SOFAB

by Telecom Valley

Lopy4
Prise en main

Jacques Chambon

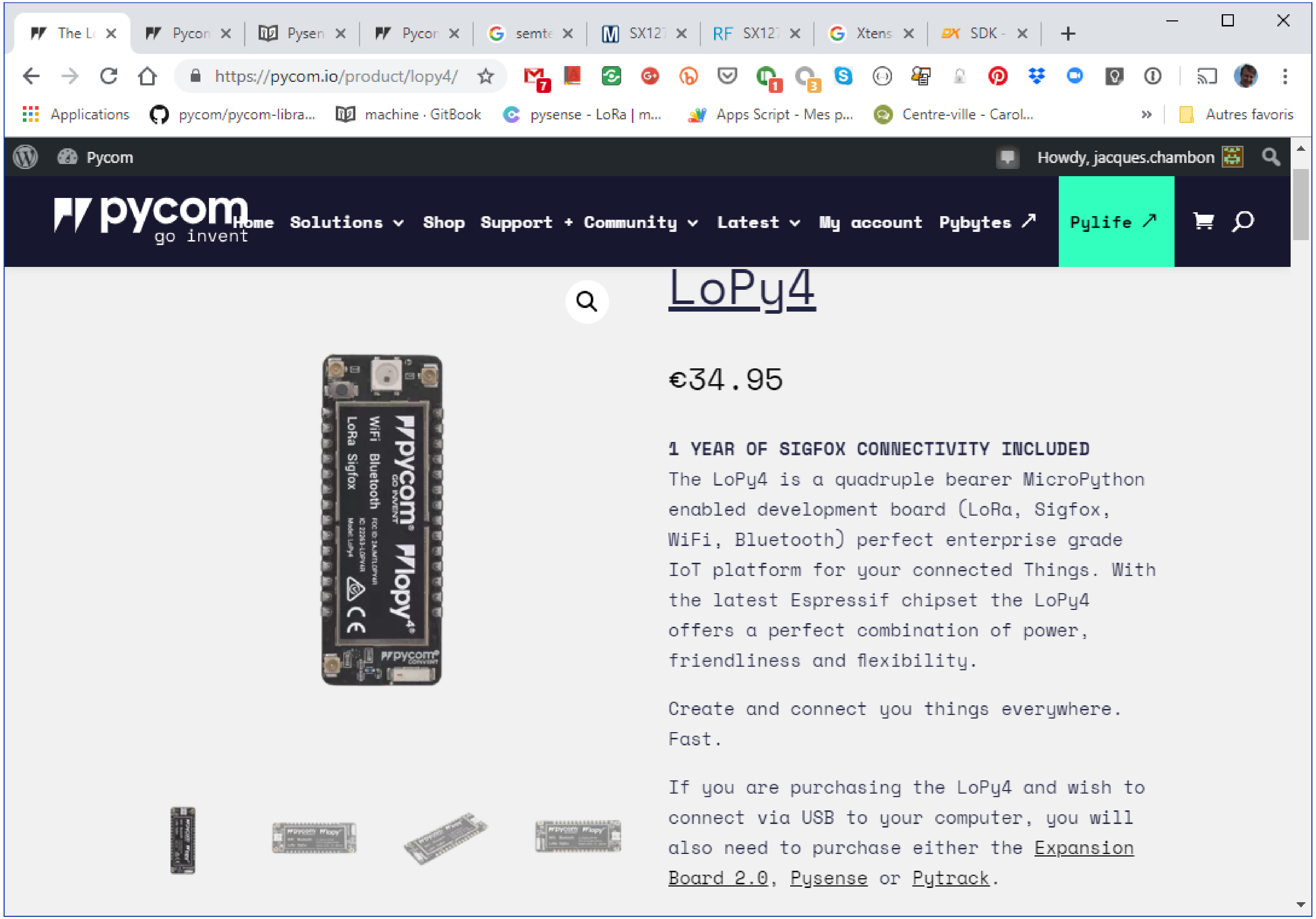
https://github.com/jacquesChambon/Lopy4

Atelier Lopy4

- Rappel sur les ateliers SoFab « IOT »
- Présentation Lopy4
  - Description, architecture, caractéristiques
  - Préparation Hardware/Software
  - TP

Prochains ateliers SoFab « IOT » (à confirmer)

- Jeudi 11/04 : Lopy4 (Jacques Chambon)
- mardi 23/04 : Lora/Sigfox sur Lopy4 (Rémi Jolin)
- mercredi 24/04 : The Things Network (TTN) (Fabien Ferrero)
- Jeudi 25/04 : Présentation Busit (Guillaume Meriel)
- vendredi 26/04 : Propriété Intellectuelle (Benjamin Delsol)



LoPy4

€34.95

1 YEAR OF SIGFOX CONNECTIVITY INCLUDED

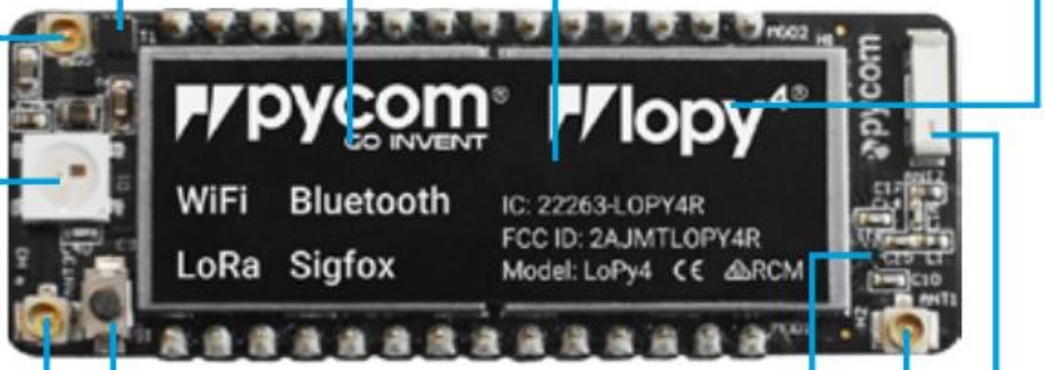
The LoPy4 is a quadruple bearer MicroPython enabled development board (LoRa, Sigfox, WiFi, Bluetooth) perfect enterprise grade IoT platform for your connected Things. With the latest Espressif chipset the LoPy4 offers a perfect combination of power, friendliness and flexibility.

Create and connect you things everywhere. Fast.

If you are purchasing the LoPy4 and wish to connect via USB to your computer, you will also need to purchase either the Expansion Board 2.0, Pysense or Pytrack.

Size: 55mm x 20mm x 3.5mm

Operating temperature:
-40 to 85 degrees celsius



3V3 Ultra-Low -Noise switching regulator

LoRa / Sigfox transceiver

ESP32 Dual Core Microcontroller and WiFi/Bluetooth 4.2 radio

8MB flash memory

LoRa / Sigfox 868-915 MHz antenna

WS2812 RGB multi-colour LED

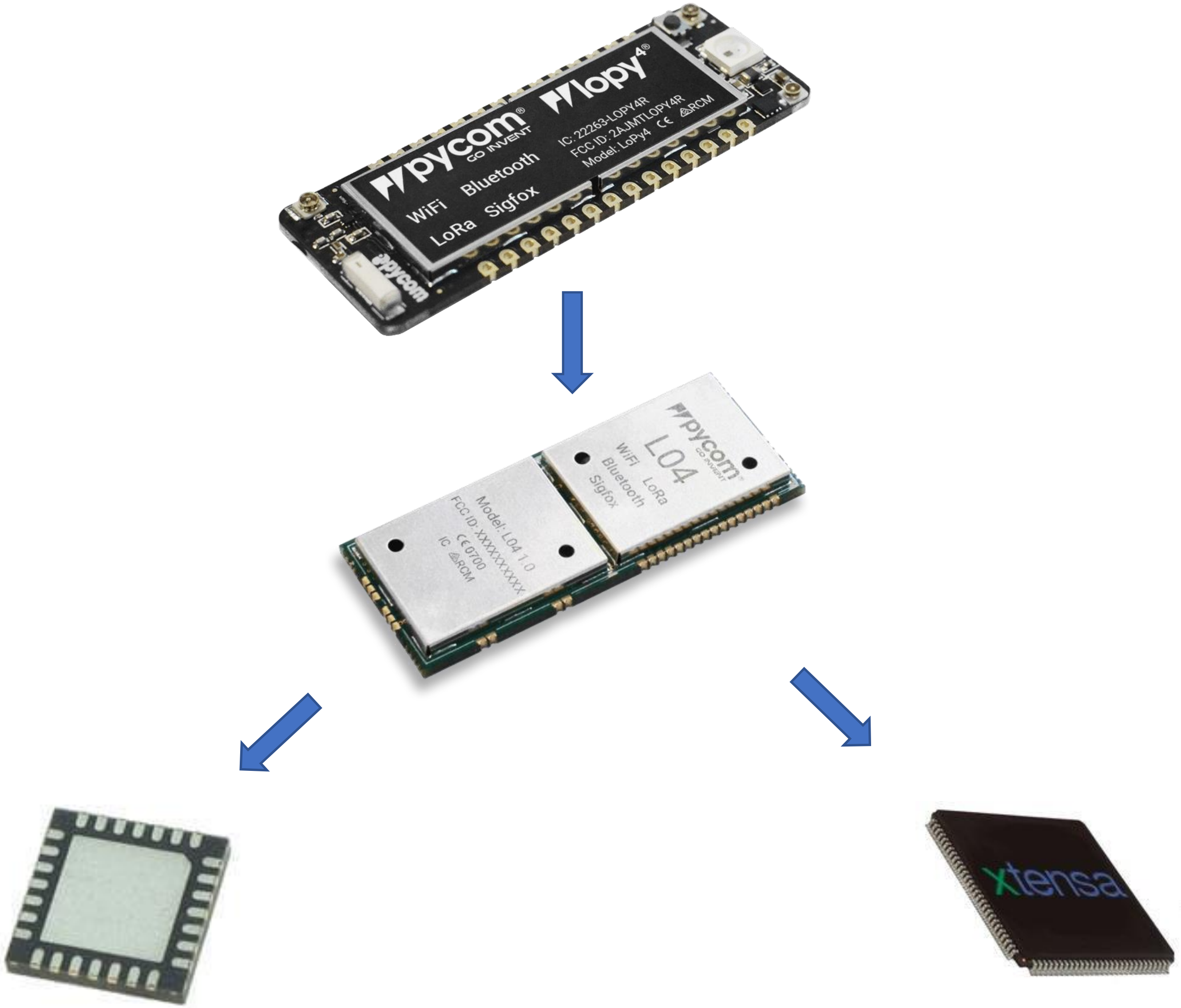
RF switch

U.FL connector

LoRa 433-510 MHz antenna

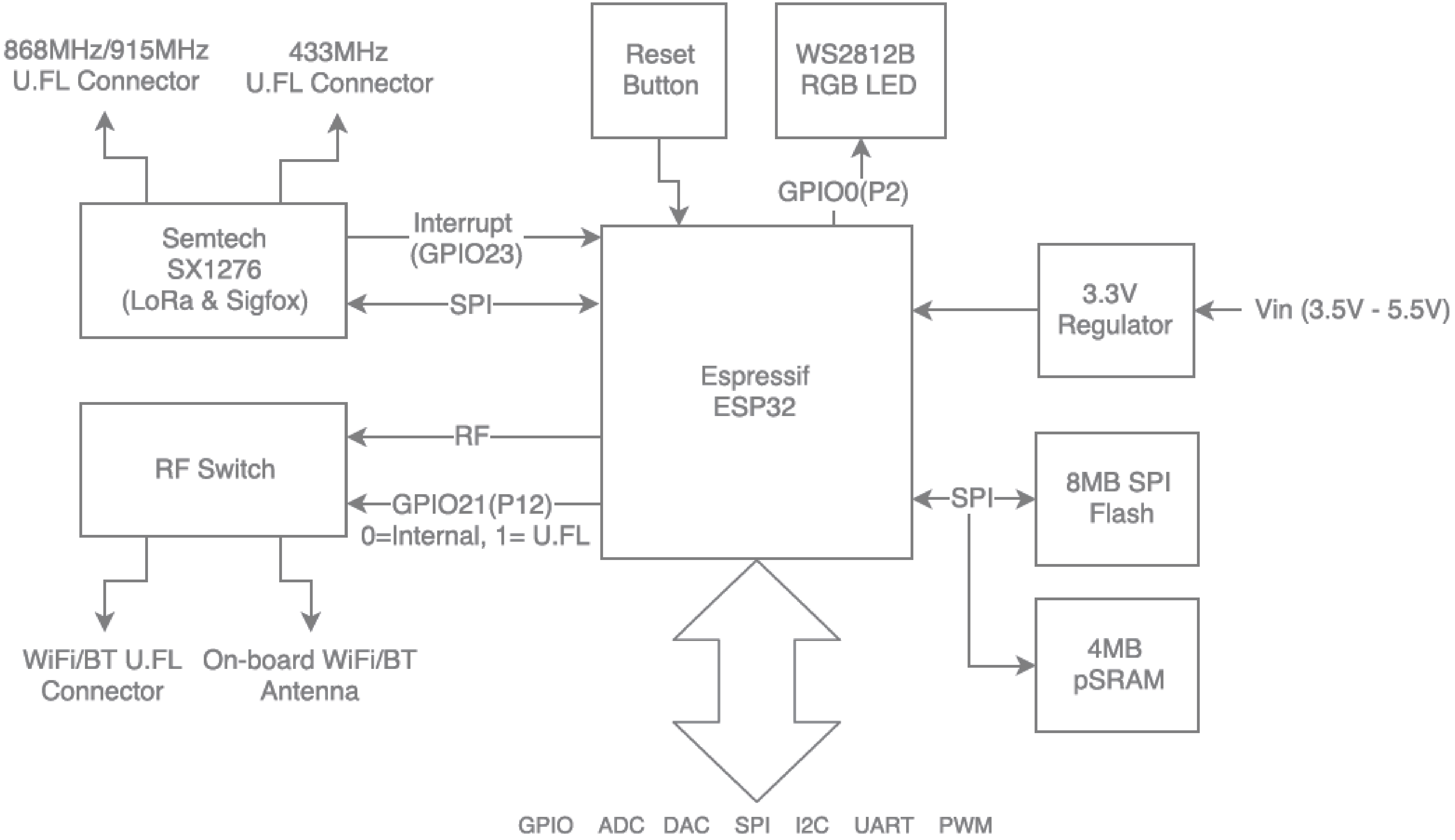
Reset switch

Internal WiFi and Bluetooth Antenna



Semtech SX 1276

Espressif ESP32



868MHz/915MHz U.FL Connector

433MHz U.FL Connector

Reset Button

WS2812B RGB LED

GPIO0(P2)

Semtech SX1276 (LoRa & Sigfox)

Interrupt (GPIO23)

SPI

Espressif ESP32

3.3V Regulator

Vin (3.5V - 5.5V)

RF Switch

RF

GPIO21(P12)

