How to use your own payloads with R4S dongle

1.- First you need to do is download and install Arduino IDE.

You can downlaod it from Arduino website:

https://www.arduino.cc/en/Main/Software

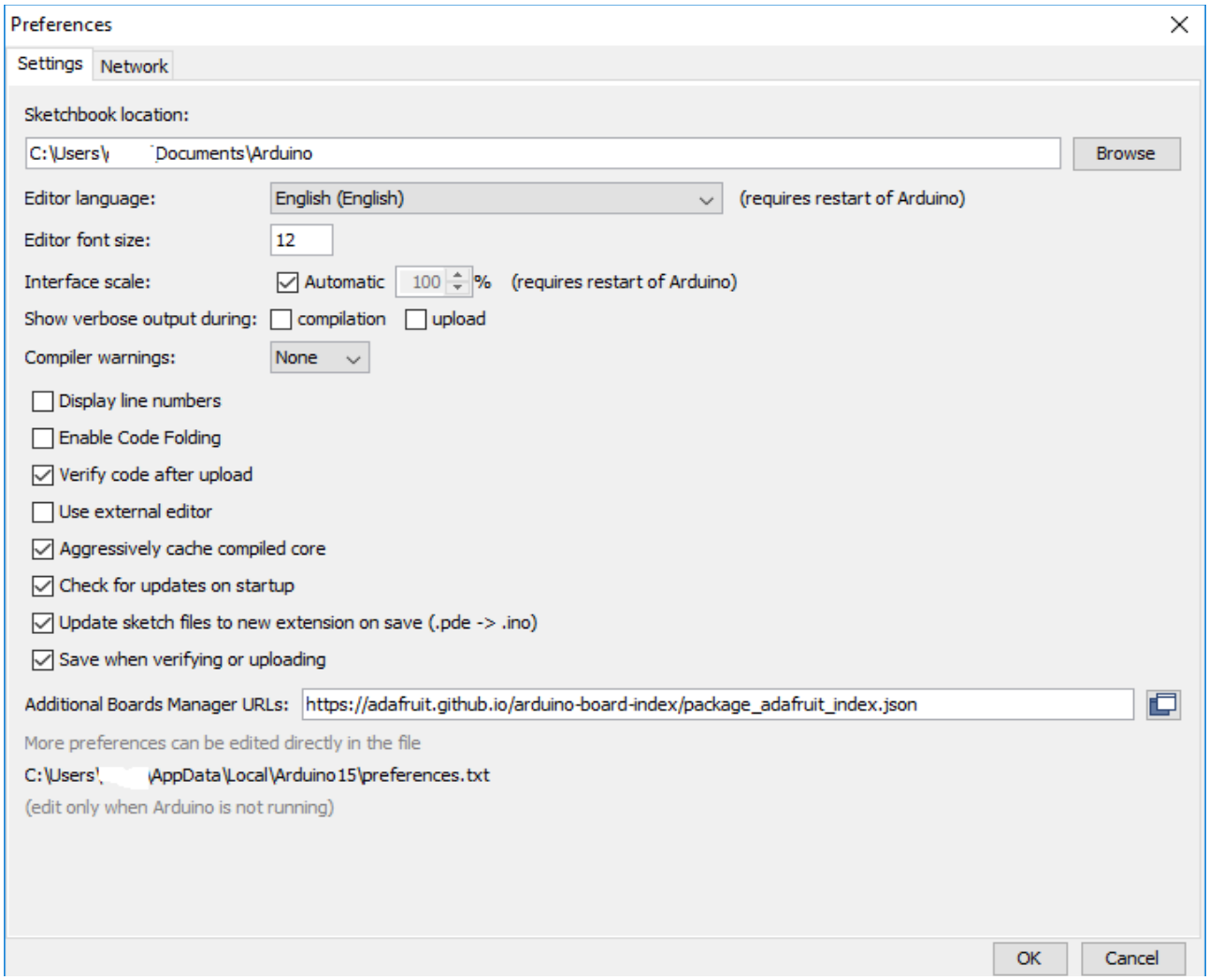
2.- Once installed, donwload this repo, unzip it and copy "fusee-launcher_for_R4S_dongle" directory in the Arduino projects directory (documentos/arduino in Windows).

