HW 4 – 313288763 Rom Hirsch

1.

1.)

a. $x[n] = 10\sin(0.25\pi n)u[n]$

$$X(z) = \frac{10z\sin(0.25\pi)}{z^2 - 2z\cos(0.25\pi) + 1} = \frac{7.071z}{z^2 - 1.414z + 1}$$

b. $x[n] = e^{-0.1n}\cos(0.25\pi n)u[n]$

$$X[z] = \frac{z^2 - e^{-0.1}z\cdot\cos(0.25\pi)}{z^2 - 2e^{-0.1}\cos(0.25\pi) + e^{-0.2}} = \frac{z(z-0.639)}{z^2 - 1.2794z + 0.81}$$

2.

2.

$$Y(z) = \frac{z^2(z+1)}{(z-1)(z^2 - z + 0.5)} = \frac{z^2(z+1)}{(z-1)(z+\frac{1}{2}+\frac{1}{2}j)(z+\frac{1}{2}-\frac{1}{2}j)}$$

$$\frac{Y(z)}{z} = \frac{A}{z-1} + \frac{B}{(z+\frac{1}{2}+\frac{1}{2}j)} + \frac{C}{(z+\frac{1}{2}-\frac{1}{2}j)}$$

$$A = (z-1)\frac{Y(z)}{z} = \frac{z(z+1)}{z^2 - z + 0.5}\Bigg|_{z=1} = 4$$

$$B = (z-0.5-j0.5)\frac{Y(z)}{z}\Bigg|_{z=0.5+j0.5} = \frac{z(z+1)}{(z-1)(z^2-z+0.5)}\Bigg|_{z=0.5+j0.5} = -0.5+j1.5$$

$$C = B^* = -0.5 - j1.5$$

$$Y(z) = \frac{4z}{z-1} + \frac{(-0.5+j1.5)z}{z-0.5-0.5} + \frac{(-0.5-j1.5)z}{z-0.5+j0.5}$$

אפשר גם ככה -

$$y[n] = 4u[n] + (-0.5+j1.5)(-0.5-j0.5)^n u[n] + u[n](-0.5-j1.5)(-0.5+j0.5)^n$$

3.

a. Calculation :

3.

a. $F(z) = \frac{z^4 + z^3 + z^2 + z + 1}{z^4} = 1 + z^{-1} + z^{-2} + z^{-3} + z^{-4}$

$F(z) = \sum_{n=0}^{4} f[n] z^{-n} = f[0] + f[1] z^{-1} + f[2] z^{-2} + f[3] z^{-3} + f[4] z^{-4}$

