

El siguiente documento tiene como finalidad el explicar como se realizó la configuración de los dispositivos Lopy y Lopy 4, todos los pasos realizados a continuación se basan en información recopilada del primer semestre del 2020.

La configuración de estos dispositivos se divide en cuatro secciones: configuración de hardware, configuración de software, programación de los módulos y registro del dispositivo.

Configuración del Hardware

Para la configuración del dispositivo Lopy se procede a mirar los componentes necesarios para el correcto funcionamiento. Los componentes básicos que se necesitan son los siguientes:

- Tarjeta de expansión / PyTrack / PySense / PyScan
- Lopy
- Adaptador de antena
- Antena

Cabe aclarar que el Lopy en cualquiera de las versiones descritas en este documento se puede programar con la ayuda de cualquiera de las tarjetas mencionadas del primer ítem.

Luego de esto se procede con la conexión de la antena ya que dependiendo del conector donde se deje se va a tener una funcionalidad diferente. Como el objetivo es usar la tecnología de LoRa se conectará el adaptador de la antena al conector que permita el funcionamiento adecuado. La siguiente imagen (Fig.1) permite visualizar de manera más detallada en donde se debe hacer la conexión.

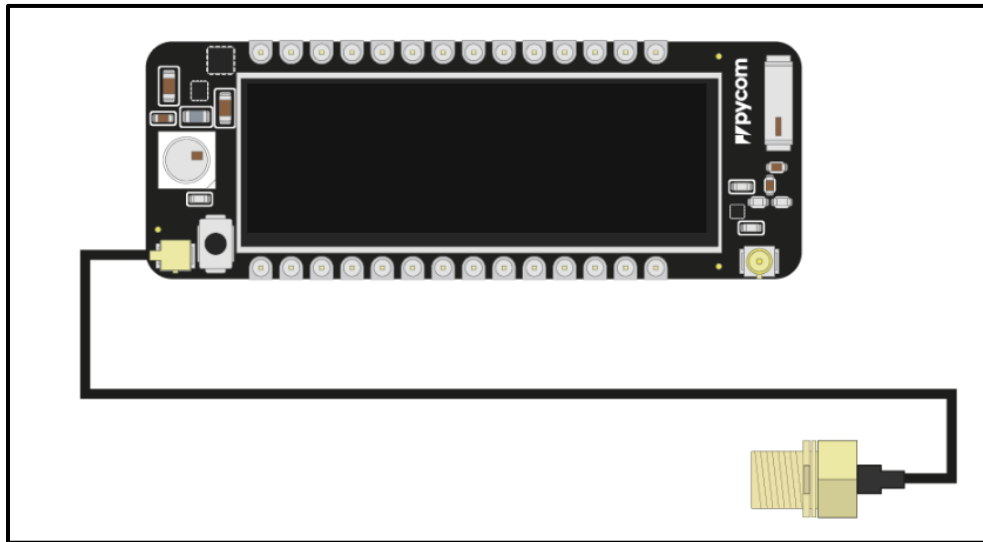


Fig. 1 - Conexión adaptador antenna al conector LoRa

Al realizar la conexión en este puerto se garantiza que se pueden usar las frecuencias de 868MHz y 915MHz. Una vez conectado el adaptador el unico restante es colocar la antena, para ello la siguiente imagen (Fig. 2) muestra cómo queda el montaje final.

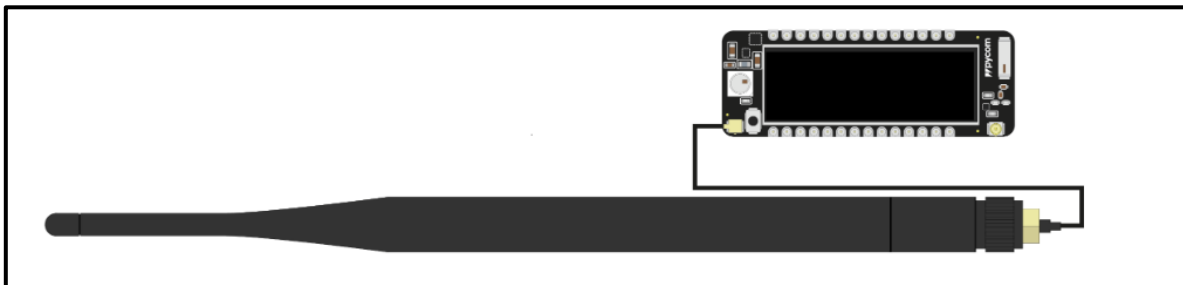


Fig. 2 – Montaje final conexión antena al Lopy

Hay que tener en cuenta que cada vez que se vaya a usar el Lopy no importa la aplicación que sea siempre debe conectarse la antena, ya que no conectarla implicara daños en el dispositivo a largo plazo. Otra cosa que se debe tener en cuenta es que al momento de realizar algún tipo de prueba que sea de envío de datos entre dos Lopy se tiene que hacer con la frecuencia de 915 MHz ya que en el territorio colombiano esta frecuencia no cuenta con restricciones.