0=Internal, 1= U.FL

SPI

8MB SPI Flash

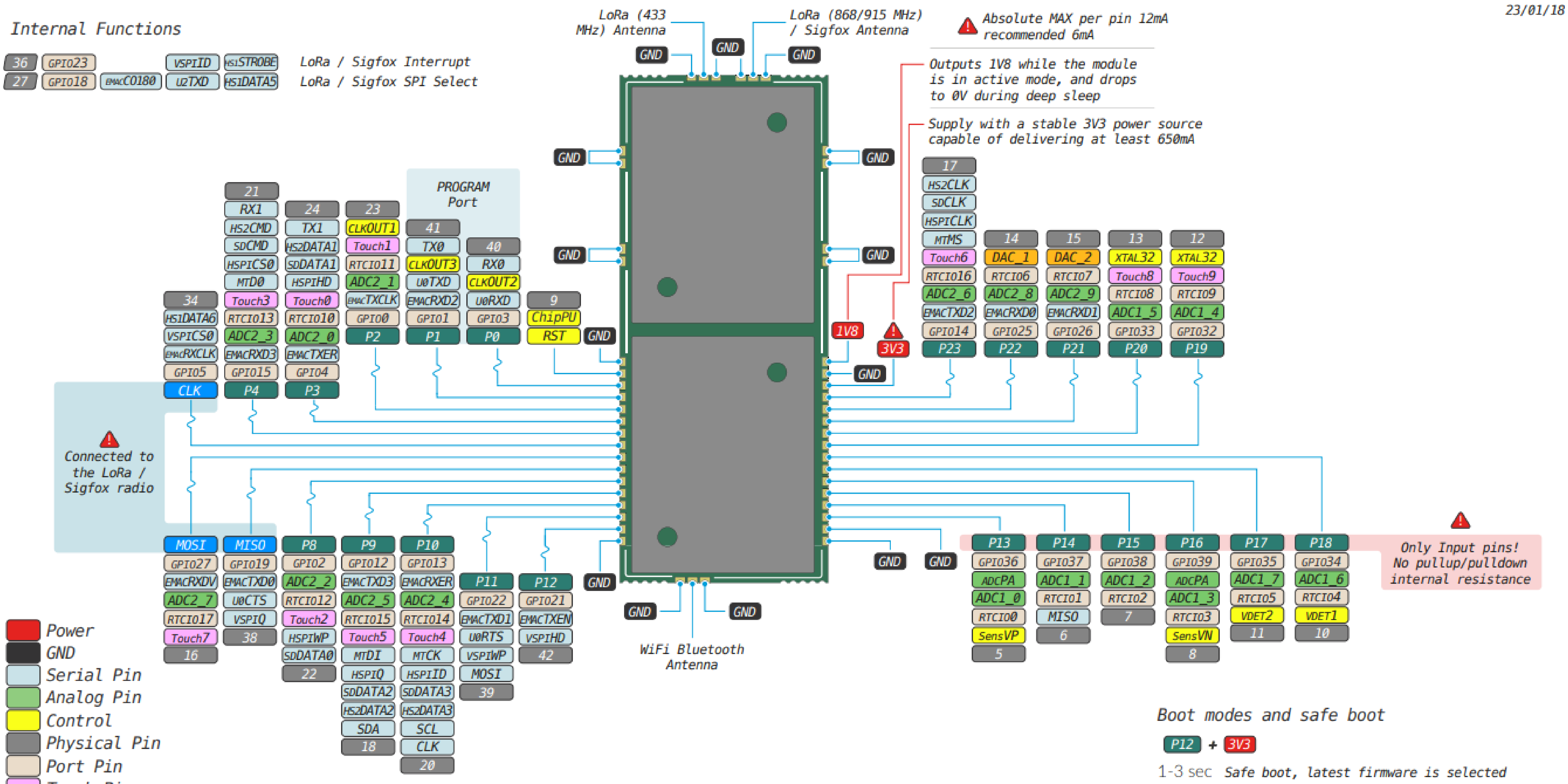
4MB pSRAM

WiFi/BT U.FL Connector

On-board WiFi/BT Antenna

GPIO ADC DAC SPI I2C UART PWM

23/01/18


Internal Functions
36 GPIO23 VSPIID HS1STROBE
27 GPIO18 EMACCO180 U2TXD HS1DATA5
LoRa (433 MHz) Antenna
LoRa (868/915 MHz) / Sigfox Antenna
GND
GND
GND
GND
GND
GND
GND
PROGRAM Port
34 HS1DATA6 VSPICS0 EMACRXCLK GPIO5 CLK
21 RX1 HS2CMD SDCMD HSPICS0 MTDO Touch3 RTCIO13 ADC2_3 EMACRXD3 GPIO15 P4
24 TX1 HS2DATA1 SDDATA1 HSPIHD Touch0 RTCIO10 ADC2_0 EMACTXER GPIO4 P3
23 CLKOUT1 Touch1 RTCIO11 ADC2_1 EMACTXCLK GPIO0 P2
41 TX0 CLKOUT3 U0TXD EMACRXD2 GPIO1 P1
40 RX0 CLKOUT2 U0RXD GPIO3 P0
9 ChipPU RST
GND
1V8
3V3
GND
17 HS2CLK SDCLK HSPICLK MTMS Touch6 RTCIO16 ADC2_6 EMACTXD2 GPIO14 P23
14 DAC_1 RTCIO6 ADC2_8 EMACRXD0 GPIO25 P22
15 DAC_2 RTCIO7 ADC2_9 EMACRXD1 GPIO26 P21
13 XTAL32 Touch8 RTCIO8 ADC1_5 GPIO33 P20
12 XTAL32 Touch9 RTCIO9 ADC1_4 GPIO32 P19
MOSI GPIO27 EMACRXDV ADC2_7 RTCIO17 Touch7 16
MISO GPIO19 EMACTXD0 U0CTS VSPIQ 38
P8 GPIO2 ADC2_2 RTCIO12 Touch2 HSPIWP SDDATA0 22
P9 GPIO12 EMACTXD3 ADC2_5 RTCIO15 Touch5 MTDI HSPIQ SDDATA2 HS2DATA2 SDA 18
P10 GPIO13 EMACRXER ADC2_4 RTCIO14 Touch4 MTCK HSPIID SDDATA3 HS2DATA3 SCL CLK 20
P11 GPIO22 EMACTXD1 U0RTS VSPIWP MOSI 39
P12 GPIO21 EMACTXEN VSPIHD 42
GND
GND
GND
WiFi Bluetooth Antenna
GND
GND
P13 GPIO36 ADCPA ADC1_0 RTCIO0 SensVP 5
P14 GPIO37 ADC1_1 RTCIO1 MISO 6
P15 GPIO38 ADC1_2 RTCIO2 7
P16 GPIO39 ADCPA ADC1_3 RTCIO3 SensVN 8
P17 GPIO35 ADC1_7 RTCIO5 VDET2 11
P18 GPIO34 ADC1_6 RTCIO4 VDET1 10

LoRa / Sigfox Interrupt

LoRa / Sigfox SPI Select

⚠ Absolute MAX per pin 12mA recommended 6mA

Outputs 1V8 while the module is in active mode, and drops to 0V during deep sleep

Supply with a stable 3V3 power source capable of delivering at least 650mA

⚠ Connected to the LoRa / Sigfox radio

⚠ Only Input pins! No pullup/pulldown internal resistance

🟥 Power
⬛ GND
🟦 Serial Pin
🟩 Analog Pin
🟨 Control
▨ Physical Pin
⬜ Port Pin
🟪 Touch Pin
🟧 DAC Pin
∿ PMW Pin

Low Level Bootloader

P2 + GND

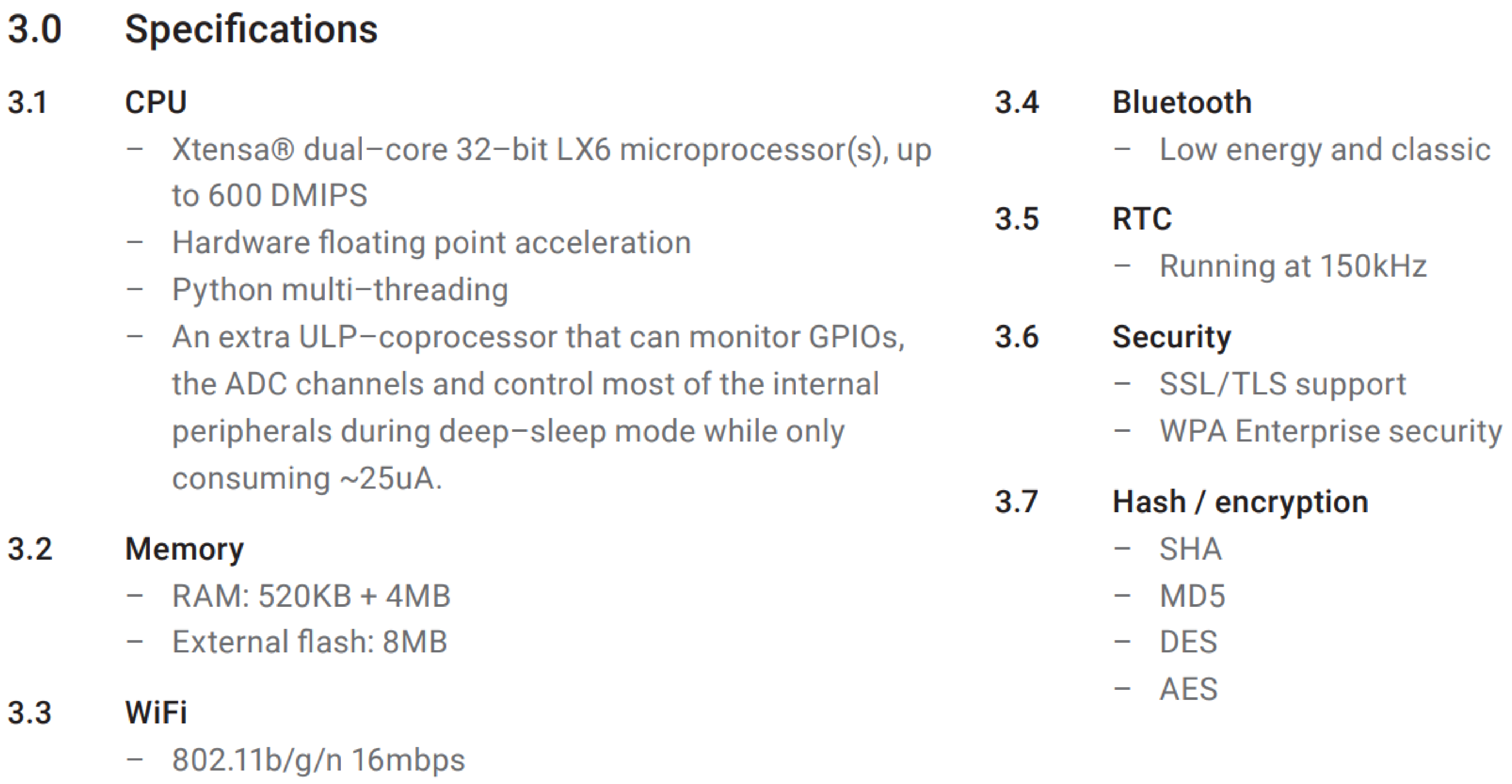
Boot modes and safe boot

P12 + 3V3

1-3 sec Safe boot, latest firmware is selected

4-6 sec Safe boot, previous user update selected

7-9 sec Safe boot, the factory firmware is selected



3.0 Specifications

3.1 CPU

- Xtensa® dual–core 32–bit LX6 microprocessor(s), up to 600 DMIPS
- Hardware floating point acceleration
- Python multi–threading
- An extra ULP–coprocessor that can monitor GPIOs, the ADC channels and control most of the internal peripherals during deep–sleep mode while only consuming ~25uA.

3.2 Memory

- RAM: 520KB + 4MB
- External flash: 8MB

3.3 WiFi

- 802.11b/g/n 16mbps

3.4 Bluetooth

- Low energy and classic

3.5 RTC

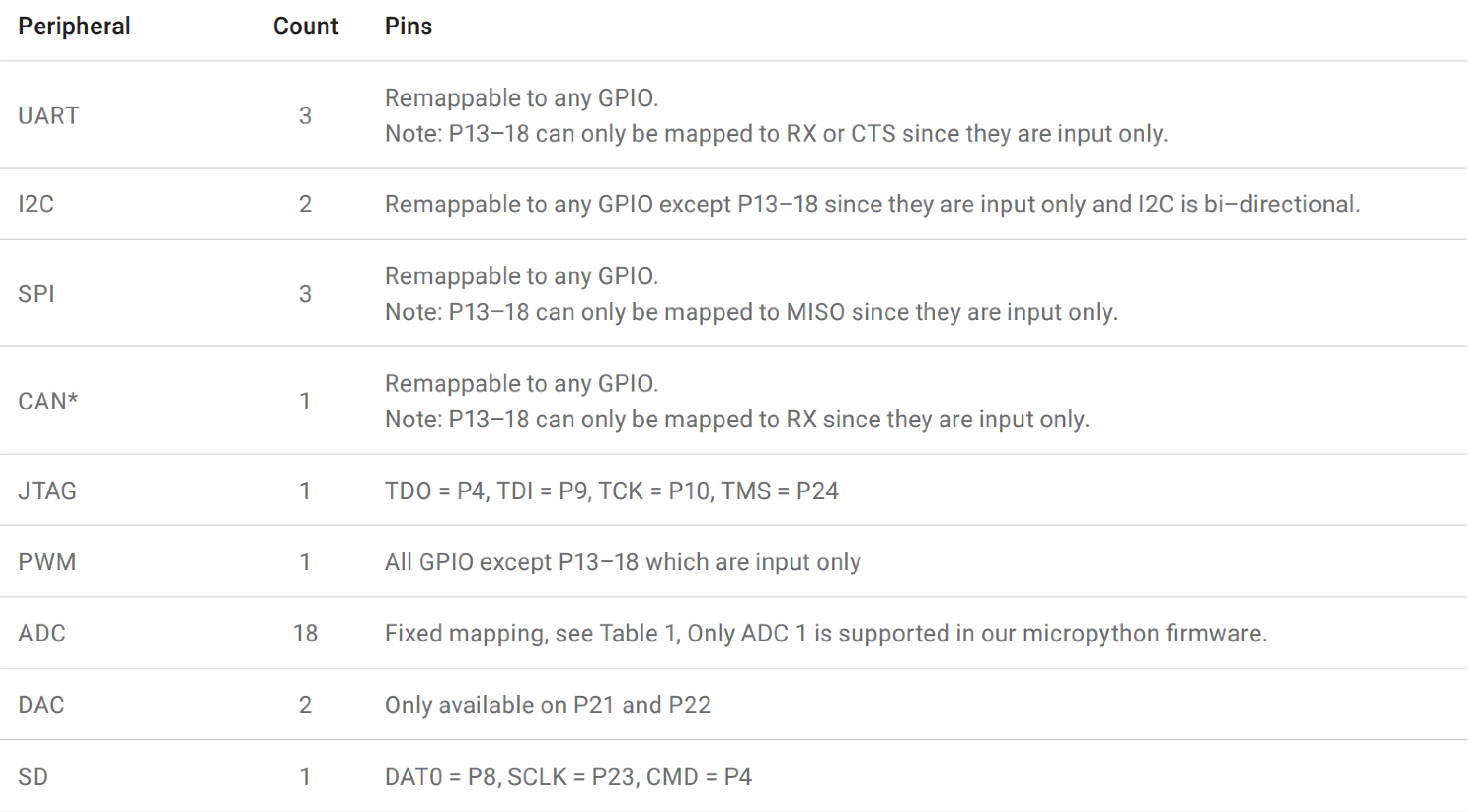
- Running at 150kHz

3.6 Security

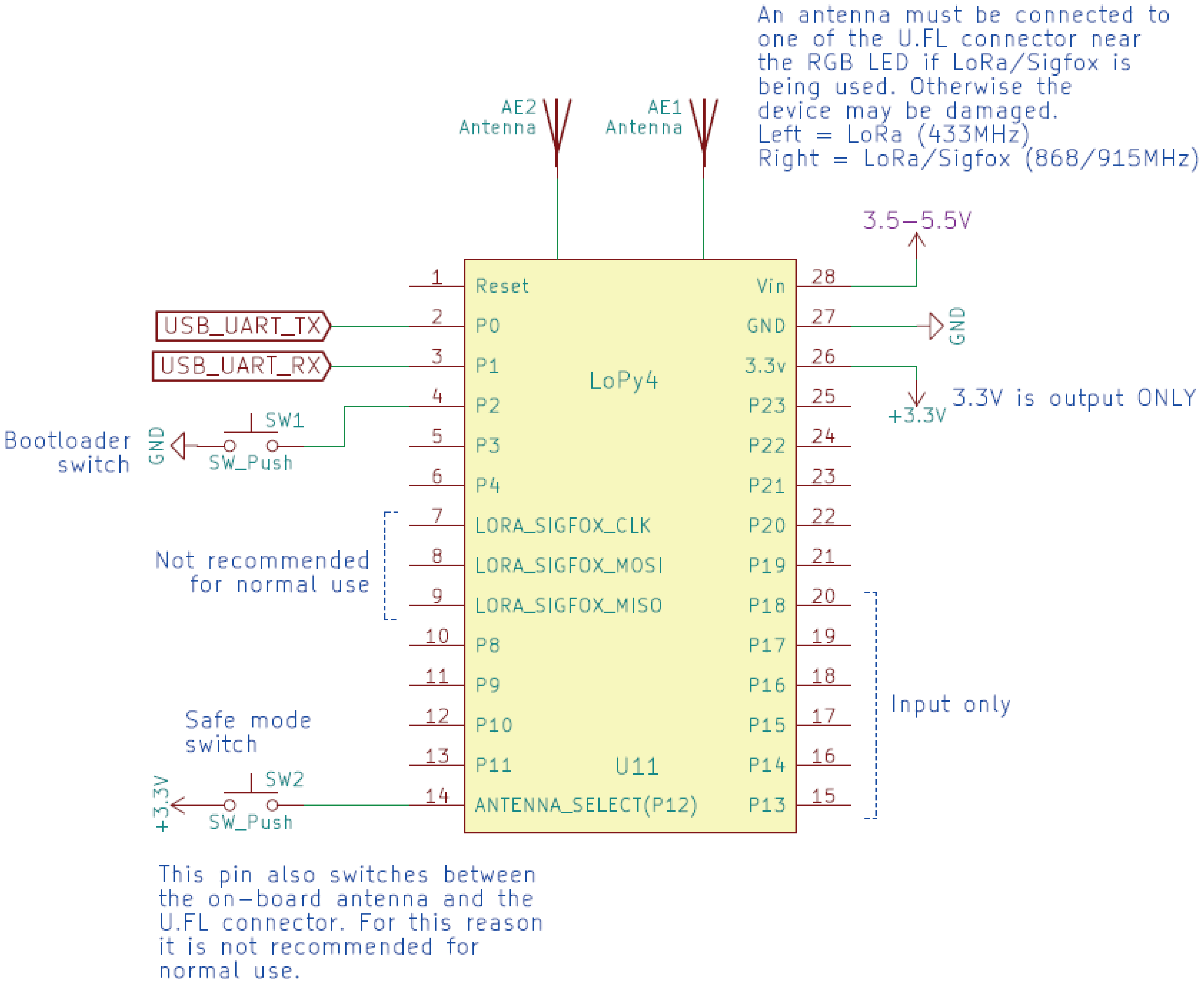
- SSL/TLS support
- WPA Enterprise security

3.7 Hash / encryption

- SHA
- MD5
- DES
- AES



Peripheral	Count	Pins
UART	3	Remappable to any GPIO. Note: P13–18 can only be mapped to RX or CTS since they are input only.
I2C	2	Remappable to any GPIO except P13–18 since they are input only and I2C is bi–directional.
SPI	3	Remappable to any GPIO. Note: P13–18 can only be mapped to MISO since they are input only.
CAN*	1	Remappable to any GPIO. Note: P13–18 can only be mapped to RX since they are input only.
JTAG	1	TDO = P4, TDI = P9, TCK = P10, TMS = P24
PWM	1	All GPIO except P13–18 which are input only
ADC	18	Fixed mapping, see Table 1, Only ADC 1 is supported in our micropython firmware.
DAC	2	Only available on P21 and P22
SD	1	DAT0 = P8, SCLK = P23, CMD = P4



