

Collecting data from IoT devices using Sigfox network

Making data collection easier

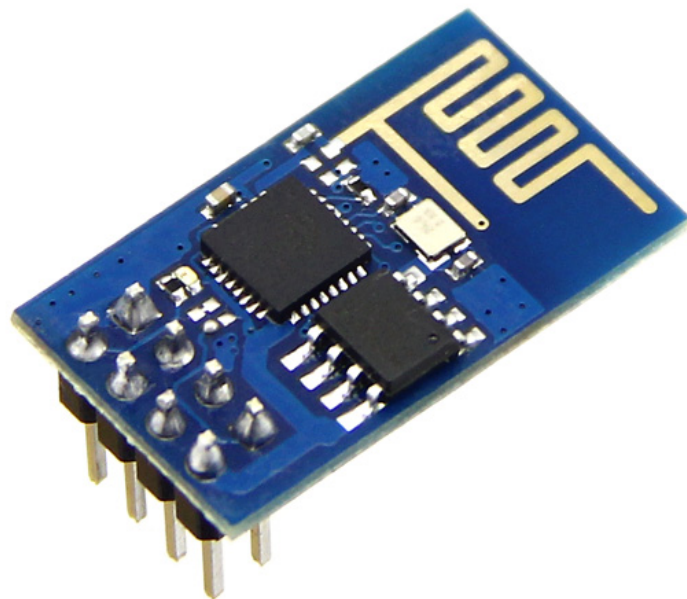
Jan Krupa
Operations Engineer
JKrupa@suse.com

The background features abstract geometric shapes in two shades of green. A large teal shape occupies the left and top portions, while a bright green shape is on the right. They are separated by white diagonal lines.

About me

IoT networks

The background features abstract geometric shapes in two shades of green. A large teal shape occupies the left and top portions, while a bright green shape is on the right. A white diagonal line separates the two green areas.









	SIGFOX	LoRa	clean slate	NB LTE-M Rel. 13	LTE-M Rel. 12/13	EC-GSM Rel. 13	5G (targets)
			cloT				

Range (outdoor) MCL	<13km 160 dB	<11km 157 dB	<15km 164 dB	<15km 164 dB	<11km 156 dB	<15km 164 dB	<15km 164 dB
Spectrum Bandwidth	Unlicensed 900MHz 100Hz	Unlicensed 900MHz <500kHz	Licensed 7-900MHz 200kHz or dedicated	Licensed 7-900MHz 200kHz or shared	Licensed 7-900MHz 1.4 MHz or shared	Licensed 8-900MHz 2.4 MHz or shared	Licensed 7-900MHz shared
Data rate	<100bps	<10 kbps	<50kbps	<150kbps	<1 Mbps	10kbps	<1 Mbps
Battery life	>10 years	>10 years	>10 years	>10 years	>10 years	>10 years	>10 years
Availability	Today	Today	2016	2016	2016	2016	beyond 2020