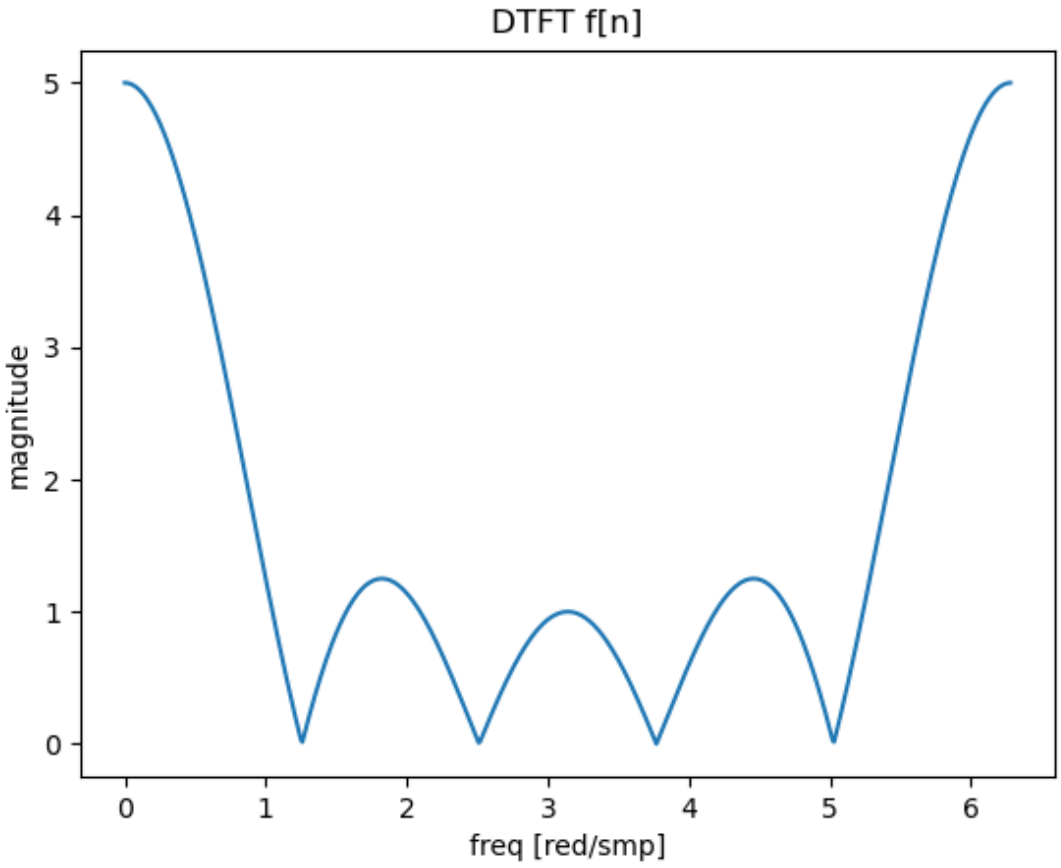
: כך נקבל

$f[n] = [1, 1, 1, 1, 1]$

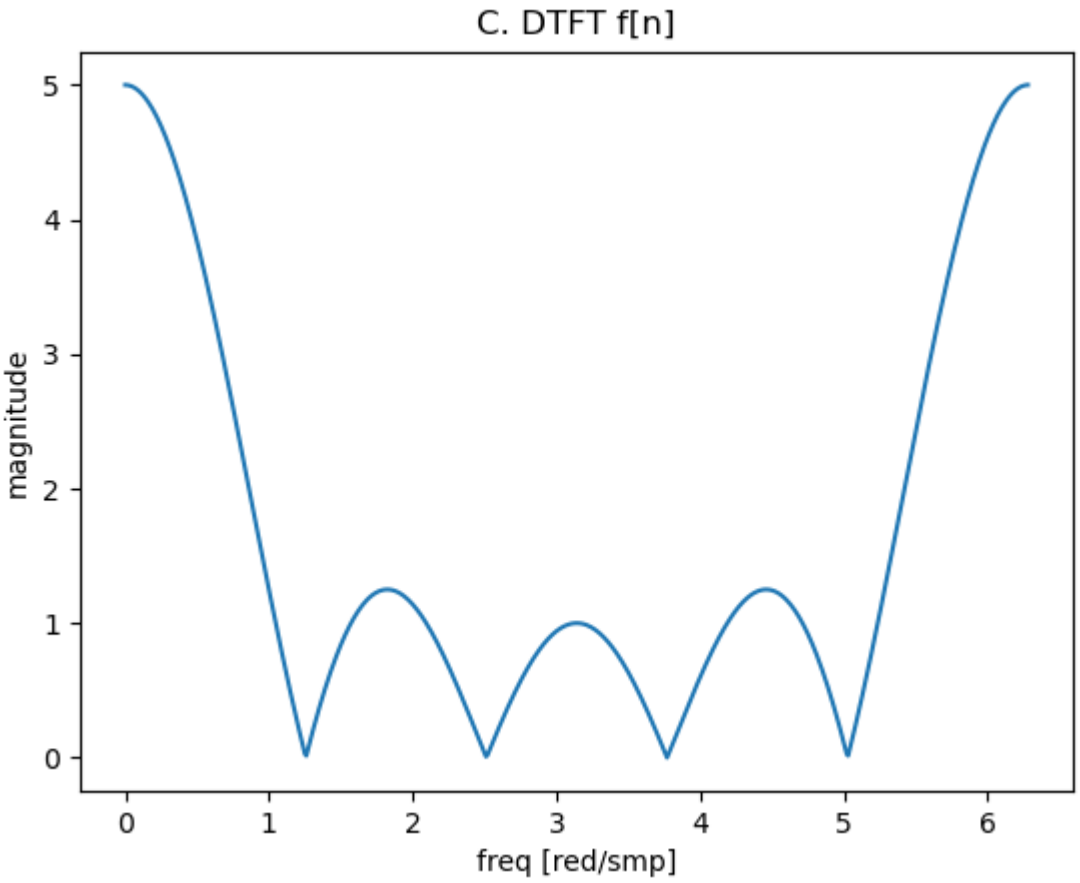
b. Plot DTFT :

לפי הנוסחה הבאה :

