

Procesamiento de señales

TP1.

1) Sistemas LTI.

1. Demuestre si los siguientes son LTI.

Considerando desplazamiento en el tiempo:

 $t \rightarrow t - t_0$

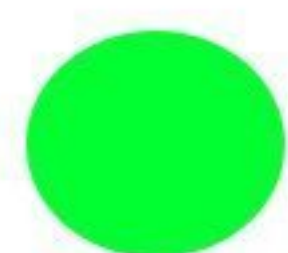
a. $y(t) = x(t) * \cos(t)$

Reemplazando:

Salida: $y(t) \rightarrow y(t - t_0) \rightarrow$ Sistema $\rightarrow = x(t - t_0) * \cos(t - t_0)$

Entrada: $x(t) \rightarrow x(t - t_0) \rightarrow$ Sistema $\rightarrow = x(t - t_0) * [\cos(t)]$

No es LTI



b. $y(t) = \cos(x(t))$

Salida: $y(t) \rightarrow y(t - t_0) \rightarrow$ Sistema $\rightarrow = \cos[x(t - t_0)]$

Entrada: $x(t) \rightarrow x(t - t_0) \rightarrow$ Sistema $\rightarrow = \cos[x(t - t_0)]$

Es LTI

c. $y(t) = e^{x(t)}$

Salida: $y(t) \rightarrow y(t - t_0) \rightarrow$ Sistema $\rightarrow = e^{x(t - t_0)}$

Entrada: $x(t) \rightarrow x(t - t_0) \rightarrow$ Sistema $\rightarrow = e^{x(t) - t_0}$

No es LTI

SI es invariante en el tiempo pero NO es lineal o sea, $e^{ax_1(t)+bx_2(t)} = e^{ax_1(t)} * e^{ax_2(t)}$ que no es igual a la suma de las salidas independientes que serian $e^{ax_1(t)} + e^{bx_2(t)}$. En cualquier caso NO es LTI, pero a causa de la no linealidad

d. $y(t) = 1 / 2 x(t)$

Salida: $y(t) \rightarrow y(t - t_0) \rightarrow$ Sistema $\rightarrow = 1 / 2 x(t - t_0)$

Entrada: $x(t) \rightarrow x(t - t_0) \rightarrow$ Sistema $\rightarrow = 1 / 2 x(t - t_0)$

Es LTI

1pt

2) Ruido de cuantización.

1. Calcule la relación señal a ruido de cuantización teórica máxima de un sistema con un ADC de:

Aplicando $SNR = 1,76 + 6,02 * N$, donde 'N' es la resolución en bits, se tiene:

- **24 bits:** $24 * 6,02 + 1,76 = 146,24$ dB
- **16 bits:** $16 * 6,02 + 1,76 = 98,08$ dB
- **10 bits:** $10 * 6,02 + 1,76 = 61,96$ dB
- **8 bits:** $8 * 6,02 + 1,76 = 49,92$ dB
- **2 bits:** $2 * 6,02 + 1,76 = 13,8$ dB

2. Dado un sistema con un ADC de 10 bits, ¿qué técnica le permitiría aumentar la SNR? ¿En que consiste?

La técnica que permitiría aumentar la SNR de un ADC de 10 bits es el oversampling (sobre muestreo).

La técnica consiste en disminuir la densidad espectral del ruido mediante un incremento en la frecuencia de muestreo. De esta manera, en el caso planteado se tiene que al realizarse un oversampling x 4 se puede lograr un incremento de 6 dB en un ADC de 10 bits, logrando aproximadamente un SNR de 68 dB.

1pt

3) Filtro anti alias y reconstrucción.

Dado que el trabajo que se realizará consiste en un detector de golpes mediante el análisis de señales de audio, se considera adecuado utilizar un filtro analógico anti alias previo al digitalizador (ADC) que efectúe el filtrado de señales en frecuencias inferiores a los 20 KHz.

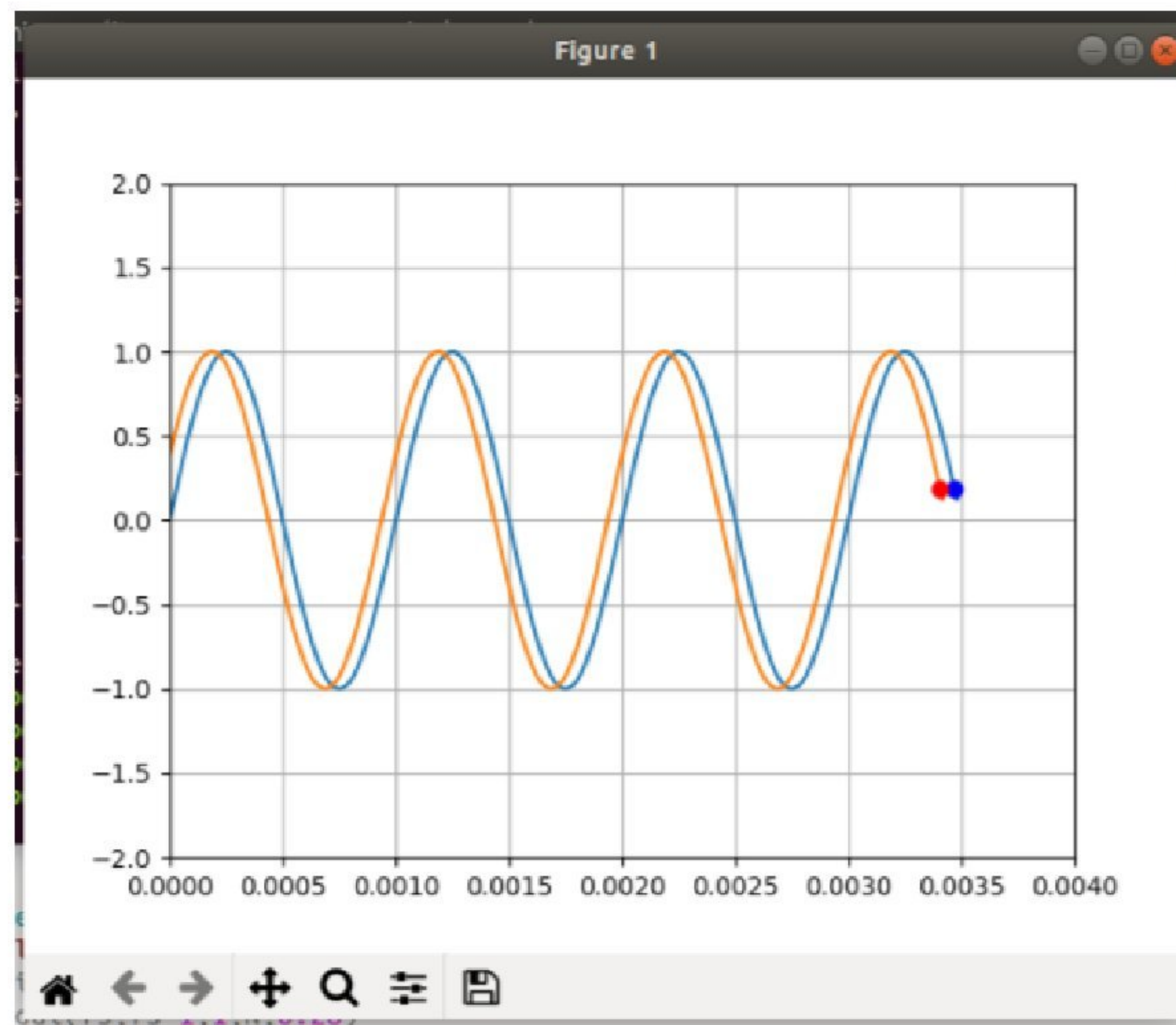
Se seleccionó dicha frecuencia porque es la frecuencia máxima del sonido audible.

4) Generación y simulación. 2.5pt

1. Genere un módulo o paquete con al menos las siguientes funciones:

Testeo del módulo con $N=1000$, Amplitud $\{-1,1\}$ y $f_s = 1000$ (Para la prueba se compara la señal senoidal con las demás funcionalidades).

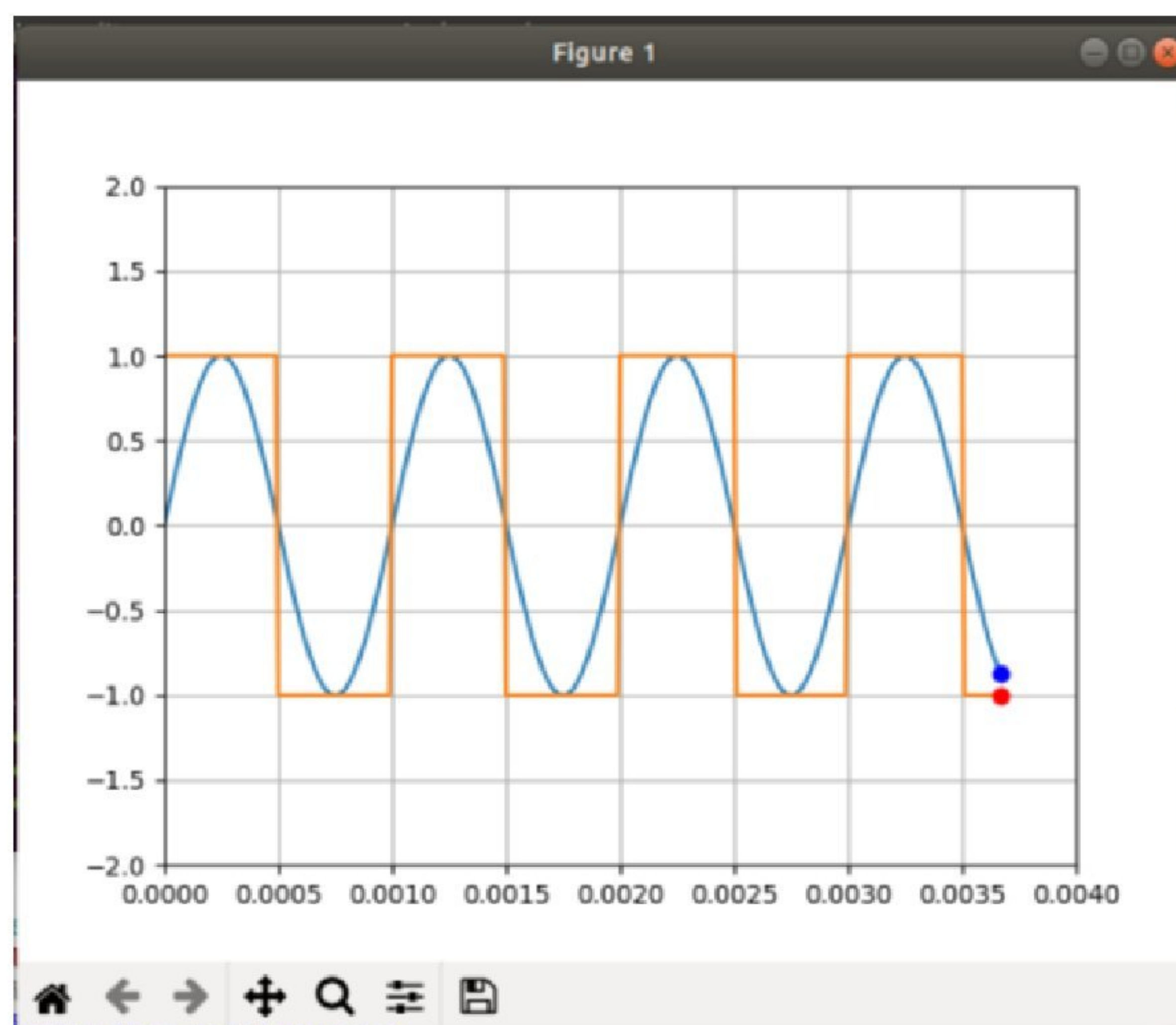
- Senoidal:



Senoidales con desplazamiento de fase

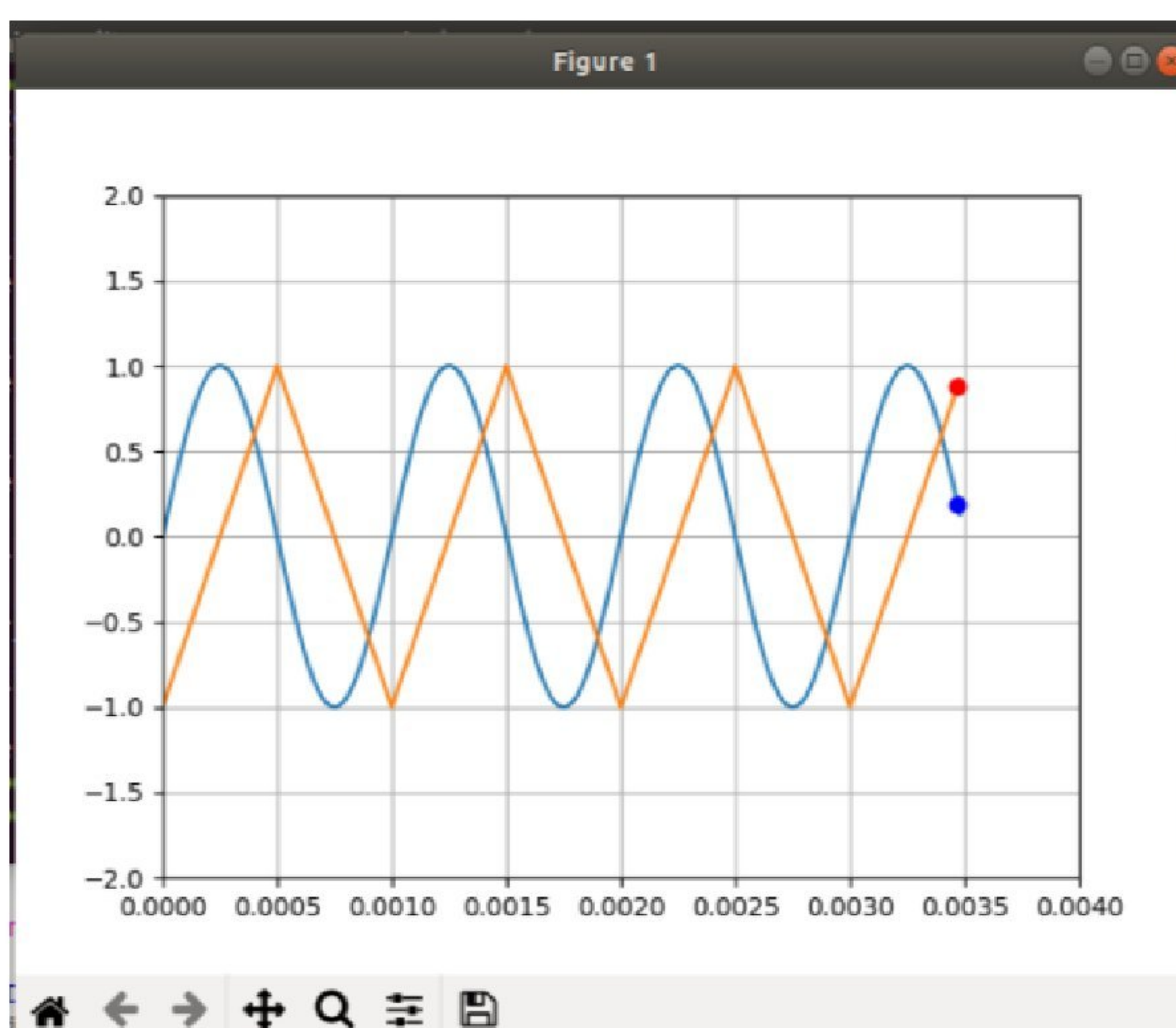
- Cuadrada:

No hace falta dibujar F_s superpuesto con las otras confunde y no le encuentro el sentido.
Si se puede poner la senial con puntos en vez de lineas para que se aprecie el espaciado $1/f_s$.



Senoidales y señal cuadrada

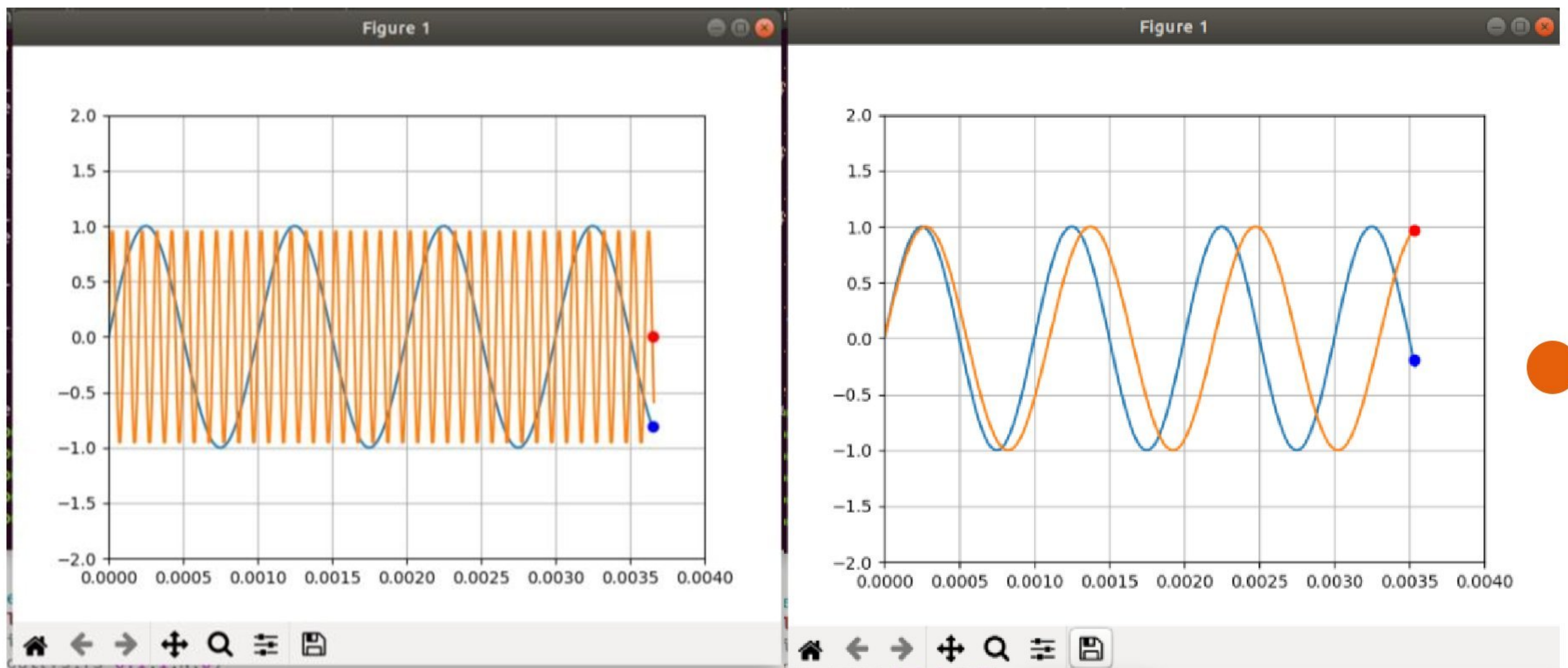
- Triangular:



Senoidales y señal triangular.