An antenna must be connected to one of the U.FL connector near the RGB LED if LoRa/Sigfox is being used. Otherwise the device may be damaged.
Left = LoRa (433MHz)
Right = LoRa/Sigfox (868/915MHz)

3.3V is output ONLY

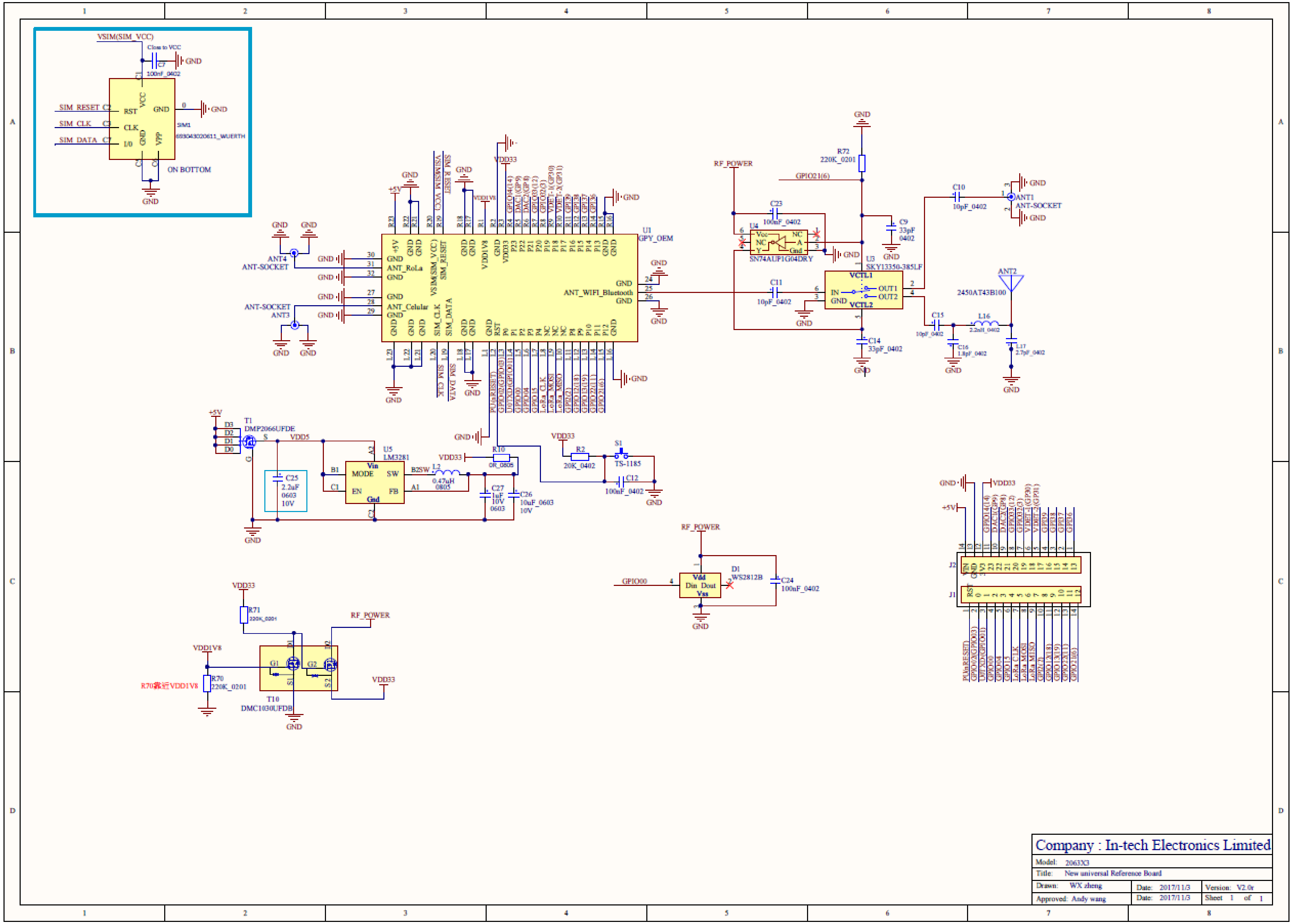
Not recommended for normal use

Input only

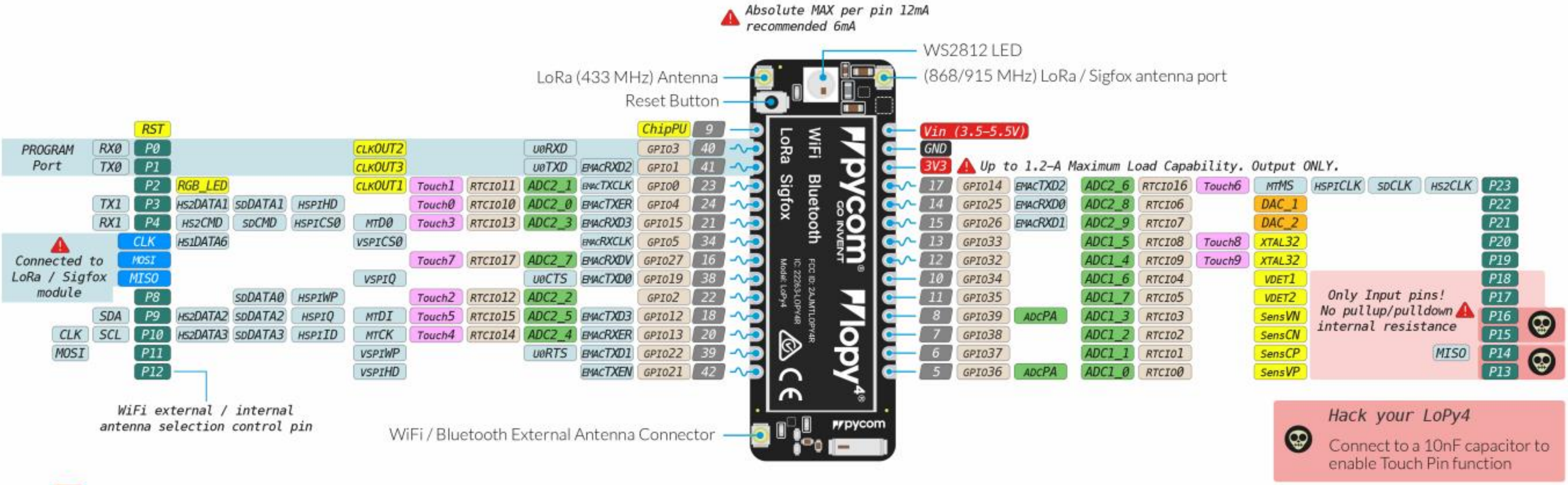
Bootloader switch

Safe mode switch

This pin also switches between the on-board antenna and the U.FL connector. For this reason it is not recommended for normal use.



SOFAB by Telecom Valley



Absolute MAX per pin 12mA recommended 6mA

WS2812 LED

LoRa (433 MHz) Antenna

(868/915 MHz) LoRa / Sigfox antenna port

Reset Button

		RST									ChipPU	9
PROGRAM Port	RX0	P0				CLKOUT2			u0RXD		GPIO3	40
	TX0	P1				CLKOUT3			u0TXD	EMACRXD2	GPIO1	41
		P2	RGB_LED			CLKOUT1	Touch1	RTCIO11	ADC2_1	EMACTXCLK	GPIO0	23
	TX1	P3	HS2DATA1	SDDATA1	HSPIHD		Touch0	RTCIO10	ADC2_0	EMACTXER	GPIO4	24
	RX1	P4	HS2CMD	SDCMD	HSPICS0	MTD0	Touch3	RTCIO13	ADC2_3	EMACRXD3	GPIO15	21
Connected to LoRa / Sigfox module		CLK	HS1DATA6			VSPICS0				EMACRXCLK	GPIO5	34
		MOSI					Touch7	RTCIO17	ADC2_7	EMACRXDV	GPIO27	16
		MISO				VSPIQ			u0CTS	EMACTXD0	GPIO19	38
		P8		SDDATA0	HSPIWP		Touch2	RTCIO12	ADC2_2		GPIO2	22
	SDA	P9	HS2DATA2	SDDATA2	HSPIQ	MTDI	Touch5	RTCIO15	ADC2_5	EMACTXD3	GPIO12	18
CLK	SCL	P10	HS2DATA3	SDDATA3	HSPIID	MTCK	Touch4	RTCIO14	ADC2_4	EMACRXER	GPIO13	20
MOSI		P11				VSPIWP			u0RTS	EMACTXD1	GPIO22	39
		P12				VSPIHD				EMACTXEN	GPIO21	42

WiFi external / internal antenna selection control pin

WiFi / Bluetooth External Antenna Connector

Vin (3.5-5.5V)

GND

3V3 ⚠ Up to 1.2-A Maximum Load Capability. Output ONLY.

17	GPIO14	EMACTXD2	ADC2_6	RTCIO16	Touch6	MTMS	HSPICLK	SDCLK	HS2CLK	P23
14	GPIO25	EMACRXD0	ADC2_8	RTCIO6		DAC_1				P22
15	GPIO26	EMACRXD1	ADC2_9	RTCIO7		DAC_2				P21
13	GPIO33		ADC1_5	RTCIO8	Touch8	XTAL32				P20
12	GPIO32		ADC1_4	RTCIO9	Touch9	XTAL32				P19
10	GPIO34		ADC1_6	RTCIO4		VDET1				P18
11	GPIO35		ADC1_7	RTCIO5		VDET2				P17
8	GPIO39	ADCPA	ADC1_3	RTCIO3		SensVN				P16
7	GPIO38		ADC1_2	RTCIO2		SensCN				P15
6	GPIO37		ADC1_1	RTCIO1		SensCP			MISO	P14
5	GPIO36	ADCPA	ADC1_0	RTCIO0		SensVP				P13

Only Input pins! No pullup/pulldown ⚠ internal resistance

Hack your LoPy4

Connect to a 10nF capacitor to enable Touch Pin function

- 🟥 Power
- ⬛ GND
- ▨ Serial Pin
- 🟩 Analog Pin
- 🟨 Control
- ▨ Physical Pin
- ▨ Port Pin
- ▨ Touch Pin
- 🟧 DAC Pin
- ∿● PWM Pin

Low Level Bootloader

P2 + GND

Boot modes and safe boot

P12 + 3V3

1-3 sec Safe boot, latest firmware is selected

4-6 sec Safe boot, previous user update selected

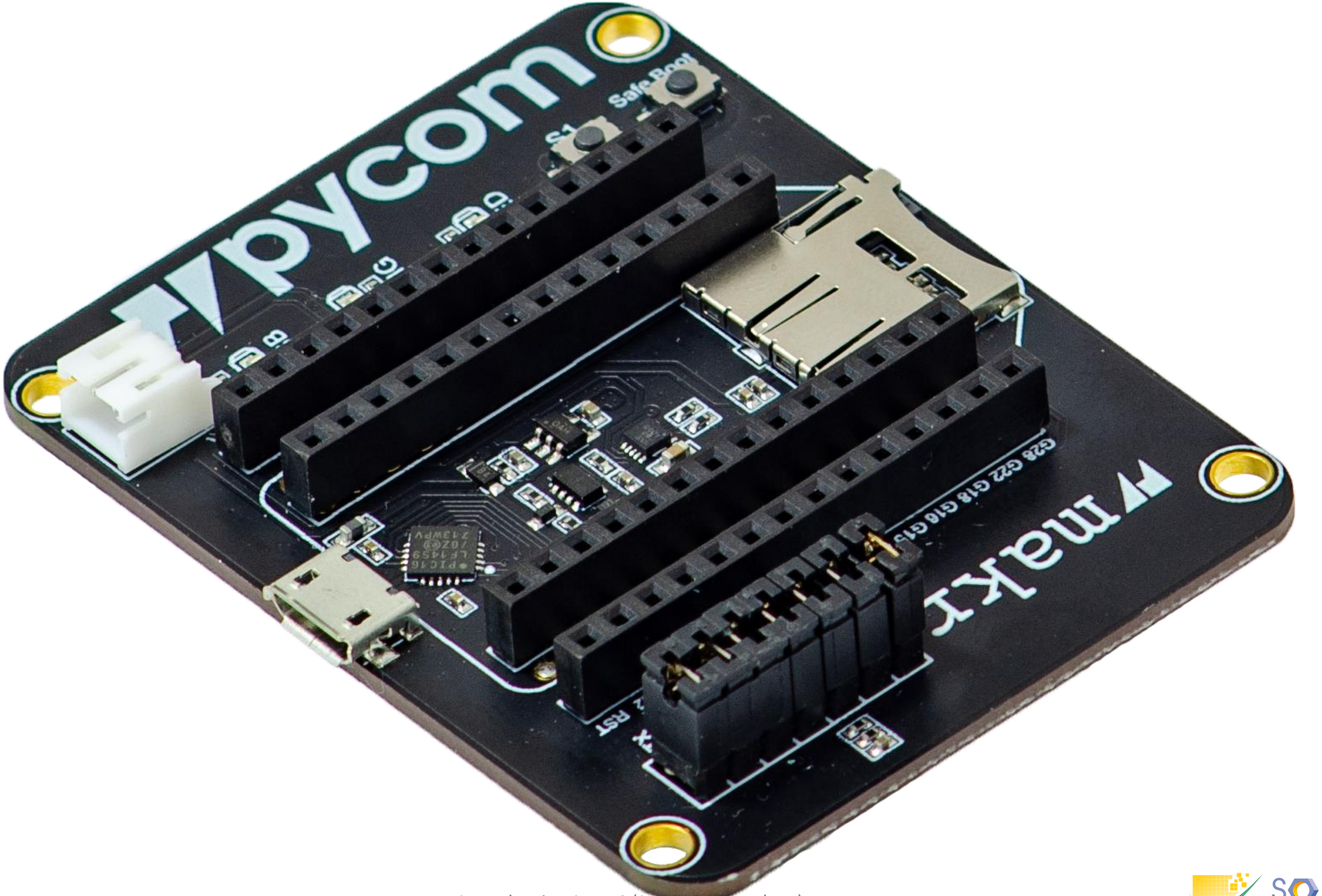
7-9 sec Safe boot, the factory firmware is selected

Internal Functions

36	GPIO23		VSPIID	HS1STROBE	LoRa / Sigfox Interrupt
27	GPIO18	EMACCO180	U2TXD	HS1DATA5	LoRa / Sigfox Select

04/07/18

SOFAB by Telecom Valley

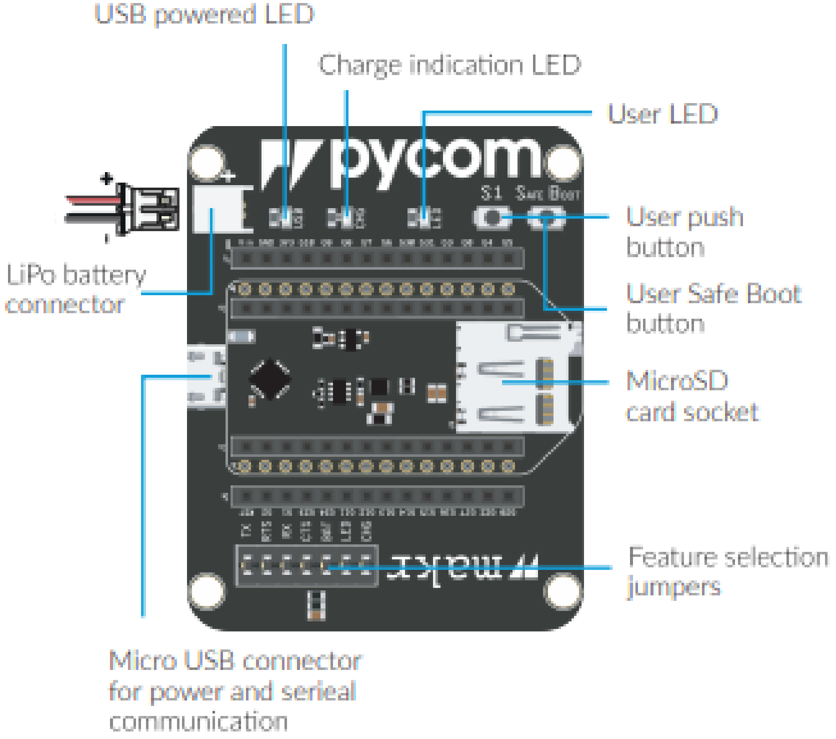

pycom
S1
Safe Boot
G28 G22 G18 G16 G15
RST
PIC16
LF1459
maker

Expansion Board 3v.0 Features

- USB and LiPo battery powered
- Custom PIC USB to serial converter with automatic bootloader mode
- LiPo battery charger (BQ24040), with options for two different charging currents (100mA and 450mA)
- TPS2115A with reverse voltage protection
- MicroSD card slot
- JST style battery connector
- Power LED and charge status LED
- One user LED and one user switch
- Button to enter into "safe mode" easily
- Battery voltage monitoring via the WiPy ADC
- Lots of jumpers to enable/disable features