$$F\left(e^{j\omega}\right) = \sum_{n=0}^{4} f[n] e^{-j\omega n}$$

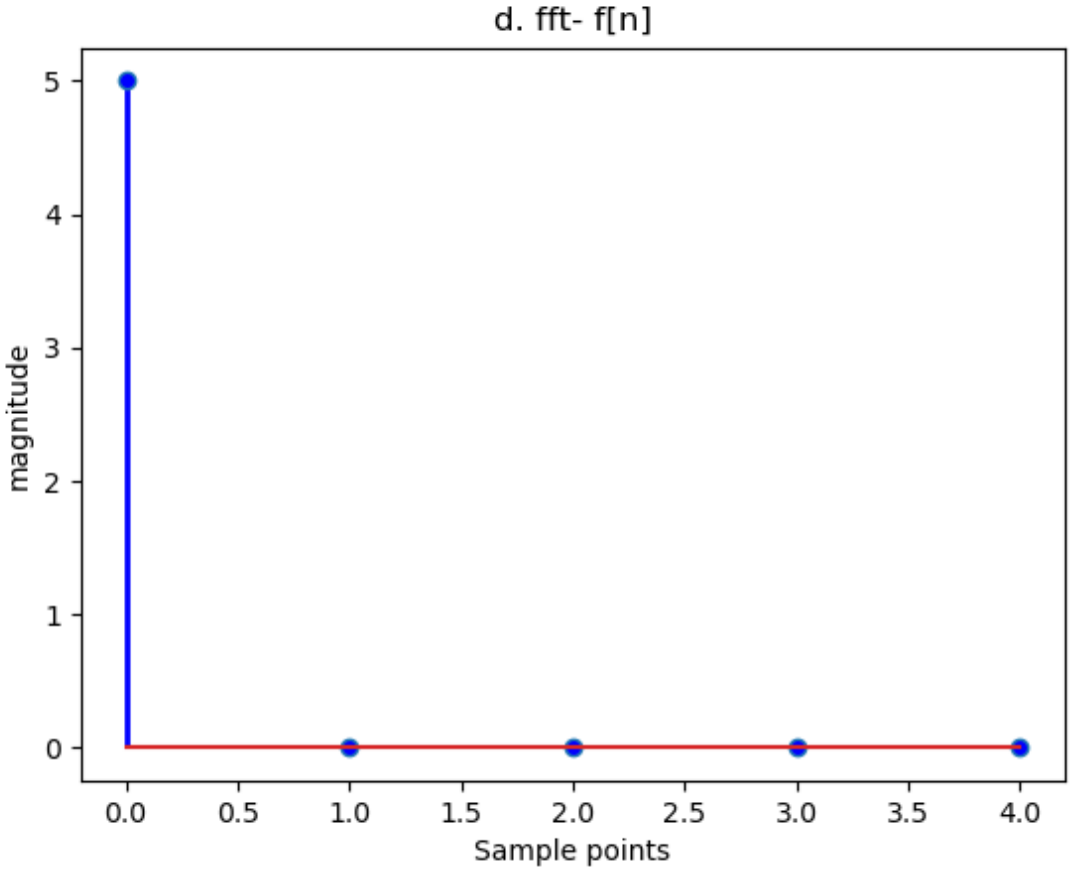


c. Plot DTFT from Fz calculation :



d. Plot FFT

DTFT זאת התמרה של DFT כאשר נבצע N דגימות שאופות לאינסוף(מבוצע על אות אינו מחזורי) לעומת התמרת DFT אשר לה מספר דגימות N סופי (התמרה זו מבצעים על אות מחזורי)

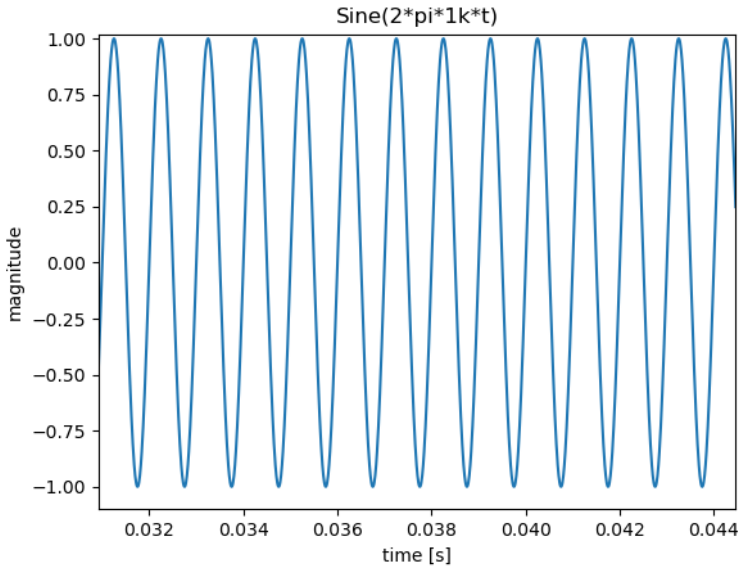


4.

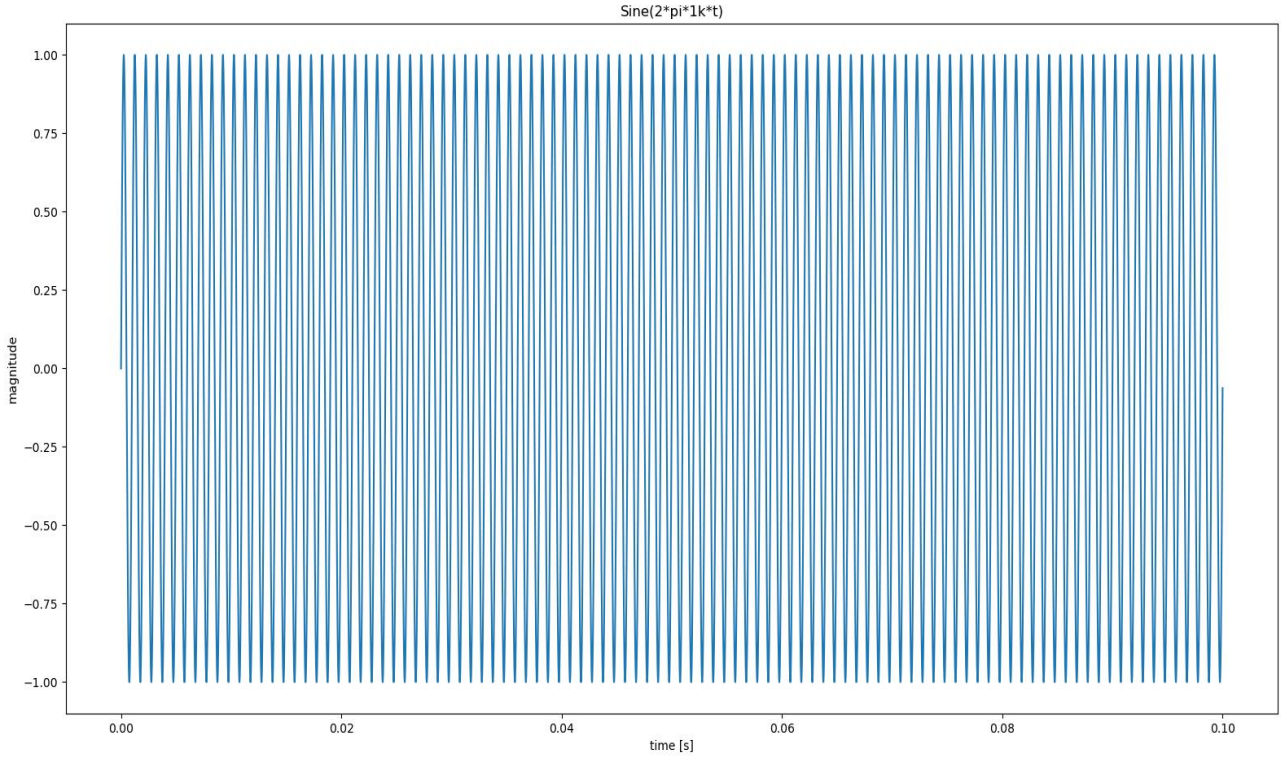
a. Create sine signal at 1khz יצירתי ב python

b. Plot sine :

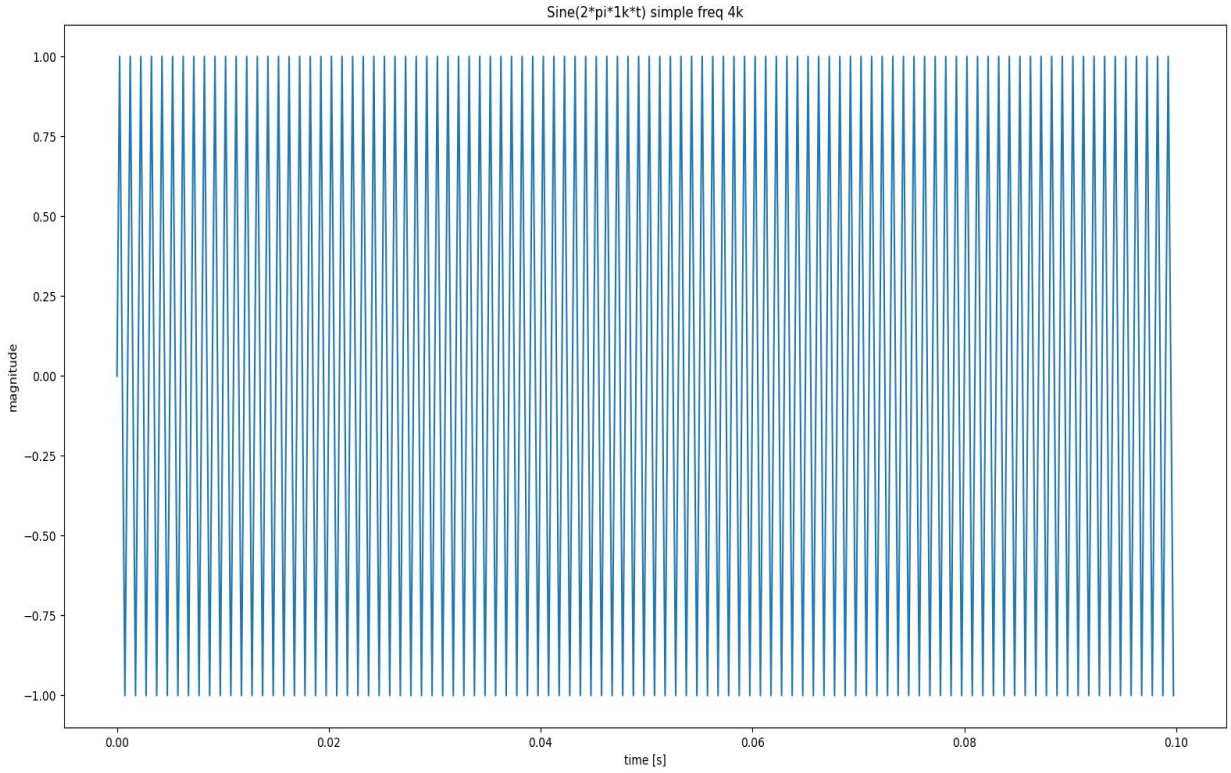
תמונה בקירוב :