2. Realice los siguientes experimentos

2.1. $f_0 = 0.1 \cdot f_s$ y $1.1 \cdot f_s$ ¿Como podría diferenciar las senoidales?



Izquierda: Azul f_s , Naranja $f_0 = 0.1 \cdot f_s$ - Derecha: Azul f_s , Naranja $f_0 = 1.1 \cdot f_s$.

Las senoides podrían diferenciarse seleccionando una velocidad de muestreo adecuada a la frecuencia de cada una.

Lo que se busca aca es que dada tu f_s , samplees a 0.1 y a 1.1, al parecer estas sampleando mucho mas rapido que f_s y por eso tu seniales se ven bien. En teoria si mostras el trazo azul que es f_s en un sistema que esta sampleado a f_s solo verias un punto por ciclo, pero vos mostras una senoide perfecta. Mirando el codigo tenes que corregir como estas interpretando f_s .

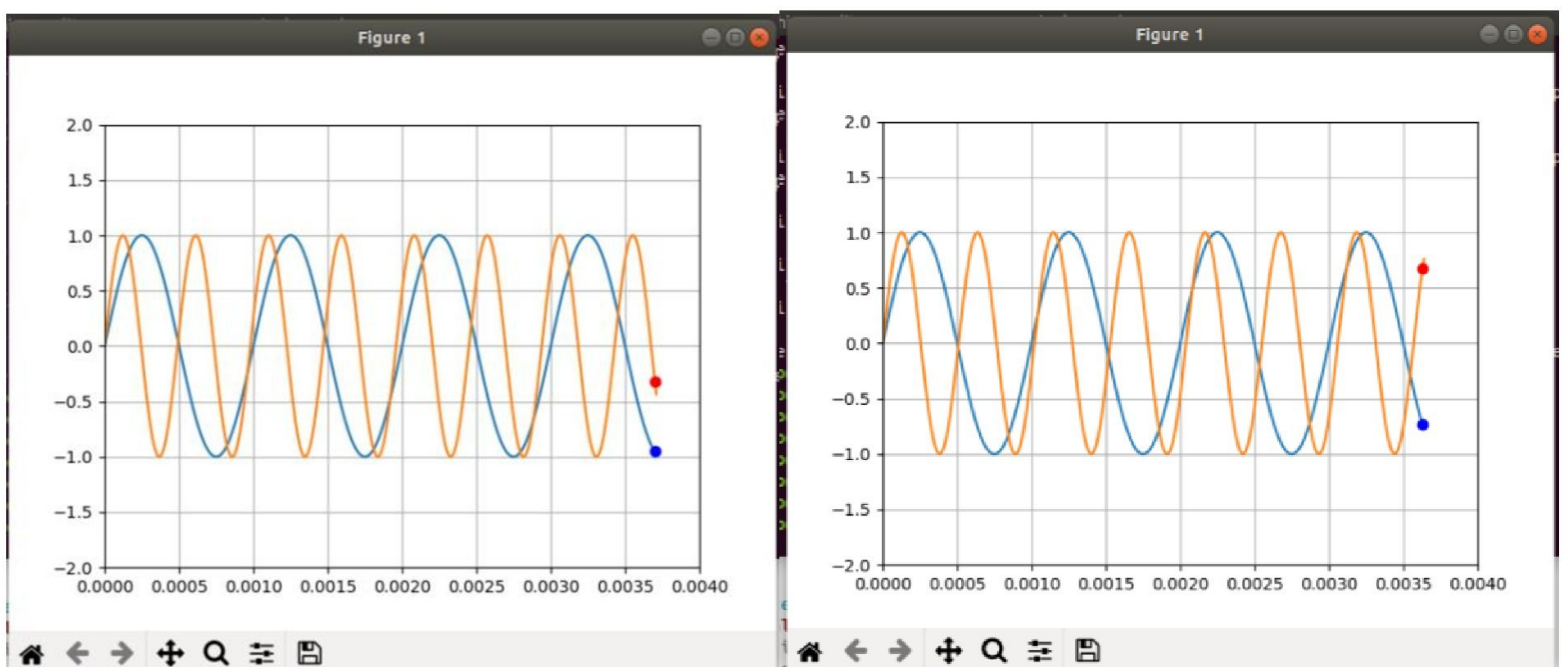
Esto es $1/f_s$ es la distancia entre cada punto que podes graficar, en el medio no hay nada. Cuando hagas eso se te van a mezclar 0.1 y 1.1 tal que no se podran reconocer una de otra.

2.2. $f_0 = 0.49 \cdot f_s$ y $0.51 \cdot f_s$ ¿Como es la frecuencia y la fase entre ambas?

intenta no mostrar como senoide f_s , no tiene mucho sentido

pero si muestra solo los puntos de la senial de interes en espacios de tiempo de $1/f_s$. con esa correccion se van a aclarar todos los graficos.

Lo mismo para el caso de las fases.



Izquierda: Azul f_s , Naranja $f_0 = 0.49 \cdot f_s$ - Derecha: Azul f_s , Naranja $f_0 = 0.51 \cdot f_s$.

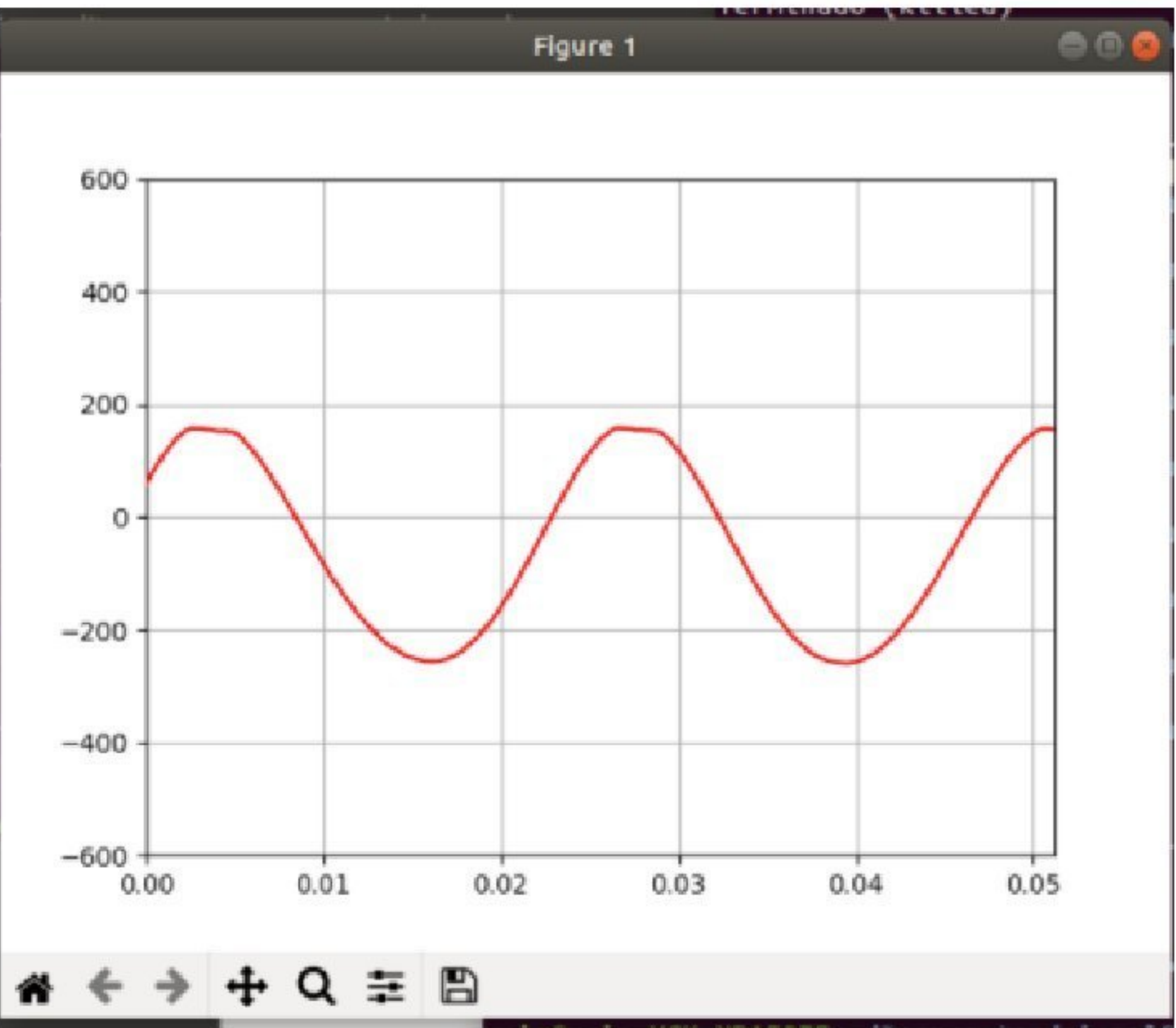
Las frecuencias y las fases son diferentes. En ambos casos se aprecia un aumento de la frecuencia y un desplazamiento de la fase.

