

Procesamiento de señales

TP1.

9.5pts

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

Es invariante en el tiempo.

Sin embargo:

$e^{x(t) + ix(t)} \neq e^{x(t)} + e^{ix(t)}$

No es LTI

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



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.



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.



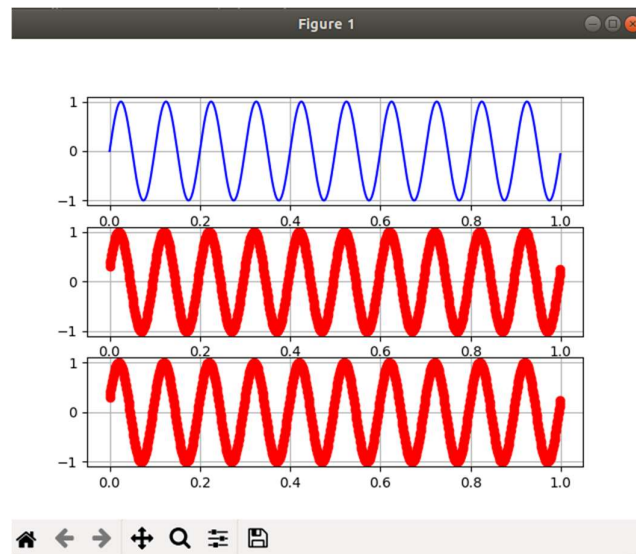
Se seleccionó dicha frecuencia porque es la frecuencia máxima del sonido audible.

4) Generación y simulación.

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

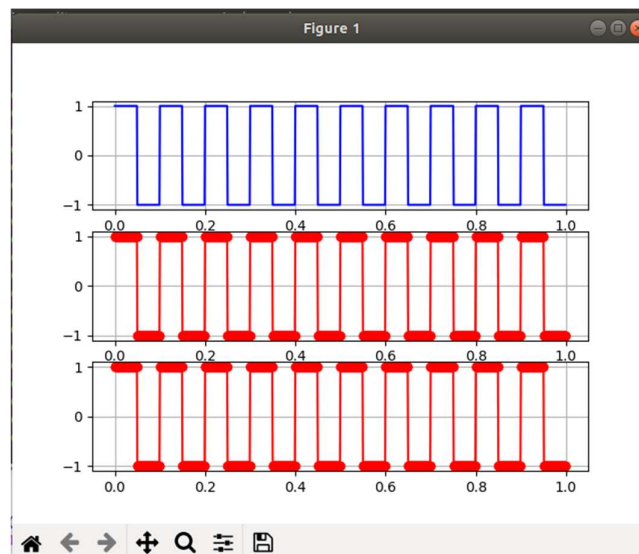
Testeo del módulo con $N=1000$, Amplitud $\{-1,1\}$ y $f_s = 1000$ (Para la prueba se compara la señal senoidal con las demás funcionalidades).

- Senoidal:



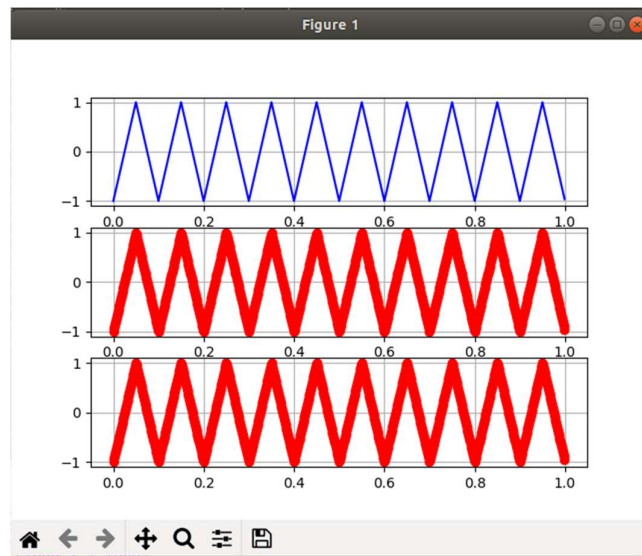
Senoidales: teórica, f_s y f_0 . Las dos últimas muestran un desplazamiento de fase.

- Cuadrada:



Señal cuadrada teórica, f_s y f_0 .

- Triangular:

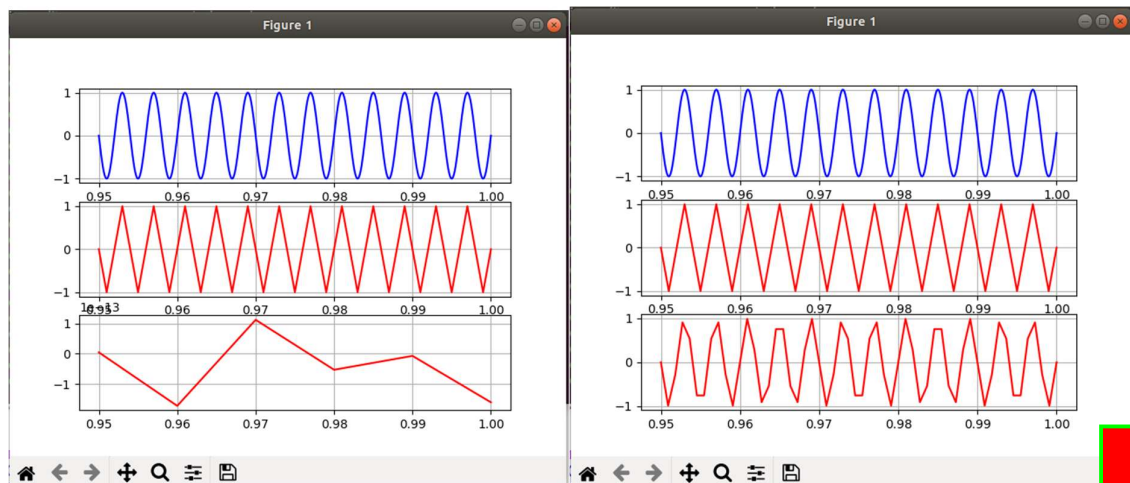


Señal triangular teórica, f_s y f_0 .

2. Realice los siguientes experimentos

* Observación: para las pruebas se utilizó una señal de 250 Hz.

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



Izquierda: $f_0 = 0.1 * f_s$ - Derecha: $f_0 = 1.1 * f_s$.

Las gráficas anteriores muestran en rojo f_s y f_0 , en ambos casos f_0 es el gráfico inferior.

En el caso de $f_0 = 0.1 * f_s$ ($f_0 = 100$ Hz) no se puede diferenciar la senoide original, mientras que con $f_0 = 1.1 * f_s$ ($f_0 = 1100$ Hz) si se la diferencia.

Aca hay varias cosas a marcar. 1) la f_s no se puede graficar, no es una señal, es solo un numero que dice cada cuanto se discretizo una señal de tiempo continuo. Pero no tiene forma de senoide ni de cuadrada, etc. Con lo cual cuando graficas f_s quiere decir que tenes otra base de tiempo mas rapida que te permite hacerlo, o sea OTRA f_s en el codigo para mostrar la f_s original. Eso no termina bien. Conceptualmente no estaria bien.

El segundo punto es en cuanto a que se pide si podes diferenciar una seúal de $0.1 * f_s$ contra una de $1.1 * f_s$, no si podes distinguir una señal de $0.1 * f_s$ por si sola. De hecho aca tenes otro error y es que una señal de $0.1 * f_s$ deberias verla perfectamente dado que esta muy por debajo de $f_s/2$.

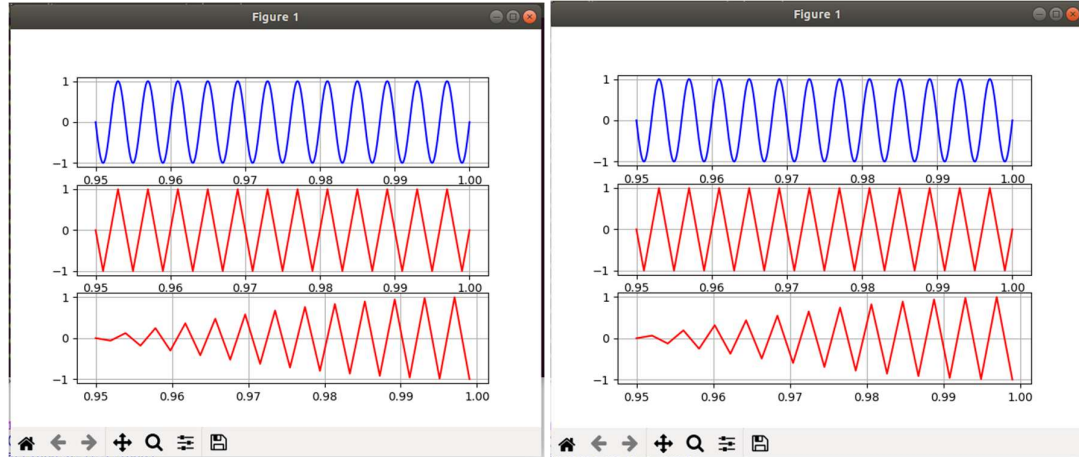
Lo ideal en este ejercicio es que superpongas $0.1 \cdot f_s$ y $1.1 \cdot f_s$ en el mismo gráfico porque así las conclusiones estarán a la vista y que NO trazas f_s en los gráficos como si fuera una señal, dado que no lo es y tampoco se podría graficar sin otra base de tiempo superior.

