AIN SHAMS UNIVERSITY

FACULTY OF ENGINEERING

Specialized Credit Hours Engineering Programs

ECE Program

ECE 451s : Digital Signal Processing - Fall 2025

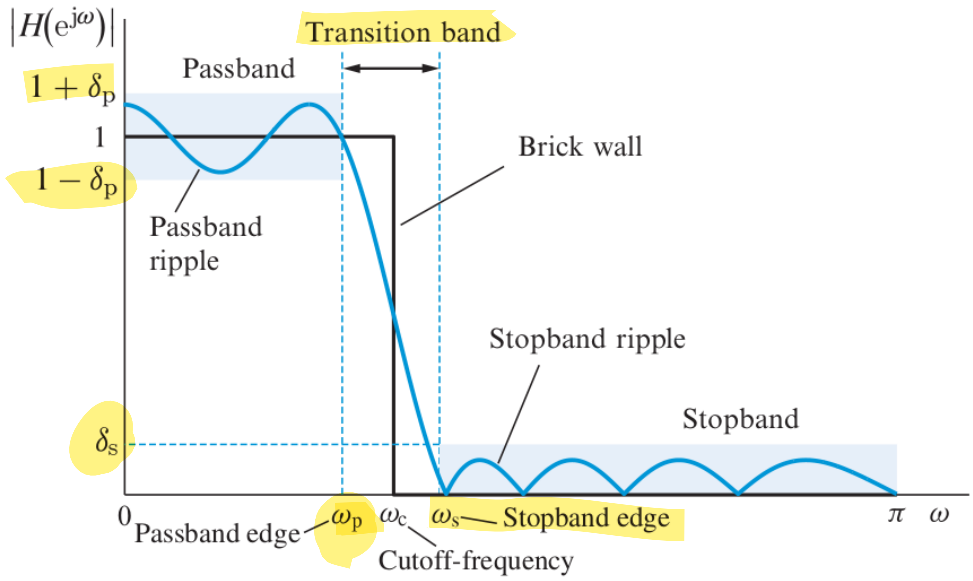
Course Project: Digital Filter Design Using Matlab / GNU-Octave

1 Introduction:

- The frequency response of a discrete-time system depends on the location of its poles and zeros in the $\mathcal{Z}$-domain.
- There are four main types of filters, namely: Butterworth filters, Chebyshev Type I filters, Chebyshev Type II filters, and Elliptic filters.
- One of the main differences among these four filter types is the location of poles and zeros.
- In order to get a filter with real coefficients, complex poles should appear in complex-conjugate pairs, same for complex zeros.
- The objective of this project is to become familiar with designing digital filters using Matlab / GNU-Octave.

2 Filter Specifications:

The specifications of practical filters usually take the form of a tolerance diagram, as shown below for a lowpass filter.



Tolerance diagram for a lowpass filter

- The passband edge is denoted by ω_p and the stopband edge is denoted by ω_s.
- The width of the transition band is $\Delta\omega = \omega_s - \omega_p$.
- The cutoff frequency is denoted by ω_c.
- The maximum allowed passband ripple A_p which indicates the maximum fluctuations in the passband.

- The minimum stopband attenuation A_s which indicates how much the signal in the stopband is attenuated w.r.t. the passband gain.

3 General Requirements:

Assume that the sampling frequency used in your digital filter is F_s.
For all the filters in this assignment use Matlab / GNU-Octave to generate the following plots:

1. Pole-zero plot for the digital filter in the $\mathcal{Z}-$domain.
2. Magnitude and phase response of the filter in dB in the frequency range $(-F_s/2 < f \leqslant F_s/2)$.
3. Phase response of the filter in the frequency range $(-F_s/2 < f \leqslant F_s/2)$.
4. Group delay of the filter in the frequency range $(-\pi < \omega \leqslant \pi)$.
5. The filter's impulse response.

Please do NOT use Matlab GUI tools like filterDesigner or fvtool.

4 Task 0: Audio File

- Download the sample audio file from LMS, and use Matlab / GNU-Octave to read this audio file and use it in all the sections of this assignment.
- Let the sampling frequency F_s of your digital filters be equal to the the sampling frequency of this audio file.
- Let the audio signal be denoted by $x[n]$.
- Use Matlab/GNU-Octave to plot the time-domain signal $x[n]$.
- Use Matlab/GNU-Octave to plot the magnitude spectrum of this audio signal.
- Use Matlab/GNU-Octave to compute the total energy in this signal from time domain and also from frequency domain.