5) Adquisición y reconstrucción.

1. Señal senoidal.

10 b.

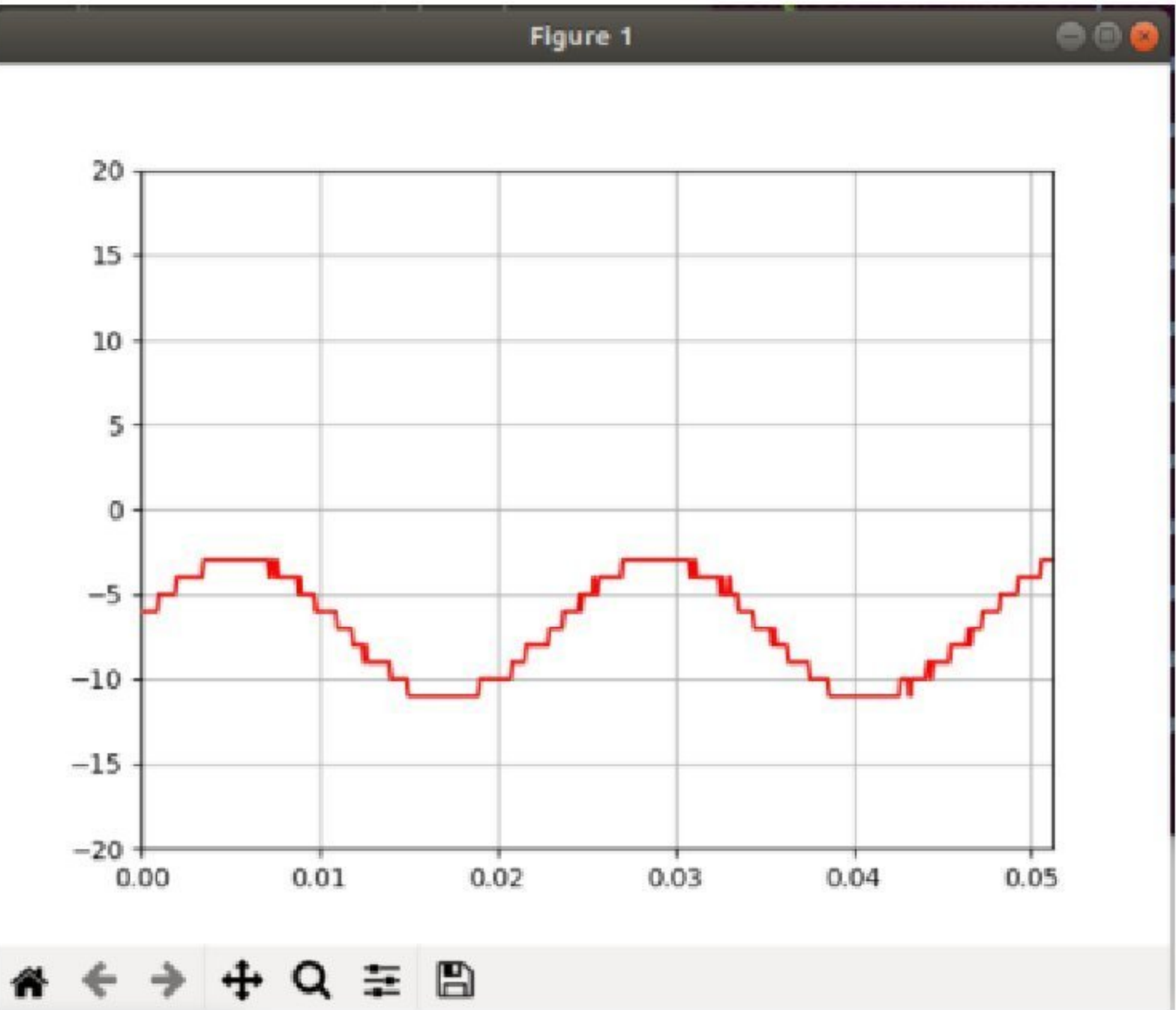
3pts



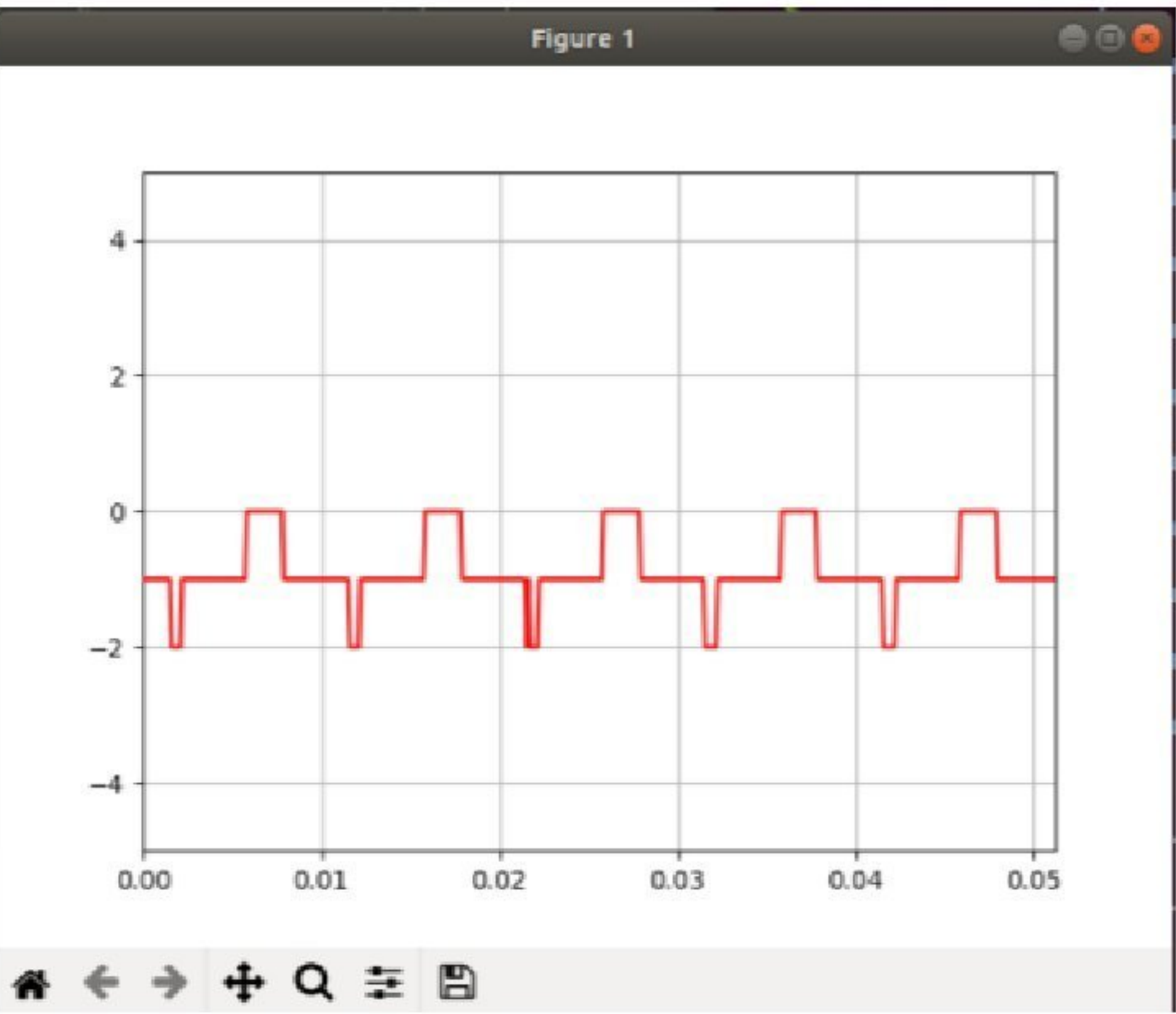
Porque te quedo en [200,-200]?
en que unidad esta el eje Y?
si son enteros, directo del ADC
no llegas a [512,-512]?



