

Contents

- [filter 1 \(analog elliptic\)](#)
- [filter 2 \(analog chebyshev 1\)](#)
- [filter 3 \(digital elliptic\)](#)
- [filter 4 \(digital chebyshev 1\)](#)

```
% SPDX-License-Identifier: GPL-3.0-or-later
%
% ECE211 problem_set05.m
% Copyright (C) 2024 Aidan Cusa <aidancusa@gmail.com>

clc;    % clear command window
clear; % clear all variables from current workspace
close all;

fs = 10e6; % sampling rate of digital filters, 10 MHz
fn = fs / 2; % nyquist band width

Wp = [1.5e6, 2e6]; % passband
Rp = 2; % peak to peak ripple in pass band

Ws = [1.4e6, 2.2e6]; % stopband
Rs = 40; % peak to peak ripple in stop band
```

filter 1 (analog elliptic)

```
% get order and normalized band
[n_eA, Wn] = ellipord(Wp, Ws, Rp, Rs, 's');
% get zeros, poles, and scaling factor k
[z_eA, p_eA, k_eA] = ellip(n_eA, Rp, Rs, Wn, 's');
% get transfer function given zpk
[b_eA, a_eA] = zp2tf(z_eA, p_eA, k_eA);

elliptic_analog_order = 2 * n_eA

% get magnitude and phase from freqs
[h_eA, wout_eA] = freqs(b_eA, a_eA, 1e3);
mag_eA = 20 * log10(abs(h_eA));
phase_eA = unwrap(angle(h_eA));
phasedeg_eA = phase_eA*180/pi;

figure;
subplot(2, 2, 2);
plot(wout_eA, mag_eA);

grid on;
xlim([0, fn]); % x limits from DC to Nyquist Bandwidth
ylim([-65, 5]);
xticklabels(0:5);
xlabel('Frequency [MHz]');
ylabel('Magnitude [dB]');
title("Magnitude Response");
```

```

subplot(2, 2, 4);
plot(wout_eA, phasedeg_eA);

grid on;
xlim([0, fn]); % x limits from DC to Nyquist Bandwidth
xticklabels(0:5);
xlabel('Frequency [MHz]');
ylabel('Phase [Degrees]');
title("Phase Response");

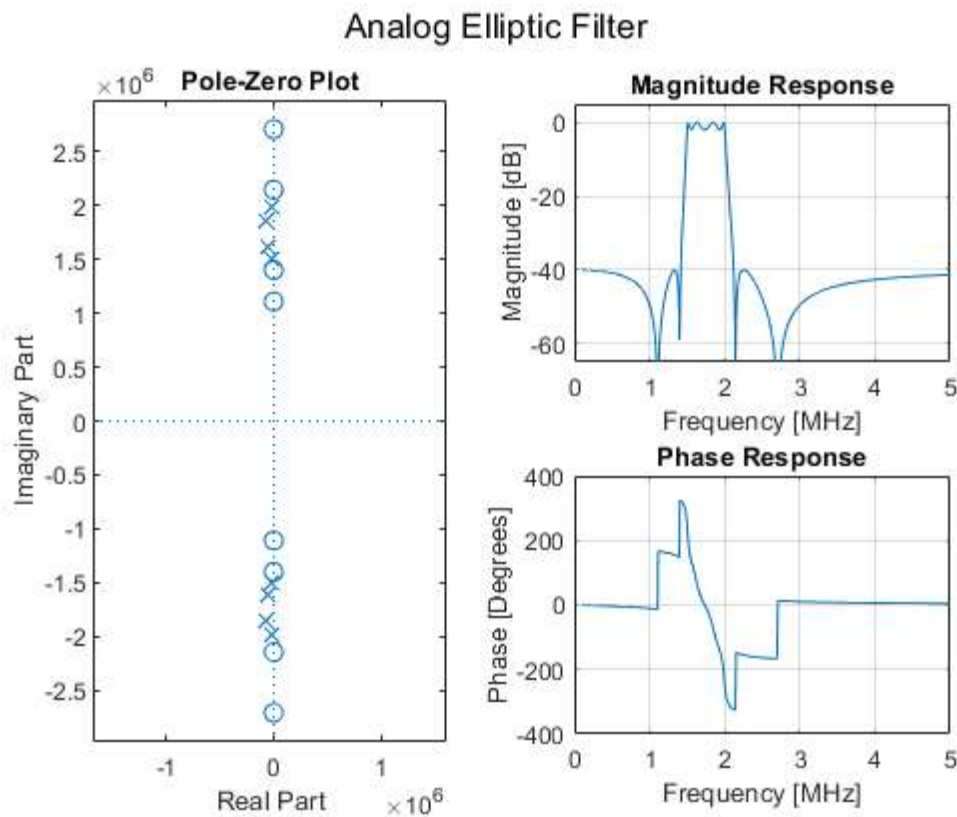
subplot(2, 2, [1,3]);
zplane(z_eA, p_eA);

sgtitle("Analog Elliptic Filter");