Finalmente, para dejar el modulo conectado a cualquiera de las placas (placa de expansión, pytrack, pysence, pyscan) se muestra en las siguientes imágenes (Fig.3 a Fig.6) como quedaria la conexión.



Fig.5 – Conexión Lopy a pysense

Colocar las demás imágenes una vez sean tomadas con los dispositivos en campo.

Para el Lopy 4 se tiene los mismos componentes que para el Lopy, lo que varía para este caso es que se tienen tres conectores debido a que este dispositivo es uno de los más completos.

La siguiente imagen (Fig.7) muestra cómo sería la conexión del conector de la antena si se quiere usar la tecnología LoRa en su frecuencia de 868 MHz o 915 MHz.

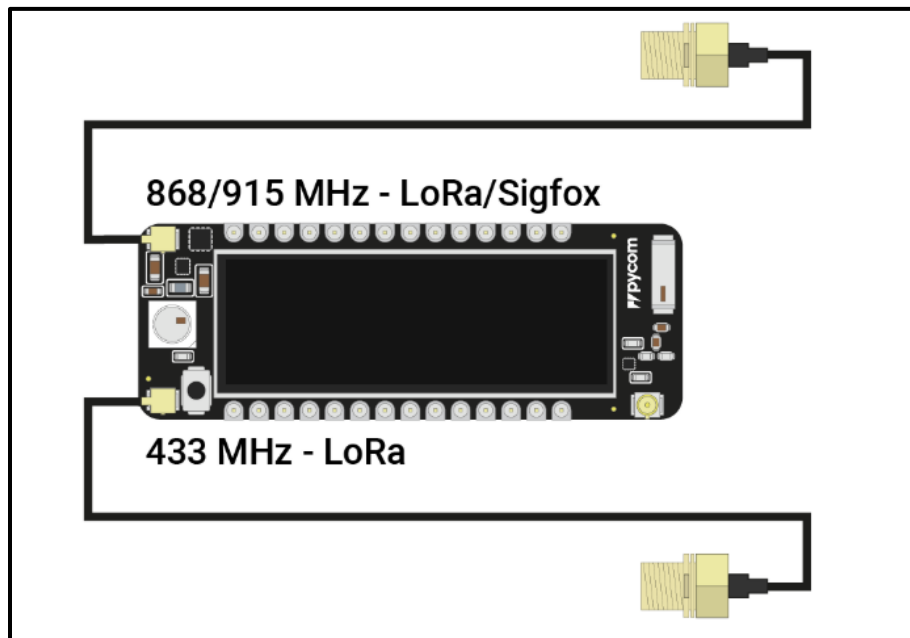


Fig.7 – Conexión de adaptador de antena

Finalmente se realiza la conexión de la antena, la conexión deberá quedar como la mostrada a continuación (Fig. 8)

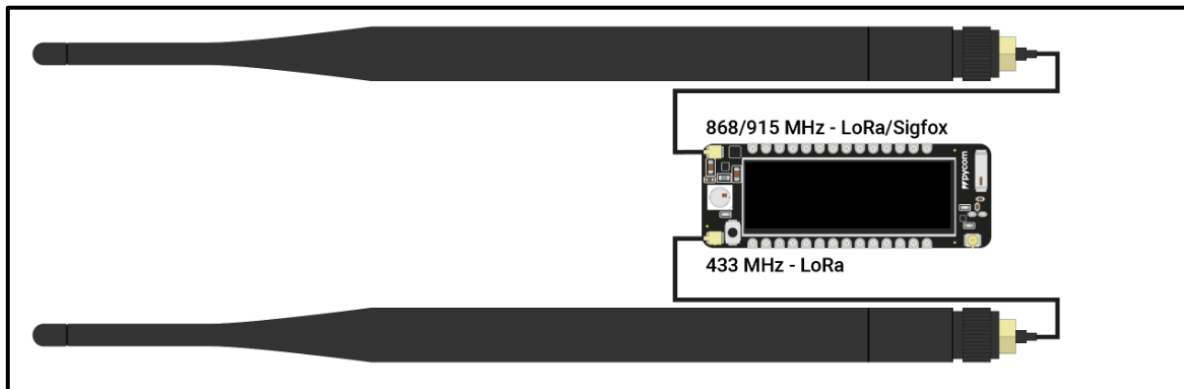


Fig.8 – Conexión completa del dispositivo.

Al igual que con el anterior dispositivo se debe tener en cuenta que al momento de usarlo se debe hacer la conexión de la antena para no dañar el módulo. Con respecto a como funciona este con las placas mencionadas en la lista se sigue teniendo en cuenta las imágenes mostradas para la conexión de Lopy.

Para las demás configuraciones que se mencionan a continuación serán iguales para ambos dispositivos.

Configuración de software

En esta sección se miran tres factores diferentes los cuales son: drivers, firmware y pymakr. Lo primero que se debe definir es que tipo de sistema operativo se va a utilizar, haciendo la correspondiente búsqueda y análisis la mejor elección de SO que se puede usar es Linux ya que este no requiere de muchos pasos para hacer uso del dispositivo.

A partir de este punto todo lo referente a drivers, firmware y pymakr se hará teniendo en cuenta que el SO implementado va a ser hecho en una consola de Linux ya sea en maquina virtual (como es el caso actualmente) o teniendo dicho SO como sistema base.

El usuario esta en la libre potestad de usar la maquina virtual de preferencia llámese virtual box, vmware, entre otros. Lo recomendable es usar una versión de Linux que cuente con soporte ya que esto puede influenciar a la hora de hacer la configuración del pymakr.

La instalación de los drivers en SO de Linux (Ubuntu) solo se requiere cambiar un parámetro relacionado a la activación del puerto serial, esto con el fin de poder permitir que el dispositivo pueda ser usado en ambientes Linux. La forma de poder hacer la activación del puerto serial, se debe abrir una terminal y digitar la siguiente línea de código:

```
sudo chmod 777 /dev/ttyACM0
```

Código. 1 – Activación de puerto serial

Luego de insertar este código en la terminal de Linux, se procede a la instalación de un editor de texto para ello se tendrá que elegir entre Visual Studio Code o ATOM, ya que estos dos editores tienen un plugin que nos ayudara para la programación del dispositivo.

Para este caso se utilizó Visual Studio Code, ya que es un editor de texto del cual ya se conoce la mayoría de funciones que este posee. Para la instalación del plugin se procede a abrir el editor de texto e ir al icono de extensiones tal como se muestra en la siguiente imagen (Fig.7) o bien se puede usar la combinación de teclas Ctrl + Shift + X.

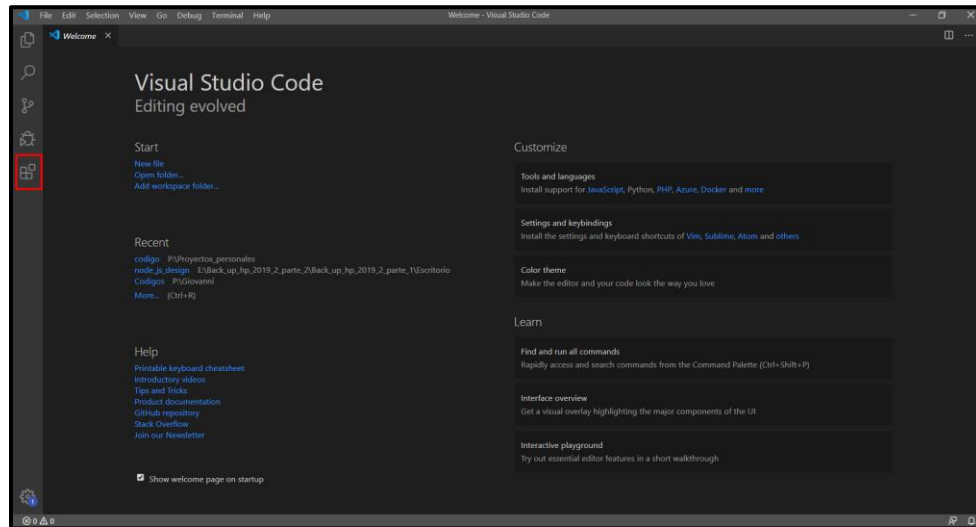


Fig. 7 – Ubicación icono de extensiones

Una vez encontrado dicho icono, se procede a instalar el plugin llamado "pymakr". Debe aparecer algo similar a lo mostrado en la siguiente imagen (Fig. 8):