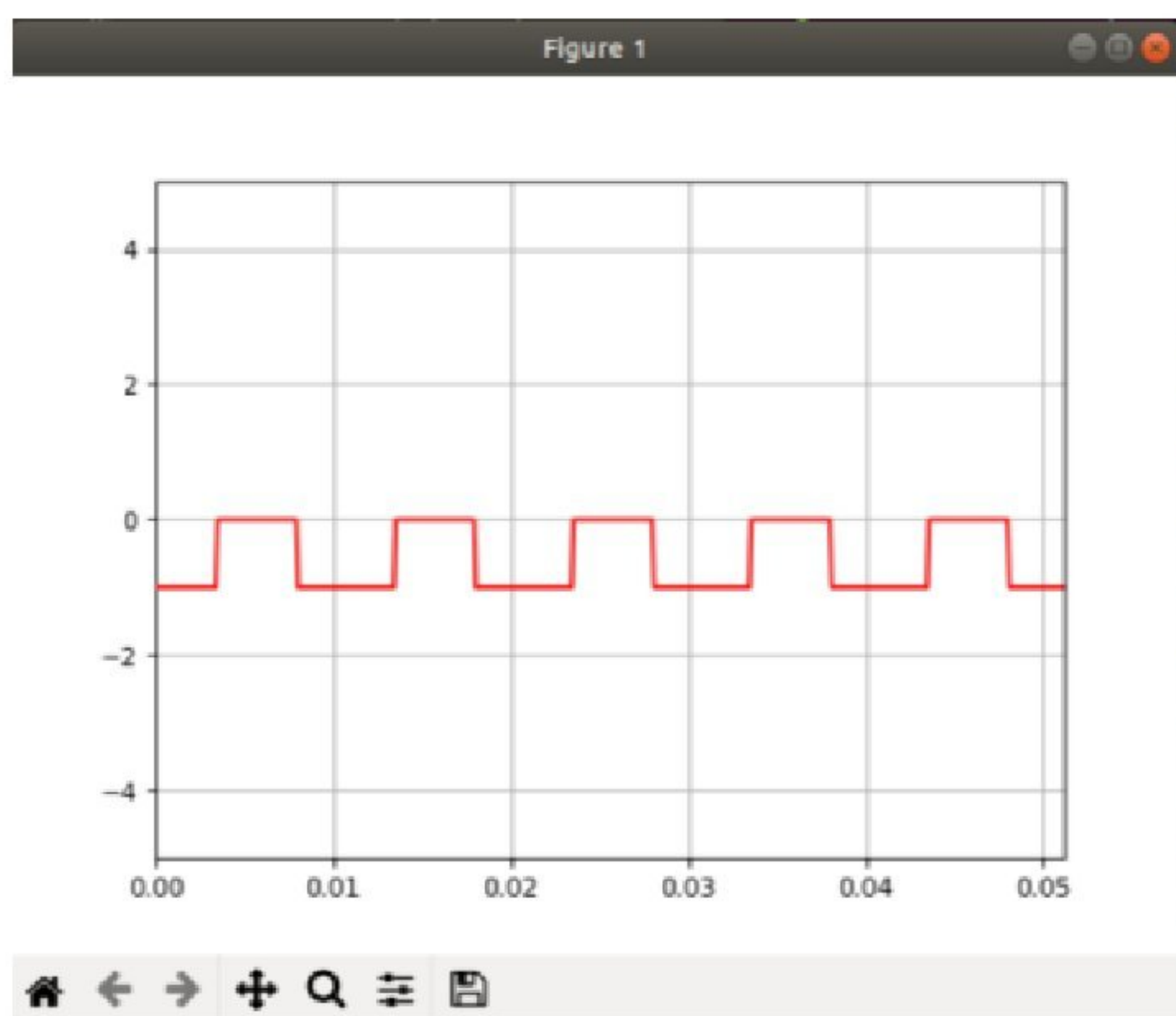
8b



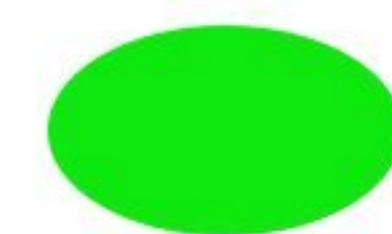
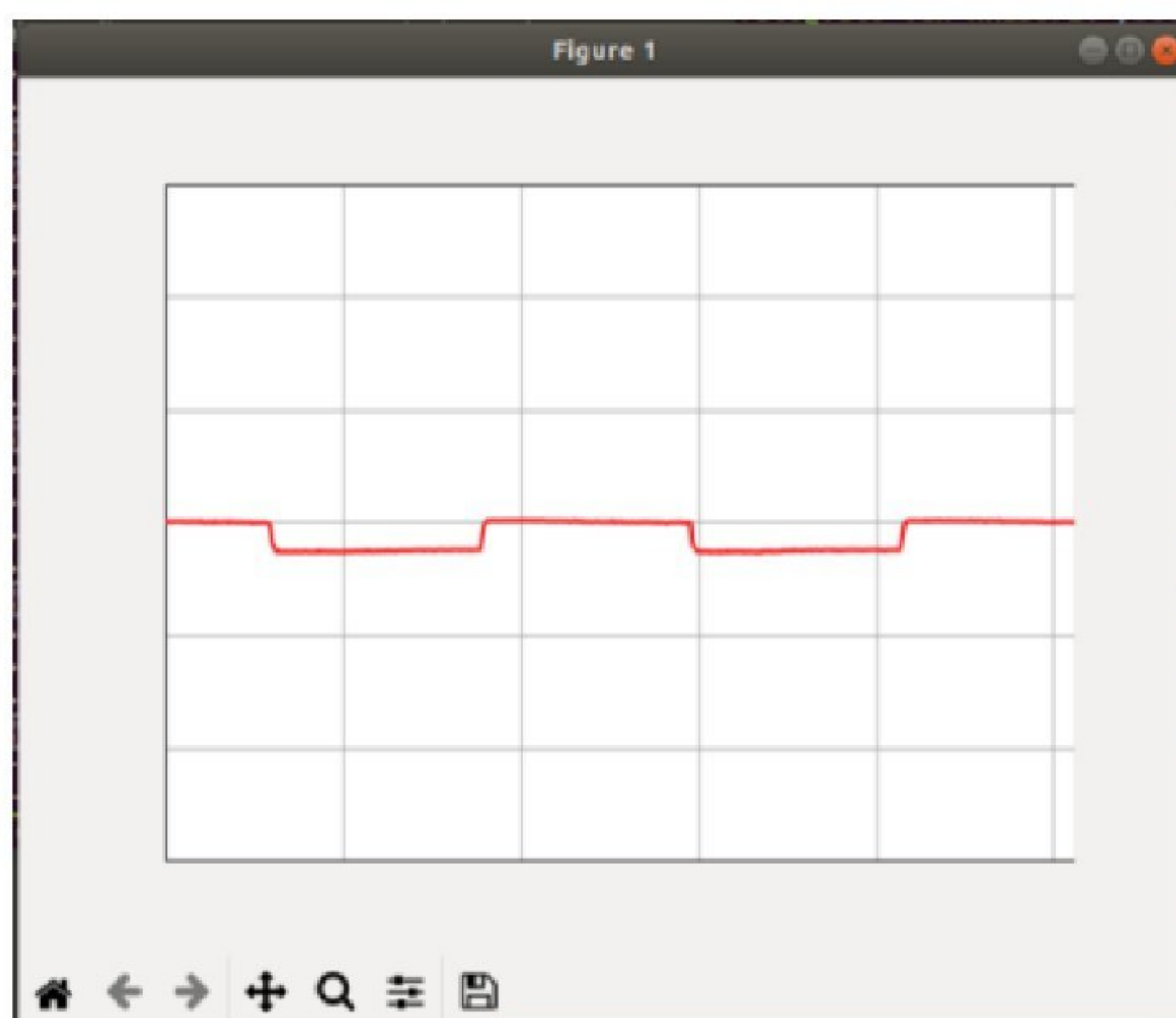
4b