```

```
elliptic_analog_order =
```

8



filter 2 (analog chebyshev 1)

```

% order of cheb1ord and normalized band
[n_cA, Wn] = cheb1ord(Wp, Ws, Rp, Rs, 's');
% get zeros, poles, and scaling factor k
[z_cA, p_cA, k_cA] = cheby1(n_cA, Rp, Wn, 's');
% get transfer function given zpk
[b_cA, a_cA] = zp2tf(z_cA, p_cA, k_cA);

```

```

cheby1_analog_order = 2 * n_cA

[h_cA, wout_cA] = freqs(b_cA, a_cA, 1e3);
mag_cA = 20 * log10(abs(h_cA));
phase_cA = unwrap(angle(h_cA));
phasedeg_cA = phase_cA*180/pi;

figure;

subplot(2, 2, 2);
plot(wout_cA, mag_cA);

grid on;
xlim([0, fn]); % x limits from DC to Nyquist Bandwidth
ylim([-300, 5]);
xticklabels(0:5);
xlabel('Frequency [MHz]');
ylabel('Magnitude [dB]');
title("Magnitude Response");

subplot(2, 2, 4);
plot(wout_cA, phasedeg_cA);

grid on;
xlim([0, fn]); % x limits from DC to Nyquist Bandwidth
xticklabels(0:5);
xlabel('Frequency [MHz]');
ylabel('Phase [Degrees]');
title("Phase Response");

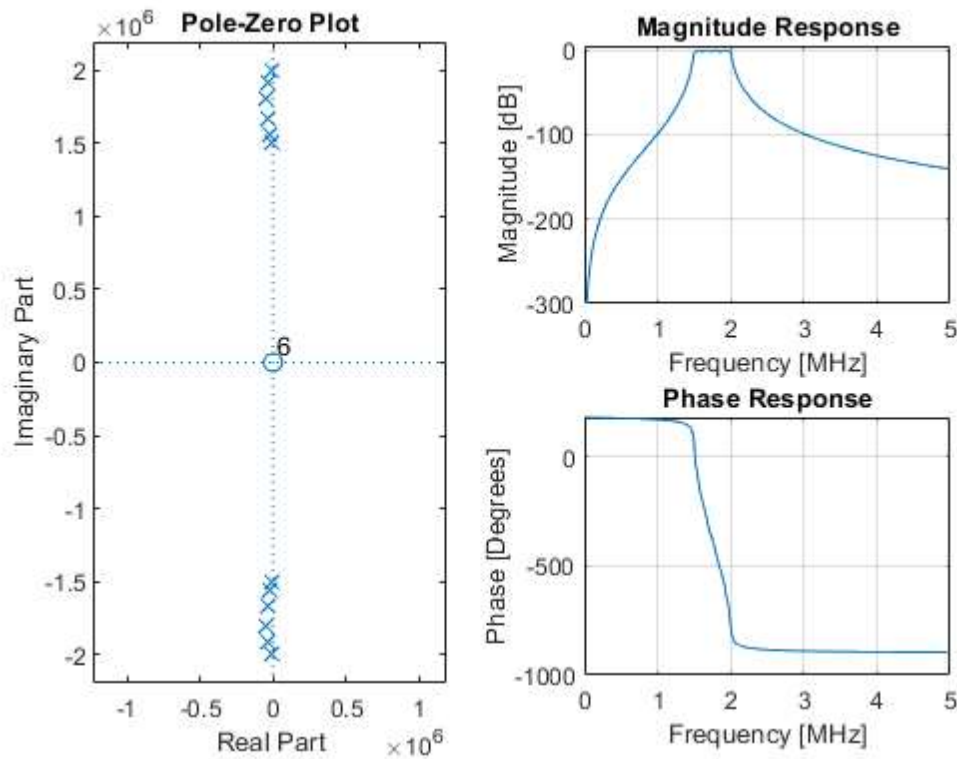
subplot(2, 2, [1,3]);
zplane(z_cA, p_cA);

sgtitle("Analog Chebyshev I Filter");