5 Task 1: Digital Echo System

A digital echo system $\mathcal{H}$ can be represented by a general impulse response

$$h[n] = \sum_{k=0}^{\infty} b_k \delta[n - kD]$$

The added "echos" cause a distortion of the original signal. A digital **equalizer** system $\mathcal{G}$ can be used in order to remove this distortion. The impulse response of the equalizer is denoted by $g[n]$, and it is related to $h[n]$ as follows,

$$h[n] * g[n] = \delta[n]$$

- Let $b_k = \{\underline{1}, 0.9, 0.8, 0.7\}$, and let $D = 1000$.
- Using hand analysis, determine the LCCDE and the system function $H(z)$ of the digital echo system $\mathcal{H}$.
- Use Matlab/GNU-Octave to generate the plots described in section 3 for the digital echo system system $\mathcal{H}$.
- Use the audio signal $x[n]$ as an input to this digital echo system $\mathcal{H}$, let's denote the output as $y_1[n]$.
- Compute the mean-square-error of $y_1[n]$ as follows:

$$\text{MSE}_{y1} = \frac{1}{(N+1)} \sum_{n=0}^{n=N} (y_1[n] - x[n])^2$$

- Using hand analysis, determine the LCCDE and the system function $G(z)$ of the equalizer system $\mathcal{G}$.
- Use Matlab/GNU-Octave to generate the plots described in section 3 for the equalizer system $\mathcal{G}$.
- Apply the signal $y_1[n]$ (which has echos) to the equalizer system, and let's denote the output as $y_2[n]$.
- Compute the mean-square-error of $y_2[n]$ as follows:

$$\text{MSE}_{y2} = \frac{1}{(N+1)} \sum_{n=0}^{n=N} (y_2[n] - x[n])^2$$

6 Task 2: Design of a Digital LPF

Passbandedge stopbandedge

- ## Design a minimum order digital IIR LPF with the following specifications: $f_p = 3$ KHz, $f_s = 4$ KHz, $A_p \leqslant 1$ dB, and $A_s \geqslant 50$ dB.

↳ ripples pass ⟶ Stopband attenuation

- ## Repeat your design using the four filter types: Butterworth, Chebyshev Type I, Chebyshev Type II, and the Elliptic filters.
- Compare the filter order for the four types.

2

Design Func.

- Use the audio signal $x[n]$ as an input to each of the four filters.
- For each one of the four filters generate the plots described in section 3
- For each one of the four filters, compute the mean-square error between the input and output signals.
- For each one of the four filters, compute the percentage of energy lost due to the filtering process.
- Based on your results, which one of the four filters is better in terms of the minimum mean-square error between the input and output signals? which filter type results in a minimum percentage of lost power?

7 Task 3: Frequency Transformation using Pole-Zero Pattern Rotation

Rotating the pole-zero pattern can be used to transform a lowpass filter into a highpass or a bandpass filter. This is based on the frequency shift property of the DTFT.

- Transform the Butterworth LPF you have designed in section 6 into a highpass filter by rotating the pole-zero pattern by π. Generate the plots described in section 3 for your HPF.
- Employ the same concept to transform the Butterworth LPF in section 6 into a bandpass filter centered around $\pi/2$. What would you do to make sure that the filter coefficients are real valued? Generate the plots described in section 3 for your BPF.

8 Notes & Deliverables:

- Each **group of 3/4 students** should work together and submit one report.
- Please prepare one compressed file that includes the following items:
  1. Your Matlab codes (*.m files).
  2. A report (pdf files Only) that includes the Matlab codes, results, plots, and hand analysis for each section of the project.
- Upload your compressed file to LMS before 11 : 59 PM on Thursday December 11^{th}, 2025 (Week 12)

9 Useful Matlab Commands:

1. freqz
2. filter
3. impz
4. zplane
5. buttord, butter
6. cheb1ord, cheby1
7. cheb2ord, cheby2
8. ellipord, ellip
9. fft, fftshift
10. audioread, audiowrite
11. upsample

Good Luck
Dr. Michael Ibrahim