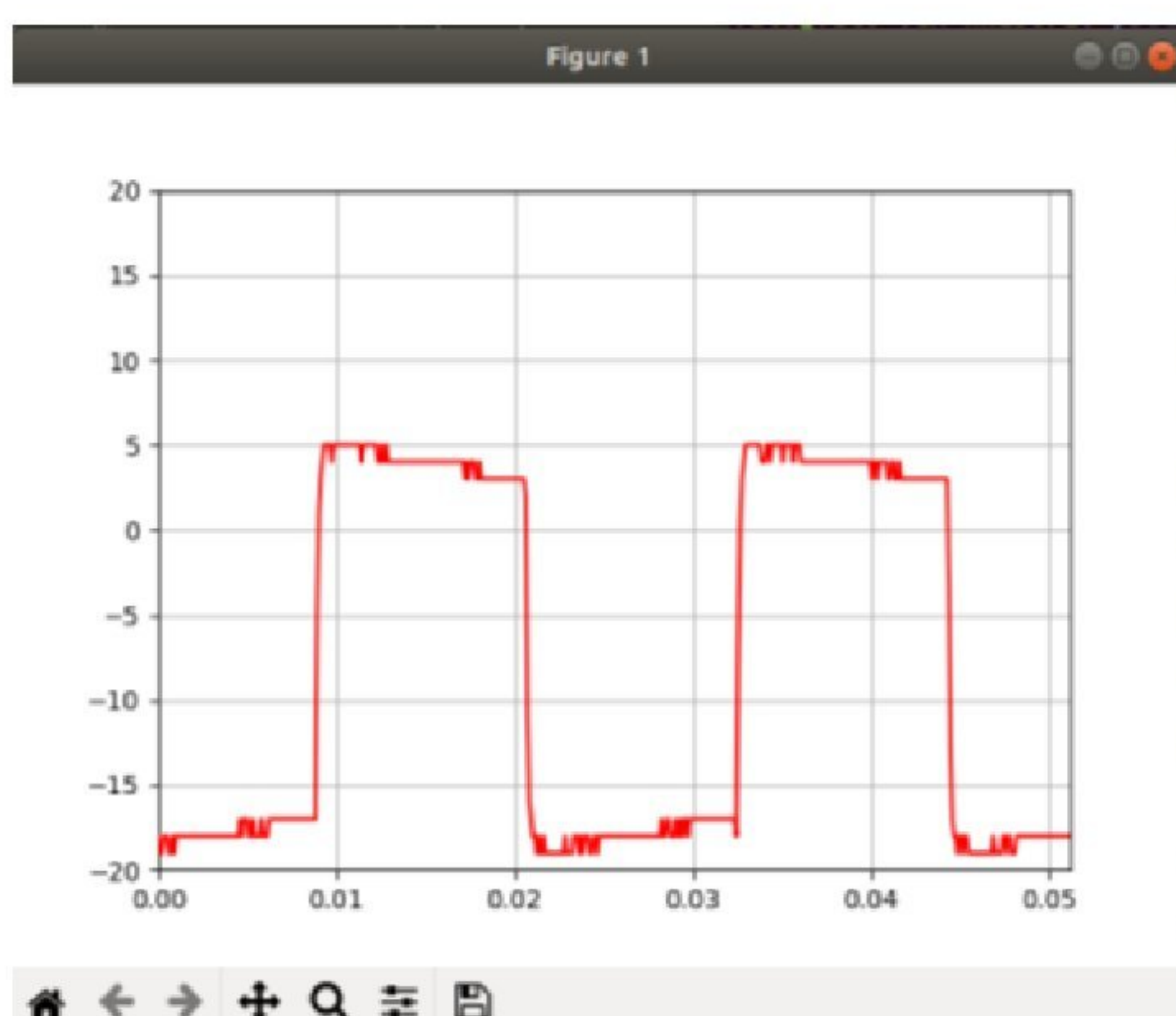
2b



2. Señal cuadrada.
10 b.



8b

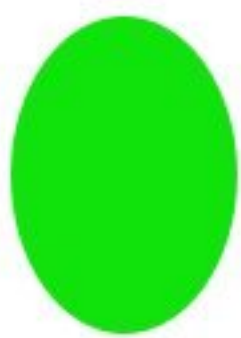
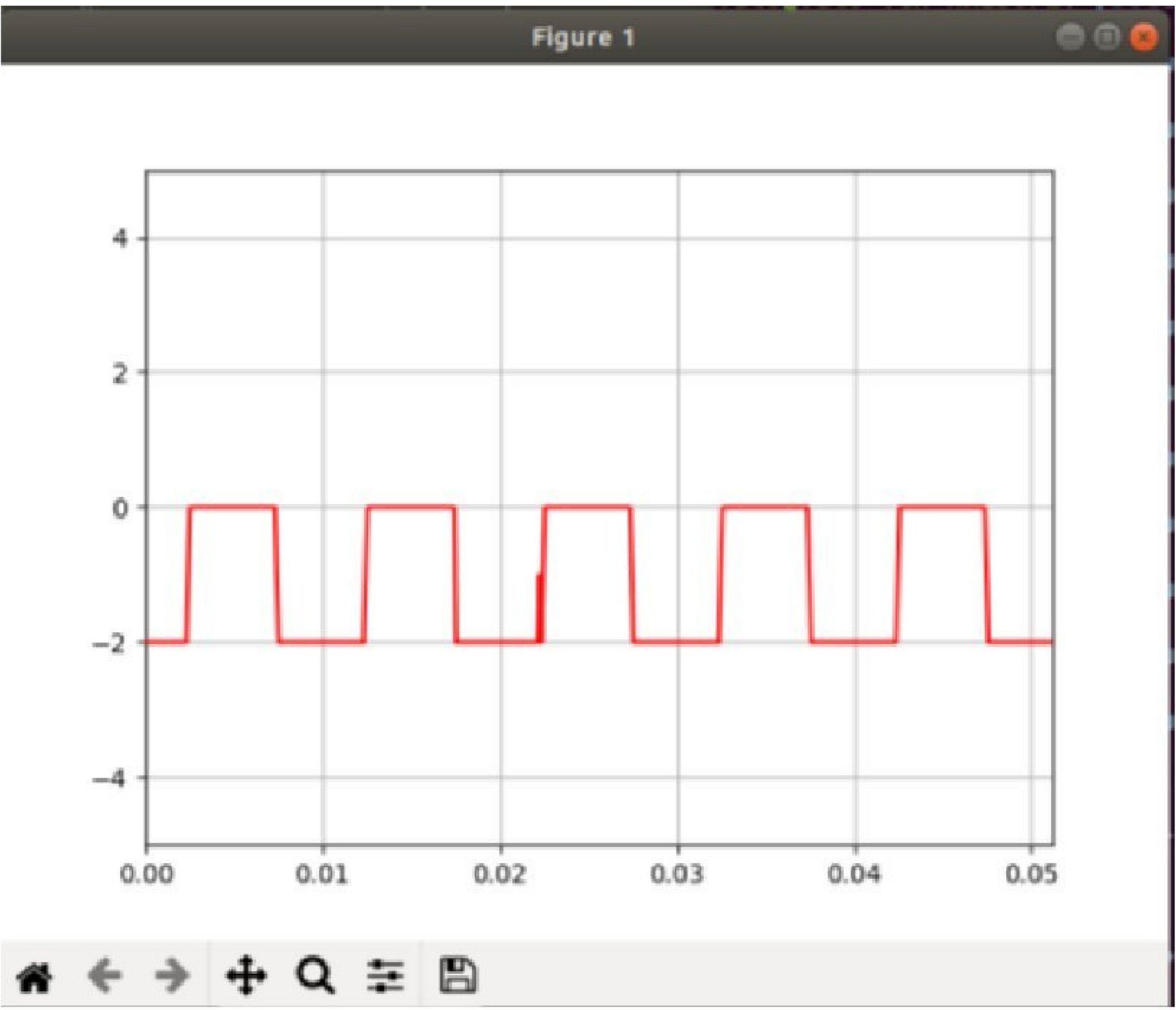


Ojo que se ve que tienes un offset hacia los valores negativos
Deberían quedarte las señales centradas en 0, con una sutil variación