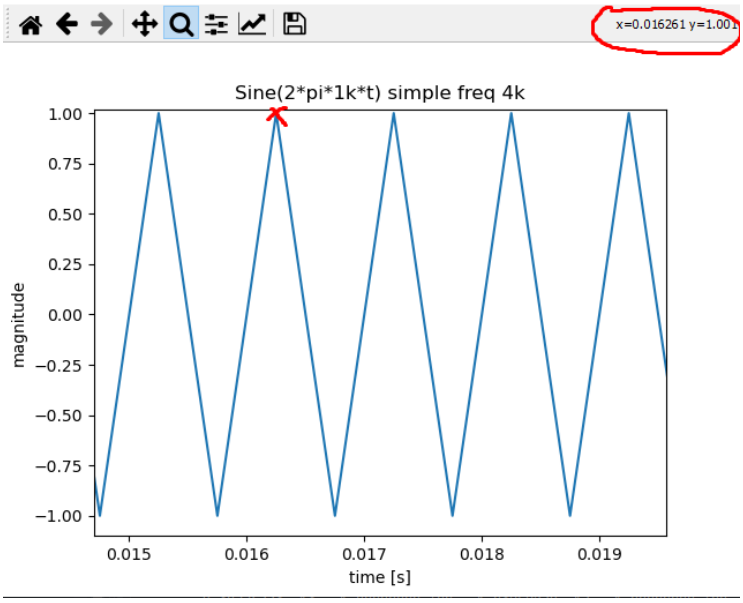
c. plot sine

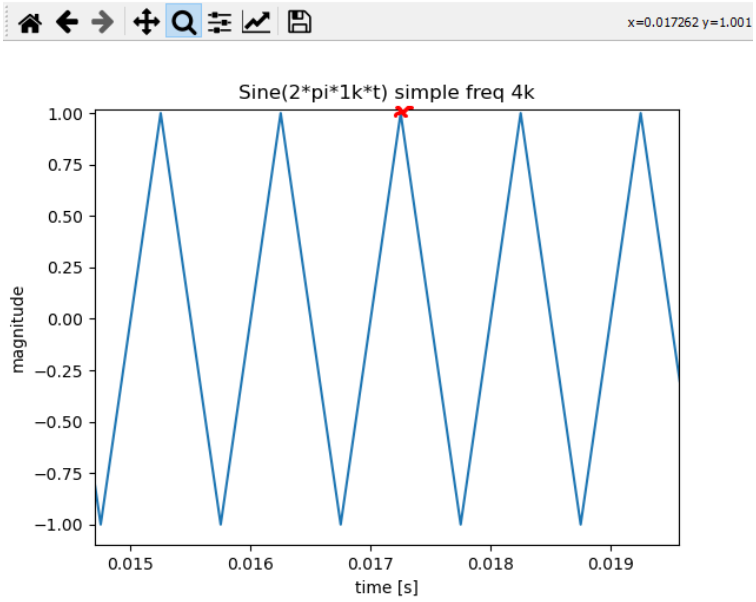


d. sampling 4khz
Plot fsemp=4khz :



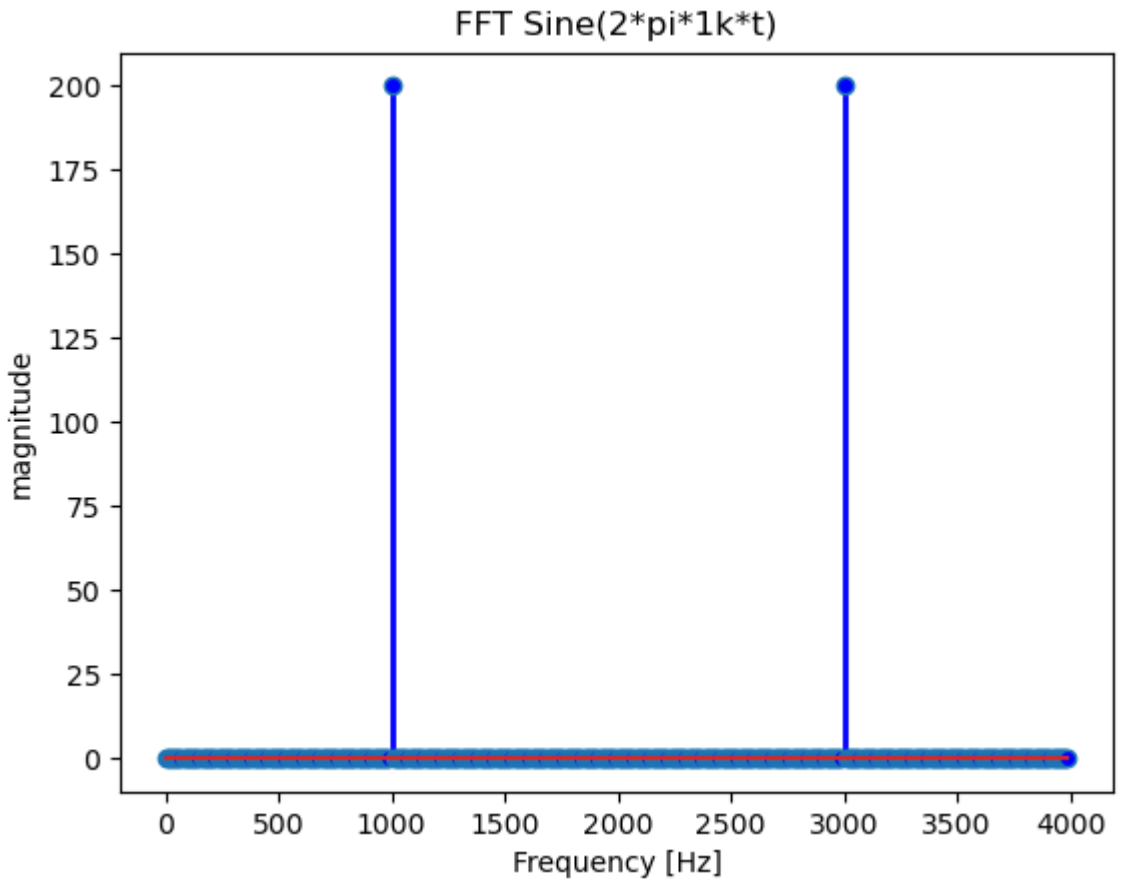
e. Measure the period time :



