

TP1 PDS: Fundamentos

Ruido de cuantización

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

- 24 bits = $1,76 + 6,02 \cdot 24 = \mathbf{146,2 \text{ dB}}$ ✓
- 16 bits = $1,76 + 6,02 \cdot 16 = \mathbf{98,08 \text{ dB}}$ ✓
- 10 bits = $1,76 + 6,02 \cdot 10 = \mathbf{61,96 \text{ dB}}$
- 8 bits = $1,76 + 6,02 \cdot 8 = \mathbf{49,92 \text{ dB}}$
- 2 bits = $1,76 + 6,02 \cdot 2 = \mathbf{13,8 \text{ dB}}$

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

- a. Sobremuestreo - "Oversampling" ✓
- b. Consiste en tomar muchas más muestras y posteriormente dividir por el factor de sobremuestreo. Esto se debe a que al aumentar la cantidad de muestras, el espacio entre estas disminuye y por ende el error de cuantificación también lo hace. Por ejemplo, si se tiene un ADC de 10 bits de resolución, se obtiene que el SNR teórico aproximado es de 62 dB. Si se realizara un sobremuestreo con un factor de 4 se obtendrían 6 dB extra, lo equivale a haber aumentado la resolución en 1 bit. ✓ ✓

Esto se debe a que....

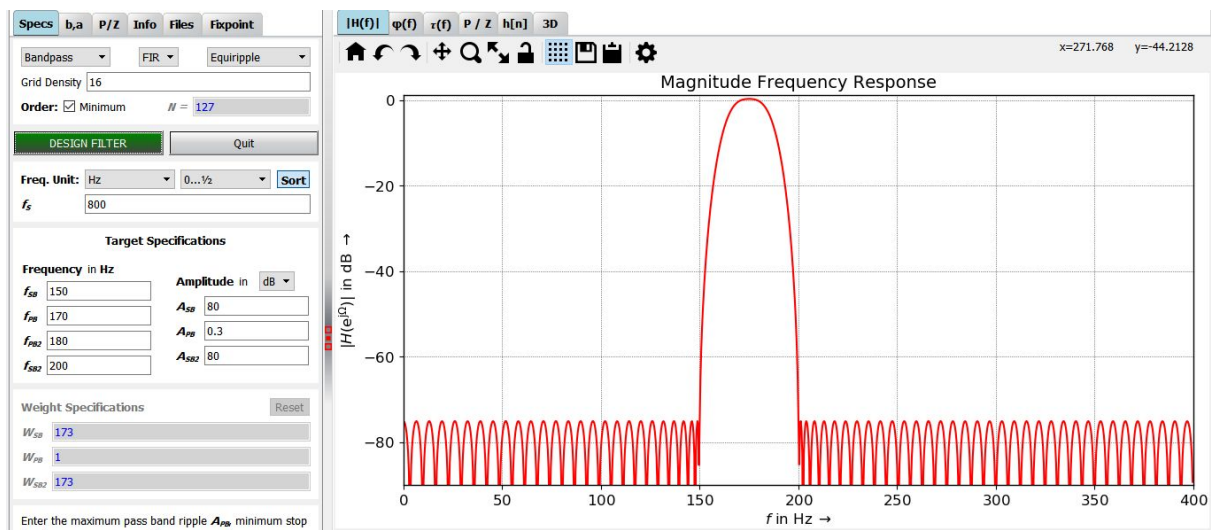
Adicional a esto, de [1] se encontró que la frecuencia de sobremuestreo (f_{sm}) se puede establecer como:

$$f_{sm} = 4^w \cdot f_s$$

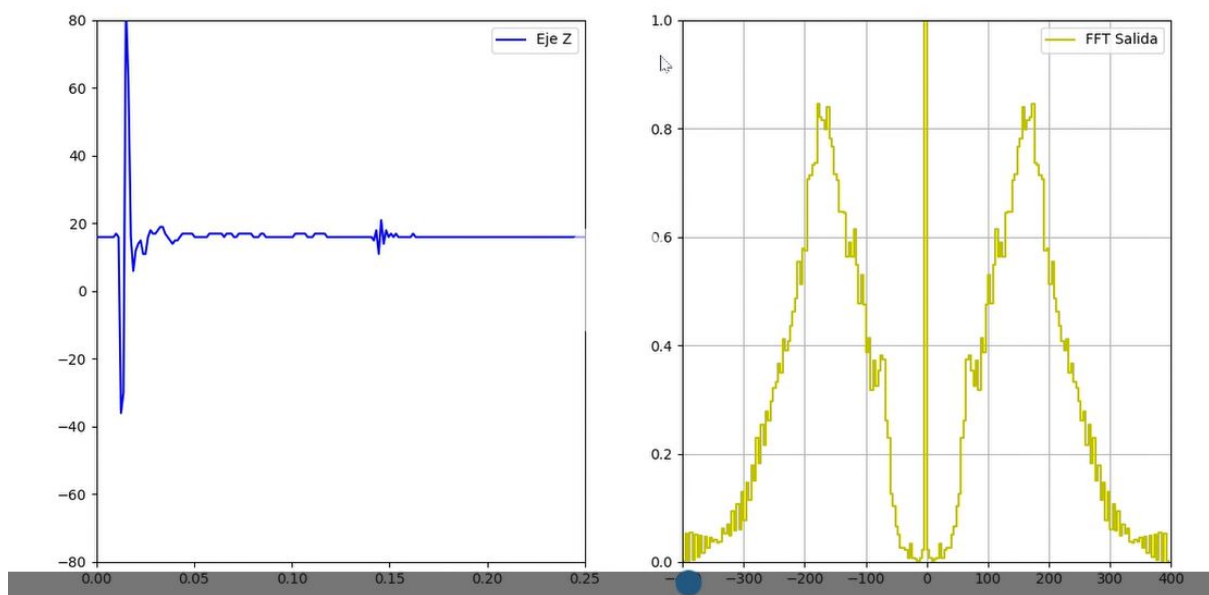
En donde w es la cantidad de bits extra de resolución que se quiere llegar a obtener con el sobremuestreo. Es decir, si se tiene un ADC de 12 bits y se quiere obtener una resolución de 16 bits, se debe sobremuestrear la señal en un factor de 256 (4^4). La desventaja de esta aproximación radica en que la frecuencia efectiva de salida también se ve reducida por este factor de sobremuestreo, además de un ligero aumento de uso de procesador. ✓

Filtro antialias y reconstrucción

Para el proyecto final no se usará un filtro antialias ya que los datos se obtienen digitalmente de un acelerómetro. El diseño del sistema es el siguiente: ✓



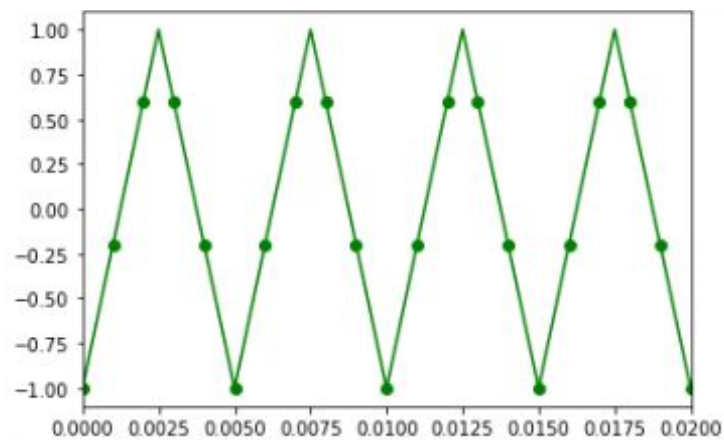
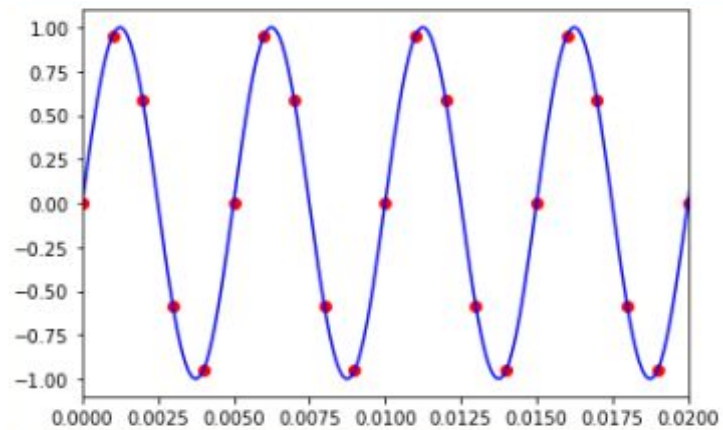
Ya que la señal de entrada tiene la siguiente forma:

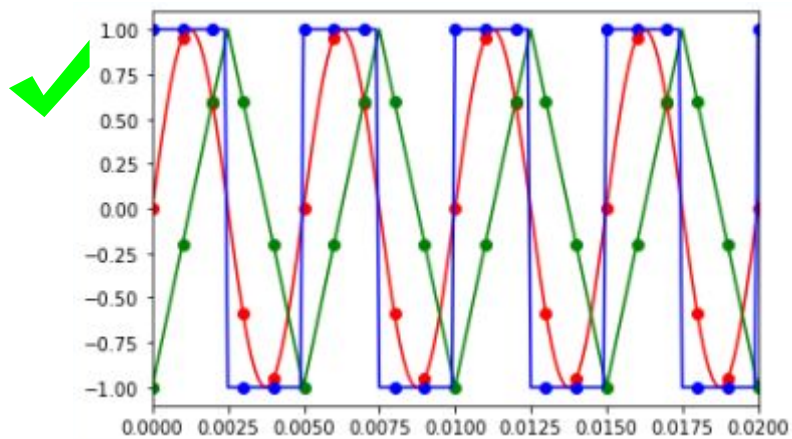
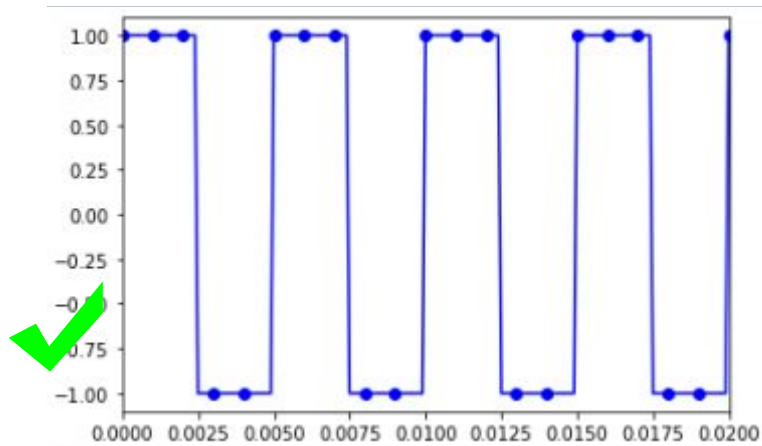


Generación y simulación

Adjunto a este archivo se encuentran los archivos usados para generar la información acá plasmada.

1. Genere un módulo o paquete con al menos las siguientes funciones
 - a. senoidal
 - b. cuadrada
 - c. triangular





Nota: las señales fueron generadas con:

- $f_s = 1000$
- $N = 1000$
- $f = 0.2 \cdot f_s$
- $\text{amp} = 1$
- $\text{fase} = 0$

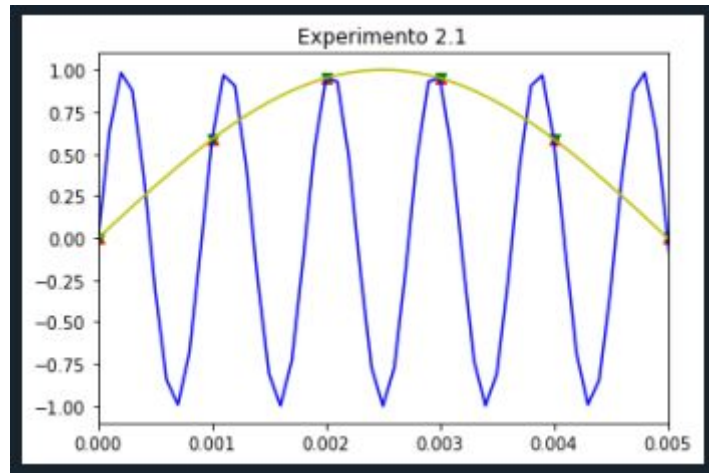
2. Realice los siguientes experimentos con:

- a. $f_s = 1000$
- b. $N = 1000$
- c. $\text{fase} = 0$
- d. $\text{amp} = 1$

- $f_0 = 0.1 \cdot f_s$ y $f_1 = 1.1 \cdot f_s$. Cómo podría diferenciar las senoidales?

Como se en la siguiente figura, las señales muestreadas (en verde y rojo) coinciden en frecuencia y fase. Esto debido a que hay una señal que tiene frecuencia mayor a la frecuencia de muestreo, lo que hace que genere un efecto de alias en la captura. Una de

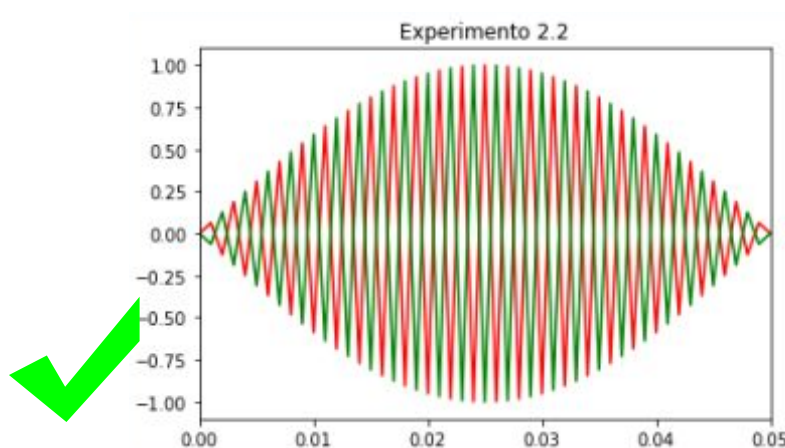
las soluciones para evitar este inconveniente sería usar un filtro antialias a la entrada de la señal para que no entren este tipo de señales. Otra alternativa sería incrementar la frecuencia de muestreo para que se pueda muestrear este tipo de señales que están por encima de la frecuencia de muestreo original, aunque puede que no sea la más óptima.

