The primary goals of this course are to

1. Introduce signals, systems, time and frequency domain concepts and the associated mathematical tools that are fundamental to all DSP techniques;
2. Provide a thorough understanding and working knowledge of design, implementation, analysis and comparison of digital filters for processing of discrete time signals.

Upon successful completion of this class, you will be able to propose, design, implement and validate appropriate DSP techniques for a broad spectrum of real-world applications. In order to achieve these overall goals, you must be able to meet the following individual instructional objectives. These objectives explain exactly what you need to be able to do if you understand the concepts. My expectations of you will therefore be limited to these objectives. All student performance evaluation modalities that will be used in this course (homework assignments, exams, projects) will be geared towards testing whether you have achieved these course objectives. If we discuss additional material during the semester that is not covered below, this document will be appropriately updated, and you will be notified of that change.

**INTRODUCTIONS – SIGNALS & SYSTEMS**

- Describe specific components of a typical DSP system, explain what each component does, and why they are necessary.
- Describe different types of discrete signals (e.g., how they are characterized, how to represent any discrete signal using unit impulse sequences)
- Determine whether a signal is periodic or not. Explain the properties of periodic signals in Cartesian as well as polar coordinates. Explain the concept of digital frequency
- Perform discrete convolution of given two sequences, know the properties of the convolution process, interpret the meaning of the convolution.
- Determine whether a signal is an energy or power signal, explain what this means.
- Determine whether a system is linear, time invariant, stable, causal or has memory
- Explain the meaning of impulse response, determine the output of a system given its input and impulse response. Properly use appropriate properties to determine impulse response of cascaded or parallel systems.
- Determine whether a system is finite (FIR) or infinite impulse response (IIR) system given its constant coefficient linear difference equations (and vice versa when appropriate).

**REPRESENTATION OF SIGNALS IN FREQUENCY DOMAIN**

- Determine the frequency domain representation of a signal using the appropriate form of Fourier transform (Fourier series, transforms, DTFT, DFT, etc.), and interpret the outcome based on the magnitude and phase spectra. Explain under what conditions each can be computed.
- Explain the differences and relationships between different forms of Fourier transforms.
• Know and apply appropriately the properties of the Fourier transforms.
• Know the important Fourier transform pairs for commonly used signals.
• Explain (and implement) the relationship between analog and digital frequencies, impulse response and frequency response.
• Describe which Fourier transforms are periodic in frequency domain and why, interpret the meaning, reason and consequences of this periodicity.
• Describe and derive the sampling theorem.
• Explain what aliasing is, when and why it happens, and its consequences.
• Explain (and implement) how to recover a continuous time signal from its sampled version.

**THE Z-TRANSFORM**

• Define the z-transform, explain its relationship to Laplace transform as well as other types of Fourier transforms.
• Know and apply the convergence conditions of the z-transform, determine the region of convergence of the z-transform of a given signal.
• Know and apply appropriately the properties of the z-transform
• Know the important z-transform pairs for commonly used signals.
• Determine the zeroes and poles of a system based on the z-transform of its impulse response, plot these on the z-plane, explain the connection between the zeroes and poles and the ROC of a system, determine whether the system is stable based on this analysis.
• Compute the inverse z-transform based on partial fraction expansion.

**LINEAR TIME INVARIANT SYSTEMS AND THE CONCEPT OF FILTERING**

• Compute the frequency response and system response given impulse response and/or CCLDE and explain the difference between the two.
• Interpret the system response based on the poles and zeroes – determine whether the system is lowpass, highpass, bandpass, or bandstop based on this interpretation.
• Describe the properties of an ideal filter, and why ideal filters cannot be realized. Explain the changes that need to be made for an ideal filter to be realizable.
• Determine whether a system is zero phase, linear phase, generalized linear phase or non-linear phase based on its system response. Why would we prefer zero phase? Linear phase? How can we implement a zero-phase filter – under what conditions?
• Explain the concepts of phase and group delay. Compute and interpret these quantities.
• Determine whether an FIR system is type I, II, III or IV. Explain the difference between these different types and which ones can be used to implement what kind of (low/high/band) pass systems.
• Implement second / third or higher order low/high/band pass or comb filters by cascading first order or other simple filter structures.
• Describe the properties of minimum and maximum phase and allpass filters. When and how can these type of systems be used / implemented?
FILTER STRUCTURES & REALIZATIONS

- Know different types of FIR filter structures and their implementations (such as linear phase, cascade, canonic form, etc.).
- Know different types of IIR filter structures and their implementations (such as direct form I, direct form II, cascade of second order systems, parallel form implementations, etc.).
- Implement and interpret lattice form structures.

FILTER DESIGN & IMPLEMENTATION

- Determine and implement the appropriate type of design method (e.g. windowing based [along with what kind of windows to use], direct design, frequency sampling, Remez exchange, etc.) to design an FIR filter for a given set of filter specifications.
- