

```

# -*- coding: utf-8 -*-
"""
Created on Sun May  5 17:08:34 2019
Filtro FIR (Prueba1)
Tecnológico Nacional de México
Tecnológico de Estudios Superiores de Ixtapaluca
División de Ingeniería Electrónica
@author: M. en C. Rogelio Manuel Higuera Gonzalez
"""

import numpy as np
from scipy import signal
import matplotlib.pyplot as plt
f1=50 #Frecuencia de la señal 1
f2=200 #Frecuencia de la señal 2
A1=1 #Amplitud de la señal 1
A2=0.5 #Amplitud de la señal 2
w1=2*np.pi*f1 #Frecuencia angular señal 1
w2=2*np.pi*f2 #Frecuencia angular de la señal 2
T=1/f1 #Periodo de la señal 1
n=10000 #Numero de muestras
t=np.linspace(0,250*T,n) #Vector de tiempo
V1=A1*np.sin(w1*t) #Señal 1
V2=A2*np.sin(w2*t) #Señal 2
Vin=V1+V2
fs=1/(t[10]-t[9]) #Frecuencia de muestreo
fc1=100 #Frecuencia de corte 1
fc2=300 #Frecuencia de corte 2
wn1=fc1/(fs/2) #Frecuencia Normalizada 1
wn2=fc2/(fs/2) #Frecuencia Normalizada 2
N=51 #Orden del Filtro
taps1 = signal.firwin(N,[wn1,wn2],pass_zero=False, window=('barthann'))
taps2 = signal.firwin(N,[wn1,wn2],pass_zero=False, window=('bartlett'))
taps3 = signal.firwin(N,[wn1,wn2],pass_zero=False, window=('blackman'))
taps4 = signal.firwin(N,[wn1,wn2],pass_zero=False, window=('bohman'))
taps5 = signal.firwin(N,[wn1,wn2],pass_zero=False, window=('hanning'))
w_1, h_1 = signal.freqz(taps1, worN=8000)
w_2, h_2 = signal.freqz(taps2, worN=8000)
w_3, h_3 = signal.freqz(taps3, worN=8000)
w_4, h_4 = signal.freqz(taps4, worN=8000)
w_5, h_5 = signal.freqz(taps5, worN=8000)
fig,axes1=plt.subplots()
axes1.plot(w_1, 20 * np.log10(np.absolute(h_1)), label='Barthann', linewidth=2)
axes1.plot(w_2, 20 * np.log10(np.absolute(h_2)), label='Bartlett', linewidth=2)
axes1.plot(w_3, 20 * np.log10(np.absolute(h_3)), label='Blackman', linewidth=2)
axes1.plot(w_4, 20 * np.log10(np.absolute(h_4)), label='Bohman', linewidth=2)
axes1.plot(w_5, 20 * np.log10(np.absolute(h_5)), label='Hanning', linewidth=2)
axes1.set_ylabel('Normalized magnitude [dB]')
axes1.set_xlabel('Normalized frequency [cycles per sample]')
plt.legend(bbox_to_anchor=(0.18,0.31),loc=2,borderaxespad=0.1)
plt.grid(True)
plt.savefig('FIRven1.eps',dpi=1000,bbox_inches='tight')

```