Mechanical

Size: 65 x 50 x 8mm



USB powered LED

Charge indication LED

User LED

User push button

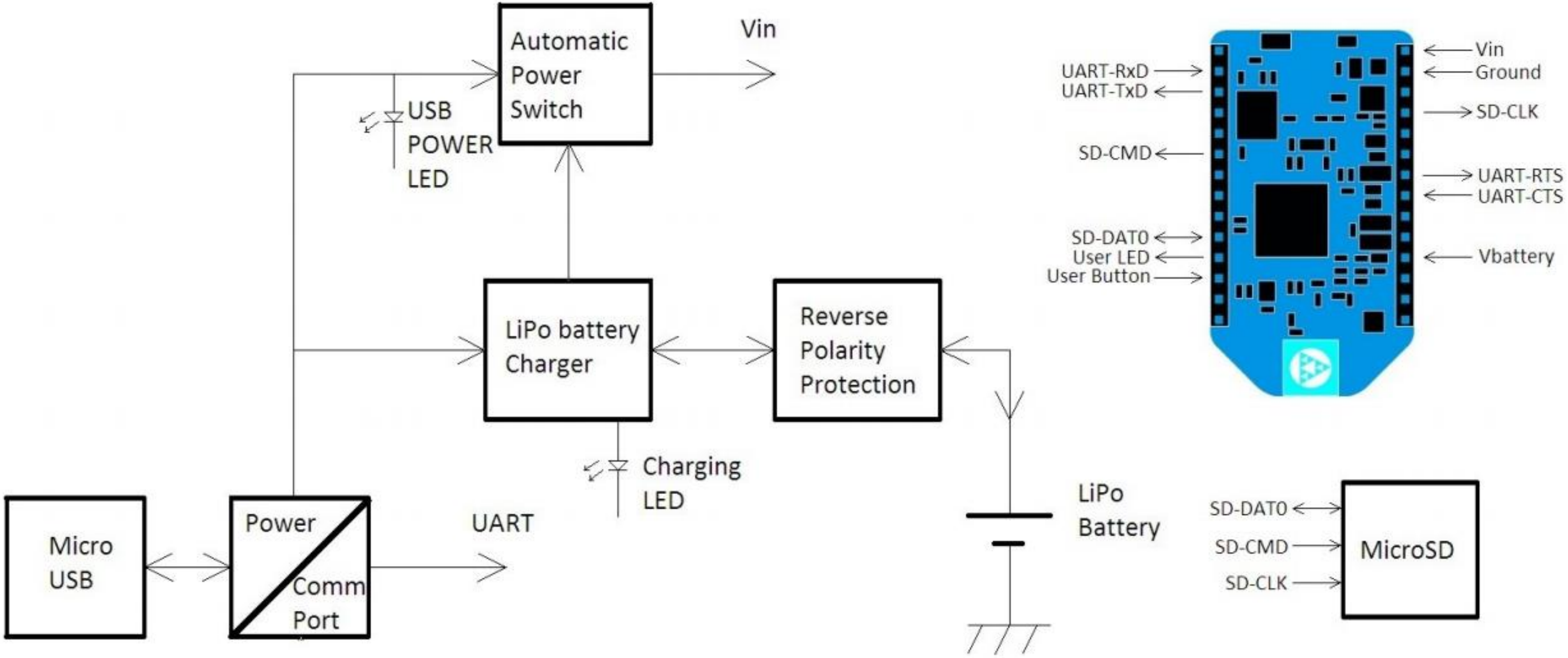
User Safe Boot button

MicroSD card socket

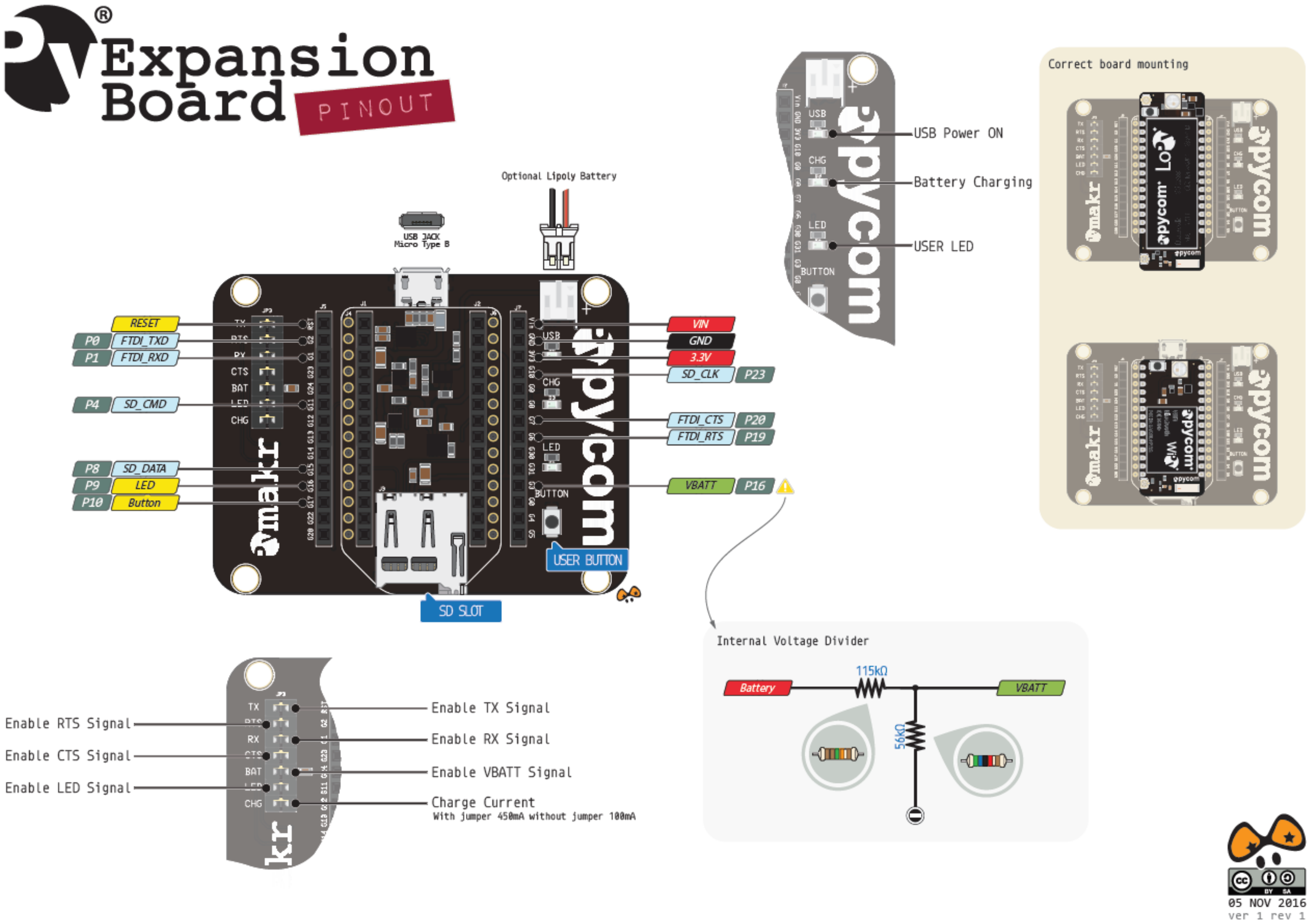
LiPo battery connector

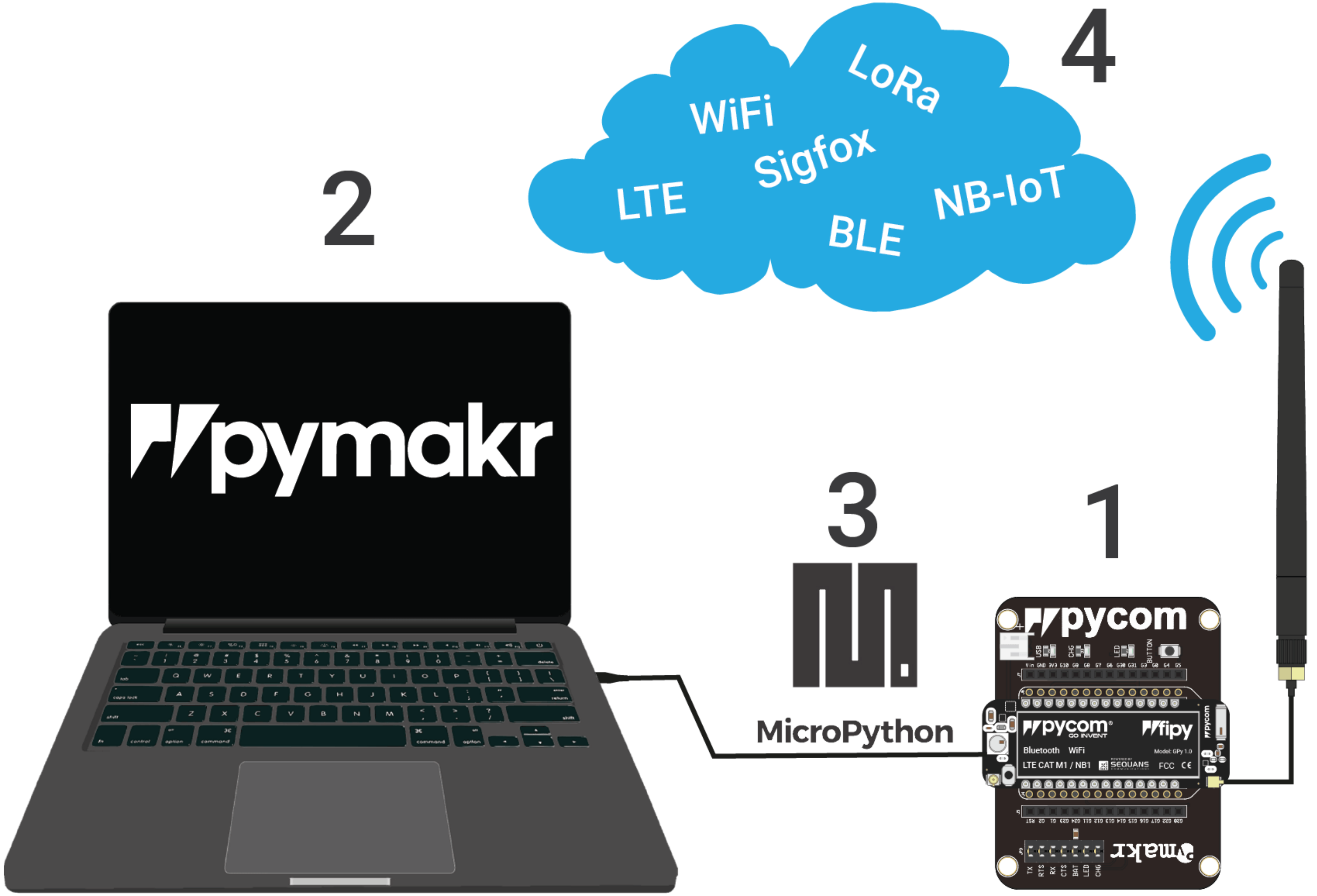
Feature selection jumpers

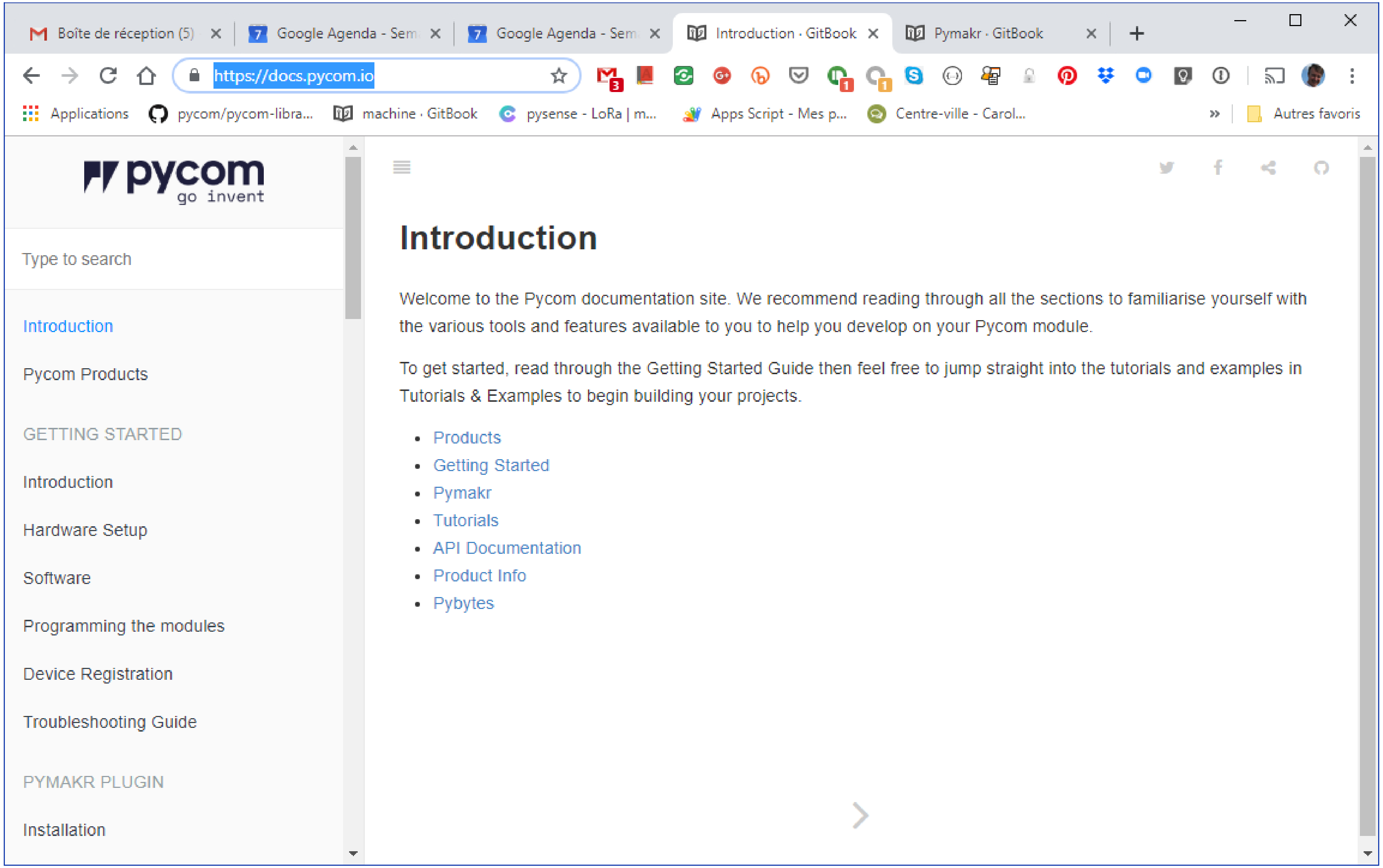
Micro USB connector for power and serieal communication


Automatic Power Switch
Vin
USB POWER LED
LiPo battery Charger
Reverse Polarity Protection
Charging LED
LiPo Battery
Micro USB
Power
Comm Port
UART
UART-RxD
UART-TxD
SD-CMD
SD-DAT0
User LED
User Button
Vin
Ground
SD-CLK
UART-RTS
UART-CTS
Vbattery
SD-DAT0
SD-CMD
SD-CLK
MicroSD

https://docs.pycom.io/.gitbook/assets/expansion2-pinout.pdf




4
LoRa
WiFi
Sigfox
LTE
NB-IoT
BLE
2
pymakr
3
MicroPython
1
pycom
fipy
makr



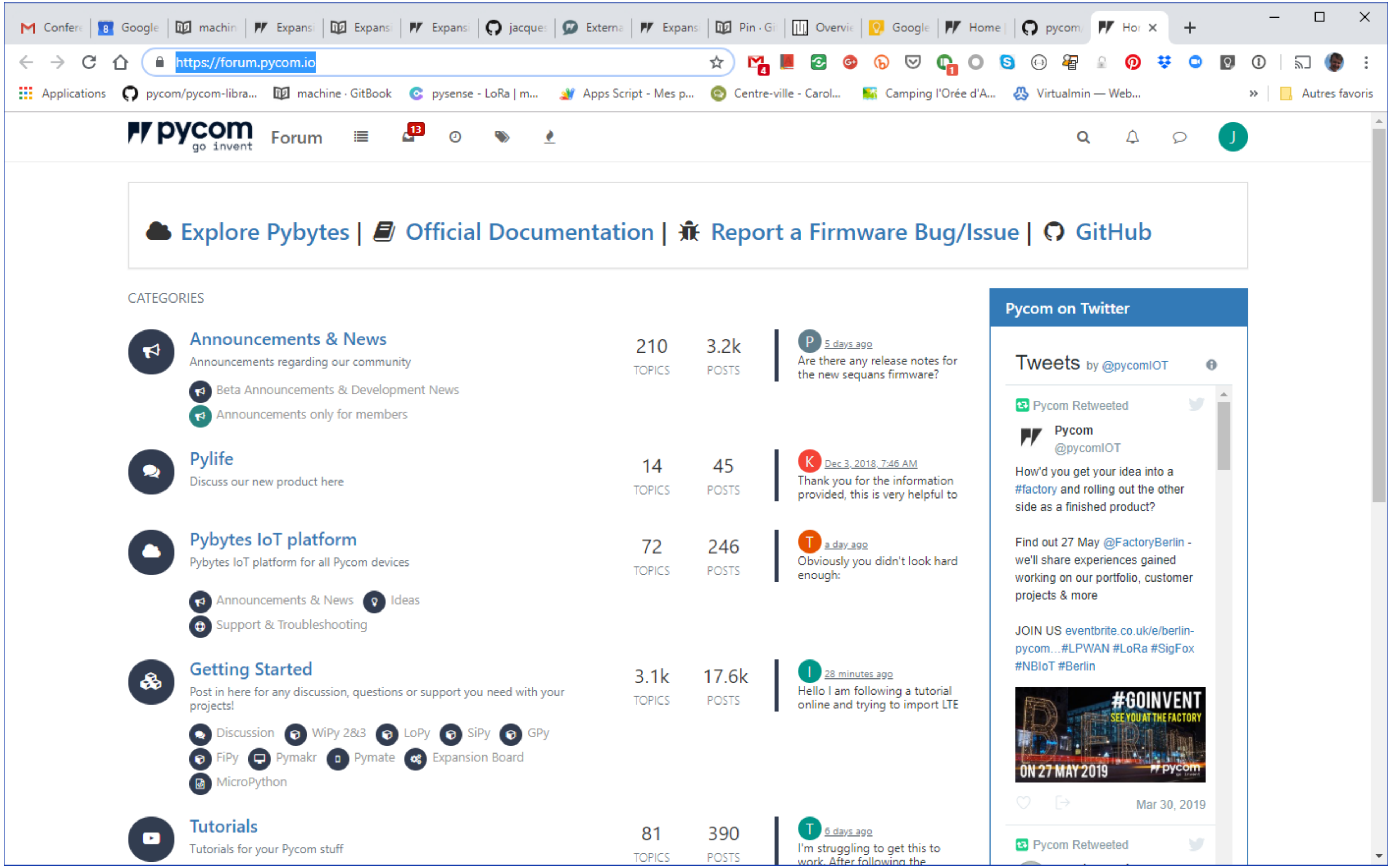
Introduction

Welcome to the Pycom documentation site. We recommend reading through all the sections to familiarise yourself with the various tools and features available to you to help you develop on your Pycom module.