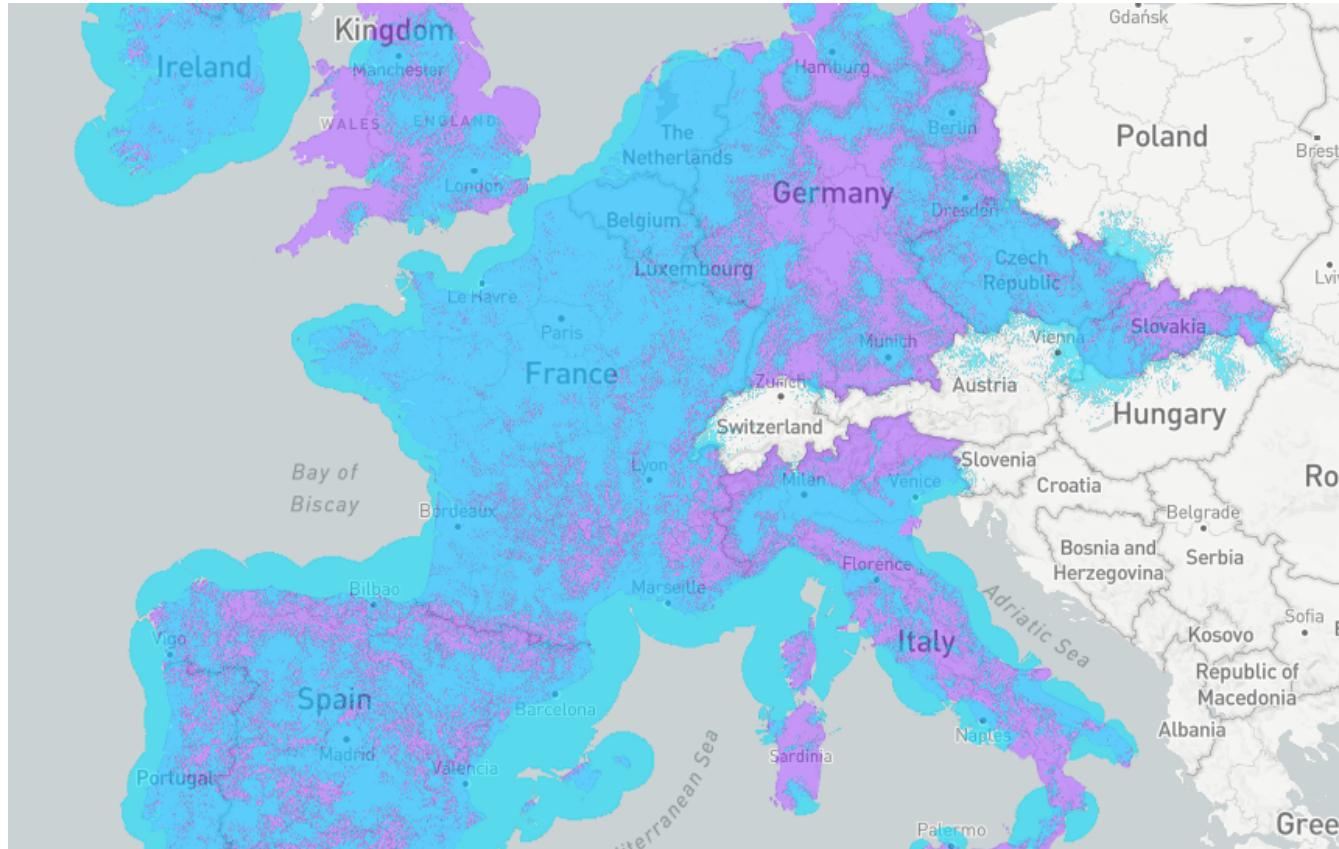
Sigfox network

What is Sigfox

“Sigfox employs a proprietary technology that enables communication using the Industrial, Scientific and Medical ISM radio band which uses **868MHz in Europe** and **902MHz in the US**. It utilizes a wide-reaching signal that passes freely through solid objects, called “ultra narrowband” and requires little energy, being termed “Low-power Wide-area network (LPWAN)”. The network is based on one-hop star topology and requires a mobile operator to carry the generated traffic. The signal can also be used to easily cover large areas and to reach underground objects.”



Sigfox coverage



What I can transfer

- uplink message limit is 12 bytes (140 per day)
- downlink message limit is 8 bytes (4 per day)
- 140 messages per day = 1 message per 10 minutes



Pricing

- One year free subscription with the hardware
- After that €0.50 - €1.00 per month

(depending on country, local provider and amount of transferred data)



Roaming

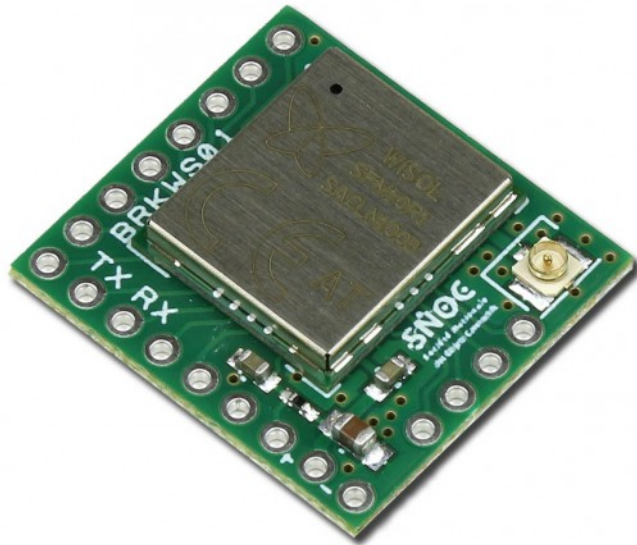
- Roaming is included in the monthly fee and is free of charge



