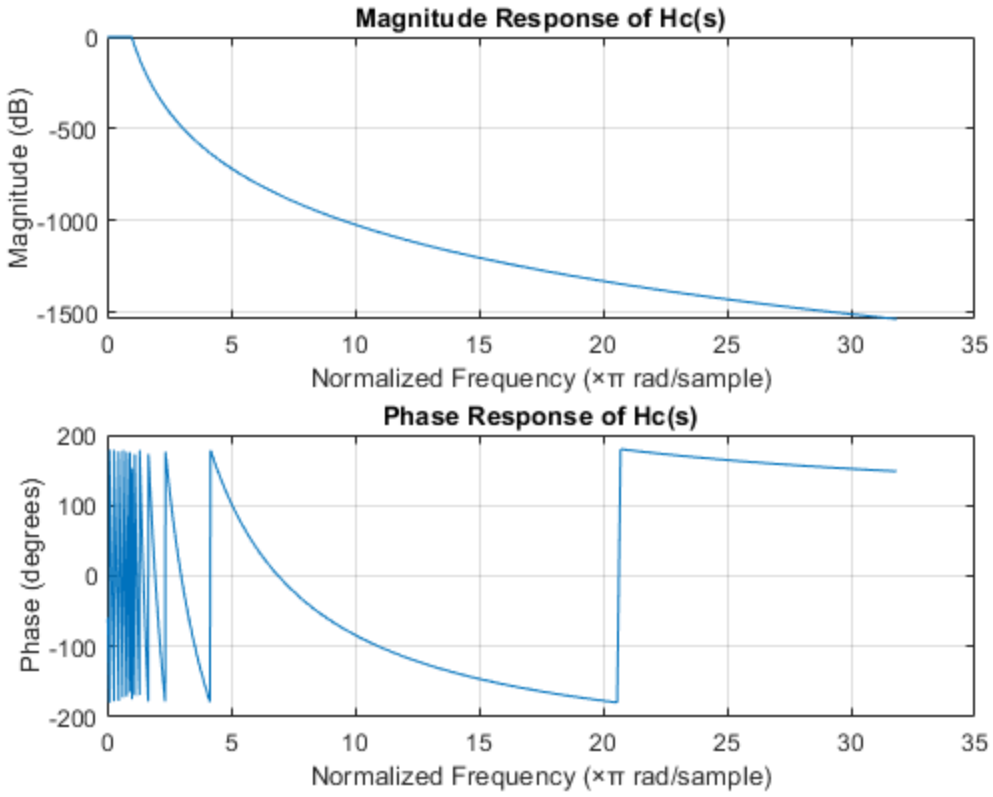
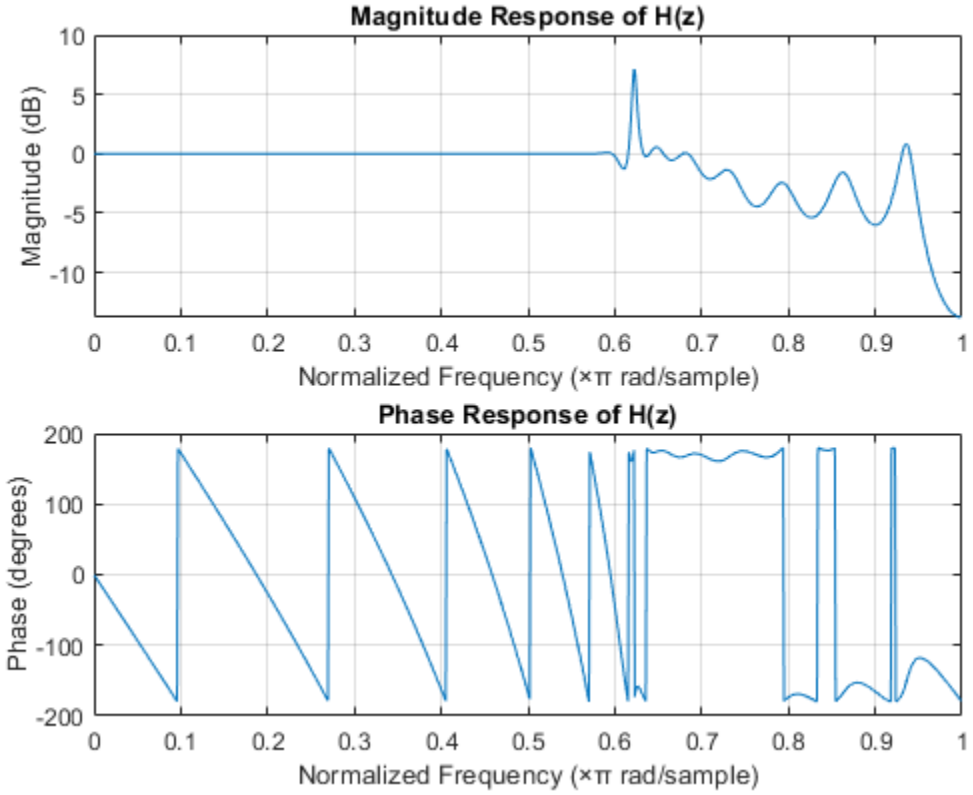
```
% Given specifications
n = 51;
Omega_c = 3.14315;
[z, p, k] = buttap(n);  %returns poles and zeros and gain of Butterworth
filter with parameter n
p = p * Omega_c;
k = k * Omega_c^n;

[num, den] = zp2tf(z, p, k); %convert to transfer function form

Hc = tf(num, den); %transfer function of analog butterworth filter
[Hc_freq, W_c] = freqs(num, den, 1024);
magnitude = 20*log10(abs(Hc_freq));
phase = angle(Hc_freq) * (180/pi);

figure;
subplot(2,1,1); % Magnitude plot
plot(W_c/pi, magnitude);
title('Magnitude Response of Hc(s)');
xlabel('Normalized Frequency (×π rad/sample)');
ylabel('Magnitude (dB)');
grid on;

subplot(2,1,2); % Phase plot
plot(W_c/pi, phase);
title('Phase Response of Hc(s)');
xlabel('Normalized Frequency (×π rad/sample)');
ylabel('Phase (degrees)');
grid on;

fs = 1;
T = 1/fs;
[num_z, den_z] = bilinear(num, den, fs); % Convert Hc(s) to H(z) using
bilinear transform
Hz = tf(num_z, den_z, T);
disp(Hz);
[H_freq, W] = freqz(num_z, den_z, 1024); % Compute frequency response