Ahora que ya dominas DFT, puedes jugar también viendo los espectros de $0.1 \cdot f_s$ y $1.1 \cdot f_s$ para ver donde te quedan los bins y tal vez te ayude a comprender y concluir.

Creo que en el video de repaso del TP1 también puedes revisar este punto a ver si te aclara un poco más

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

Más allá de la nota, me interesa que razones este punto, es importante y no importa cuantas veces interactuemos, me gustaría que lo puedas asimilar.



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

Las gráficas anteriores muestran en rojo f_s y f_0 , en ambos casos f_0 es el gráfico inferior.

En ambos casos, al compararse el muestreo de f_0 a 0.49 y 0.51 respecto de f_s , se aprecia que se pierde la señal original, notándose variaciones de fases.

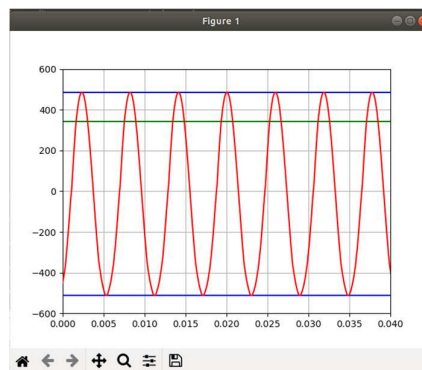
5) Adquisición y reconstrucción.

Para el ejercicio de generación y adquisición se utilizó como generador de señales a la EDU-CIAA-NXP, la cual generó las señales con DAC y las leyó mediante ADC.

En todos los gráficos se ilustra máximos y mínimos con líneas azules. La línea verde corresponde al cálculo de RMS en base al valor máximo.

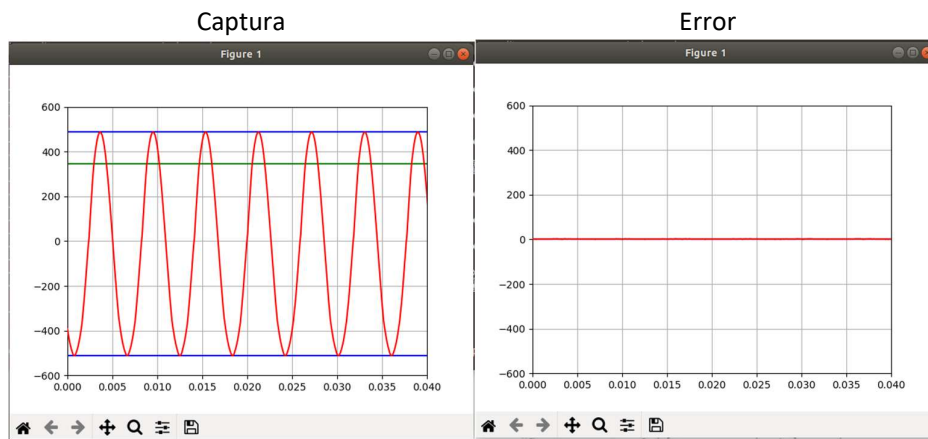
1. Señal senoidal.

Señal generada:

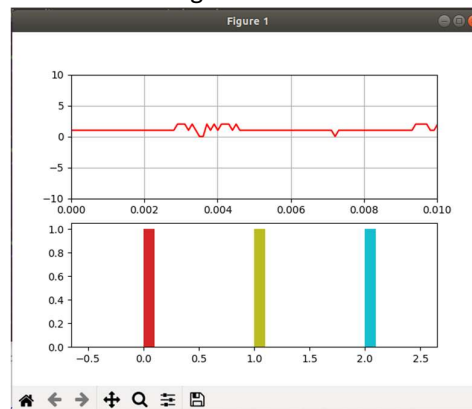


Capturas:

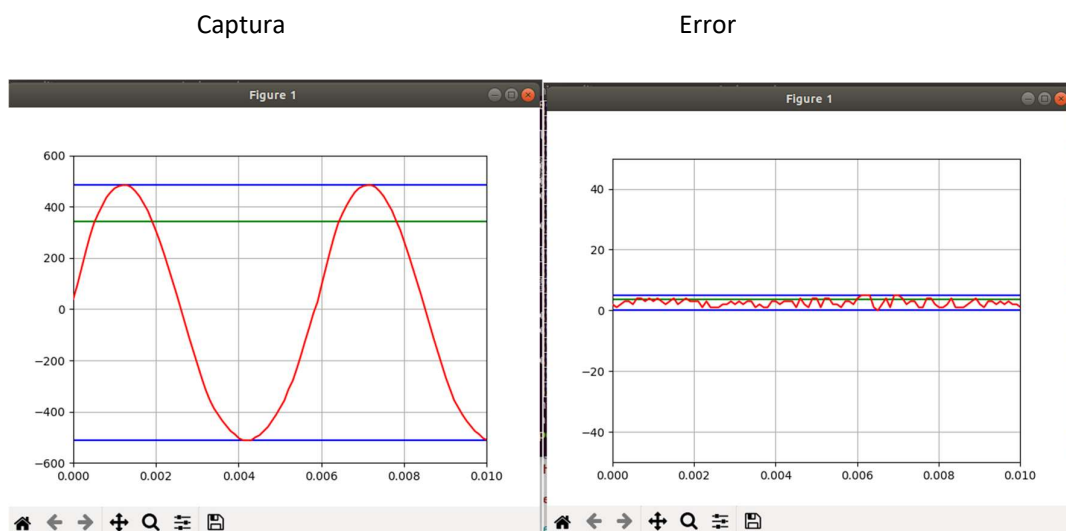
10 bits:



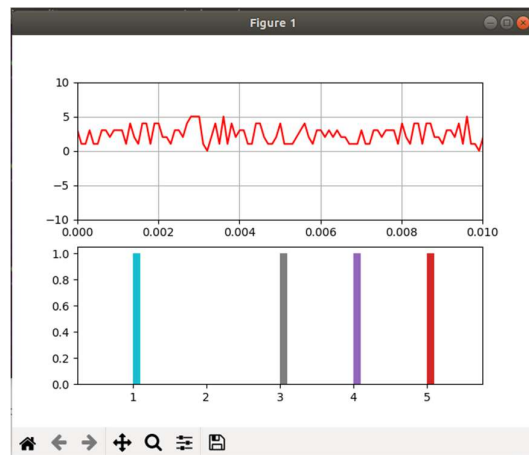
Histograma del error



8 bits:

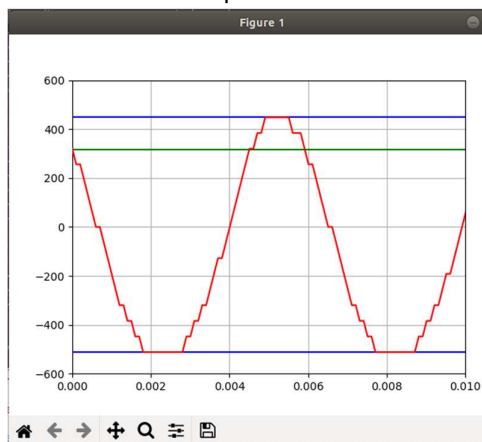


Histograma del error

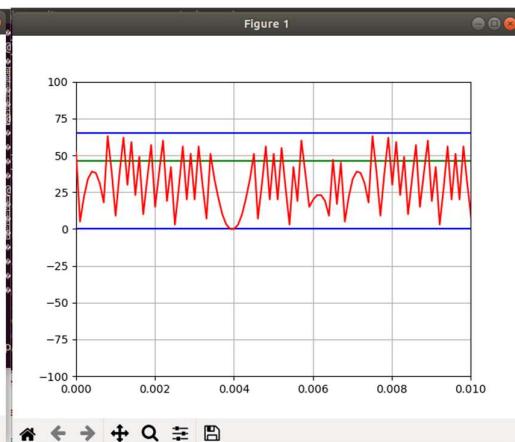


4 bits:

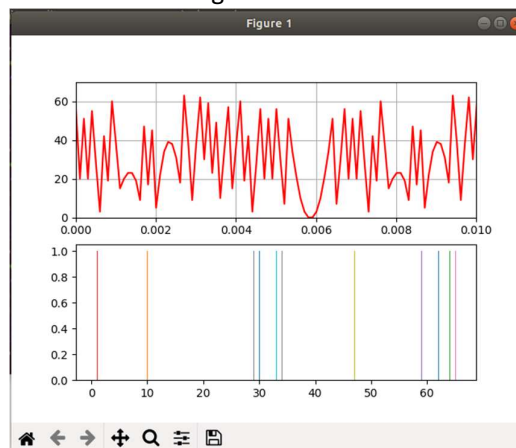
Captura



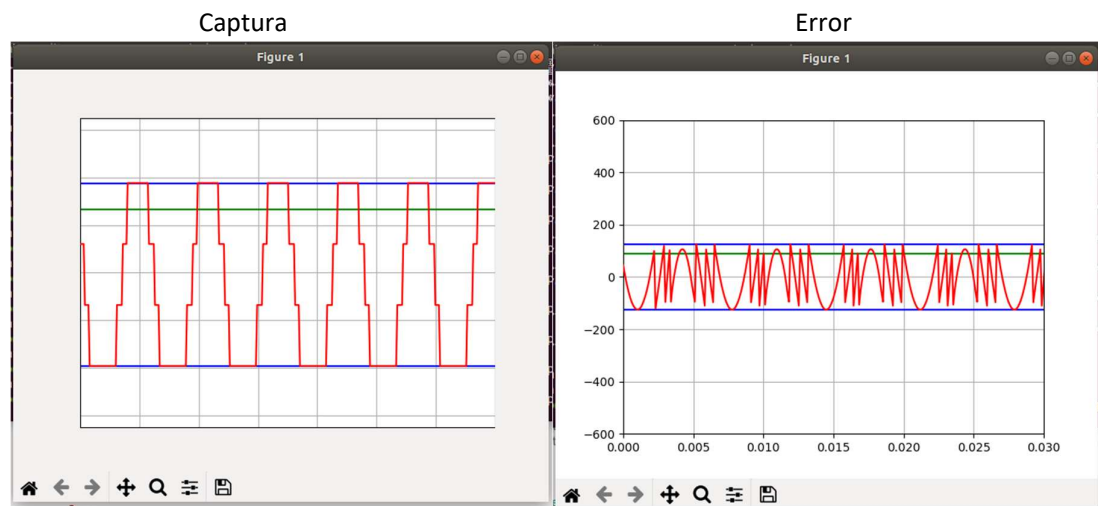
Error



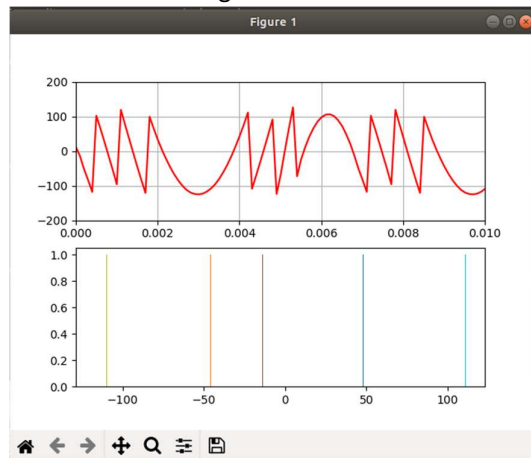
Histograma del error



2 bits:



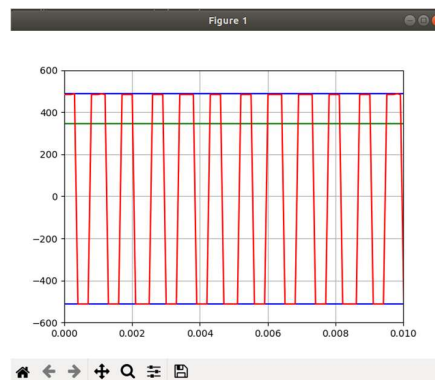
Histograma del error



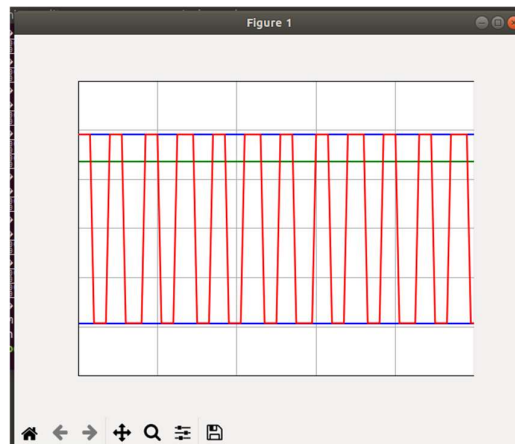
Comentarios: Se observa que al disminuir los bits de resolución en la captura se pierde definición de la señal. Además aumenta considerablemente el nivel de error en las capturas.

