder as a new Drive (called " **boot** ").
- Create a text file called **wpa_supplicant.conf** – see https://www.raspberrypi.org/documentation/configuration/wireless/headless.md for details of what to put in it, and then copy it to the microSD card. I used an app called Brackets instead. http://brackets.io as I had been advised not to use the macOS app TextEdit as it can sometimes produce a strange file / unwanted characters.
- Create a blank text file called **ssh** but ensure it has no extension (such as **.txt**). Copy the file to the microSD card. [NB: ssh access to the Raspberry Pi is not essential, but it does help with any problem solving if you have any later on].
- Eject the microSD card from your computer and insert it in to the Raspberry Pi Zero W
- Boot up the Pi Zero; when it appears in the router's control panel after a short delay, ssh in to that IP address with Terminal. I had to type **sudo chmod go+r /etc/wpa_supplicant/wpa_supplicant.conf** to be able to then sign in to emonCMS on the Pi Zero via a browser.
- Register with a new emonCMS User profile, and then log in to the emonCMS.

- I needed to make a couple of changes to my emonHub config:
    - [interfacers] section: either delete or "hash out" the entire RFM2PI section to prevent that section from running, and therefore possibly conflicting with the wifi messages
    - [[SerialTx]] section: add this text below the hashed out / deleted RFM2PI section described above:

      ```
      [[SerialTx]]
          Type = EmonHubTx3eInterfacer
           [[[init_settings]]]
              com_port= /dev/ttyAMA0
              com_baud = 115200
          [[[runtimesettings]]]
              pubchannels = ToEmonCMS,

              nodeoffset = 0
              nodename = Serial_PiZ
      ```

    - Give the Pi Zero a distinct name in the **nodename = Serial_PiZ** (or whatever you want to call it & for it to appear as a Node in the emonPi's emonCMS Inputs page).
    - [[MQTT]] section: either delete or "hash out" the entire MQTT section to prevent that section from running, and replace it with

      ```
      [[MQTT]]

      Type = EmonHubMqttInterfacer
         [[[init_settings]]]
            mqtt_host = 192.168.2.50
            mqtt_port = 1883
            mqtt_user = emonpi
            mqtt_passwd = emonpimqtt2016

         [[[runtimesettings]]]
            subchannels = ToEmonCMS,

            timestamped = True

            node_JSON_enable = 1
            node_JSON_basetopic = emon/
      ```

    - You need to change the settings for **mqtt_host, mqtt_port, mqtt_user and mqtt_passwd** details to match those of your MQTT broker (in my case it was the IP address of my emonPi, and Port 1883).
- Save the changes in emonHub, then Restart emonHub
- Shut down the Pi Zero within the Admin section of emonCMS, and after a short wait remove the power supply
- Connect the Pi Zero W to the emonTx using the 2nd diagram shown in pb66's Feb 2018 post at https://community.openenergymonitor.org/t/4-ct-emonbase-using-emontx-and-pi-zero-w/6580/17 . The Pi Zero W will get its power from the emonTx so does NOT need a separate power supply.

- Connect the emonTx to power, and wait for the Pi Zero to boot up
- Log in to your emonPi / emonCMS and you should see the Inputs coming from the new Node (named above).

---++++++++++---

Update the emonTx to run the latest firmware (Optional)

I wanted to take the opportunity of having my emonTx on my desk to update the firmware from Tx V3.4 Discrete Sampling V3.10 to Tx V3.4 Continuous Monitoring V2.0

Requirements

- A USB to serial UART Programmer, available from the OpenEnergyMonitor shop - https://shop.openenergymonitor.com/programmer-usb-to-serial-uart/
- A copy of the Arduino IDE software (free download)

Steps

- To install the Arduino IDE follow the steps shown in https://learn.openenergymonitor.org/electricity-monitoring/arduino-ide/windows10ide
- Connect the Programmer to the emonTx (having removed the other power supply), and connect the programmer to your computer. Ensure you fit it the correct way up!!!
- Open the Arduino IDE and click on the magnifying glass (top left); at the foot of the window that opens select **115200** baud and "**Both CR & LF**"
- Ensure you are communicating via the right protocol (Tools / Port should be **/dev/cu.SLAB_USBtoUART)**
- When you have the three panes showing in the Arduino IDE main window (**EmonTxV3CM §**, **config**, and **rfm**) click on the 2nd icon top left (the arrow) & the new firmware should be uploaded.

Optional

- If you want to disable the radio on the emonTx, use the reset button on the emonTx to restart it & wait for the Arduino's white window pane to ask you to type **+++** & **Enter**. Do that, then type **r** to restore the default calibration.
- Then type **+++** and **Enter** at the prompt, then type **w0**, then **s** to save
- Then press the reset button again, and the Continuous Monitoring firmware should be running.
- Remove the Programmer from the emonTx, re-connect the Raspberry Pi & connect to power.
- Log in to your emonPi's emonCMS, and all should be running fine!

If you can bear to read the journey I took, there is more info on
https://community.openenergymonitor.org/t/avoiding-wireless-connections/15056