magnitude = 20*log10(abs(H_freq)); % Convert to dB
phase = angle(H_freq) * (180/pi); % Convert to degrees

figure;
subplot(2,1,1); % Magnitude plot
plot(W/pi, magnitude);
title('Magnitude Response of H(z)');
xlabel('Normalized Frequency (×π rad/sample)');
ylabel('Magnitude (dB)');
grid on;

subplot(2,1,2); % Phase plot
plot(W/pi, phase);
title('Phase Response of H(z)');
xlabel('Normalized Frequency (×π rad/sample)');
```

```
ylabel('Phase (degrees)');
grid on;
```

```
  tf with properties:

      Numerator: {[2.4065e-09 1.2273e-07 3.0683e-06 … ] (1×52 double)}
    Denominator: {[1 14.2001 104.7289 529.1458 2.0457e+03 … ] (1×52 double)}
       Variable: 'z'
        IODelay: 0
     InputDelay: 0
    OutputDelay: 0
      InputName: {''}
      InputUnit: {''}
     InputGroup: [1×1 struct]
     OutputName: {''}
     OutputUnit: {''}
    OutputGroup: [1×1 struct]
          Notes: [0×1 string]
       UserData: []
           Name: ''
             Ts: 1
       TimeUnit: 'seconds'
   SamplingGrid: [1×1 struct]
```





Published with MATLAB® R2024b