

Micropython + LoRaWAN = PyLoRaWAN

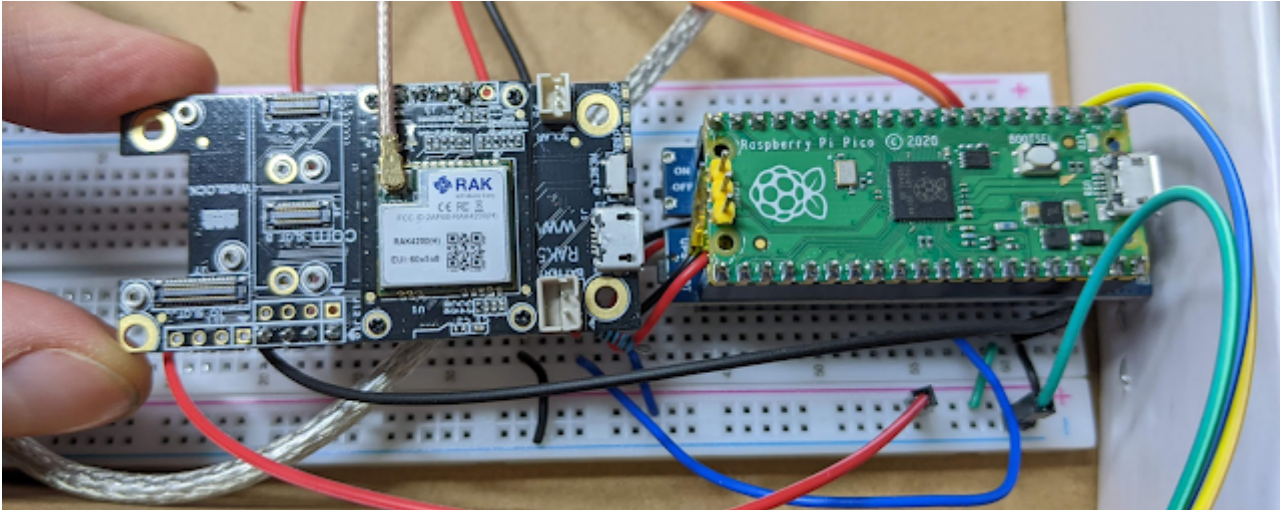
11 February 2022 · automation, devops, embedded, iot, micropython

I recently open sourced a simple Micropython library for LoRaWAN on the Raspberry Pi Pico. (If you are interested, You can find it [on GitHub \(https://github.com/phoughton/pylorawan\)](https://github.com/phoughton/pylorawan).) If you are unsure what that all means, let me unpack it for you...

Micropython is a slimmed down version of Python 3.x that works on microcontrollers like the Raspberry Pi Pico, and a [host of other microcontroller boards \(https://micropython.org/download/\)](https://micropython.org/download/).

LoRaWAN is a [wireless communication standard \(https://loro-alliance.org/about-lorawan/\)](https://loro-alliance.org/about-lorawan/) that is ideal for long range, low power & low band width data transmission. Its based on a clever technique for making signals work well over distance, called LoRa.

The library I've shared is a wrapper around the existing LoRaWAN support provided by the RAK Wireless 4200 board. The [RAK4200 \(https://rakwireless.kckb.st/821e96d1\)](https://rakwireless.kckb.st/821e96d1) (affiliate link) essentially provides a modem, that can establish a connection to the network and relay messages. It uses the traditional [AT command syntax \(https://en.wikipedia.org/wiki/Hayes_command_set\)](https://en.wikipedia.org/wiki/Hayes_command_set) (used by the modems of yore!)



The Pico and RAK4200 Evaluation board (there is also a UPS under the Pico there - that's optional)

The library [pylorawan \(https://github.com/phoughton/pylorawan\)](https://github.com/phoughton/pylorawan) provides a convenient way to use/control the device directly from your Micropython code. You can just provide the keys to the network, device etc and the code will handle the connection and retries etc. [This is the simplest \(https://github.com/phoughton/pylorawan/blob/main/simple_example.py\)](https://github.com/phoughton/pylorawan/blob/main/simple_example.py) example, where we just connect and send a couple of hex bytes.