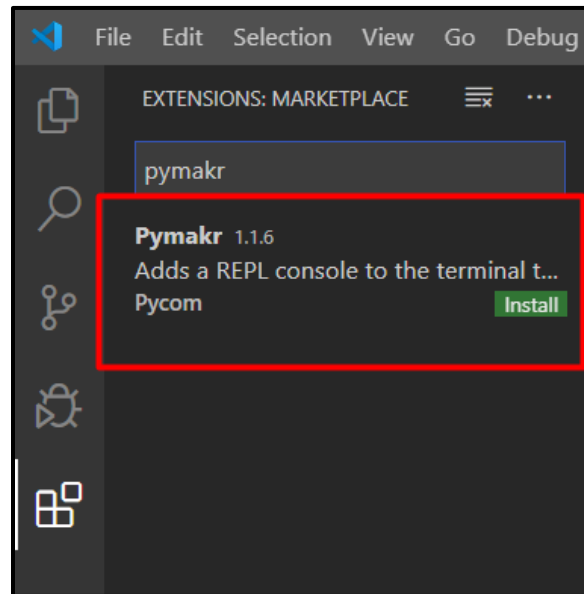


Fig.8 – Plugin pymakr

Una vez instalado este plugin nos aparecerá una interfaz como la siguiente (Fig.9) en donde podremos realizar el código y realizar las diferentes configuraciones necesarias para garantizar el funcionamiento del dispositivo.

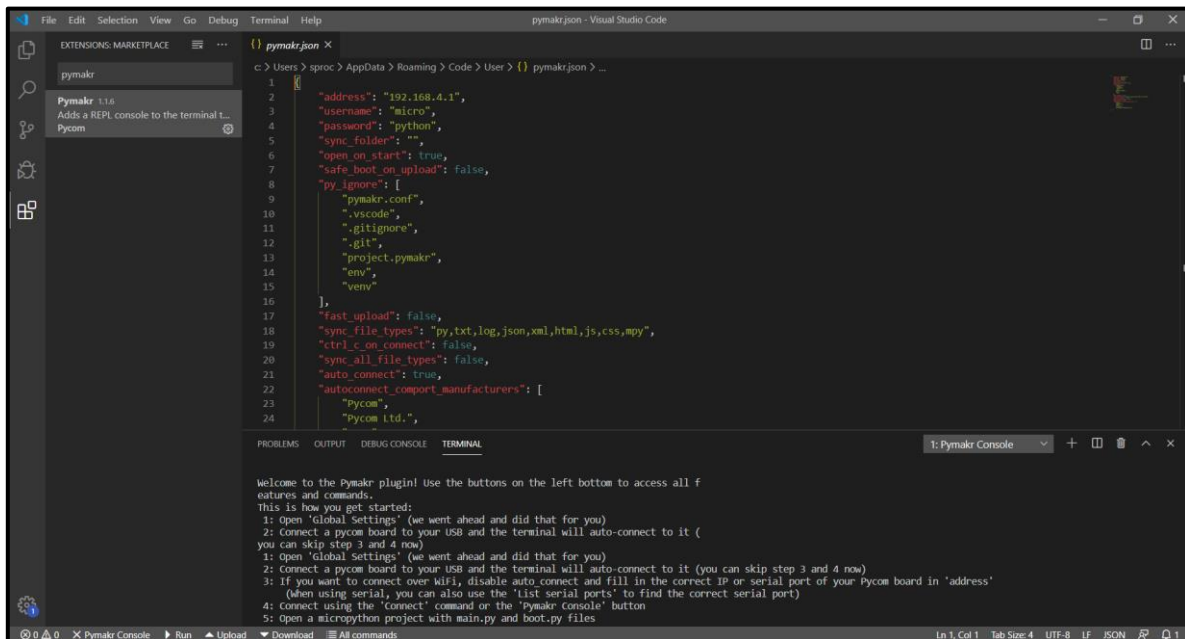


Fig. 9 – Interfaz del plugin pymakr.

Para que este plugin funcione de manera correcta se deben instalar dos programas más: el primero es Python esto para poder programar el dispositivo y el segundo es nodejs que permitirá el uso correcto de la herramienta.