```

```
cheby1_analog_order =
```

Analog Chebyshev I Filter



filter 3 (digital elliptic)

```
% get order and normalized band to nyquist bandwidth
[n_eD, Wn] = ellipord(Wp / fn, Ws / fn, Rp, Rs);
% get zeros, poles, and scaling factor k
[z_eD, p_eD, k_eD] = ellip(n_eD, Rp, Rs, Wn);
% get transfer function given zpk
[b_eD, a_eD] = zp2tf(z_eD, p_eD, k_eD);

elliptic_digital_order = 2 * n_eD

w = linspace(0, pi, 10000);

[h_eD, wout_eD] = freqz(b_eD, a_eD, w);
mag_eD = 20 * log10(abs(h_eD));
phase_eD = unwrap(angle(h_eD));
phasedeg_eD = phase_eD*180/pi;

f_dig = wout_eD * fs / (2 * pi);

figure;
subplot(2, 2, 2);
plot(f_dig, mag_eD);

grid on;
xlim([0, fn]);
ylim([-65, 5]);
xticklabels(0:5);
xlabel('Frequency [MHz]');
ylabel('Magnitude [dB]');
title("Magnitude Response");
```

```

subplot(2, 2, 4);
plot(f_dig, phasedeg_eD);

grid on;
xlim([0, fn]);
xticklabels(0:5);
xlabel('Frequency [MHz]');
ylabel('Phase [Degrees]');
title("Phase Response");

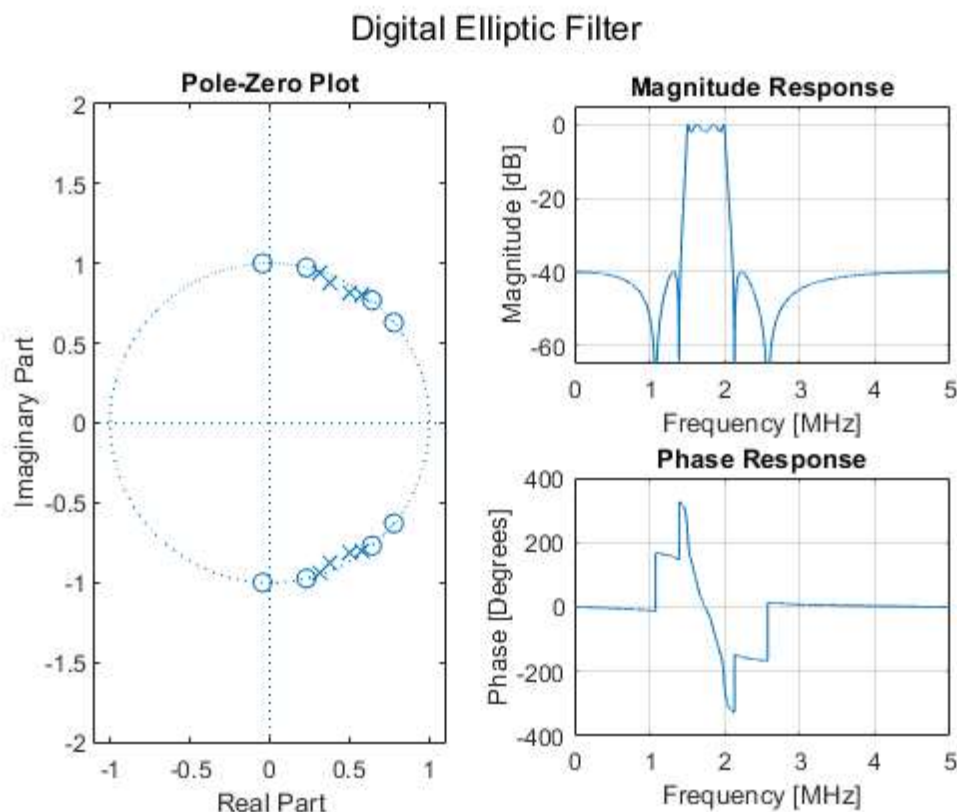
subplot(2, 2, [1,3]);
zplane(z_eD, p_eD);

sgtitle("Digital Elliptic Filter");

```

```
elliptic_digital_order =
```

8



filter 4 (digital chebyshev 1)

```

% order of cheb1ord and normalized band to nyquist bandwidth
[n_cD, Wn] = cheb1ord(Wp / fn, Ws / fn, Rp, Rs);
% get zeros, poles, and scaling factor k
[z_cD, p_cD, k_cD] = cheby1(n_cD, Rp, Wn);
% get transfer function given zpk
[b_cD, a_cD] = zp2tf(z_cD, p_cD, k_cD);

```

```

cheby1_digital_order = 2 * n_cD

w = linspace(0, pi, 10000);

[h_cD, wout_cD] = freqz(b_cD, a_cD, w);
mag_cD = 20 * log10(abs(h_cD));
phase_cD = unwrap(angle(h_cD));
phasedeg_cD = phase_cD*180/pi;

f_dig = wout_cD * fs / (2 * pi);

figure;
subplot(2, 2, 2);
plot(f_dig, mag_cD);

grid on;
xlim([0, fn]);
ylim([-300, 5]);
xticklabels([0:5]);
xlabel('Frequency [MHz]');
ylabel('Magnitude [dB]');
title("Magnitude Response");

subplot(2, 2, 4);
plot(f_dig, phasedeg_cD);

grid on;
xlim([0, fn]);
ylim([-1500, 25]);
xticklabels([0:5]);
xlabel('Frequency [MHz]');
ylabel('Phase [Degrees]');
title("Phase Response");

subplot(2, 2, [1,3]);
zplane(z_cD, p_cD);

sgtitle("Digital Chebyshev I Filter");

```

```
cheby1_digital_order =
```

Digital Chebyshev I Filter