To get started, read through the Getting Started Guide then feel free to jump straight into the tutorials and examples in Tutorials & Examples to begin building your projects.

- Products
- Getting Started
- Pymakr
- Tutorials
- API Documentation
- Product Info
- Pybytes

https://docs.pycom.io/



Explore Pybytes | Official Documentation | Report a Firmware Bug/Issue | GitHub

CATEGORIES

Announcements & News
Announcements regarding our community
Beta Announcements & Development News
Announcements only for members
210 TOPICS 3.2k POSTS
5 days ago
Are there any release notes for the new sequans firmware?

Pylife
Discuss our new product here
14 TOPICS 45 POSTS
Dec 3, 2018, 7:46 AM
Thank you for the information provided, this is very helpful to

Pybytes IoT platform
Pybytes IoT platform for all Pycom devices
Announcements & News · Ideas · Support & Troubleshooting
72 TOPICS 246 POSTS
a day ago
Obviously you didn't look hard enough:

Getting Started
Post in here for any discussion, questions or support you need with your projects!
Discussion · WiPy 2&3 · LoPy · SiPy · GPy · FiPy · Pymakr · Pymate · Expansion Board · MicroPython
3.1k TOPICS 17.6k POSTS
28 minutes ago
Hello I am following a tutorial online and trying to import LTE

Tutorials
Tutorials for your Pycom stuff
81 TOPICS 390 POSTS
6 days ago
I'm struggling to get this to work. After following the

Pycom on Twitter

Tweets by @pycomIOT

Pycom Retweeted
Pycom
@pycomIOT
How'd you get your idea into a #factory and rolling out the other side as a finished product?

Find out 27 May @FactoryBerlin - we'll share experiences gained working on our portfolio, customer projects & more

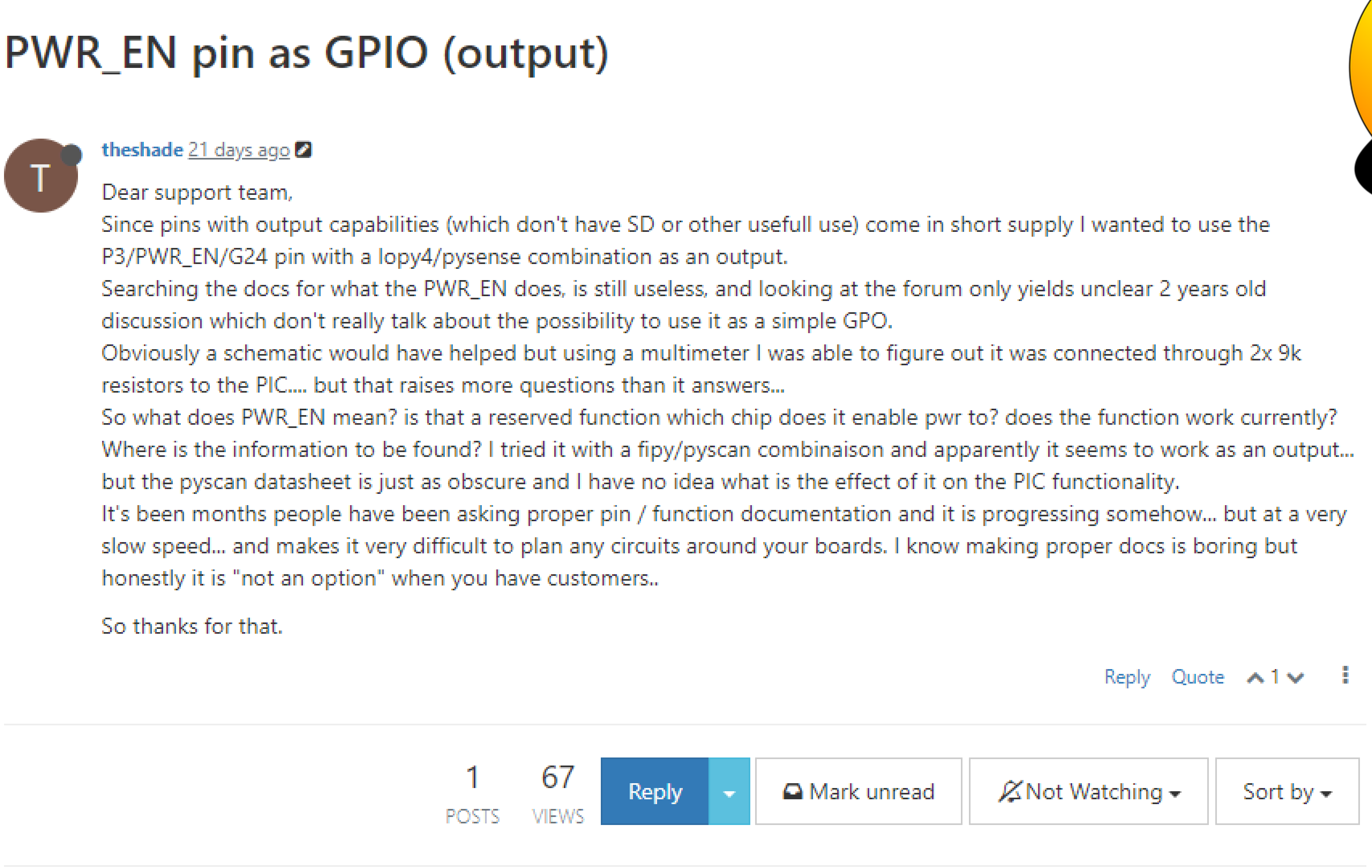
JOIN US eventbrite.co.uk/e/berlin-pycom…#LPWAN #LoRa #SigFox #NBIoT #Berlin

Mar 30, 2019

Pycom Retweeted

https://forum.pycom.io/

La documentation laisser parfois à désirer

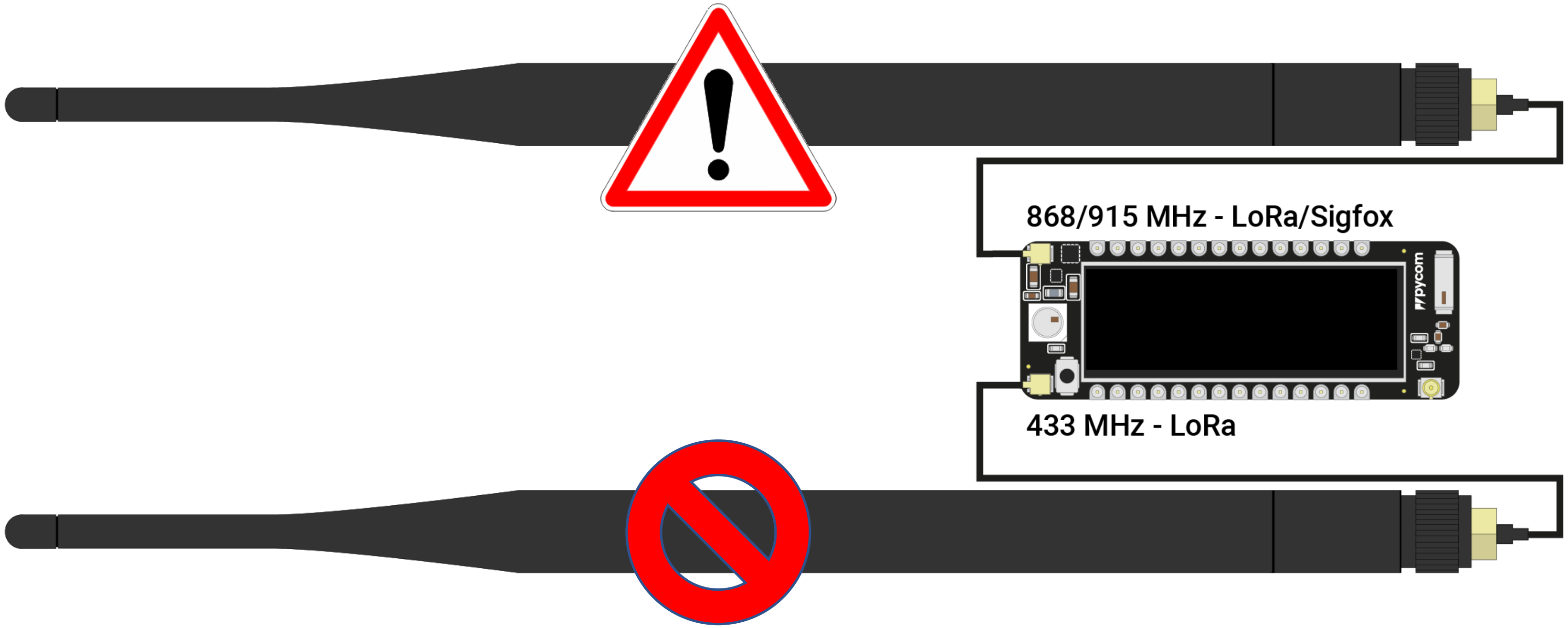


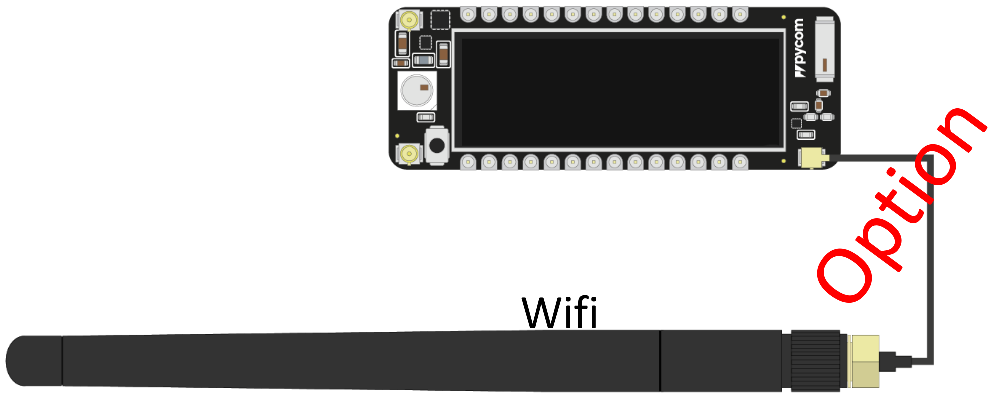
PWR_EN pin as GPIO (output)

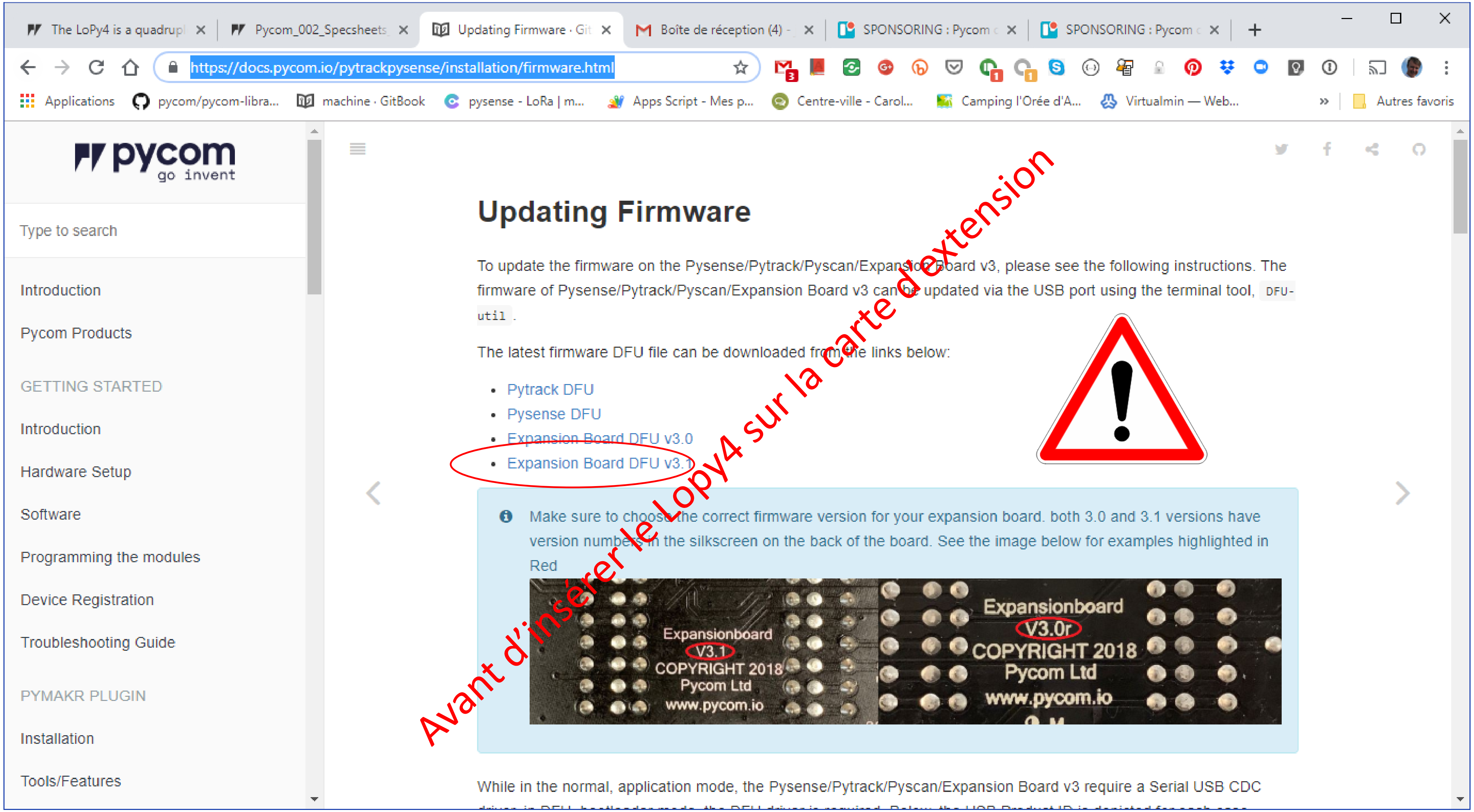
theshade 21 days ago

Dear support team,
Since pins with output capabilities (which don't have SD or other usefull use) come in short supply I wanted to use the P3/PWR_EN/G24 pin with a lopy4/pysense combination as an output.
Searching the docs for what the PWR_EN does, is still useless, and looking at the forum only yields unclear 2 years old discussion which don't really talk about the possibility to use it as a simple GPO.
Obviously a schematic would have helped but using a multimeter I was able to figure out it was connected through 2x 9k resistors to the PIC.... but that raises more questions than it answers...
So what does PWR_EN mean? is that a reserved function which chip does it enable pwr to? does the function work currently? Where is the information to be found? I tried it with a fipy/pyscan combinaison and apparently it seems to work as an output... but the pyscan datasheet is just as obscure and I have no idea what is the effect of it on the PIC functionality.
It's been months people have been asking proper pin / function documentation and it is progressing somehow... but at a very slow speed... and makes it very difficult to plan any circuits around your boards. I know making proper docs is boring but honestly it is "not an option" when you have customers..

So thanks for that.


868/915 MHz - LoRa/Sigfox
433 MHz - LoRa
pycom


Wifi
Option
pycom



Updating Firmware

To update the firmware on the Pysense/Pytrack/Pyscan/Expansion Board v3, please see the following instructions. The firmware of Pysense/Pytrack/Pyscan/Expansion Board v3 can be updated via the USB port using the terminal tool, `DFU-util`.