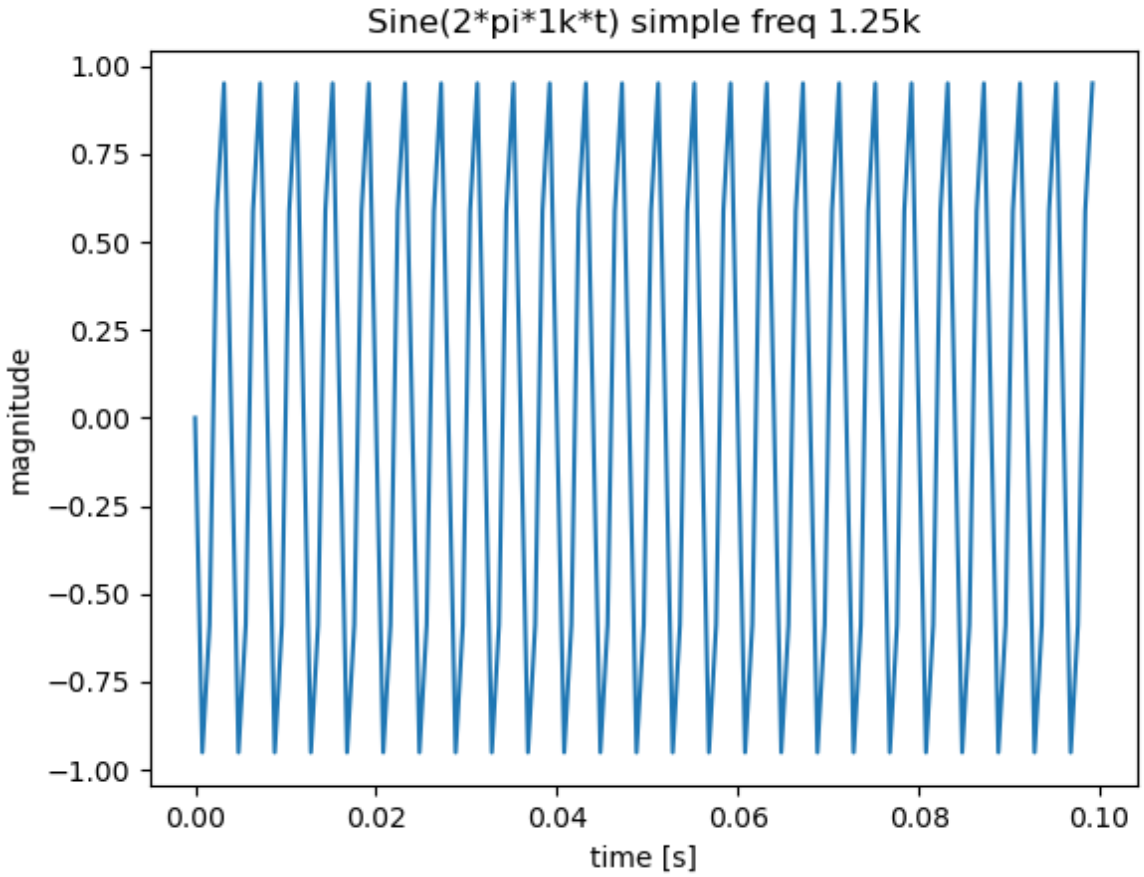


<u>קיבלנו : 1kHz תדר מהמדידה.</u>

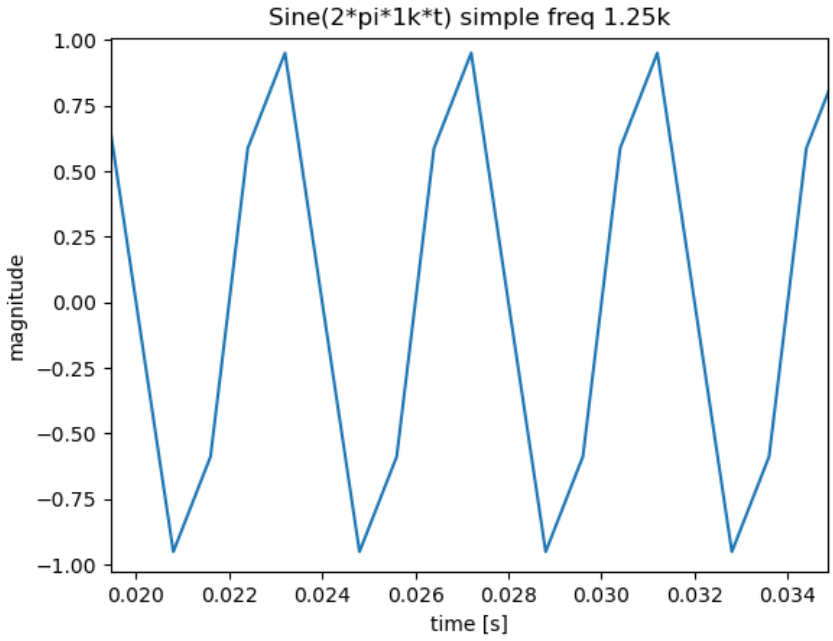
f. FFT :