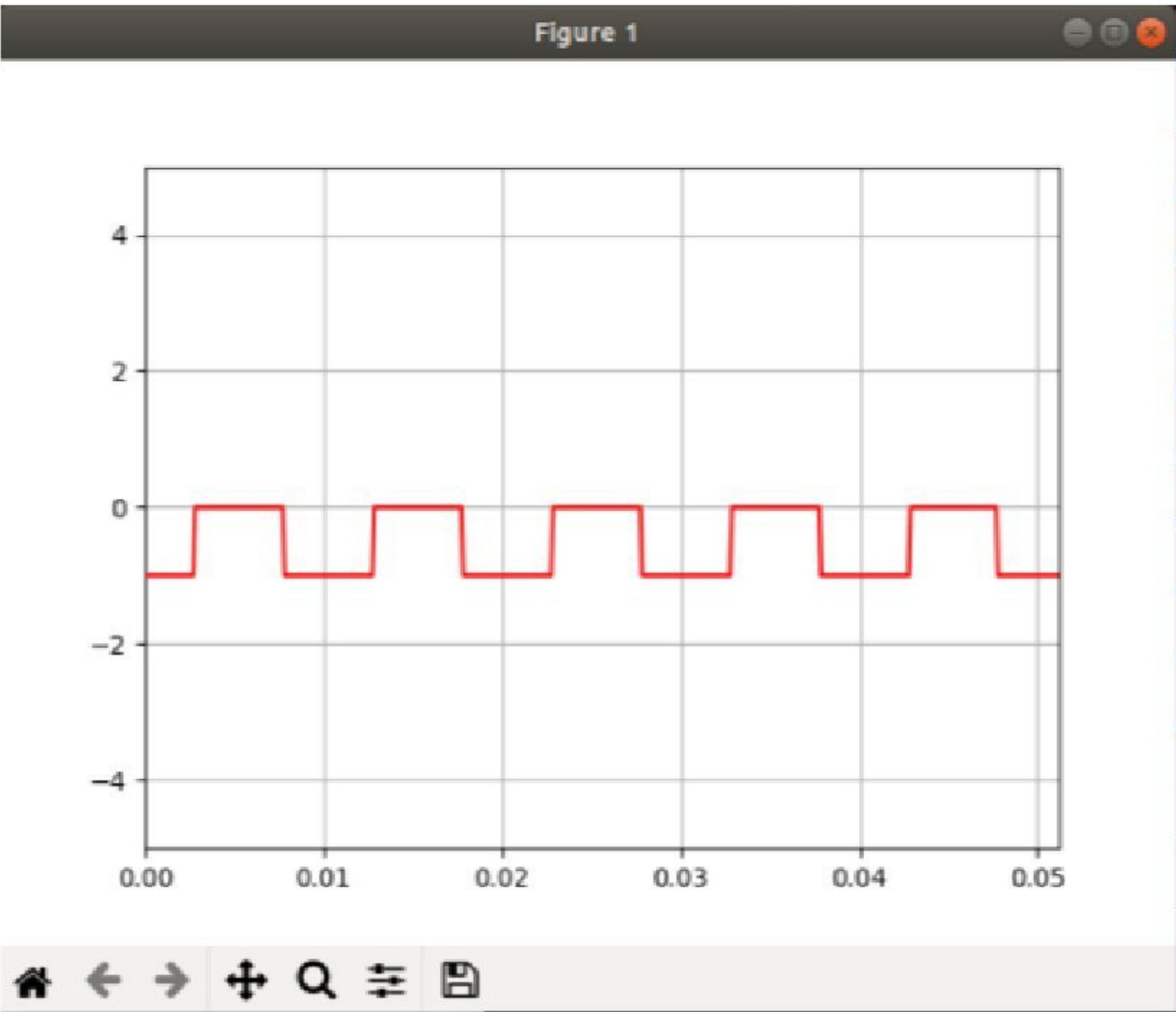


Puedes revisar el divisor que este ok, el cap y sino correjilo por firmw

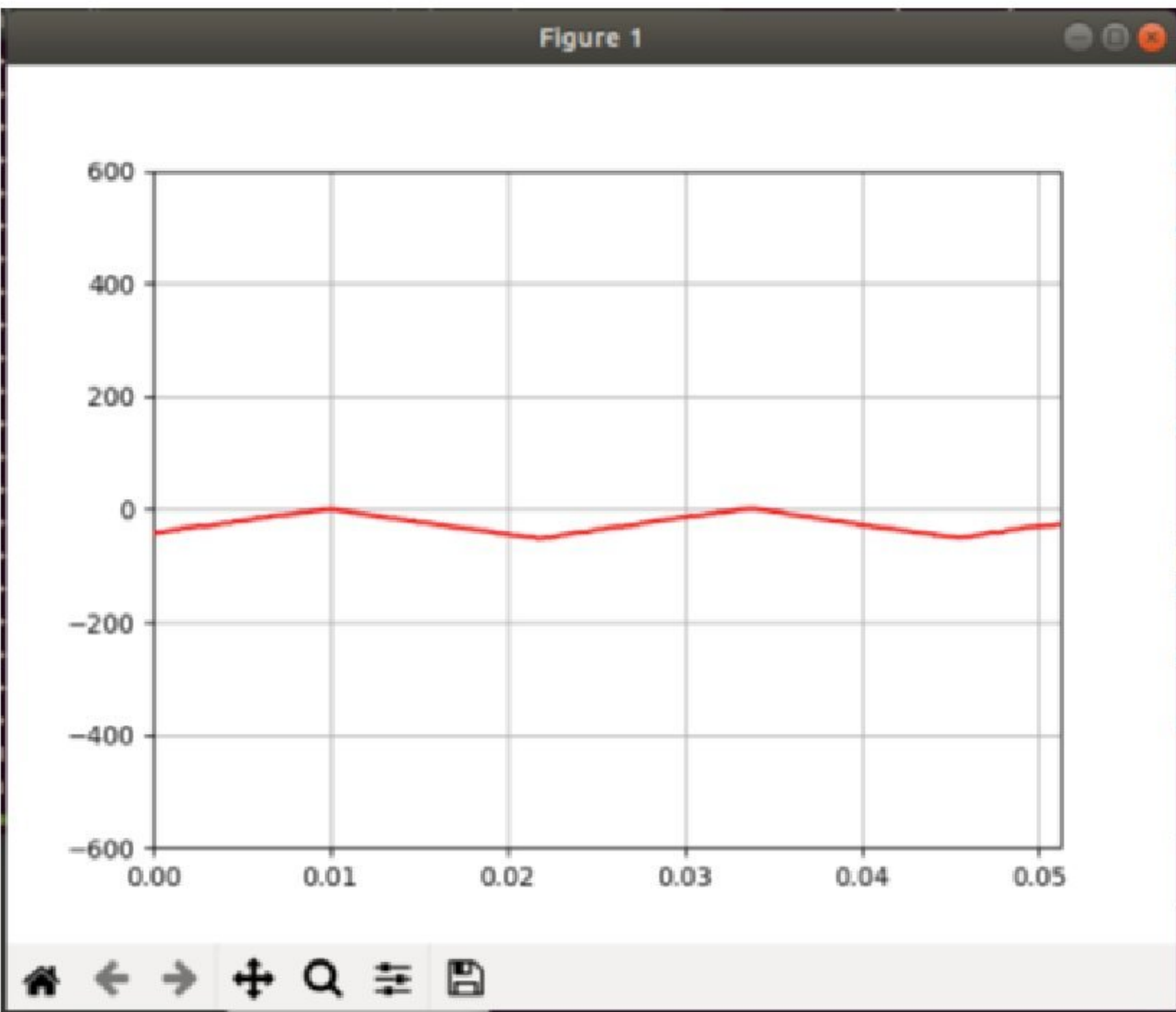
4b



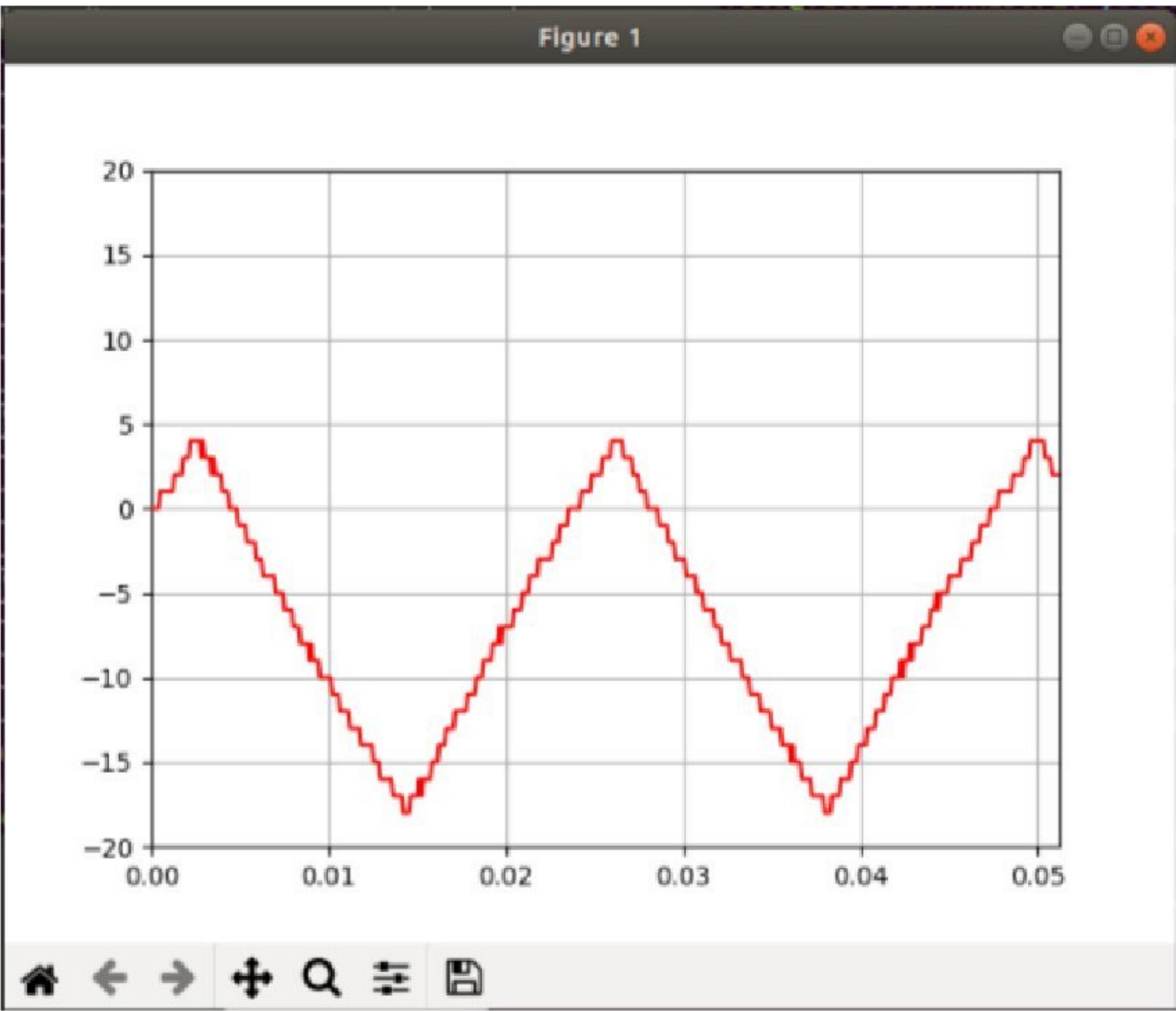
2b



3. Señal triangular.
10 b.

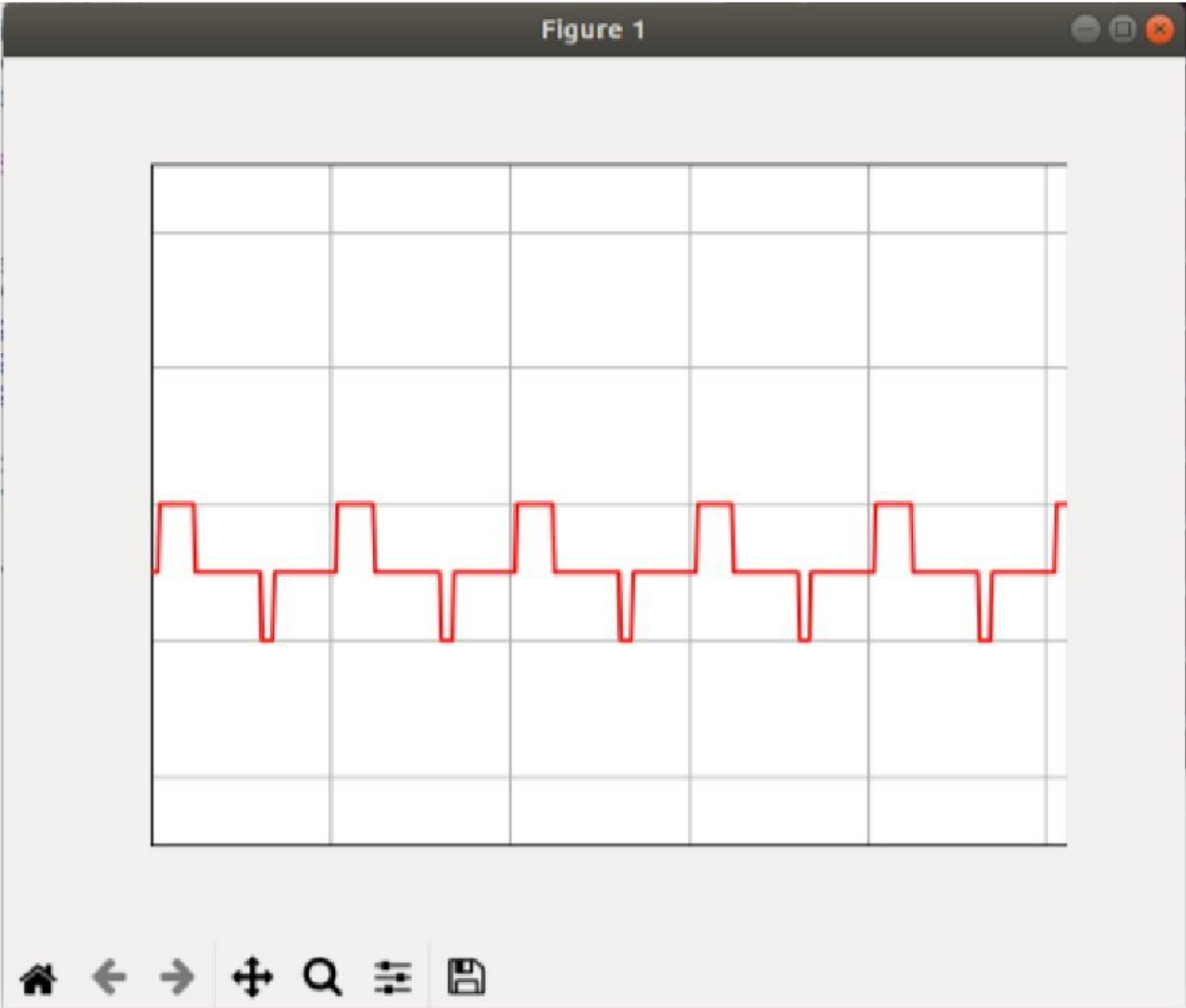


8b

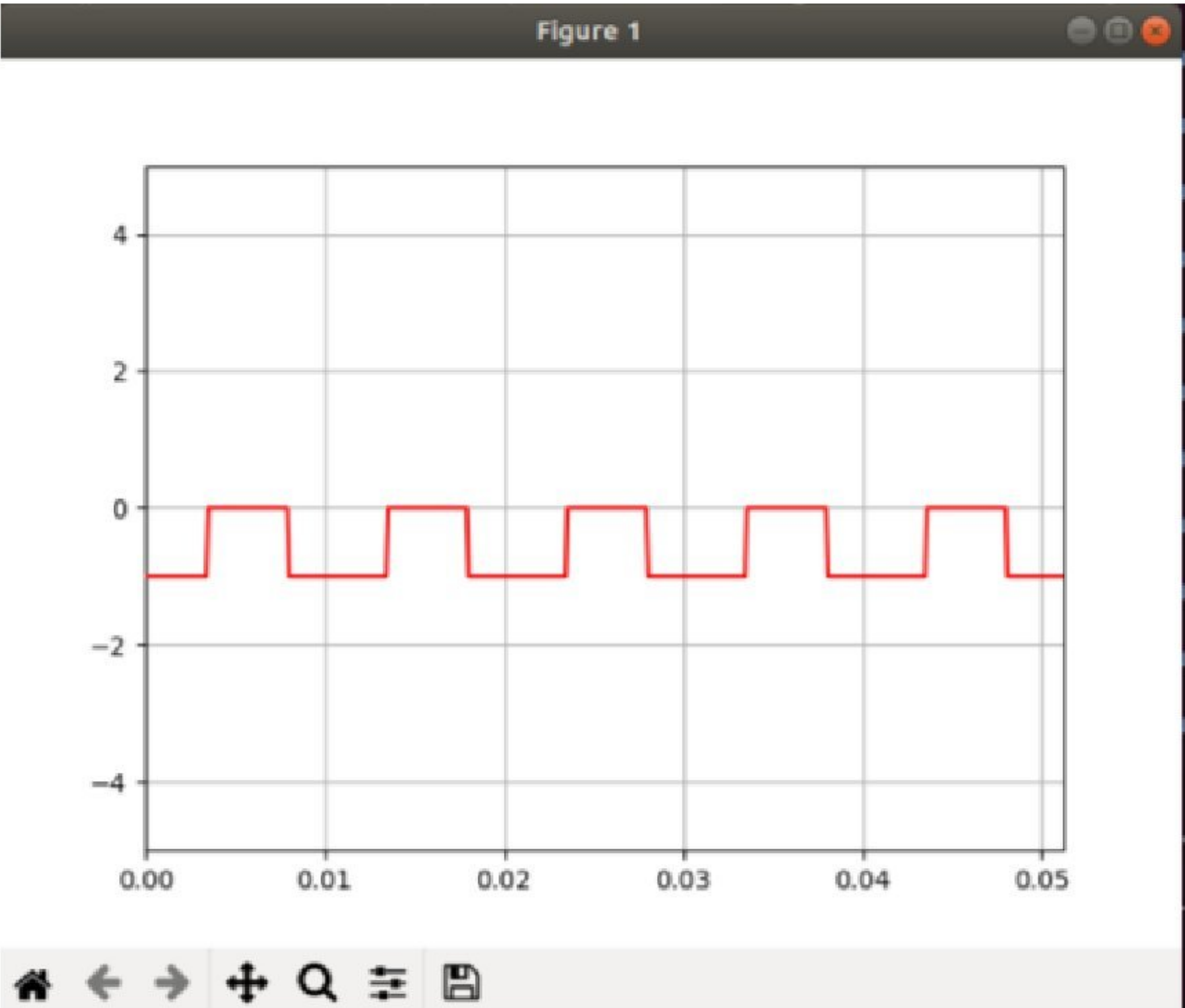


de nuevo atento con el offset

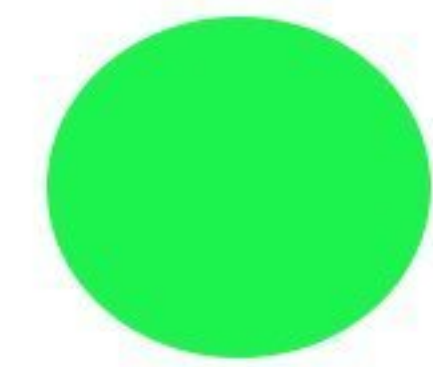
4b



2b



Conclusiones generales: al disminuir la resolución en bits se pierde definición de la señal. Finalmente con 2 bit de resolución, los tres tipos de señales se reconstruyen como una señal cuadrada, sin poder distinguir entre las mismas las diferencias.



6) Sistema de números. 0.75pt

1. Explique brevemente algunas de las diferencias entre la representación flotante de simple precisión (32b) y el sistema de punto fijo Qn.m

La principal diferencia radica en que el sistema de representación de punto flotante de 32 bits los números son representados de manera similar a la notación científica, con una mantisa (A) y un exponente (B) de la siguiente manera $A * 2^B$. En el caso del sistema de punto fijo, los números se representan con cantidades fijas de bits antes y después del punto decimal.