The latest firmware DFU file can be downloaded from the links below:

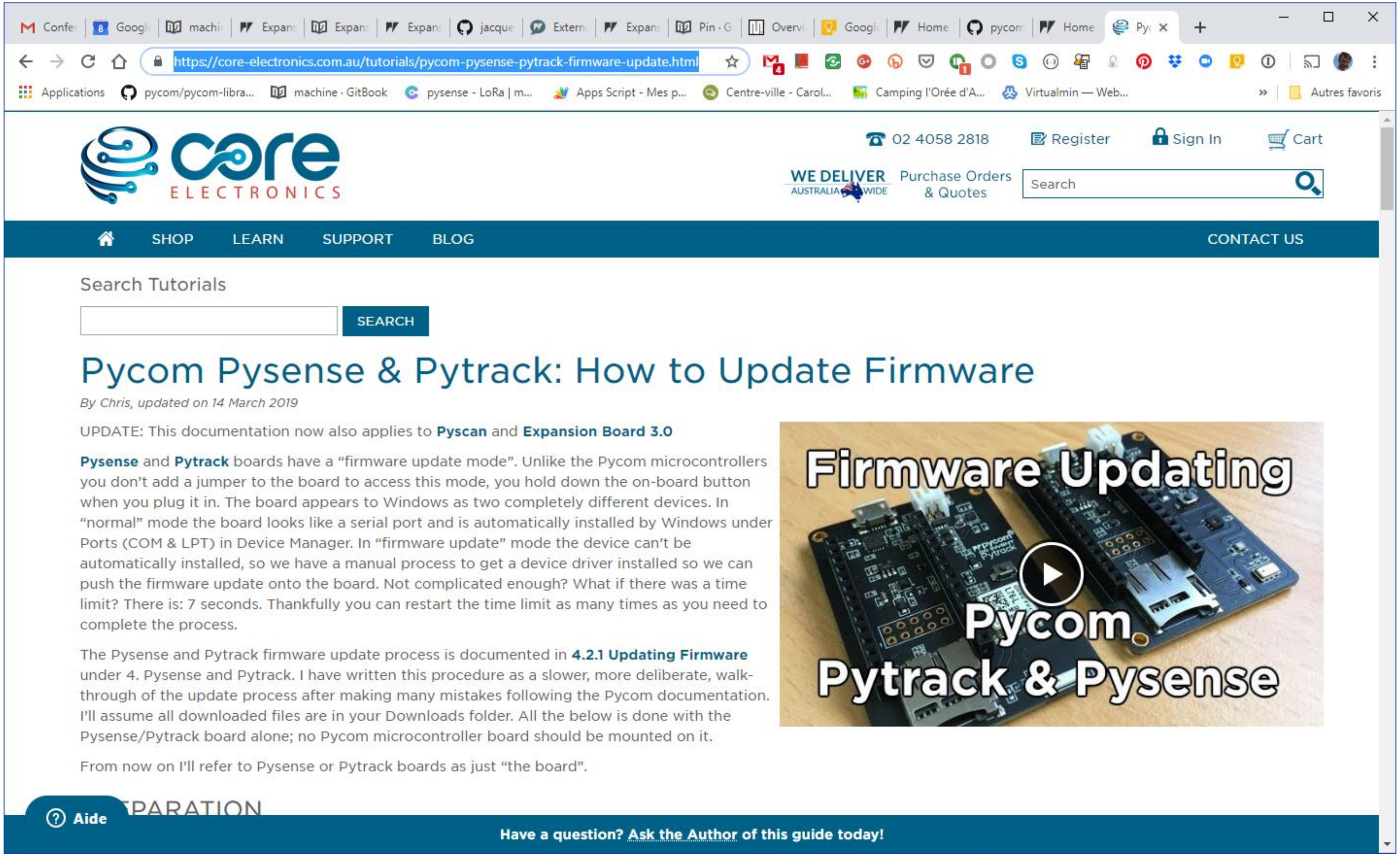
- Pytrack DFU
- Pysense DFU
- Expansion Board DFU v3.0
- Expansion Board DFU v3.1

Make sure to choose the correct firmware version for your expansion board. both 3.0 and 3.1 versions have version numbers in the silkscreen on the back of the board. See the image below for examples highlighted in Red

While in the normal, application mode, the Pysense/Pytrack/Pyscan/Expansion Board v3 require a Serial USB CDC

Avant d'insérer le Lopy4 sur la carte d'extension

https://docs.pycom.io/pytrackpysense/installation/firmware.html



Pycom Pysense & Pytrack: How to Update Firmware

By Chris, updated on 14 March 2019

UPDATE: This documentation now also applies to **Pyscan** and **Expansion Board 3.0**

Pysense and **Pytrack** boards have a "firmware update mode". Unlike the Pycom microcontrollers you don't add a jumper to the board to access this mode, you hold down the on-board button when you plug it in. The board appears to Windows as two completely different devices. In "normal" mode the board looks like a serial port and is automatically installed by Windows under Ports (COM & LPT) in Device Manager. In "firmware update" mode the device can't be automatically installed, so we have a manual process to get a device driver installed so we can push the firmware update onto the board. Not complicated enough? What if there was a time limit? There is: 7 seconds. Thankfully you can restart the time limit as many times as you need to complete the process.

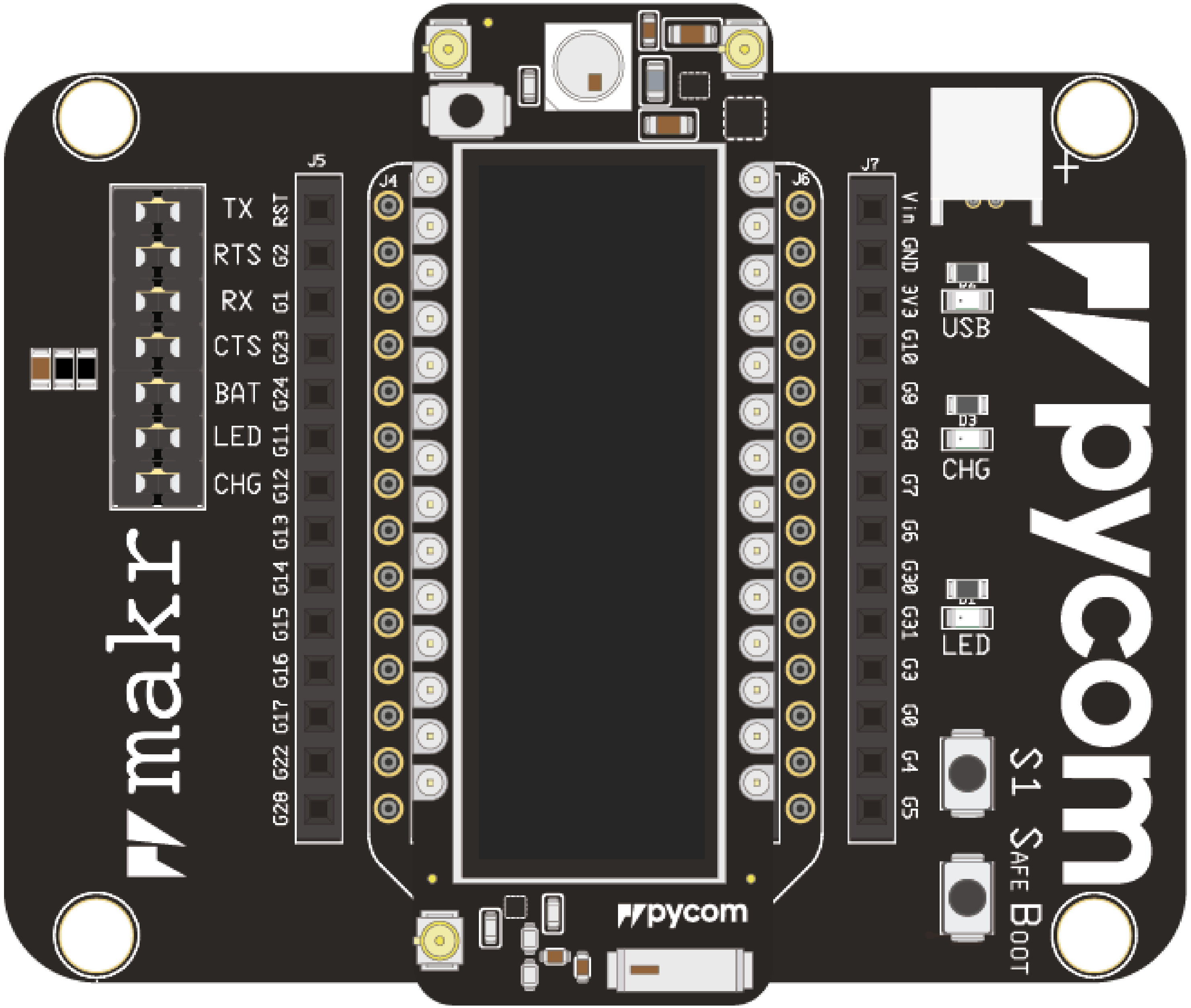
The Pysense and Pytrack firmware update process is documented in **4.2.1 Updating Firmware** under 4. Pysense and Pytrack. I have written this procedure as a slower, more deliberate, walk-through of the update process after making many mistakes following the Pycom documentation. I'll assume all downloaded files are in your Downloads folder. All the below is done with the Pysense/Pytrack board alone; no Pycom microcontroller board should be mounted on it.

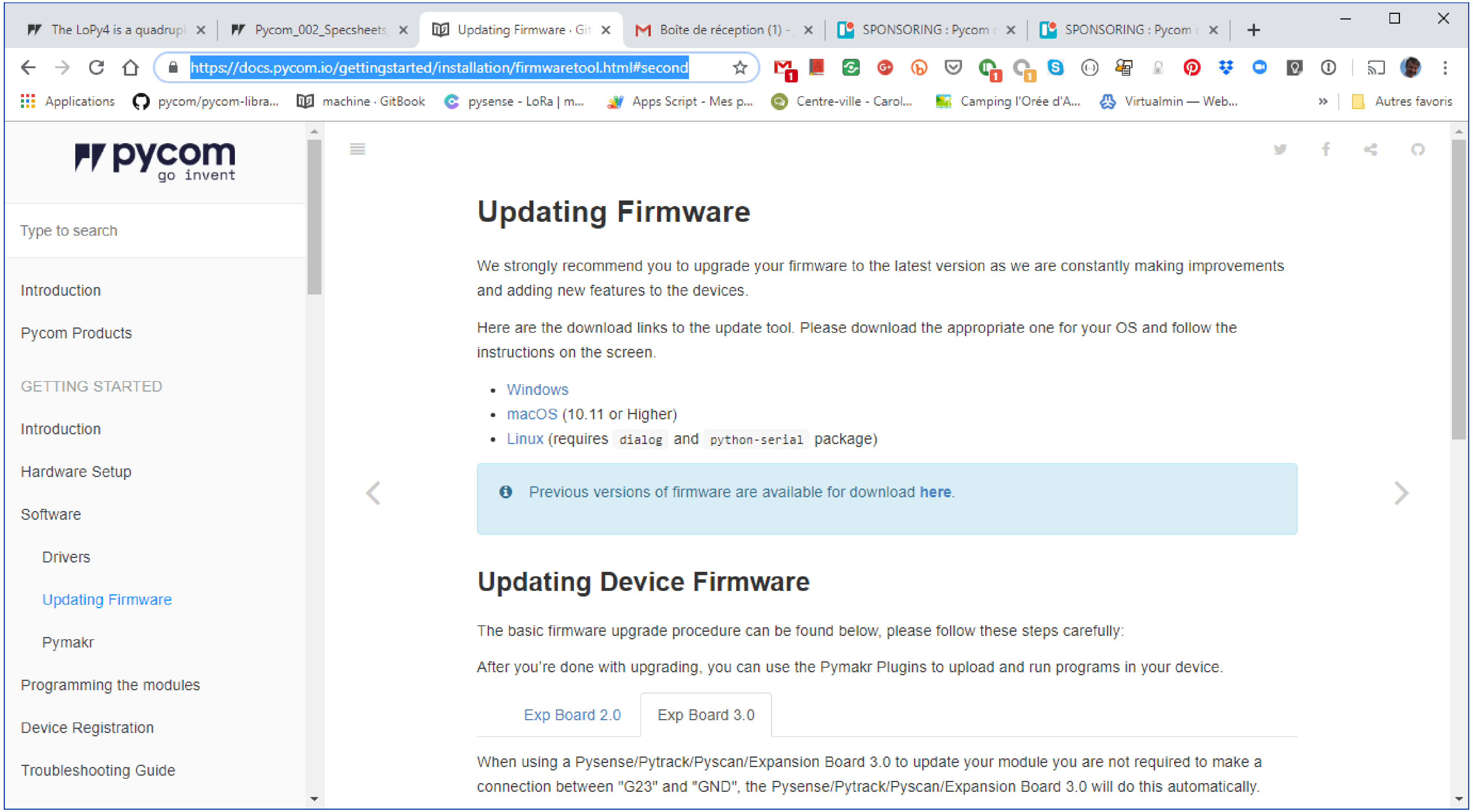
From now on I'll refer to Pysense or Pytrack boards as just "the board".

PARATION

Have a question? Ask the Author of this guide today!

https://core-electronics.com.au/tutorials/pycom-pysense-pytrack-firmware-update.html


TX
RTS
RX
CTS
BAT
LED
CHG
J5
RST
G2
G1
G23
G24
G11
G12
G13
G14
G15
G16
G17
G22
G28
J4
J6
J7
Vin
GND
3V3
G10
G9
G8
G7
G6
G30
G31
G3
G0
G4
G5
USB
D3
CHG
LED
+
pycom
S1
Safe Boot
makr
pycom



Updating Firmware

We strongly recommend you to upgrade your firmware to the latest version as we are constantly making improvements and adding new features to the devices.

Here are the download links to the update tool. Please download the appropriate one for your OS and follow the instructions on the screen.

- Windows
- macOS (10.11 or Higher)
- Linux (requires `dialog` and `python-serial` package)

ℹ Previous versions of firmware are available for download **here**.

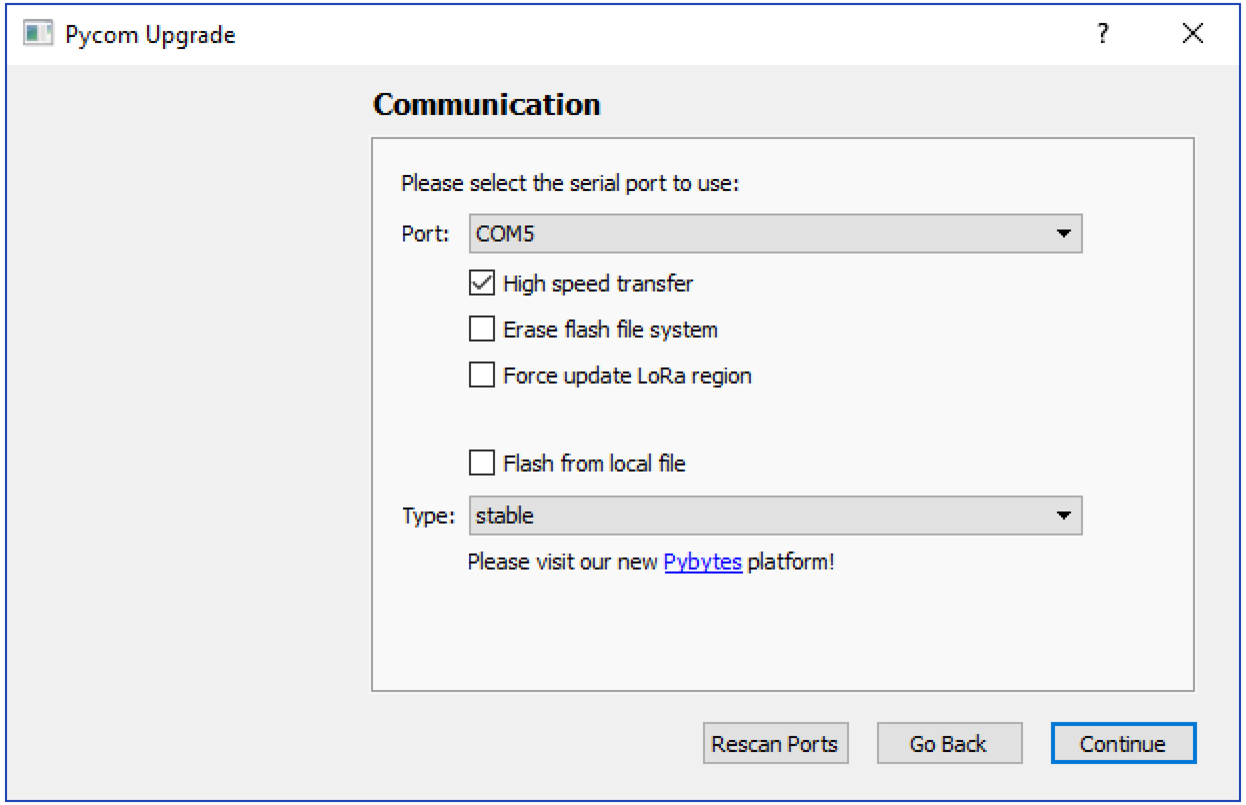
Updating Device Firmware

The basic firmware upgrade procedure can be found below, please follow these steps carefully:

After you're done with upgrading, you can use the Pymakr Plugins to upload and run programs in your device.