La instalación de Python se resuelve con el siguiente comando:

```
sudo apt-get update  
sudo apt-get install python3
```

Código 2 – Instalación de Python 3

Una vez se tenga instalado Python se procede a la instalar node js, para ello se utilizaron las siguientes líneas de código.

```
sudo apt-get update  
sudo apt-get install curl  
curl -sL https://deb.nodesource.com/setup_12.x | sudo -E bash -  
sudo apt-get install -y nodejs
```

Código 3 – Instalación de nodejs

Para la actualización del firmware del modulo se tuvieron en cuenta los siguientes pasos. Lo que se va a explicar a continuación fue realizado desde SO de Windows ya que no se necesita instalar ningún driver realizar la actualización.

Se hizo la descarga de la herramienta desde la página oficial de pycom^f en la parte donde dice firmware updates tal como lo muestra la siguiente imagen (Fig.10).

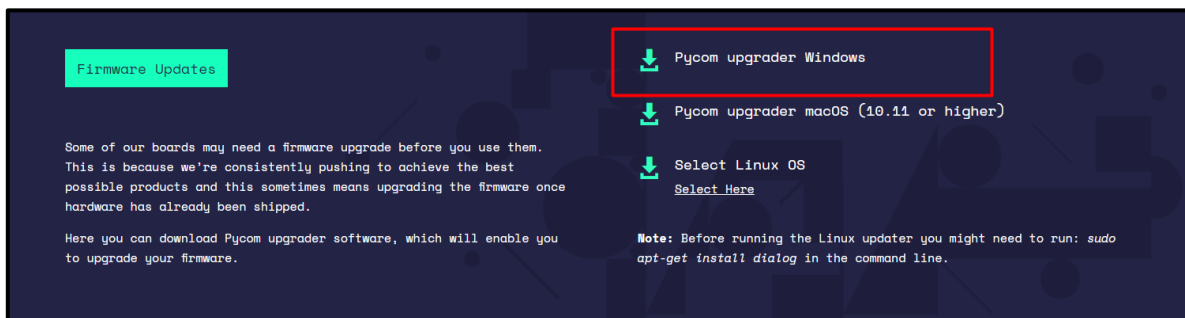


Fig. 10 – Descarga de herramienta

Aquí se elegirá el sistema operativo correspondiente donde se va hacer la actualización del módulo. Una vez se haya realizado la descarga se procede con la instalación de la herramienta y se procederá a ejecutarla. Esta abrirá un recuadro como se muestra a continuación (Fig.11)

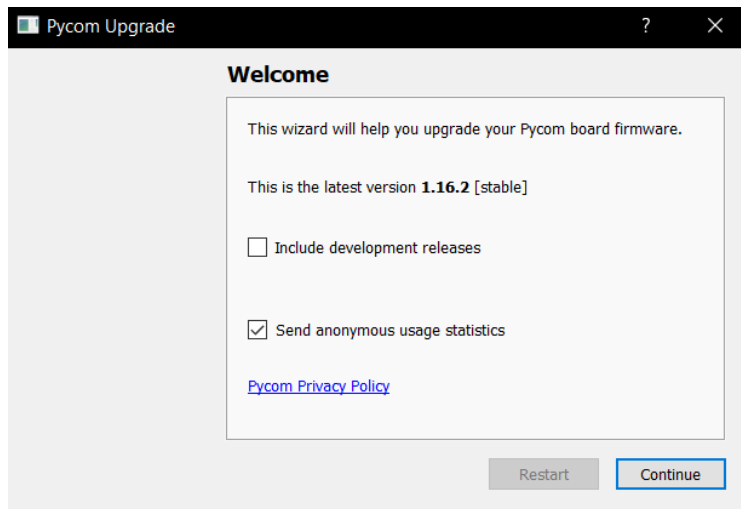


Fig. 11 – Interfaz de la herramienta

Ejecutada la herramienta, procedemos a conectar la placa (pysense/pytrack) y el modulo (lopy/lopy 4) al computador, cuando lo hayamos hecho aparecerá un recuadro como el siguiente (Fig.12), en donde automáticamente se detecta el puerto serial para este caso será el COM4 y la velocidad del serial que se dejó 115200.