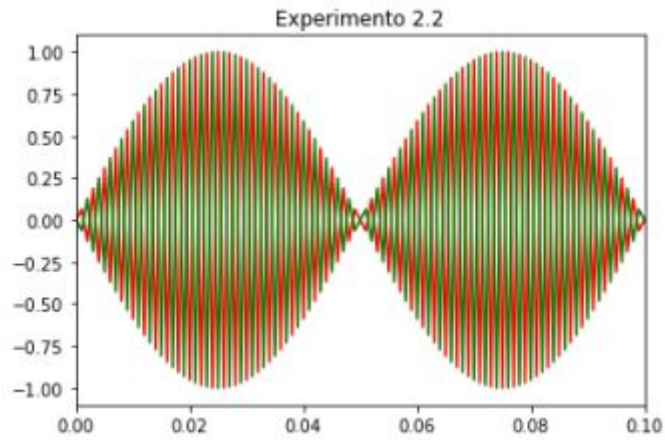


En la figura se ve en azul la señal “original” a $1.1 \cdot f_s$ y en amarillo la señal “original” a $0.1 \cdot f_s$. En rojo se ve la señal muestreada de la señal azul (que tiene frecuencia = $1.1 \cdot f_s$) y en verde la señal muestreada de la señal amarilla (que tiene frecuencia = $0.1 \cdot f_s$).

- $f_0 = 0.49 \cdot f_s$ y $0.51 \cdot f_s$ Cómo es la frecuencia y la fase entre ambas?

El resultado del experimento es el siguiente:





En rojo se ve la señal muestreada a 0.49fs y en verde la señal muestreada a 0.51fs. De estas dos se ve que tienen la misma frecuencia pero con una fase de 180 grados.



Te faltó el punto de adquisición

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

- a. El sistema de punto flotante permite representar números muy grandes o números muy pequeños sin alterar la estructura de la codificación. El sistema de punto fijo está restringido a un rango dinámico mucho menor. ✓
- b. El sistema de punto flotante no tiene una resolución uniforme, el sistema de punto fijo sí. A su vez, el sistema de punto flotante tiene mejor resolución que el sistema de punto fijo. ✓
- c. Con el sistema de punto flotante no se debe saber a priori el rango de las entradas del sistema, con el sistema de punto fijo se debe saber para elegir el n y el m. ✓
- d. El espaciado entre dos números sucesivos es mucho mayor en el sistema de punto fijo, eso conlleva a que un sistema que muestree una señal con punto fijo tenga más posibilidades de sufrir errores de cuantización. ✓
- e. El costo de procesamiento es mayor para el sistema de punto flotante y más aún si no se tiene una unidad de punto flotante dedicada en hardware. ✓
- f. Los costos de sistemas de DSP con punto fijo suelen ser menores a los de punto flotante. Esto se debe a que no se requiere una unidad de punto flotante para maximizar los tiempos de procesamiento, con la ALU interna de cualquier micro se pueden hacer los cálculos ya que sigue siendo un registro plano de n+m bits. ✓
- g. Los sistemas de punto flotantes están optimizados para aplicaciones especializadas y computacionalmente intensivas. Por otro lado, los sistemas de punto fijo están enfocados para manejar grandes volúmenes de datos en aplicaciones de propósito general de forma más eficiente. ✓

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

Número	Float (32b)	Q1.15	Q2.14
0.5	0b 0 0111 1110 000 0000 0000 0000 0000 0000 ✓	0b 0 100 0000 0000 0000 ✓ (16384)	0b 00 10 0000 0000 ✓ 0000 (8192)
-0.5	0b 1 0111 1110 000 0000 0000 0000 0000 0000 ✓	0b 1 100 0000 0000 0000 ✓ (-16384)	0b 11 10 0000 0000 ✓ 0000 (-8192)
-1.25	0b 1 0111 1111 010 0000 0000 0000 0000 0000 ✓	Fuera del rango ✓	0b 10 11 0000 0000 ✓ 0000 (-20480)
0.001	0b 0 0111 0101 000 0011 0001 0010 0110 1111 ✓	0b 0 000 0000 0010 0001 ✓ (33)	0b 00 00 0000 0001 0000 ✓ (16)

-2.001	0b 1 1000 0000 000 0000 0001 0000 0110 0010 ✓✓	Fuera del rango ✓✓	Fuera del rango ✓
20400000 0	0b 0 1001 1010 100 0010 1000 1100 1011 0000 ✓✓	Fuera del rango ✓✓	Fuera del rango ✓✓