2. Señal cuadrada.

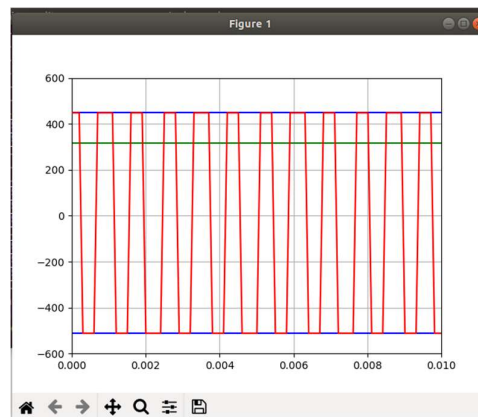
10 bits.



8 bits:



4 bits

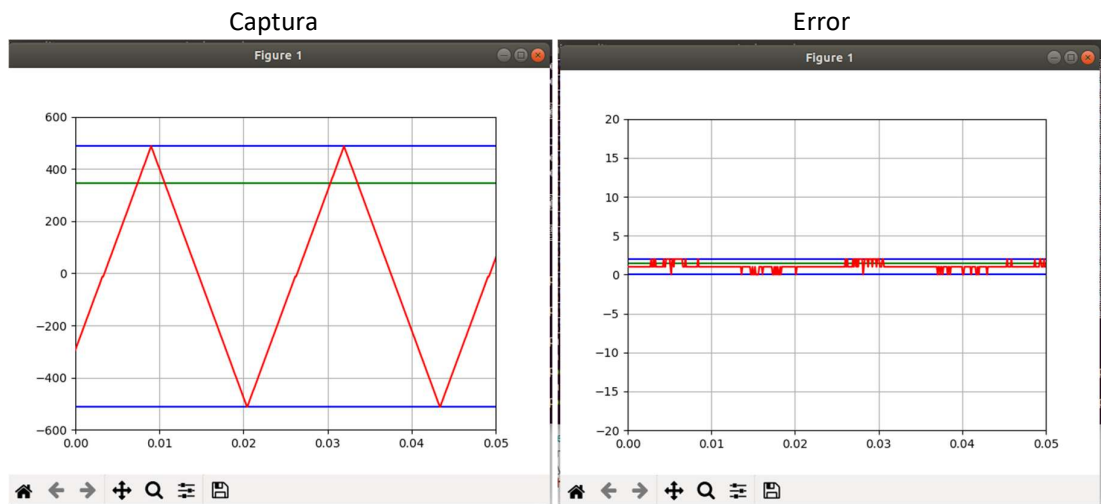


En el caso de la señal cuadrada se notó que no se registraban errores al comparar la señal generada con la medida, por lo cual no se realizaron los gráficos de error.

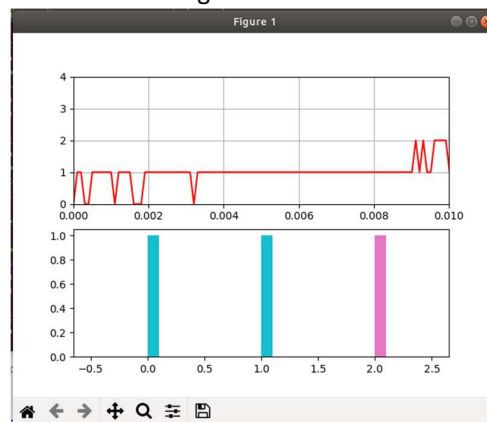
Cabe señalar, que para igualar las amplitudes se multiplicó el resultado de la salida por un escalar proporcional a la diferencia numérica con la representación de mayor precisión.

En todos los casos, se pudo ver que la señal cuadrada no varió su forma.