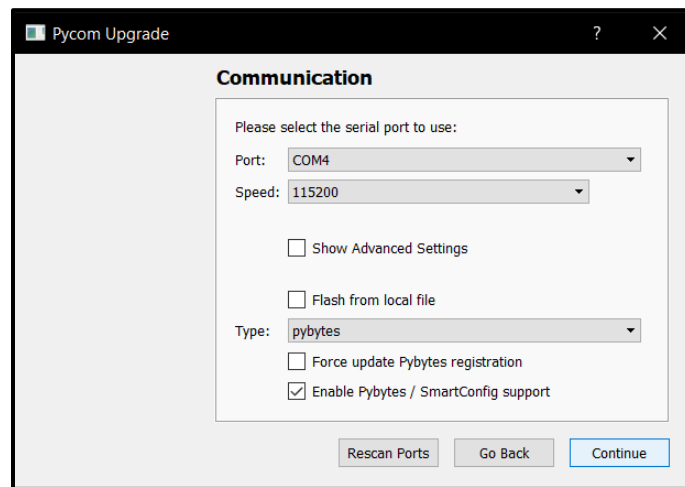


Fig.12 – Configuración del dispositivo.

Cuando se tenga todo configurado como en la imagen anterior se procederá a darle al botón de continue, aparecerá un recuadro en donde nos permitirá escoger la placa que se quiere actualizar (Fig.13), después nos pedirá ingresar el país en donde estamos y automáticamente este elegirá la frecuencia permitida para usar (Fig.14), después aparecerá un recuadro pidiendo el registro pybytes al cual se le da skip y automáticamente se empezará con la instalación del firmware, a continuación se deja en evidencia los cuadros que aparecerán al momento de hacer la instalación (Fig.15 a Fig.17)

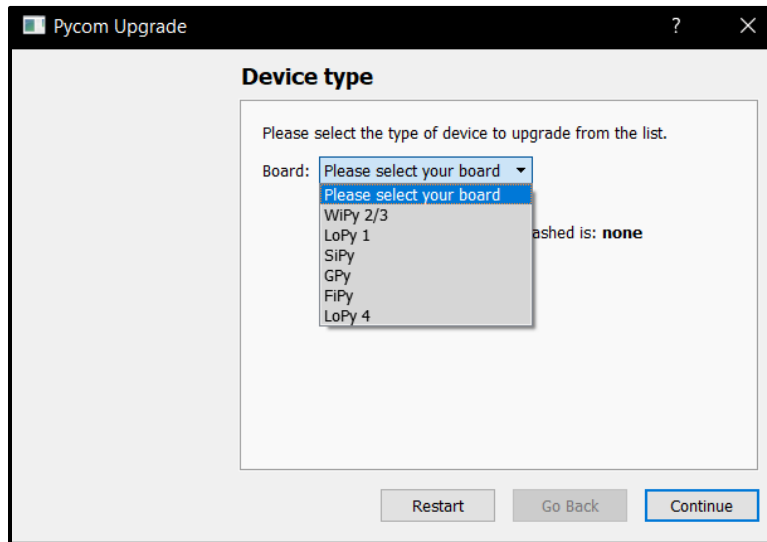


Fig.13 – Elección de moduló

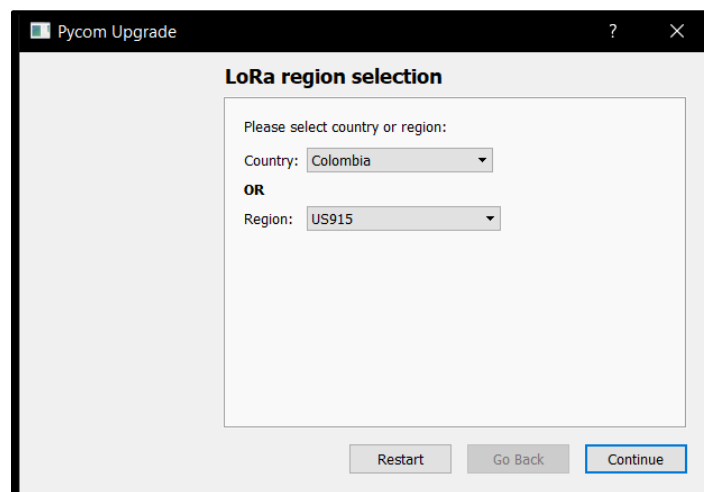


Fig.14 – Elección de país.