The background features abstract geometric shapes in two shades of green. A large teal shape occupies the left and top portions, while a bright green shape is on the right. They are separated by a white diagonal line.

Sigfox hardware

Sigfox BRKWS01 (Wisol SFM10R1)



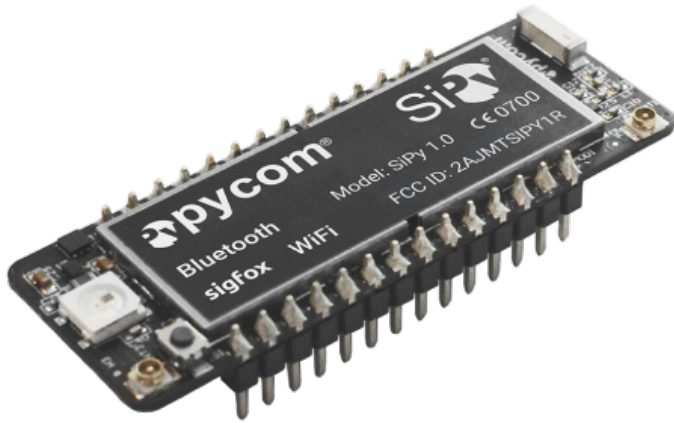
- Easily connectable to Arduino
- Price: €4
- Price including breakout board and antenna: €24

Pycom SiPy – overview



- Espressif ESP32 chipset
- Dual processor + WiFi and Bluetooth radio System on Chip.
- Network processor handles the WiFi connectivity and the IPv6 stack
- Price: €30

Pycom SiPy – hardware



- RAM: 512kb
- Flash Memory: 4MB
- GPIO: up to 24
- Python multi-threading



Pycom SiPy – interfaces



- 2 x UART, 2 x SPI, I2C, I2S, micro SD card
- Analog channels: 8x12 bit ADCs
- DMA on all peripherals
- GPIO: up to 24

