

UNIVERSIDAD POLITÉCNICA DE VICTORIA

Nombre del Reporte:

Explorando la frecuencia en una señal sinusoidal continua

Que presenta el alumno:

De la Cruz Rivera Yan Carlo Guadalupe

De la carrera de:

Ingeniería en Mecatrónica

Asignatura:

Adquisición y Procesamiento Digital de Señales

Impartida por:

M.C. CARLOS ANTONIO TOVAR GARCIA

Cd. Victoria, Tamaulipas, Mayo de 2025

Objetivos y descripción de señales

El propósito de esta actividad es entender cómo la frecuencia influye en la forma de una onda senoidal. Al modificar este valor, se podrá observar visualmente cómo cambia el número de ciclos que aparecen en un mismo intervalo de tiempo. La idea no es solo hacer una gráfica, sino también reflexionar sobre lo que significa aumentar o disminuir la frecuencia en una señal que se repite de forma periódica.

La señal que se va a graficar es una onda senoidal continua, que se define con la fórmula:

$$x(t)=A\cdot\sin(2\pi ft)$$

donde:

- $A=1A = 1A=1$ es la amplitud (constante),
- f es la frecuencia en Hertz (valor que el usuario ingresa),
- t representa el tiempo.

Proceso de graficación

1. **Ingreso de datos:**
 - El programa se ejecuta desde consola usando el archivo main.py.
 - El usuario debe escribir la frecuencia como argumento al momento de correr el script.
2. **Generación de la onda:**
 - Se crea un rango de tiempo desde 0 hasta 2 segundos, usando numpy, con muchos puntos (al menos 1000) para que la gráfica sea suave.
3. **Cálculo de la señal:**
 - Con el tiempo y la frecuencia, se calcula el valor de la onda en cada punto del tiempo usando la fórmula matemática.
4. **Visualización:**
 - Se muestra la onda con matplotlib, añadiendo un título que incluya la frecuencia, etiquetas para los ejes y una cuadrícula para facilitar la lectura.
5. **Pruebas sugeridas:**
 - Para observar mejor los efectos de la frecuencia, se recomienda probar con varios valores (por ejemplo: 2 Hz, 5 Hz, 10Hz).

Se usan las siguientes librerías de Python:

- `numpy`: para generar el tiempo y calcular la señal.
- `matplotlib.pyplot`: para hacer la gráfica.
- `sys`: para leer el valor de la frecuencia desde la terminal.