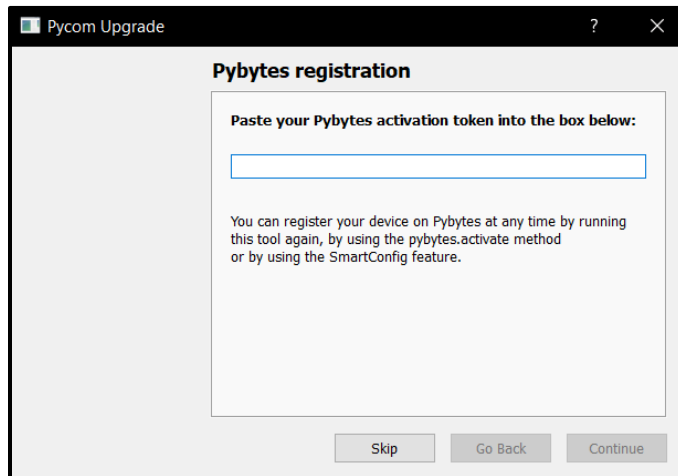


Fig.15 – Registro del modulo

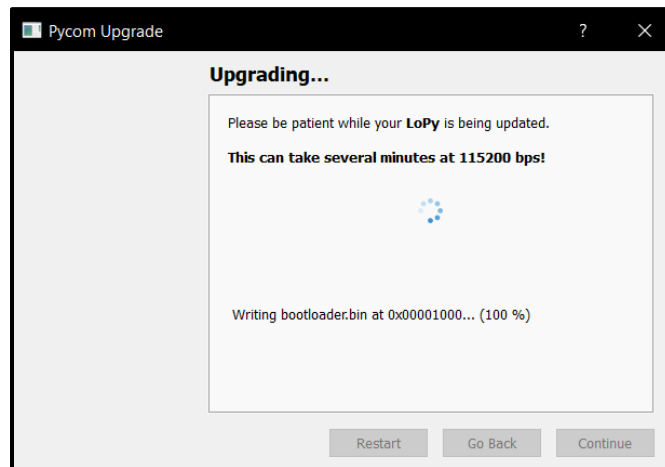


Fig.16 – Pantalla de carga

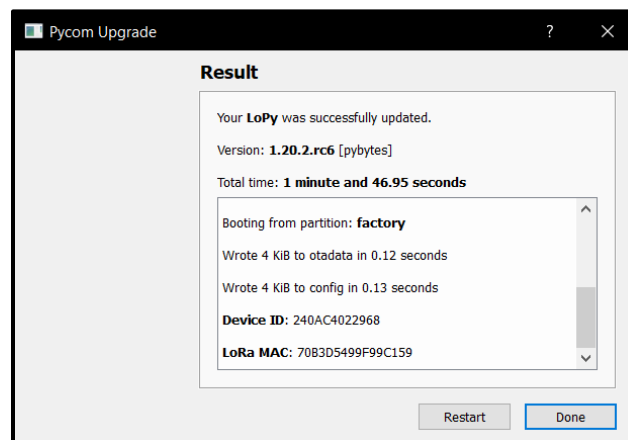


Fig. 17 – Resultado de la instalación de firmware.

Si usted precisa usar la conexión de LoRa en Lopy, este paso es de vital importancia ya que al momento de implementar el código mostrará un error en donde menciona que el dispositivo no puede usar el atributo LoRa.

Programación de los módulos

Esta sección se basa en realizar un proyecto básico haciendo el uso del lopy, para conocer como es la estructura que se van a tener en los proyectos posteriores, como se sube y quema un proyecto en lopy, entre otros.

Para ello primero procedemos a abrir el editor de texto de preferencia, para este caso se usará Visual Studio Code, cuando abra procedemos a conectar el dispositivo, una vez lo lea iremos a la parte inferior de la pantalla donde encontraremos los siguientes iconos (Fig.18)



Fig.18 – Iconos pymakr

Aquí procedemos ir a la opción de “All commands” y buscar la opción “Connect” para conectar el dispositivo, si en la consola no aparece el siguiente símbolo “>>>” se procederá abriendo una terminal e ingresando el comando del apartado código 1 para activar el puerto serial y volver a repetir el procedimiento.

Una vez se tenga lista la conexión procedemos a crear en nuestro tablero la siguiente estructura una carpeta que tendrá el nombre de lib, en donde dejaremos alojado la mayor cantidad de código (funciones) con el fin de reciclar código que ya tengamos funcional con el dispositivo, los archivos que se creen dentro de esta carpeta deben llevar la extensión “.py” ya que será el lenguaje Python el que utilizaremos para los dispositivos.

