

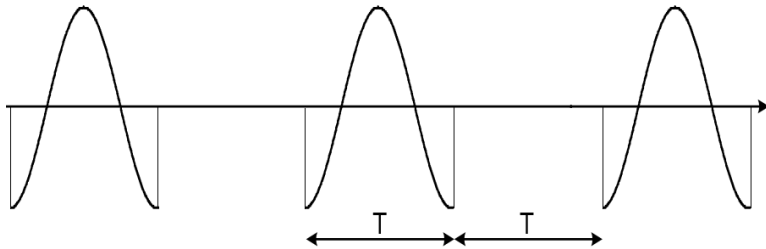
SEÑALES ELÉCTRICAS

Parcial N°2 06/06/2025 **Tema 2** Alumno:.....

P1. Se tiene un amplificador de con 15dB de ganancia, 500MHz de ancho de banda y $F=10\text{dB}$; también un tramo de 60m de cable coaxial con atenuación de 12dB. La temperatura de ruido de la señal es $T_i=400^\circ\text{K}$. [2p]

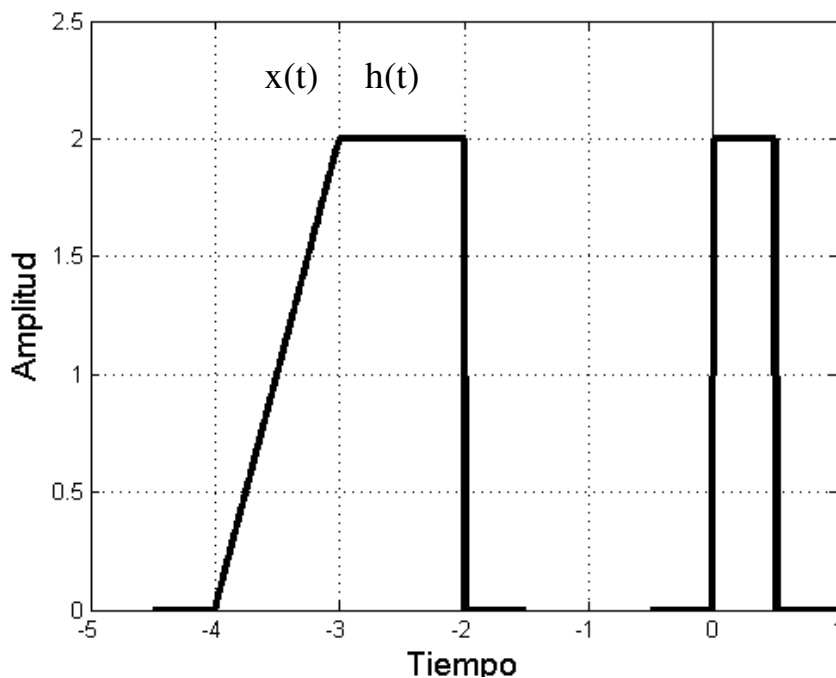
- a) Encuentre la temperatura de ruido equivalente y el factor de ruido para la configuración más conveniente de conexión. (¿Cable-Amp ó Amp-Cable?)
- b) La potencia de ruido que se vuelca sobre la carga suponiendo todas las uniones de los componentes con impedancias adaptadas.
- c) La potencia de señal (en la entrada del conjunto) necesaria para tener una SNR de 40dB sobre la carga.

P2. Para la señal periódica de la figura, determine cuáles de las seis primeras armónicas tendrá amplitud 0. (Dibuje el espectro de la señal al menos hasta la sexta armónica). [3p]



Sugerencia: considere que la forma de onda proviene del producto de una señal **cosenoidal** y una cuadrada con extremos entre cero y uno.

P3. La respuesta a un impulso aplicado en $t=0$ de un sistema LTI es $h(t)$, si el sistema es excitado por una señal $x(t)$ como la de la figura. Encuentre respuesta $R(t)$. [2,5p]



sigue \longrightarrow

P4. Se desea calcular numéricamente los primeros 30 coeficientes de la Serie de Fourier de una señal periódica formada por un pulso rectangular. La señal tiene una $f_s = 1500$ Hz, un $T_0 = 0.80$ s, Duty Cycle $D = 0.1$ y una Amplitud: $A = 1.0$. **Complete y ordene** las siguientes líneas de código para calcular los coeficientes de la serie de Fourier de la señal descrita: [2,5p]

```

___ t = .....(0, T, N, endpoint=False)
___ A = 1.0
___ coef = get_fourier_coef(30, senal, t)
___ def get_fourier_coef(n, signal, t):
    coef = []
    for k in range(n):

        exponencial = .....(-1j*2*np.pi*k*t / T)
        ck = (1 / T) * np.sum(signal * exponencial) * (T / N)
        coef.append(ck)
    return np.array(coef)

___ senal = .....([pulso, np.zeros(N-duracion_pulso)])
___ fs = 1500
___ duracion_pulso = int(D * N)
___ D = 0.1
___ N = int(fs * T)
___ pulso = A * .....(duracion_pulso)
___ T = 0.80

```

*Opciones de funciones a utilizar `np.arange()`, `np.linspace()`, `np.stack()`, `np.ones()`, `np.concatenate()`, `np.zeros()`, `np.full()`.