FFT Sine(2*pi*1k*t)
magnitude
200
175
150
125
100
75
50
25
0
0
500
1000
1500
2000
2500
3000
3500
4000
Frequency [Hz]

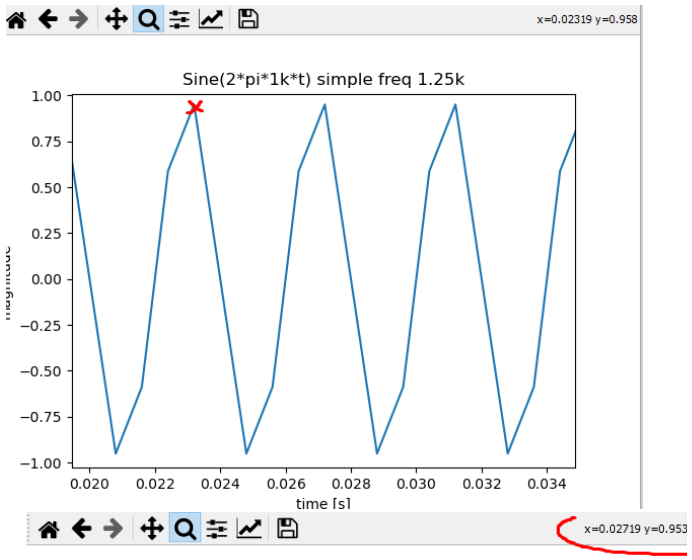
g. יצירתי את האות בקוד
h. Plot :

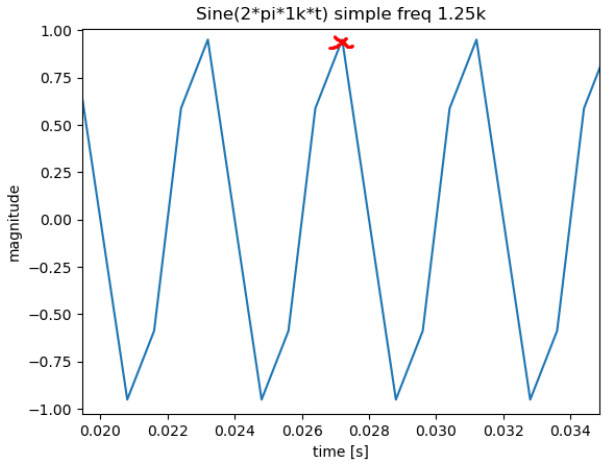


תמונת מקרוב :



i. Measure the period time :





קיבלנו :

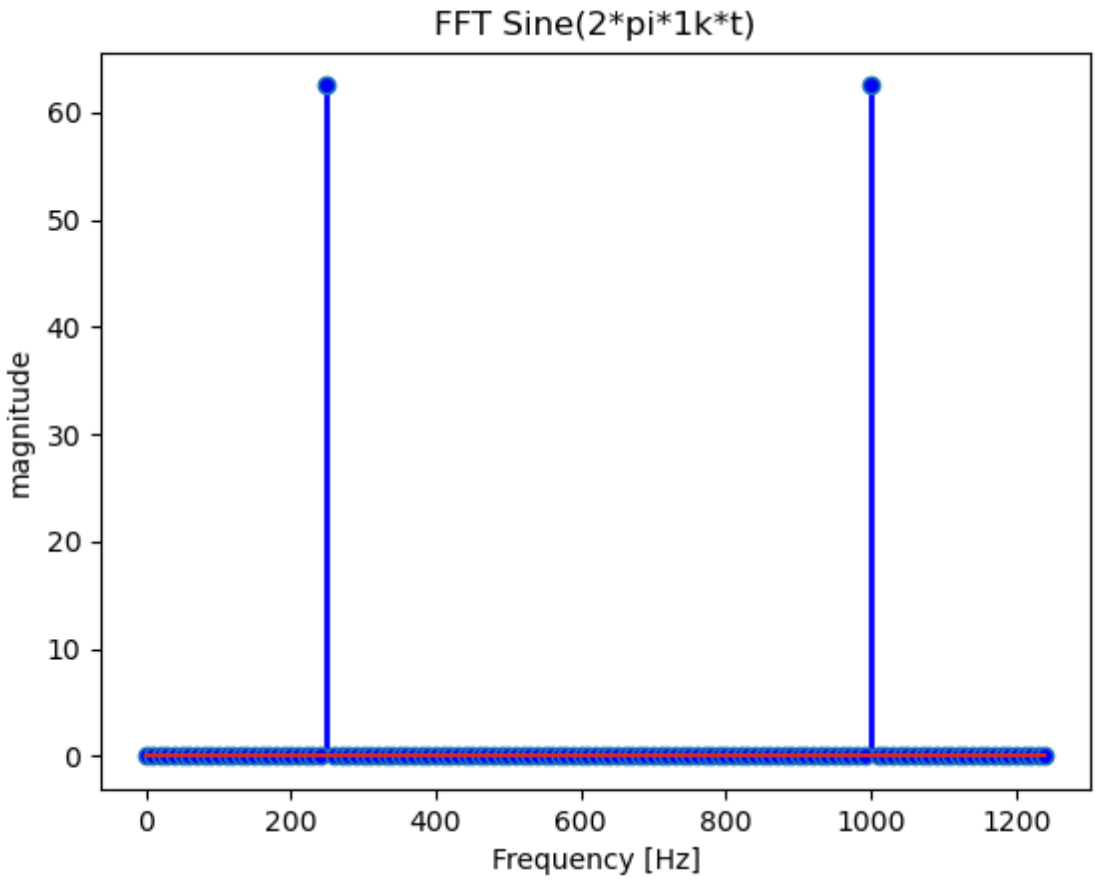
t= 0.004

נקבל :

f=250hz

קיבלנו תדר שונה מאשר התדר של אות הסינוס בגלל תדר הדגימה לכן קיבלנו התחזות.