When using a Pysense/Pytrack/Pyscan/Expansion Board 3.0 to update your module you are not required to make a connection between "G23" and "GND", the Pysense/Pytrack/Pyscan/Expansion Board 3.0 will do this automatically.

https://docs.pycom.io/gettingstarted/installation/firmwaretool.html#second



Communication

Please select the serial port to use:

Port: COM5

☑ High speed transfer

□ Erase flash file system

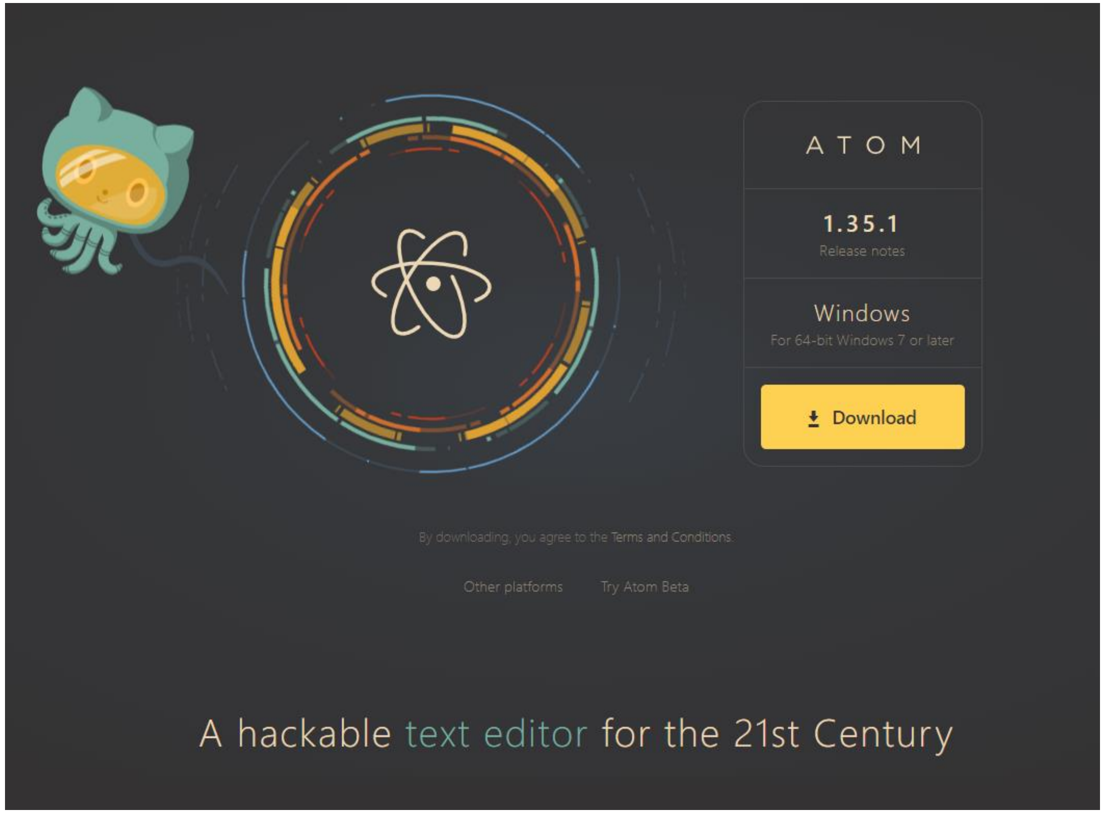
□ Force update LoRa region

□ Flash from local file

Type: stable

Please visit our new Pybytes platform!

https://atom.io/
https://github.com/pycom/pymakr-atom/



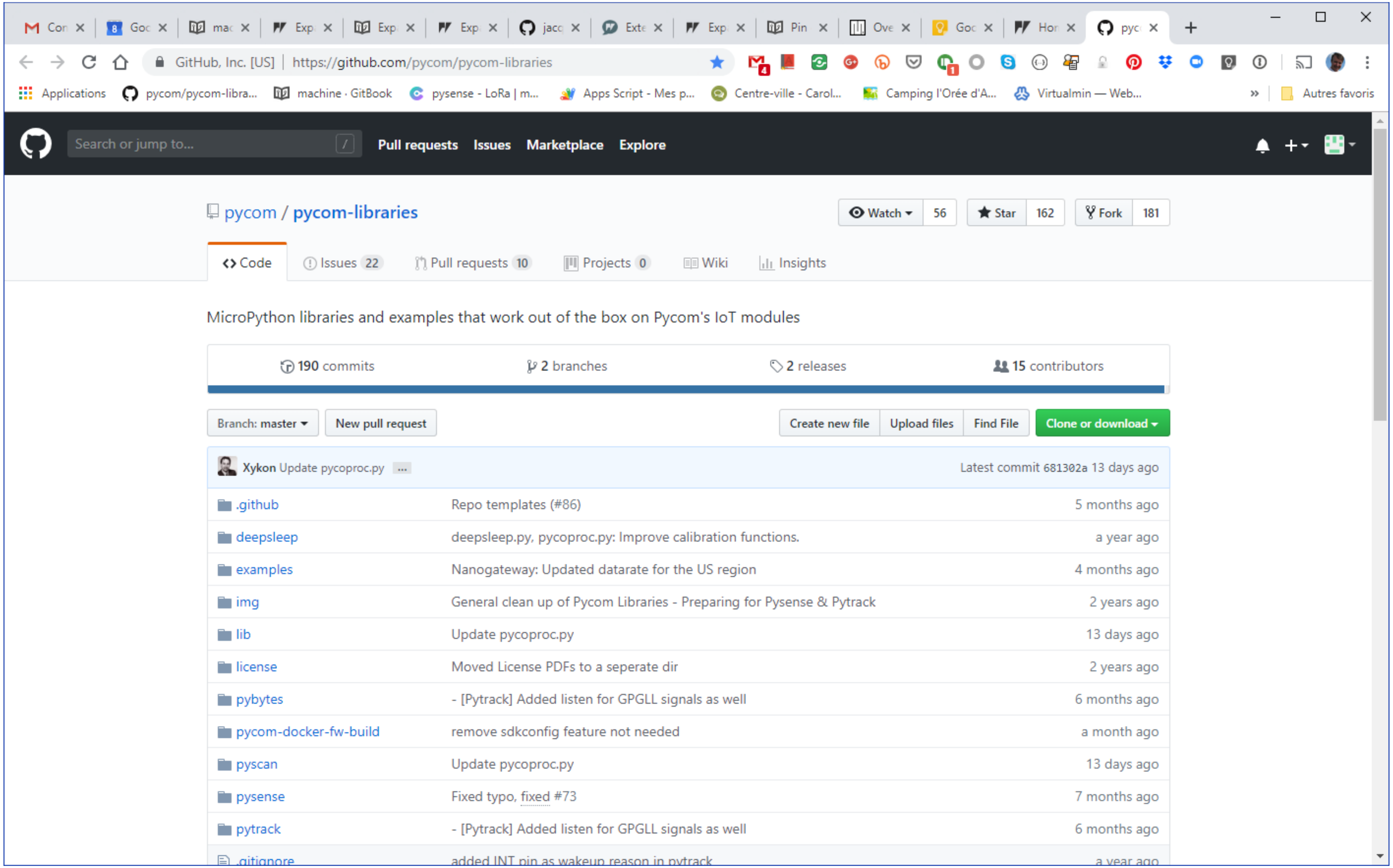
By downloading, you agree to the Terms and Conditions.

A hackable text editor for the 21st Century

pymakr
plugins

Micropython

- http://docs.micropython.org/en/latest/
  - Micropython is a lean and efficient implementation of the Python 3 programming language that includes a small subset of the Python standard library and is optimized to run on microcontrollers and in constrained environments.
  - Micropython is packed full of advanced features such as an interactive prompt, arbitrary precision integers, closures, list comprehension, generators, exception handling and more. Yet it is compact enough to fit and run within just 256k of code space and 16k of RAM.
  - Micropython aims to be as compatible with normal Python as possible to allow you to transfer code with ease from the desktop to a microcontroller or embedded system.
- Micropython differences from CPython:
  - http://docs.micropython.org/en/latest/genrst/index.html



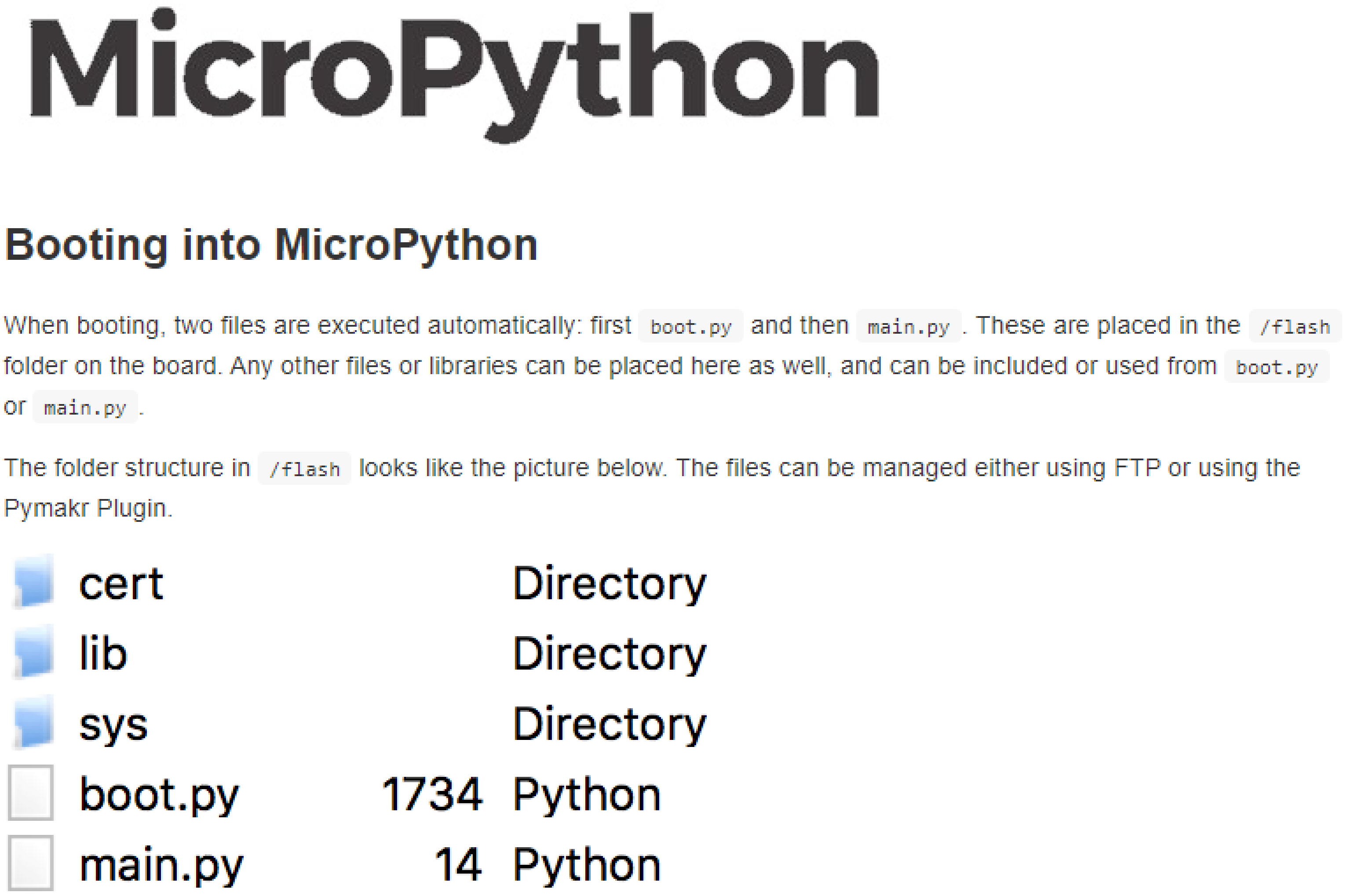
MicroPython libraries and examples that work out of the box on Pycom's IoT modules

190 commits | 2 branches | 2 releases | 15 contributors

Xykon Update pycoproc.py … Latest commit 681302a 13 days ago

.github	Repo templates (#86)	5 months ago
deepsleep	deepsleep.py, pycoproc.py: Improve calibration functions.	a year ago
examples	Nanogateway: Updated datarate for the US region	4 months ago
img	General clean up of Pycom Libraries - Preparing for Pysense & Pytrack	2 years ago
lib	Update pycoproc.py	13 days ago
license	Moved License PDFs to a seperate dir	2 years ago
pybytes	- [Pytrack] Added listen for GPGLL signals as well	6 months ago
pycom-docker-fw-build	remove sdkconfig feature not needed	a month ago
pyscan	Update pycoproc.py	13 days ago
pysense	Fixed typo, fixed #73	7 months ago
pytrack	- [Pytrack] Added listen for GPGLL signals as well	6 months ago
.gitignore	added INT pin as wakeup reason in pytrack	a year ago

https://github.com/pycom/pycom-libraries



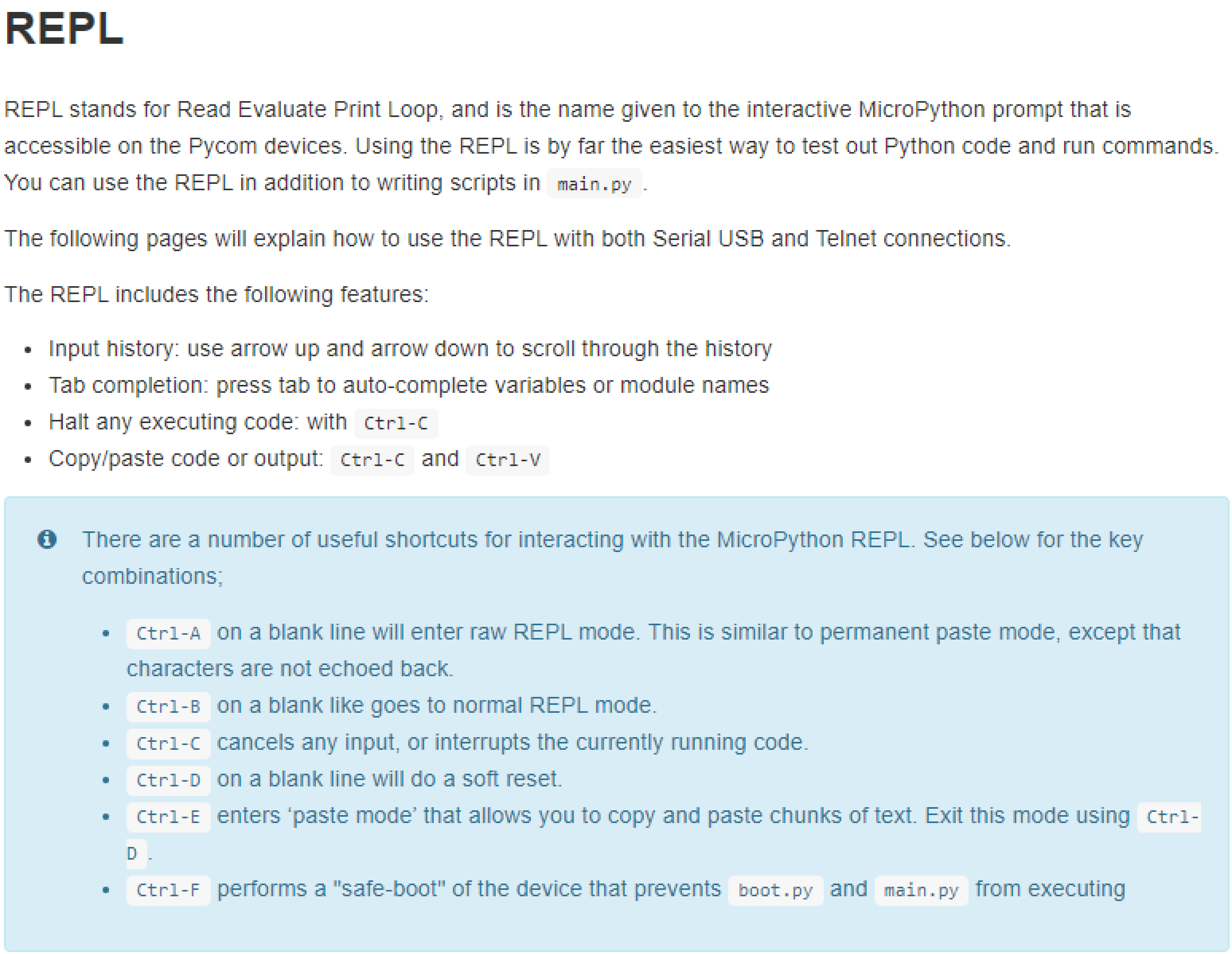
MicroPython

Booting into MicroPython

When booting, two files are executed automatically: first `boot.py` and then `main.py`. These are placed in the `/flash` folder on the board. Any other files or libraries can be placed here as well, and can be included or used from `boot.py` or `main.py`.

The folder structure in `/flash` looks like the picture below. The files can be managed either using FTP or using the Pymakr Plugin.

📁	cert		Directory
📁	lib		Directory
📁	sys		Directory
□	boot.py	1734	Python
□	main.py	14	Python



REPL

REPL stands for Read Evaluate Print Loop, and is the name given to the interactive MicroPython prompt that is accessible on the Pycom devices. Using the REPL is by far the easiest way to test out Python code and run commands. You can use the REPL in addition to writing scripts in `main.py`.