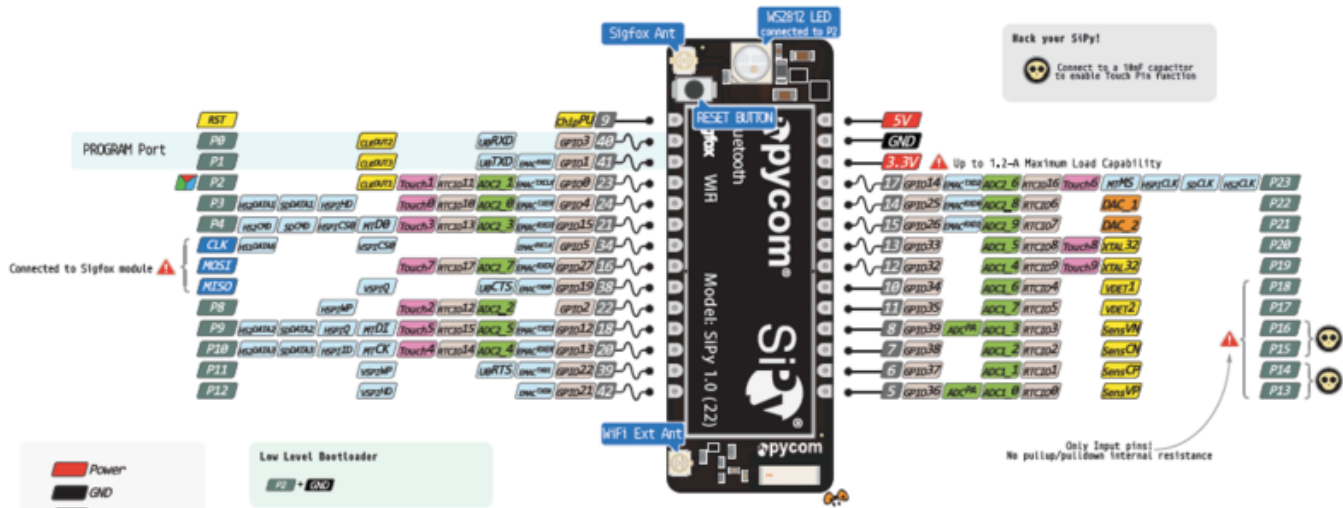




⚠ Absolute MAX per pin 12mA recommended 10mA

Internal Functions			
35	GPIO18	VSPI_CLK	Sigfox Reset
25	GPIO16	IMAC_CLK	External Antenna Switch
36	GPIO23	VSPI_CS	Sigfox module GPIO2
27	GPIO17	IMAC_CS	Sigfox Select

Back your SiPy!
 ☹ Connect to a 100µF capacitor to enable Touch Pin Function



Only Input pins!
 No pullup/pulldown internal resistance

- Power
- GND
- Serial Pin
- Analog Pin
- Control
- Physical Pin
- Port Pin
- Touch Pin
- DAC Pin
- PWM Pin

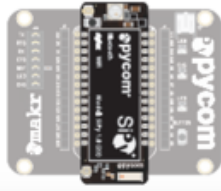
Low Level Bootloader

Boot modes and safe boot

1-3sec Safe boot, Latest firmware is selected

4-6sec Safe boot, previous user update selected

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



The background features abstract geometric shapes in two shades of green. A large, dark teal shape occupies the left and top-left portions of the frame. To its right, a lighter green shape is partially visible, separated by a white diagonal line. The overall design is modern and minimalist.


How to use it

Registering to Sigfox network

```
>>> from network import Sigfox
>>> import binascii
>>> sigfox = Sigfox(mode=Sigfox.SIGFOX, rcz=Sigfox.RCZ1)
>>> print(binascii.hexlify(sigfox.id()))
b'12345678'
>>> print(binascii.hexlify(sigfox.pac()))
b'1234567890123456'
```




Registering to Sigfox network


 [Lost password](#)


Dev Kit Activation


Sigfox has partnered with a large number of companies that provide development kits or evaluation boards in order to test and prototype on Sigfox network. Most of them come with an included subscription. By using this form you'll be able to activate your subscription and create an account on Sigfox backend.


Choose your kit provider



ARDUINO


















sigfox KeyApp








ON Semiconductor®



Connect via USB

- Virtual serial port



Connect via WiFi

- Telnet
- FTP

```
$ telnet 192.168.4.1
Trying 192.168.4.1...
Connected to 192.168.4.1.
Escape character is '^]'.
MicroPython v1.8.6-556-g989d5ac9 on 2017-03-30; SiPy with ESP32
Login as: micro
Password:
Login succeeded!
Type "help()" for more information.
>>>
```



Configuring callbacks

Callbacks

Type

DATA

UPLINK

Channel

EMAIL

Send duplicate

☐

Custom payload config

str::char:6 i1::uint:16 i2::uint:32

Recipient

user@example.com

Multiple emails allowed separated by comma, semicolon or new line

Subject syntax: Subject with device {device}

Message syntax: Message containing time {time}, key1 {var1}, key2 {var2}...

Available variables: device, time, duplicate, snr, station, data, avgSnr, lat, lng, rssi, seqNumber

Custom variables: customData#str, customData#i1, customData#i2

Subject

Sigfox message from {device}

Message

```
str: {customData#str}
i1: {customData#i1}
i2: {customData#i2}

device: {device}
time: {time}
duplicate: {duplicate}
snr: {snr}
station: {station}
data: {data}
avgSnr: {avgSnr}
lat: {lat}
lng: {lng}
rssi: {rssi}
seqNumber: {seqNumber}
```

Ok

Cancel

Direct communication between devices

- You can also communicate directly between Pycom SiPy modules without using Sigfox network
- This is a feature of this particular module, not a Sigfox network feature



The background features abstract geometric shapes in two shades of green. A large teal shape occupies the left and top portions, while a bright green shape is on the right. They are separated by a white diagonal line.