j. FFT :


FFT Sine(2*pi*1k*t)
magnitude
Frequency [Hz]
0
10
20
30
40
50
60
0
200
400
600
800
1000
1200

הקוד ב python :

```python
# -*- coding: utf-8 -*-
"""
Created on Thu Aug 19 11:44:44 2021

@author: rom Hirsch
"""

import scipy
import matplotlib.pyplot as plt
from scipy import signal
import numpy as np
#%%Functions:

"""
    Function discrete time Fourier Transform

"""
def custom_DTFT(f):
    W = np.arange(0,np.pi*2,0.01)
    N = len(W)

    N2 = len(f)
    F=np.zeros(N,dtype=(complex))
    for i,w in enumerate(W):
        for n in range(N2):
            F[i] += f[n] * np.exp(-1j * w*n)
    return abs(F)


#plot stem with color
def stem_plot(n,val,color):
    markerline1, stemlines1, baseline1 = plt.stem(n,val)
    plt.setp(markerline1, 'markerfacecolor', color)
    plt.setp(stemlines1, linestyle="-", color=color, linewidth=2 )



#Create Stem plot with color
def plotStem(title,ylabel,xlabel,color,x,y):
    plt.figure()
    plt.title(title)
    plt.ylabel(ylabel)
    plt.xlabel(xlabel)
    stem_plot(x,y,color)
    plt.show()


#%%%
"""
Q3 - b
"""

x1 = [1,1,1,1,1]
h = custom_DTFT(x1)
W = np.arange(0,np.pi*2,0.01)
plt.figure()
plt.title("DTFT f[n]")
plt.ylabel("magnitude")
```

```
plt.xlabel("freq [red/smp]")
plt.plot(W,h)
plt.show()
#%%

"""
Q3 - c
"""
z=np.exp(-1*1j*W)

Fz=1 + z**-1 + 1*z**-2 + 1*z**-3 + 1*z**-4
plt.figure()
plt.title("C. DTFT f[n]")
plt.ylabel("magnitude")
plt.xlabel("freq [red/smp]")
plt.plot(W,abs(Fz))
plt.show()

#%%
"""
Q3 - d
"""


plt.figure()

h = np.fft.fft(x1)
n = np.arange(len(h))#get the axis x for plot
plt.figure()

plotStem("d. fft- f[n]","magnitude","Sample points",'blue',n,h)
plt.show()
#%%

"""
Q4 -b and a
"""
f = 1e3 #1KHz
stepPerCycle=1/f
step = stepPerCycle/100
t = np.arange(0,stepPerCycle*100,step)
sine1=np.sin(2*np.pi*f*t)
plt.figure()
plt.title("Sine(2*pi*1k*t)")
plt.ylabel("magnitude")
plt.xlabel("time [s]")
plt.plot(t,sine1)
plt.show()


#%%

"""
Q4 -c and d
"""
f = 1e3 #1KHz
fsemp= 4e3
t = np.arange(0,0.1,1/fsemp)
sine1=np.sin(2*np.pi*f*t)
plt.figure()
plt.title("Sine(2*pi*1k*t) simple freq 4k")
```

```
plt.ylabel("magnitude")
plt.xlabel("time [s]")
plt.plot(t,sine1)
plt.show()
#%%
"""
Q4  -f
"""
f = 1e3 #1KHz
fsemp= 4e3
t = np.arange(0,0.1,1/fsemp)
sine1=np.sin(2*np.pi*f*t)
h = np.fft.fft(sine1)
n = np.arange(len(h))#get the axis x for plot

plt.figure()
plotStem("FFT Sine(2*pi*1k*t)","magnitude","Frequency
[Hz]",'blue',10*n,abs(h))
plt.show()

#%%
"""
Q4  -h
"""
f = 1e3 #1KHz
fsemp= 1.25e3
t = np.arange(0,0.1,1/fsemp)
sine1=np.sin(2*np.pi*f*t)
plt.figure()
plt.title("Sine(2*pi*1k*t) simple freq 1.25k")
plt.ylabel("magnitude")
plt.xlabel("time [s]")
plt.plot(t,sine1)
plt.show()

#%%
"""
Q4  -j
"""
f = 1e3 #1KHz
fsemp= 1.25e3
t = np.arange(0,0.1,1/fsemp)
sine1=np.sin(2*np.pi*f*t)
h = np.fft.fft(sine1)
n = np.arange(len(h))#get the axis x for plot

plt.figure()
plotStem("FFT Sine(2*pi*1k*t)","magnitude","Frequency
[Hz]",'blue',10*n,abs(h))
plt.show()
```