Finalmente, por fuera de la carpeta se creará un nuevo archivo llamado “main.py” este nombre es obligatorio ya que cuando se vaya a subir el programa al dispositivo, este buscara este nombre para ejecutar la función que se el haya colocado. La estructura principal deberá ser similar la mostrada en la siguiente imagen (Fig.19), lo cual es bastante similar a lo mostrado en la documentación del dispositivo (Fig.20).

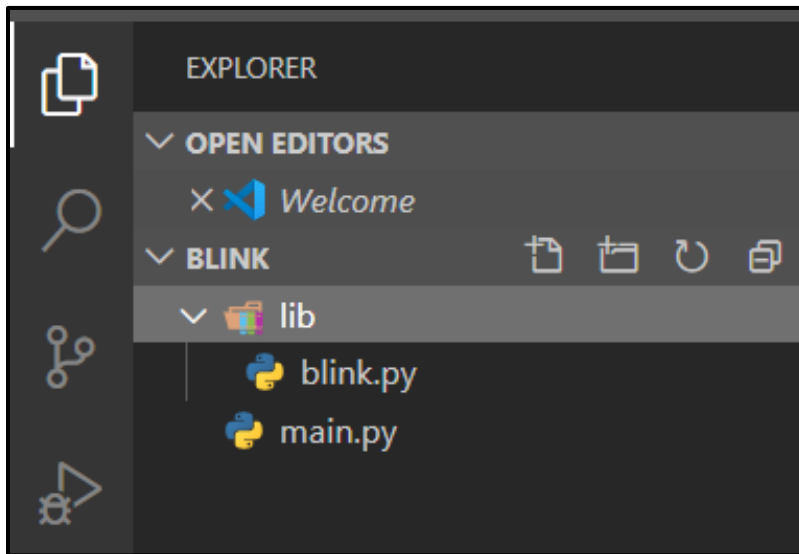


Fig.19 – Estructura realizada por el usuario

```

RGB-Blink
|-lib
|  |- some_library.py
|-boot.py
|-main.py

```

Fig.20 – Estructura de la documentación. Tomado de []

Llamaremos el archivo donde se deja alojado la función de blink como “blink” y en este colocaremos el siguiente código (Codigo.4), el cual ira imitando la funcionalidad de un semáforo.

```

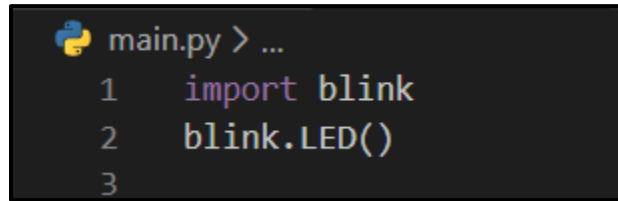
import pycom
import time

def LED():
    pycom.heartbeat(False)
    while True:
        pycom.rgbled(0xFF0000)
        time.sleep(1)
        pycom.rgbled(0x7f7f00)
        time.sleep(1)
        pycom.rgbled(0x0000FF)
        time.sleep(1)

```

Codigo.4 – Blink semáforo

Con esto ya realizado, se guarda y se crea el archivo main en donde solo habrá dos líneas de código, la primera que hace la importación del archivo anterior como una librería y la segunda es la función que se quiere usar de la librería importada, dejando algo similar a la siguiente figura (Codigo.5)

A screenshot of a code editor with a dark background. The Python logo is on the left. The text 'main.py > ...' is at the top. Below it, three lines of code are shown: '1 import blink', '2 blink.LED()', and '3'.

```
main.py > ...  
1 import blink  
2 blink.LED()  
3
```

Codigo.5 – Estructura del main

Una vez realizado todo el código se procede a subirlo en la placa, para ello iremos a la parte inferior del Visual Code y seleccionamos el botón que dice “upload” esto garantiza tener una funcionalidad similar a la que se obtiene de un Arduino. Si selecciona el botón que dice “run” se ejecutara un error mencionando que no se encuentra la carpeta lib y esto sucede porque la estructura que se viene trabajando funciona solo si se carga el archivo directamente al dispositivo, de lo contrario se generará un error.

Si se siguieron los pasos como se mencionaron durante el documento, el resultado que se obtiene es la visualización de un semáforo en donde el led tendrá tres estados en diferentes tiempos.

Todos los códigos y pruebas de que estos funcionan se encontrarán en el siguiente link: https://github.com/SrAlderson/Pycom_codes/wiki.