

```

1  # Demo FFT in three axes
2  # Author: Tony DiCola library Adafruit_ADXL345_Python
3  # License: Public Domain
4  import paho.mqtt.publish as publish
5  import time
6  import datetime
7  import RPi.GPIO as GPIO
8  import smbus
9  import numpy as np
10 from scipy import signal
11 import peakutils
12 ##from scipy.signal import find_peaks
13
14
15 # Import the ADXL345 module.
16 import Adafruit_ADXL345
17 GPIO.setwarnings(False)
18 GPIO.setmode(GPIO.BOARD)
19 ### Create an ADXL345 instance.
20 accel = Adafruit_ADXL345.ADXL345()
21
22 print('Waiting for the results of the FFT, press Ctrl-C to quit...')
23 while True:
24     # frecuencia de muestreo 15 Hz y 128 de tamaño de muestreo
25     fs=15
26     try:
27         i=0
28     #Matriz a rellenar con las muestras
29     msr=np.zeros((128,3))
30
31     while i<= 127:
32     #Lectura acelerómetro
33         msr[i]=accel.read()
34         x,y,z=accel.read()
35     #Se imprime por pantalla el valor capturado (rango)
36         print('{0};{1};{2};'.format(x, y, z))
37         i += 1
38         time.sleep(0.0666666666666666)
39
40     if i ==128:
41     #Paso a unidades g en función del rango y la resolución
42         msr=msr/256
43     #Cálculo de la FFT en cada uno de los ejes
44         msrfftx= np.fft.fft(msr,n=128,axis=0)
45         msrgraf=np.absolute(msrfftx/128)
46     #Detección de los picos
47         peaksx=peakutils.indexes(msrgraf[0:(128/2),0], thres=0.02, min_dist=0)
48         peaksy=peakutils.indexes(msrgraf[0:(128/2),1], thres=0.02, min_dist=0)
49         peaksz=peakutils.indexes(msrgraf[0:(128/2),2], thres=0.02, min_dist=0)
50
51
52     #Se procede a cambiar los arrays y matrices a cadenas para poder ser publicados en
    mensajes
53
54
55     peaksx1 = ';'.join(str(e) for e in peaksx)
56     peaksy1 = ';'.join(str(e) for e in peaksy)
57     peaksz1 = ';'.join(str(e) for e in peaksz)
58
59     msrfftx1 = ';'.join(str(p) for p in msrfftx[:,0][peaksx])
60     msrffty1 = ';'.join(str(p) for p in msrfftx[:,1][peaksy])
61     msrfftz1 = ';'.join(str(p) for p in msrfftx[:,2][peaksz])
62
63     #Publicación de los mensajes por ejes siguiendo los topics. Cada 10 s aproximadamente
64
65     publish.single("ADXL345/Tiempo", "Last valid input: " +
66 str(datetime.datetime.now()), hostname="test.mosquitto.org")
67     publish.single("ADXL345/FFT/X/Freq",peaksx1,
68 hostname="test.mosquitto.org")
69     publish.single("ADXL345/FFT/X/Cero Component",abs(msrfftx[0,0]),
70 hostname="test.mosquitto.org")
71     publish.single("ADXL345/FFT/X/Peaks",msrfftx1,

```

```
        hostname="test.mosquitto.org")
69
70    publish.single("ADXL345/FFT/Y/Freq",peaksy1,
        hostname="test.mosquitto.org")
71    publish.single("ADXL345/FFT/Y/Cero Component",abs(msrfftx[0,1]),
        hostname="test.mosquitto.org")
72    publish.single("ADXL345/FFT/Y/Peaks",msrffty1,
        hostname="test.mosquitto.org")
73
74    publish.single("ADXL345/FFT/Z/Freq",peaksz1,
        hostname="test.mosquitto.org")
75    publish.single("ADXL345/FFT/Z/Cero Component",abs(msrfftx[0,2]),
        hostname="test.mosquitto.org")
76    publish.single("ADXL345/FFT/Z/Peaks",msrfftz1,
        hostname="test.mosquitto.org")
77
78    print("Done")
79 except KeyboardInterrupt:
80     GPIO.cleanup()
81 GPIO.cleanup()
82
83
```