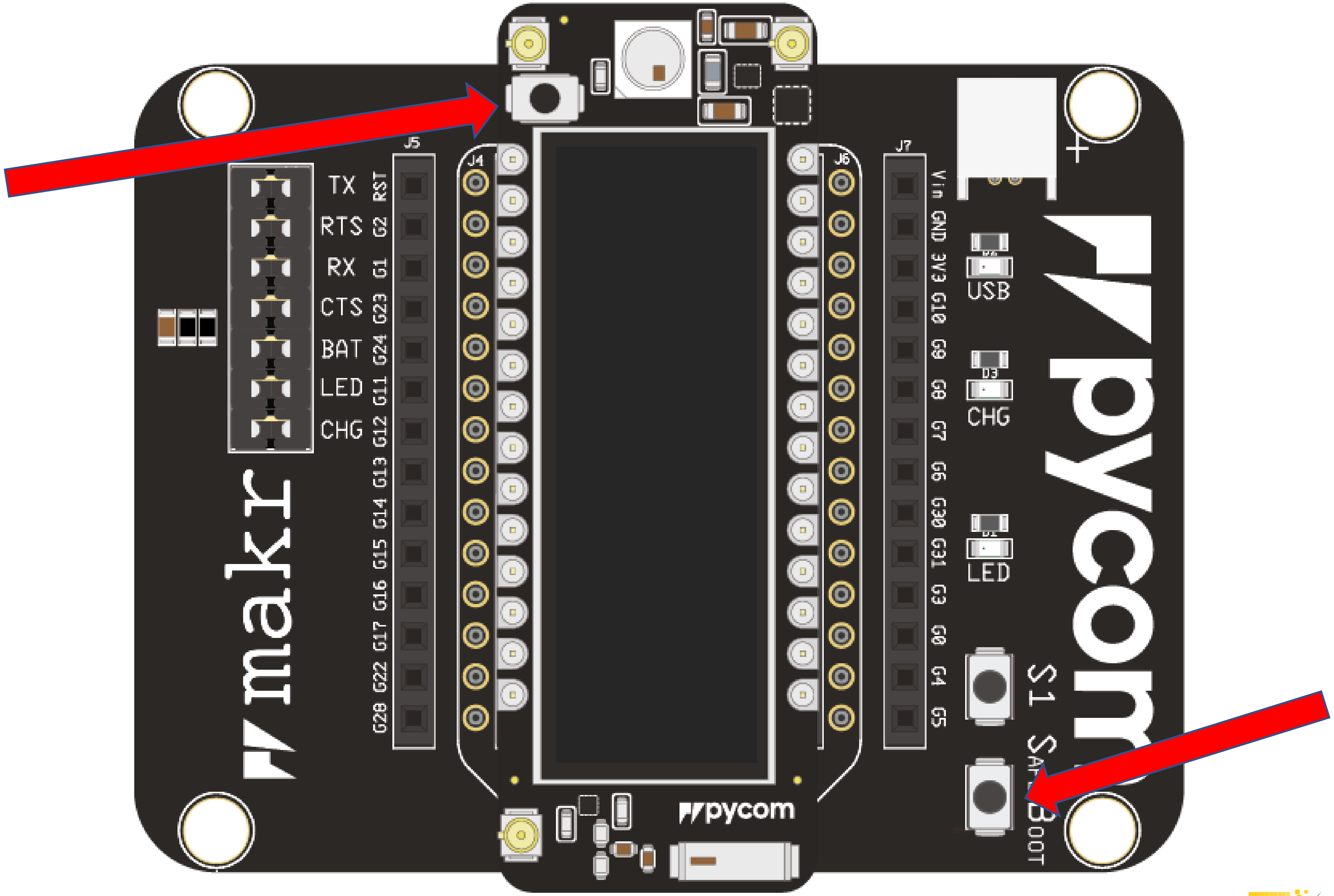
The following pages will explain how to use the REPL with both Serial USB and Telnet connections.

The REPL includes the following features:

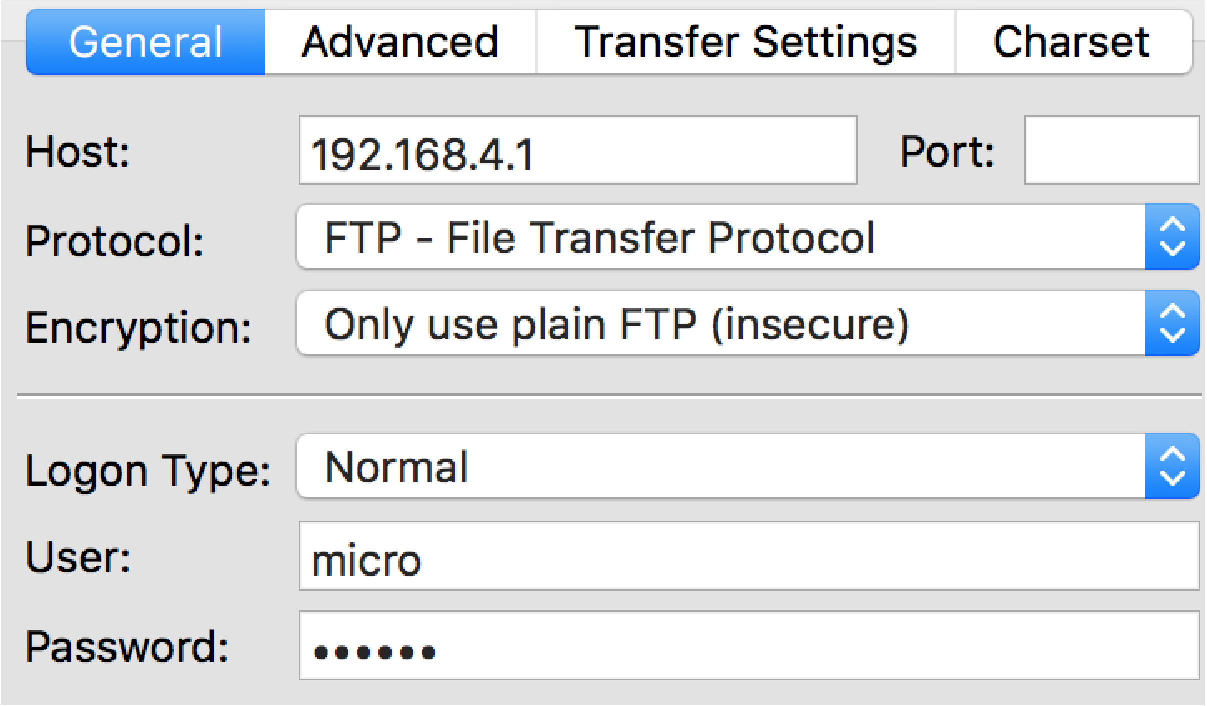
- Input history: use arrow up and arrow down to scroll through the history
- Tab completion: press tab to auto-complete variables or module names
- Halt any executing code: with `Ctrl-C`
- Copy/paste code or output: `Ctrl-C` and `Ctrl-V`

> ℹ There are a number of useful shortcuts for interacting with the MicroPython REPL. See below for the key combinations;
>
> - `Ctrl-A` on a blank line will enter raw REPL mode. This is similar to permanent paste mode, except that characters are not echoed back.
> - `Ctrl-B` on a blank like goes to normal REPL mode.
> - `Ctrl-C` cancels any input, or interrupts the currently running code.
> - `Ctrl-D` on a blank line will do a soft reset.
> - `Ctrl-E` enters ‘paste mode’ that allows you to copy and paste chunks of text. Exit this mode using `Ctrl-D`.
> - `Ctrl-F` performs a "safe-boot" of the device that prevents `boot.py` and `main.py` from executing

Reset



Safe
Boot



Host: 192.168.4.1 Port:

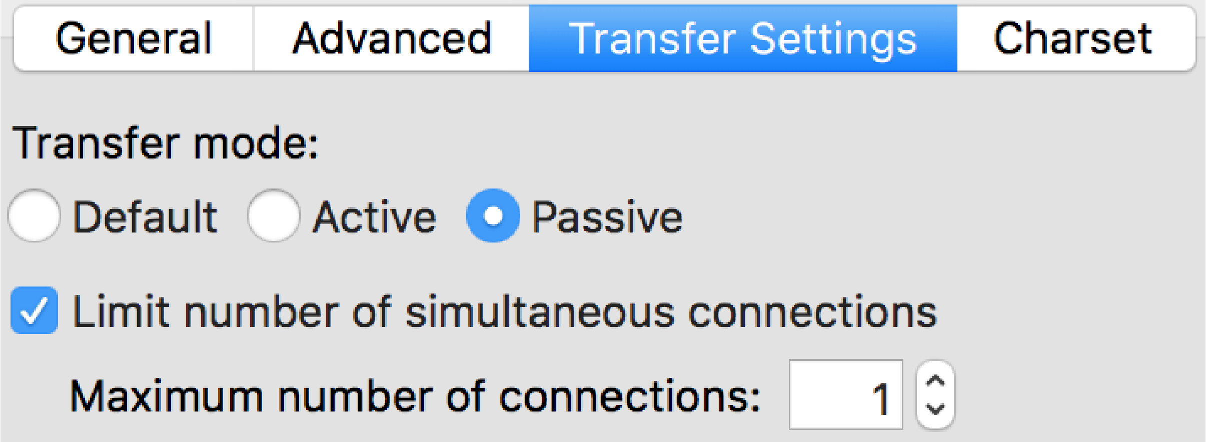
Protocol: FTP - File Transfer Protocol

Encryption: Only use plain FTP (insecure)

Logon Type: Normal

User: micro

Password: ••••••

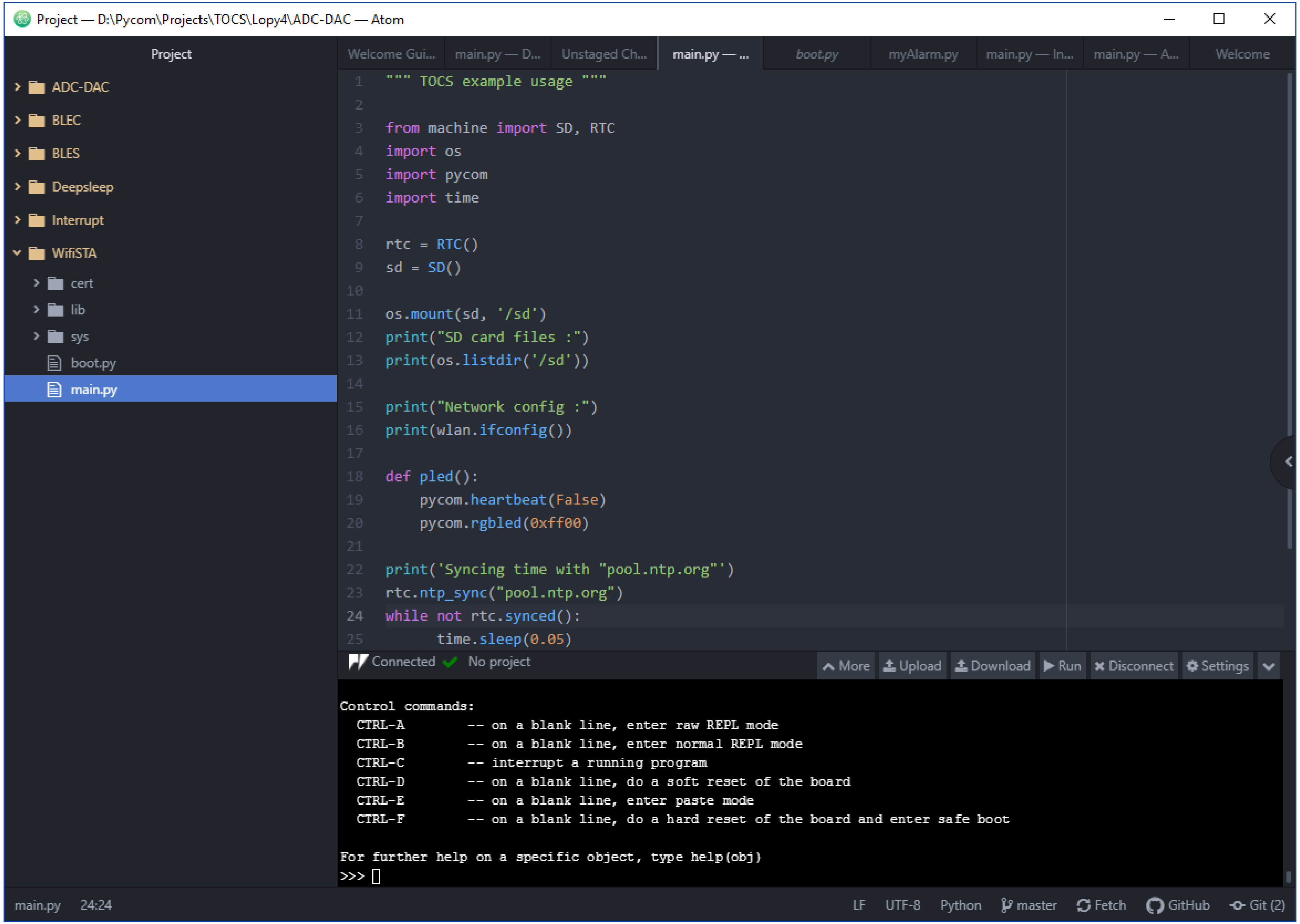


Transfer mode:

○ Default ○ Active ● Passive

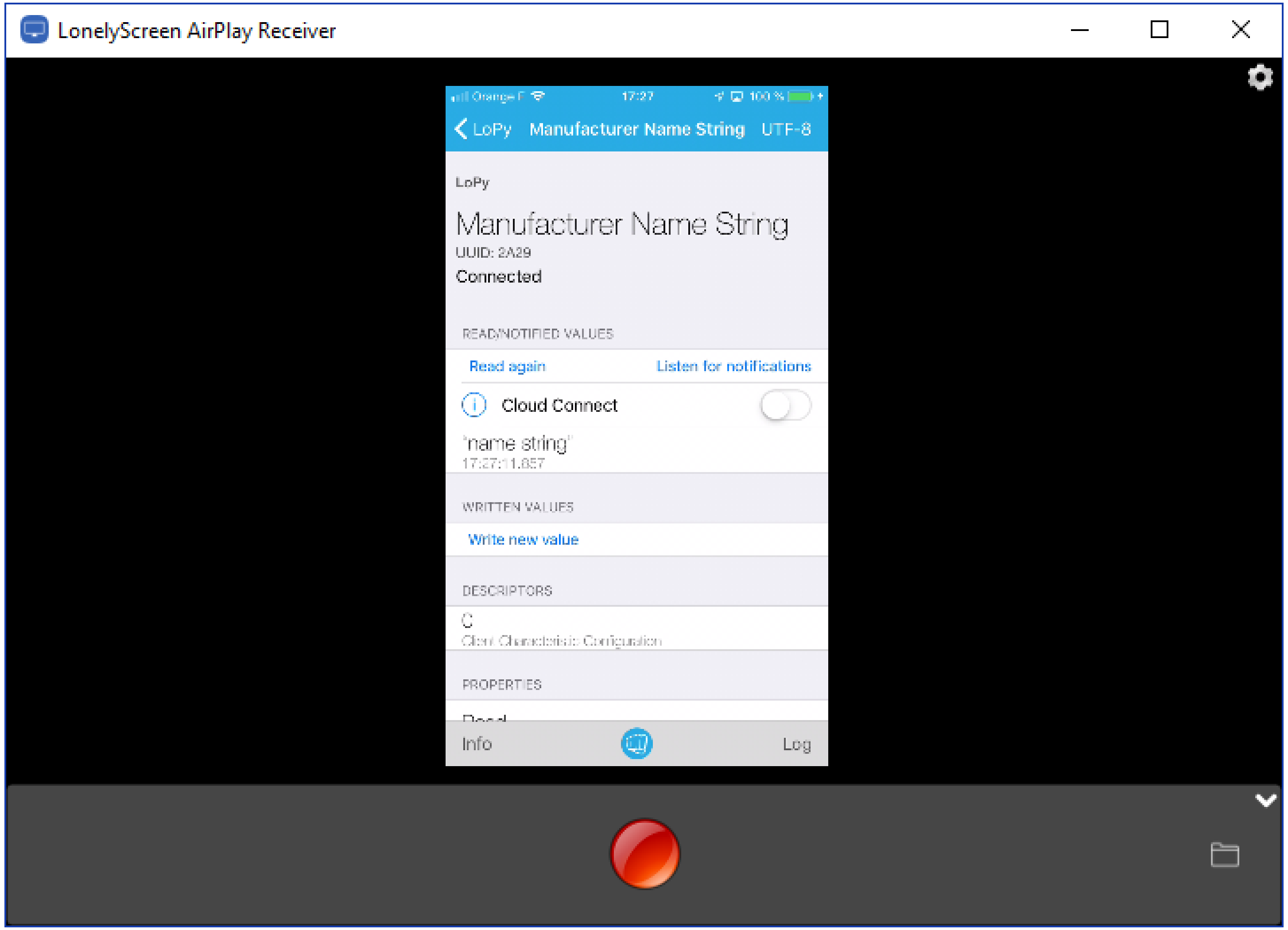
☑ Limit number of simultaneous connections

Maximum number of connections: 1



```python
""" TOCS example usage """

from machine import SD, RTC
import os
import pycom
import time

rtc = RTC()
sd = SD()

os.mount(sd, '/sd')
print("SD card files :")
print(os.listdir('/sd'))

print("Network config :")
print(wlan.ifconfig())

def pled():
    pycom.heartbeat(False)
    pycom.rgbled(0xff00)

print('Syncing time with "pool.ntp.org"')
rtc.ntp_sync("pool.ntp.org")
while not rtc.synced():
      time.sleep(0.05)
```

```
Control commands:
  CTRL-A        -- on a blank line, enter raw REPL mode
  CTRL-B        -- on a blank line, enter normal REPL mode
  CTRL-C        -- interrupt a running program
  CTRL-D        -- on a blank line, do a soft reset of the board
  CTRL-E        -- on a blank line, enter paste mode
  CTRL-F        -- on a blank line, do a hard reset of the board and enter safe boot

For further help on a specific object, type help(obj)
>>> 
```



Manufacturer Name String UTF-8

LoPy

Manufacturer Name String

UUID: 2A29

Connected

READ/NOTIFIED VALUES

Read again   Listen for notifications

Cloud Connect

'name string'
17:27:11.857

WRITTEN VALUES

Write new value

DESCRIPTORS

0
Client Characteristic Configuration

PROPERTIES

Read