

1. Simulaciones Básicas

1.0.1. Simulaciones con Python

Se utilizó el framework de *GNURadio* para programar cada módulo del sistema encerrado en una interfaz gráfica. Esta interfaz tiene la posibilidad de visualizar al mismo tiempo tanto la señal en tiempo como su espectro en cada nodo del sistema. Se puede elegir entre señales sinusoidales, triangulares, 3/2 seno o moduladas AM como entrada.

1.0.2. Simulaciones con LTSpice

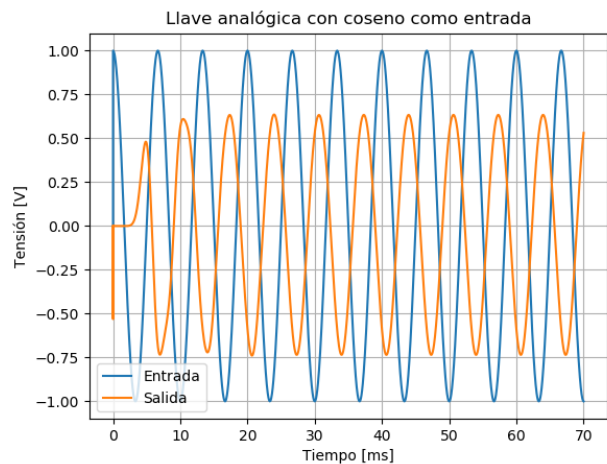
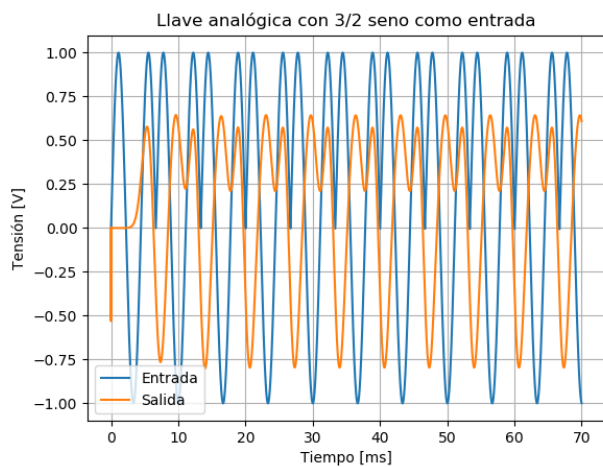
Para ambos sistemas, tanto con la llave analógica seleccionada como para el Sample and Hold elegido, se realizaron las correspondientes simulaciones con las excitaciones deseadas, obteniendo así los resultados presentados a continuación.

Aclarar bajo qué condiciones, es el punto 6a.

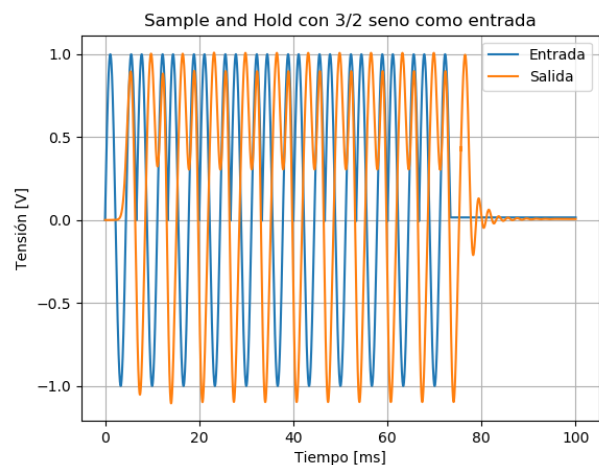
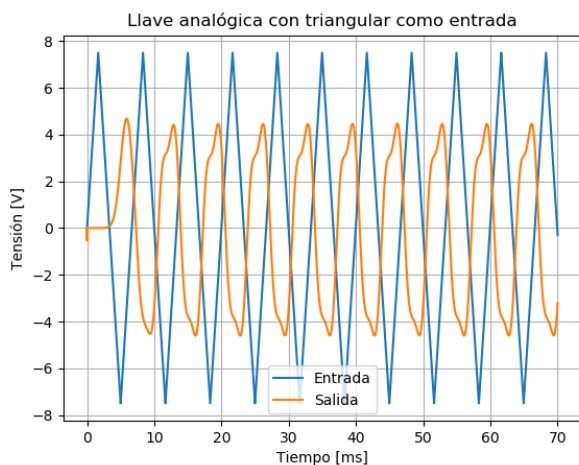
max lf398 vin = $\pm 18\text{ V}$ max 4066 vin = $\pm 7.5\text{ V}$

- 3 2.png - 3 2.png - 3 2.png - 3 2.png - 3 2.png - 3 2.png
- 3 2.png - 3 2.png - 3 2.png - 3 2.png - 3 2.png

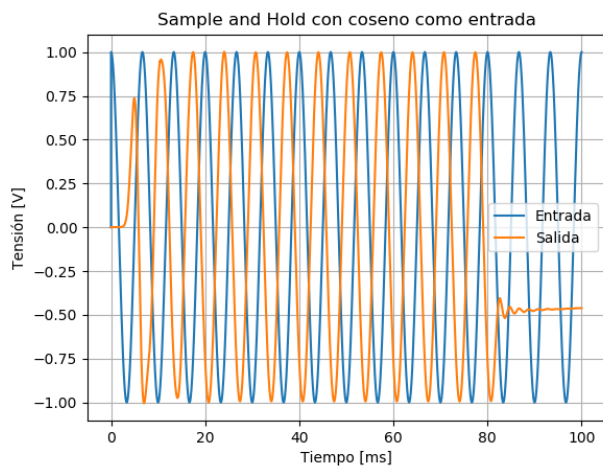
- Cos.png - Cos.png - Cos.png - Cos.png - Cos.png -
Cos.png - Cos.png - Cos.png - Cos.png - Cos.png -
Cos.png



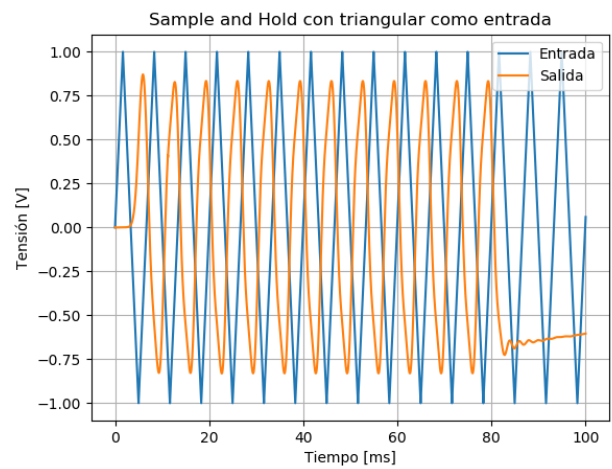
- Tri.png - Tri.png - Tri.png - Tri.png - Tri.png - Tri.png - 3 2.png - 3 2.png - 3 2.png - 3 2.png - 3 2.png - 3 2.png
- Tri.png - Tri.png - Tri.png - Tri.png - Tri.png



- Cos.png - Cos.png - Cos.png - Cos.png - Cos.png -
 Cos.png - Cos.png - Cos.png - Cos.png - Cos.png -
 Cos.png



- Tri.png - Tri.png - Tri.png - Tri.png - Tri.png - Tri.png
 - Tri.png - Tri.png - Tri.png - Tri.png - Tri.png



Nuevamente, se realizaron las mismas simulaciones variando...

Aclarar bajo qué condiciones, es el punto 6b.