El sistema de representación con punto flotante puede representar un rango más amplio de números y con mayor precisión que el sistema de representación con punto fijo. Sin embargo, el sistema de representación de punto fijo permite obtener ventajas relativas al costo de la implementación con sistemas de representación en punto flotante.

2. Escriba los bits de los siguientes números decimales (o el más cercano) en float, Q1.15, Q2.14:

- 0.5:
 - Float: 00111111000000000000000000000000
 - Q1.15: 0100000000000000
 - Q2.14: 0010000000000000

- -0.5:
 - Float: 10111111000000000000000000000000
 - Q1.15: 1100000000000000
 - Q2.14: 1110000000000000

- 1.25:
 - Float: 10111111101000
 - Q1.15: 101000000000000000 no se puede representar nada mas chico que -1 en Q1.5
 - Q2.14: 100100000000000000 este da 0xB0 00
- 0.001:
 - Float: 00111010100000110001001001101111
 - Q1.15: 0000000000100001
 - Q1.14: 0000000000010000
- 2.001:
 - Float: 1100
 - Q1.15: 1000000000100001 no se puede representar nada menos de 01
 - Q2.14: 0100000000010000 no se puede representar nada menor a -2
- 204000000:
 - Float: 01001101010000101000110010110000
 - Q1.15: 000000000000000000 no se puede o en todo caso el mas cercano es el mas grande en cada caso casi 1 para Q1.15 y casi 2 para Q2.14
 - Q2.14: 000000000000000000