3.- Install support for additional boards.

To do that, open arduino IDE and go to File->Preferences menu.

In the Preferences window you need to add the following URL in the "Additional boards manager URLs" field and click OK (if you already had some URLs there, you need to separate them with a comma).

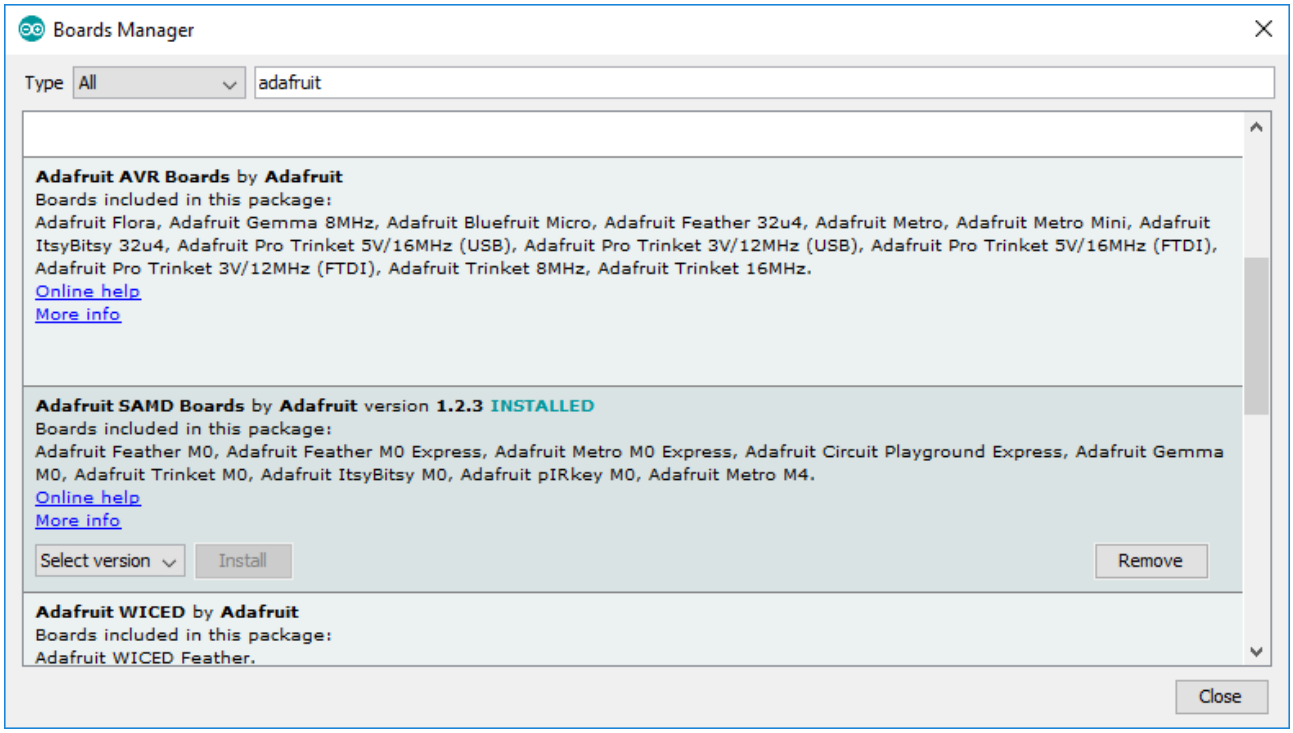
https://adafruit.github.io/arduino-board-index/package_adafruit_index.json