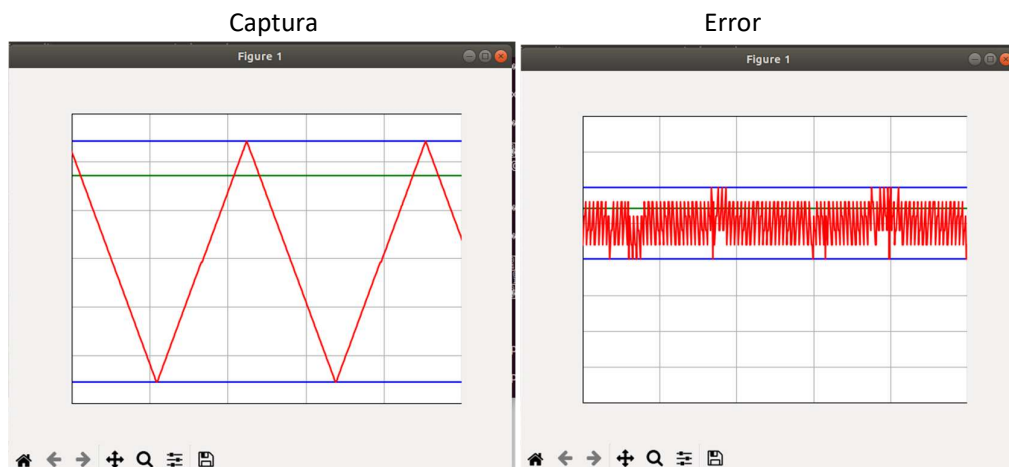
3. Señal triangular. 10 bits.



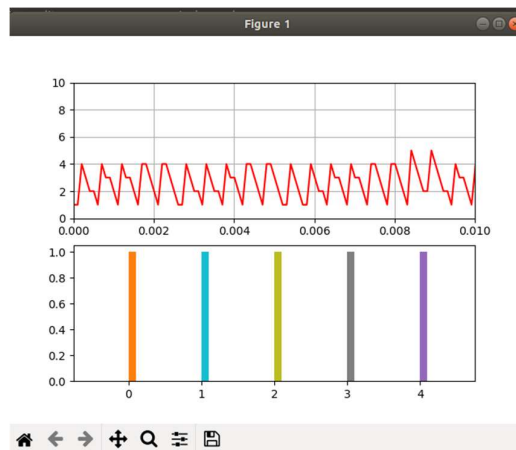
Histograma del error



8 bits

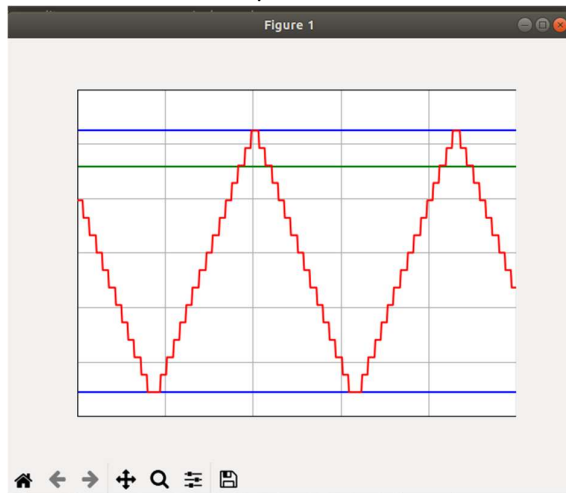


Histograma del error

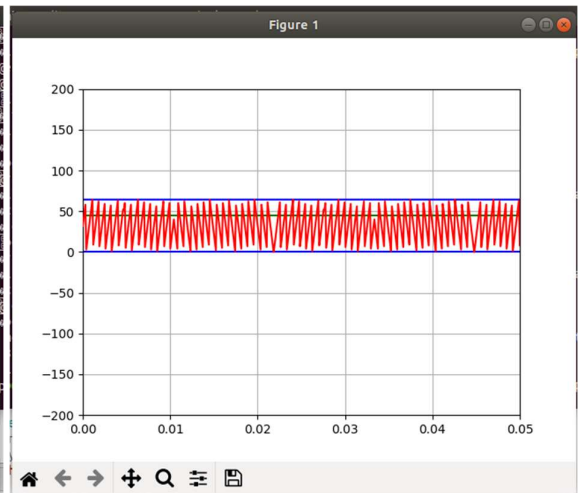


4 bits:

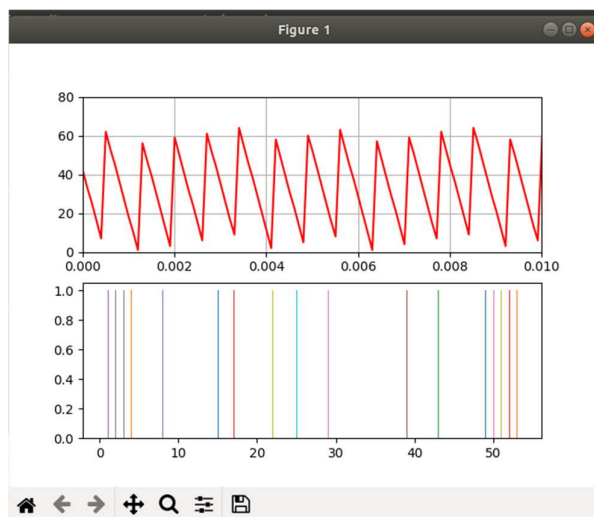
Captura



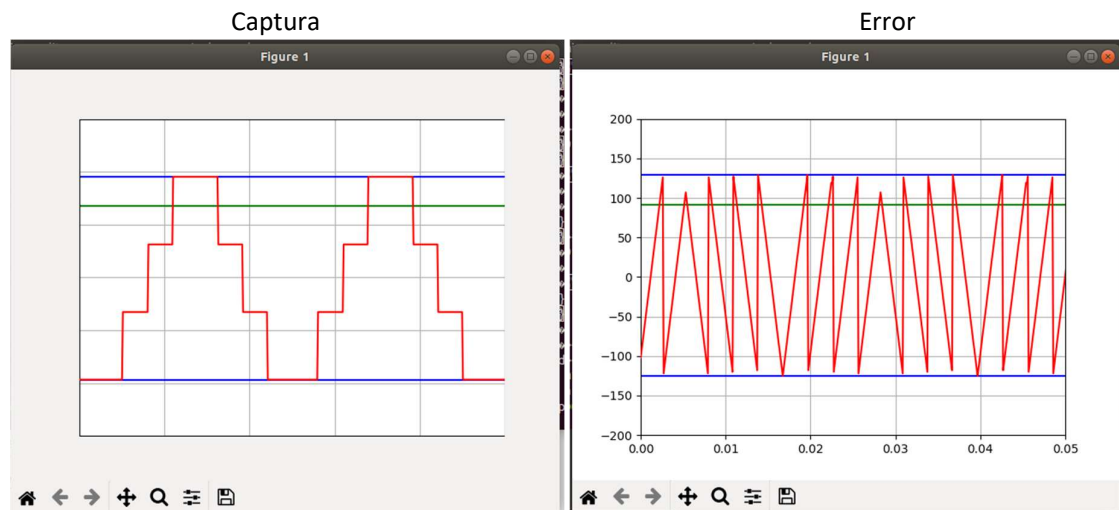
Error



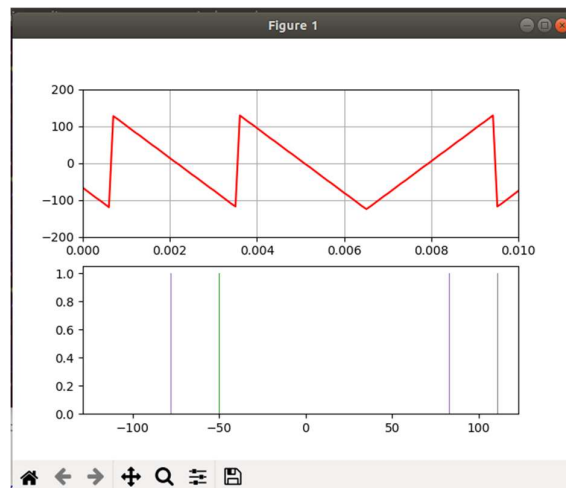
Histograma del error



2 bits



Histograma del error



En el caso de la señal triangular se notó aspectos similares a la señal senoidal, en lo relativo a la pérdida de definición.

Al disminuirse la resolución de la captura de la señal se pierde definición en la misma, y además aumentan los errores.

En general, y al observar los errores, se nota que en el mínimo de resolución (2 bits) los mismos se incrementan y agrupan en menor cantidad de valores. La distribución de valores de error se observa en 4 bits de resolución.

6) Sistema de números.

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: 10111111010000000000000000000000
- Q1.15: No se puede representar un valor a -1.
- Q2.14: 1011000000000000

- 0.001:

- Float: 00111010100000110001001001101111
- Q1.15: 0000000000100001
- Q1.14: 0000000000010000

- -2.001:
 - Float: 11000000000000000001000001100010
 - Q1.15: No se puede representar un valor menor a -1.
 - Q2.14: No se puede representar un valor menor a -2. 

- 204000000:
 - Float: 01001101010000101000110010110000
 - Q1.15: No se puede representar el valor.
 - Q2.14: No se puede representar el valor. 