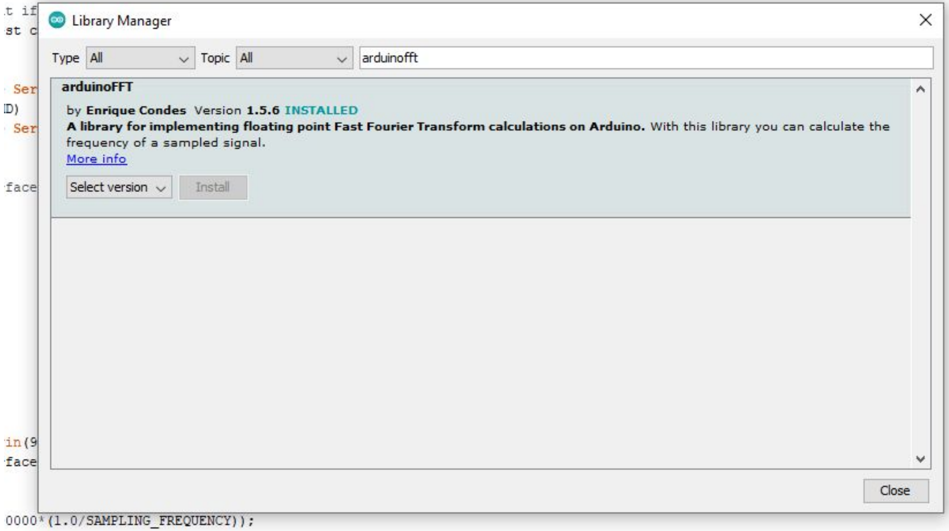
Install ArduinoFFT Library

- In the toolbar for the Arduino IDE, click **Tools->Manage Libraries…**
- Then search “**ArduinoFFT**”
- Select “**ArduinoFFT**” and click Install



Library Manager

Type All Topic All arduinofft

arduinoFFT

by **Enrique Condes** Version **1.5.6** INSTALLED

A library for implementing floating point Fast Fourier Transform calculations on Arduino. With this library you can calculate the frequency of a sampled signal.

More info

```
0000*(1.0/SAMPLING_FREQUENCY));
```

Code Initialization

- We'll be working from the WakeWordExample code on GitHub in this folder: https://github.com/kellykim5/PATHSUP_2022/tree/main/WakeWordExample
- Copy the code from the Arduino sketch here: https://github.com/kellykim5/PATHSUP_2022/blob/main/WakeWordExample/WakeWordExample.ino
- Paste the code in a new Arduino sketch in your Arduino IDE

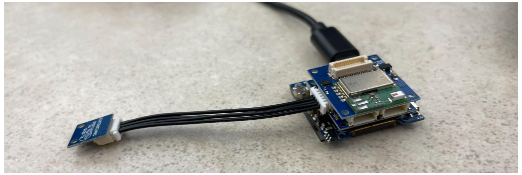
Setting Sketch To Data Collection Mode

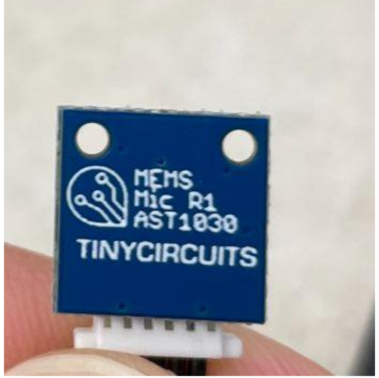
- Comment out three parts so we can collect our training data
    - We will uncomment these during classification after we have trained a machine learning model using the training data that we collect

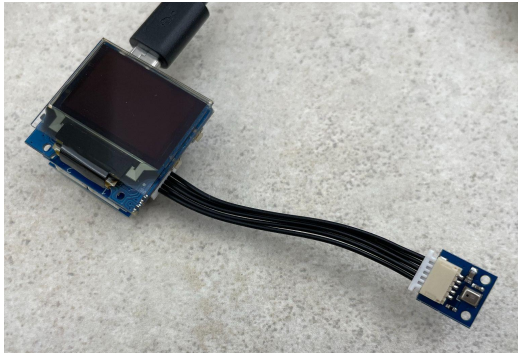
```
1:   //#include "model.h"

2:   //Eloquent::ML::Port::RandomForest classifier;

3:  /*     SerialMonitorInterface.print("You said ");


       for (uint16_t i = 0; i < NUM_SAMPLES; i++) {

           ffeatures[i] = float(features[i]);

    }


    SerialMonitorInterface.println(classifier.predictLabel(ffeatures));

   */
```

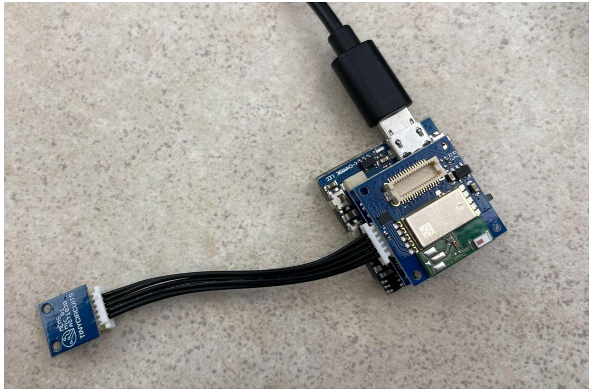
Connecting The Microphone

- Find the microphone wireling (it will have “Mic R1” on the back) and connect it to port 0 of the Wireling adapter board using a 5-pin black wire
  - The wireling adapter board should be attached to your Tinyscreen+









Collecting Training Data

- Open the serial monitor and run the script on the tiny watch
    - The serial monitor may close, make sure to open it again
- Choose two words that sound very different, such as hello and bye
- Try saying the first word you've chosen near the microphone and make sure that you are seeing output in the serial monitor
- Continue to say your first word until you have said it about 15 times
- Copy the output in the serial monitor by using the cursor to select the text and control+c
- Open Notepad++ and paste the text into a file; Click save, select all file types at the button of the saving window, and save it as the first_word.csv, e.g. "hello.csv" in the same folder as your Arduino script
- Do this same process for a second word to produce another file, e.g. "bye.csv"

Training Our Model

- Go to Google Colab - https://colab.research.google.com
- Click New notebook
- Copy the code from train.py (link) and paste it into the notebook
- Save (control+s)
- Click the folder icon on the left and click the upload icon
    - Use this to upload your two csv files
- Right click in that area to make a new folder called “dataset”, and move the two csv files into that folder
- Click the play button to run the code
- Click the Refresh icon - there should be a new file called “model.h” that has your ML model that was trained on the data you collected - right click that file and download it to the folder where your Arduino sketch is

Finally: Testing Our Model! Trying Speech Recognition!

Remove the comment (highlighted below) indicators that we placed before:

```
1:   //#include "model.h"

2:   //Eloquent::ML::Port::RandomForest classifier;

3:  /*    SerialMonitorInterface.print("You said ");


       for (uint16_t i = 0; i < NUM_SAMPLES; i++) {

           ffeatures[i] = float(features[i]);

     }


     SerialMonitorInterface.println(classifier.predictLabel(ffeatures));

    */
```

Finally: Testing Our Model! Trying Speech Recognition!

- Run the new sketch on your tiny watch
- Open the serial monitor (if not already open), and try saying your two words to see if the model is correctly identifying the word you are saying!
  - You may want to try it really loudly