BMEP 5704

Digital Lab 3

Quincy Gu

05/21/2021

Section I – Chebyshev1

1. Graph

Chart, line chart

Description automatically generated

1. Code

function Chebyshev1(Fs)

% Fs - sample rate, default be 500Hz

% passband 0-30 Hz

Wp = 30 / (Fs / 2);

% stopband 100Hz

Ws = 100 / (Fs / 2);

% 3 dB ripple in passband

% 80 dB attenuation in stopband

% cheb1ord function returns the lowest order of n of the Chebyshev Type

% I filter that loses no more than Rp dB in the passband and has at

% least Rs dB of attenuation in the stopband. the scalar (or vector) of

% corresponding cutoff frequencies Wp is also returned

[n, Wn] = cheb1ord(Wp, Ws, 3, 80);

% cheby1 function returns the transfer function coefficients of an

% nth-order lowpass digital Chebyshev Type I filter with normalized

% passband edge frequency Wp and Rp decibels of peak-to-peak passband

% ripple

[z, p, k] = cheby1(n, 3, Wn);

% zp2sos function fins a 2nd-order section matrix Sos with gain g that

% is equivalent to the transfer function H(z) whose n zeros, m poles,

% and scalar gain are specified in z, p, and k.

Sos = zp2sos(z, p, k);

fig = figure;

% graph the frequency response

freqz(Sos, 512, 500);

% add figure captions

title(sprintf('n=%d Pole Chebyshev 1 Lowpass Filter', n));

% save the figure

saveas(fig, 'Pole Chebyshev 1 Lowpass Filter.png');

Section II – Chebyshev2

1. Graph

A picture containing engineering drawing

Description automatically generated

1. Code

function Chebyshev2(Fs)

% Fs - sample rate, default be 500Hz

% passband 0-30 Hz

Wp = 30 / (Fs / 2);

% stopband 100Hz

Ws = 100 / (Fs / 2);

% 3 dB ripple in passband

% 80 dB attenuation in stopband

% cheb2ord function returns the lowest order of n of the Chebyshev Type

% II filter that loses no more than Rp dB in the passband and has at

% least Rs dB of attenuation in the stopband. the scalar (or vector) of

% corresponding cutoff frequencies Wp is also returned

[n, Wn] = cheb2ord(Wp, Ws, 3, 80);

% cheby2 function returns the transfer function coefficients of an

% nth-order lowpass digital Chebyshev Type II filter with normalized

% passband edge frequency Wp and Rp decibels of peak-to-peak passband

% ripple

[z, p, k] = cheby2(n, 80, Wn);

% zp2sos function fins a 2nd-order section matrix Sos with gain g that

% is equivalent to the transfer function H(z) whose n zeros, m poles,

% and scalar gain are specified in z, p, and k.

Sos = zp2sos(z, p, k);

fig = figure;

% graph the frequency response

freqz(Sos, 512, 500);

% add figure captions

title(sprintf('n=%d Pole Chebyshev 2 Lowpass Filter', n));

% save the figure

saveas(fig, 'Pole Chebyshev 2 Lowpass Filter.png');

Section III – Elliptical

1. Graph

A picture containing chart

Description automatically generated

1. Code

function Elliptical(Fs)

% Fs - sample rate, default be 500Hz

% passband 0-30 Hz

Wp = 30 / (Fs / 2);

% stopband 100Hz

Ws = 100 / (Fs / 2);

% 3 dB ripple in passband

% 80 dB attenuation in stopband

% ellipord returns the lowest order, n, of the digital elliptic filter

% with no more than Rp dB of passband ripple and at least Rs dB of

% attenuation in the stopband

[n, Wn] = ellipord(Wp, Ws, 3, 80);

% ellip function returns the transfer function coefficients of an

% 8th-order lowpass digital elliptic filter with normalized passband

% edge ferquency Wp.

[z, p, k] = ellip(n, 3, 80, Wn);

% zp2sos function fins a 2nd-order section matrix Sos with gain g that

% is equivalent to the transfer function H(z) whose n zeros, m poles,

% and scalar gain are specified in z, p, and k.

Sos = zp2sos(z, p, k);

fig = figure;

% graph the frequency response

freqz(Sos, 512, 500);

% add figure captions

title(sprintf('n=%d Pole Elliptical Lowpass Filter', n));

% save the figure

saveas(fig, 'Pole Elliptical Lowpass Filter.png');

Section IV – yulewalk

1. Graph

Diagram

Description automatically generated

1. Code

function yulewalk(Fs, N)

% Fs - sample rate, default be 500Hz

% N - order of the filter, default be 8

fig = figure;

% sequence of 251 points representing the filter and the frequency

F = linspace(0, 1, 251);

% stopband from 55-65 Hz

M = [ones(1, 55), linspace(1, 0, 10), zeros(1, 251-65)];

% yulewalk is a function that returns the transfer function

% coefficients of an 8th-order IIR filter whose frequency magnitude

% response approximately matches the values given in F and M

[B, A] = yulewalk(N, F, M);

% graph frequency response

[h, w] = freqz(B, A, Fs);

plot(F\*Fs/2, 20\*log10(M), w\*Fs/(2\*pi), 20\*log10(abs(h)), '--');

% figure captions and other components

legend('Ideal', 'yulewalk Designed');

title('Comparison of Frequency Response Magnitudes');

xlabel('Frequency (Hz)');

ylabel('Magnitude (dB)');

% save the figure

saveas(fig, 'CFRM.png');