Gráficas generadas

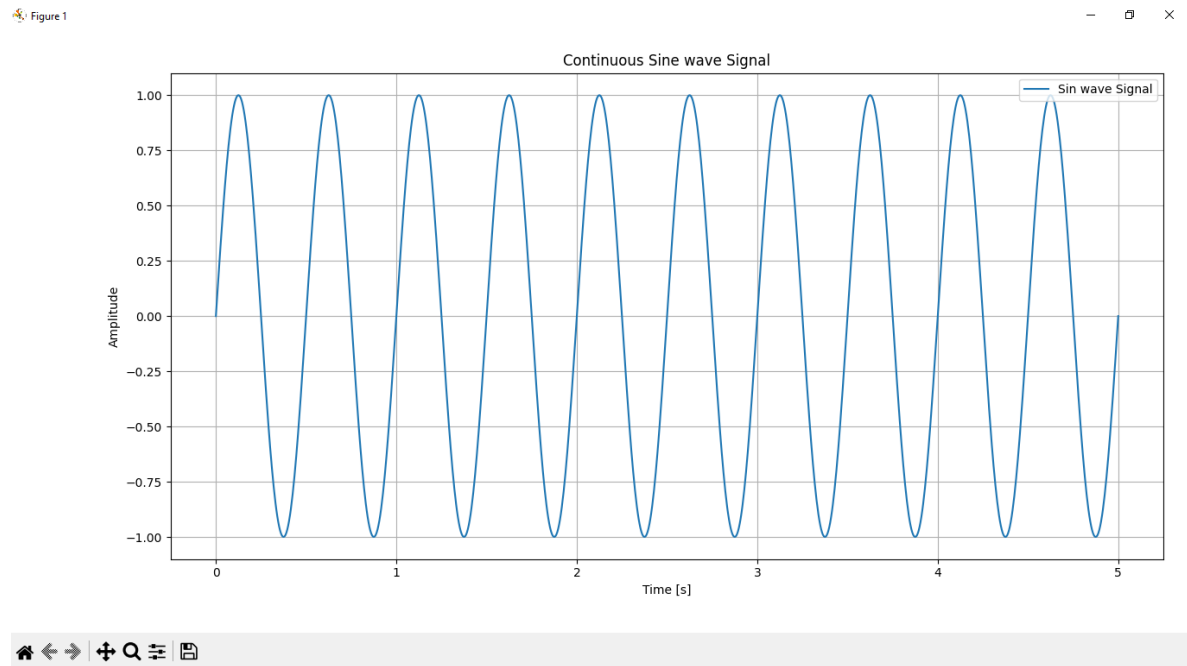


Figura1. Grafica de 2Hz

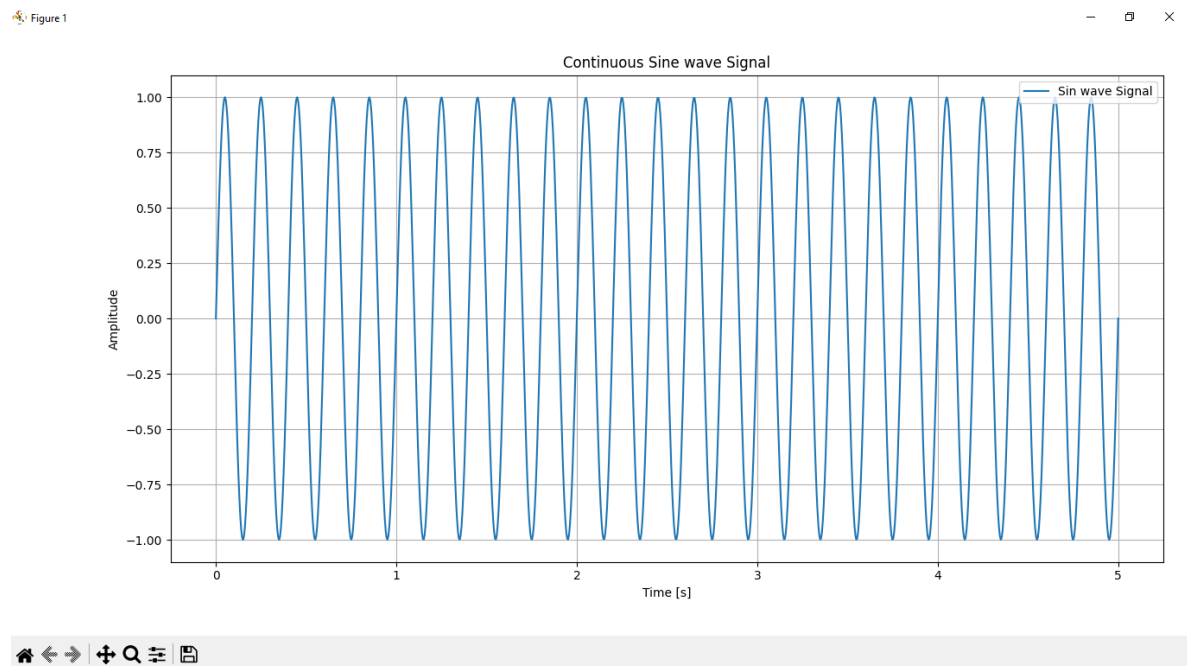


Figura2. Grafica de 5Hz

Figure 1

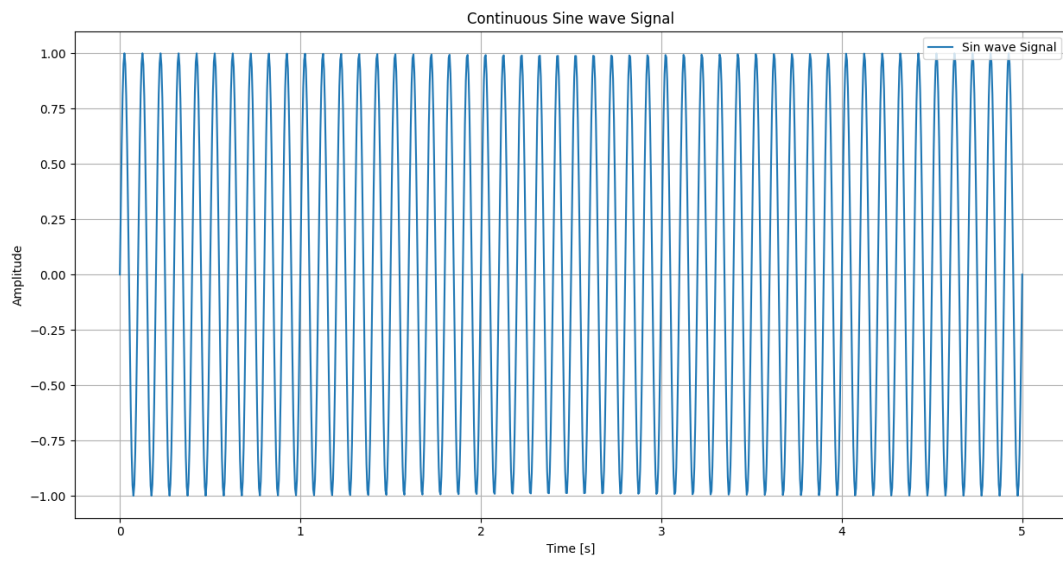


Figura3. Grafica de 10Hz

[Repositorio en GitHub](#)

URL del repositorio: [Yan-carlo-1311/Pds](https://github.com/Yan-carlo-1311/Pds)