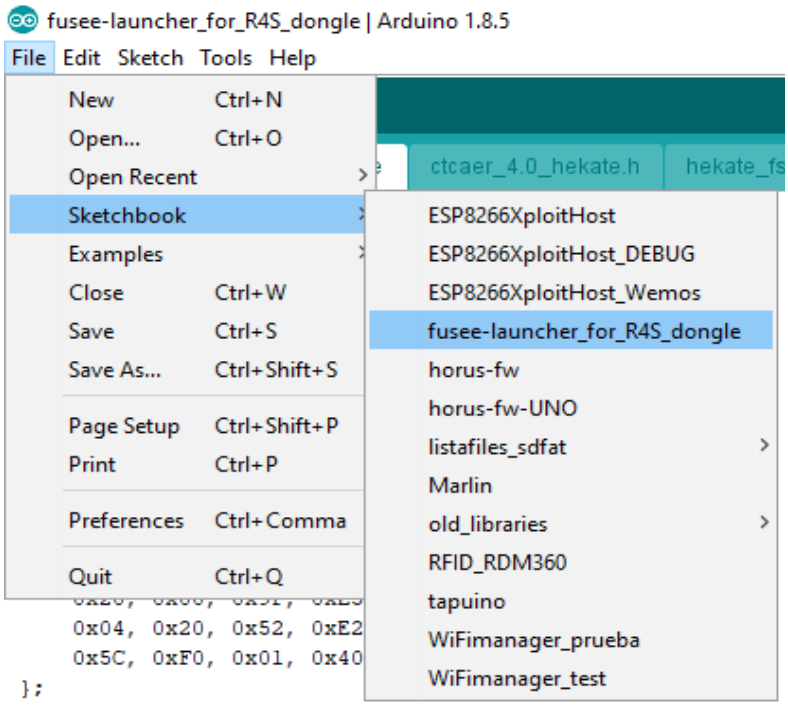
3.- Install the boards we want to work with. In this case Adafruit Trinket M0.

To do so, use the Boards manager option menu (Tools -> Board -> Boards manager)