Live demo

Sending the first message

```
import socket
from network import Sigfox
sigfox = Sigfox(mode=Sigfox.SIGFOX, rcz=Sigfox.RCZ1)
s = socket.socket(socket.AF_SIGFOX, socket.SOCK_RAW)
s.setblocking(True)
s.setsockopt(socket.SOL_SIGFOX, socket.SO_RX, False)
s.send('test12345678')
```



Sigfox portal email callback

Date: Mon, 17 Apr 2017 14:00:00 +0200 (CEST)

From: SIGFOX <backend-noreply@sigfox.com>

To: user@example.com

string: test12345678

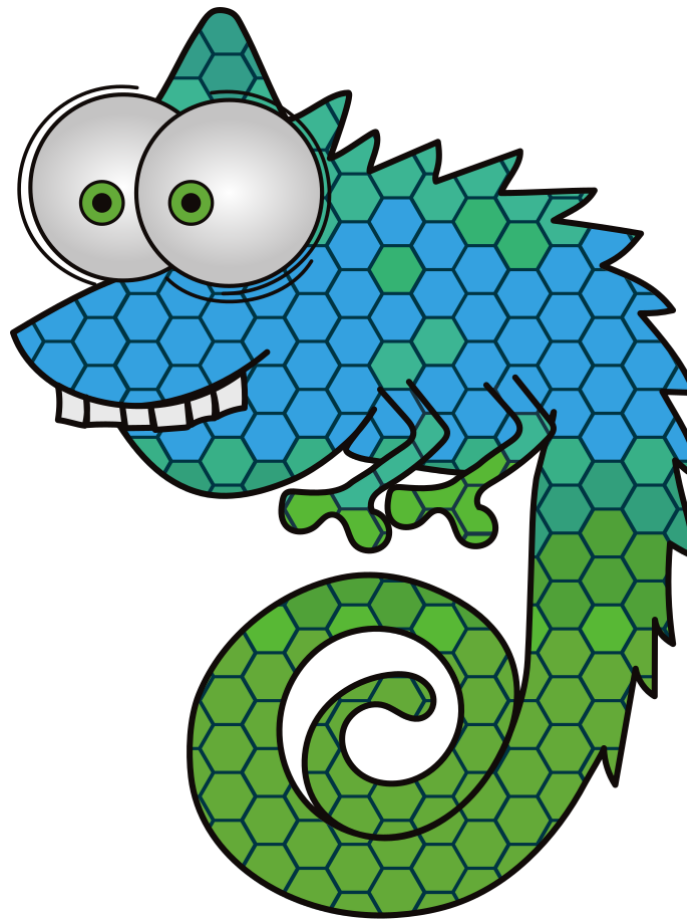


Questions?

You can find more information on my blog:
<https://jankrupa.com/>

The background features abstract geometric shapes in two shades of green. A large teal shape occupies the left and top portions, while a bright green shape is on the right. They are separated by a white diagonal line.

Thank you!



Join Us at www.opensuse.org



License

This slide deck is licensed under the Creative Commons Attribution-ShareAlike 4.0 International license. It can be shared and adapted for any purpose (even commercially) as long as Attribution is given and any derivative work is distributed under the same license.

Details can be found at <https://creativecommons.org/licenses/by-sa/4.0/>

General Disclaimer

This document is not to be construed as a promise by any participating organisation to develop, deliver, or market a product. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. openSUSE makes no representations or warranties with respect to the contents of this document, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. The development, release, and timing of features or functionality described for openSUSE products remains at the sole discretion of openSUSE. Further, openSUSE reserves the right to revise this document and to make changes to its content, at any time, without obligation to notify any person or entity of such revisions or changes. All openSUSE marks referenced in this presentation are trademarks or registered trademarks of SUSE LLC, in the United States and other countries. All third-party trademarks are the property of their respective owners.

Credits

Template

Richard Brown
rbrown@opensuse.org

Design & Inspiration

openSUSE Design Team

<http://opensuse.github.io/branding-guidelines/>