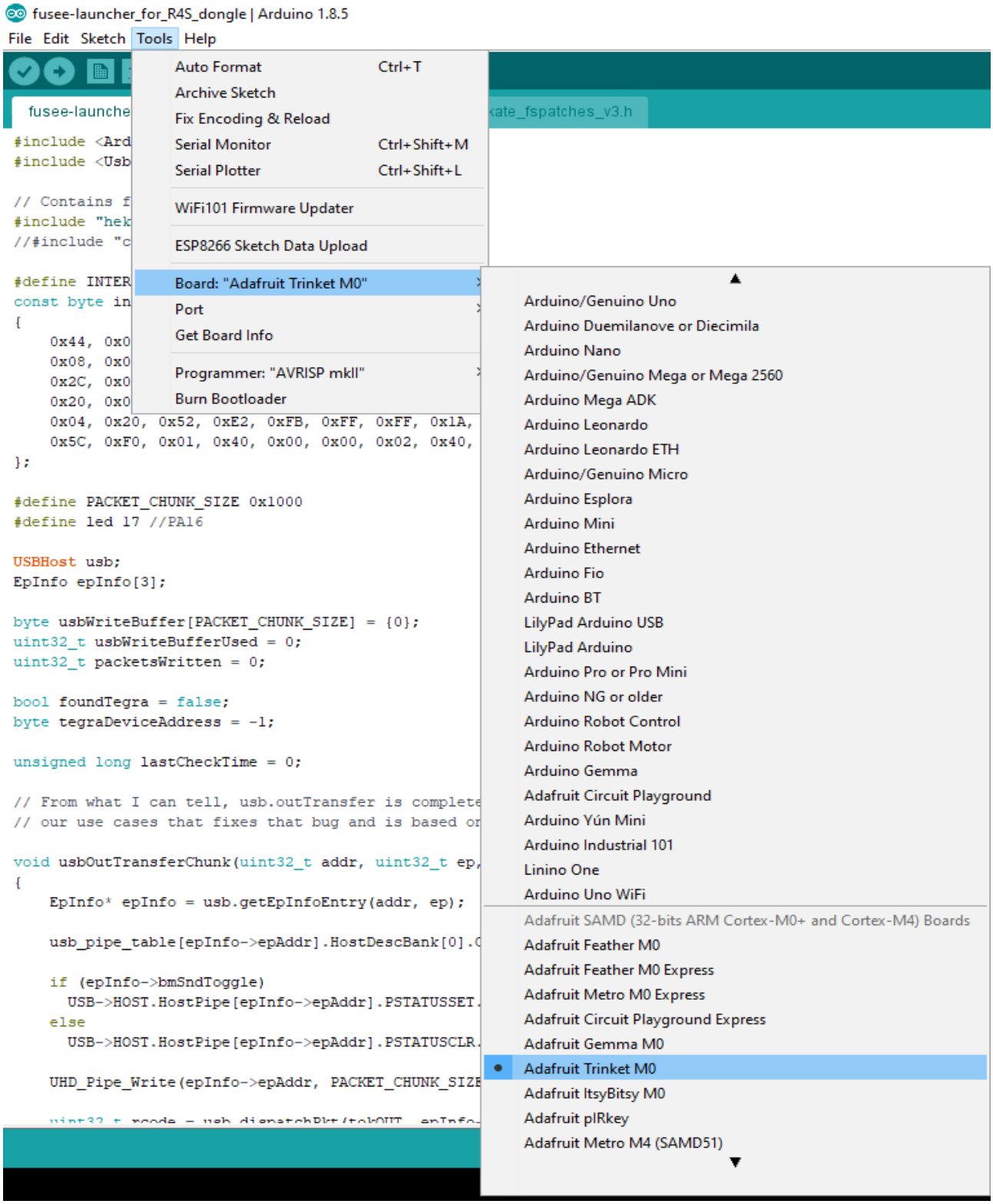
When it's done downloading data, type "adafruit" in the search field, and select "Adafruit SAMD boards" from the search results shown. Select last version (1.2.3 when writing this manual), and click Install.



4.- Now you can open the project we copied earlier from the File menu (File->Sketchbook)



Before you can compile the sketch, you need to select the board you want to compile it for. You can do it from the Tools->Board menu. Select Adafruit Trinket M0.



You will find 2 files in the project: a .ino file which contains fusee launcher code, and a .h which contains the payload we want to send. The .h file is the equivalent to the .bin file you can download from devs repos (hekate, reinx, sxos, etc...). This is the payload being sent once in RCM mode. That .bin file needs to be converted to .h so you can include it in the sketch and compile it.

To convert it you need to use the "binconverter.py" Python app included in the tools folder in my repo. I'm running it in Ubuntu with Python 2.7, i don't think it works with Python 3.x

To convert the .bin file to .h just run this:

```
python binconverter.py payload_file.bin
```

This will create a payload_file.h file that you need to copy to our sketch the directory created earlier, and then modify the following line in the sketch code to match your .h file name:

```
#include "ctcaer_4.0_hekate.h"
```

If you want the IDE to show the current files included in the sketch directory, just close it and open the sketch again. If you just want to compile it, this is not necessary (if you wrote the .h file name correctly, it will find it).

Once you edited that line to include your own .h payload file, you can compile it to get a binary file we need to convert again to UF2 format so we can copy t to the dongle.

To do this, just click the Sketch -> Export compiled binary menu

This will create a .bin file (fusee-launcher_for_R4S_dongle.ino.trinket_m0.bin) in the project directory.

You need to convert this file to UF2 now using the uf2conv.py Python app included in the Tools directory of this repo.

Just run this (again, i've run this on Python 2.7, probably won't work on Python 3.x):

```
python uf2conv.py -c -f SAMD21 -o CURRENT.UF2 fusee-launcher_for_R4S_dongle.ino.trinket_m0.bin
```

This will create a CURRENT.UF2 file you can now copy to the dongle.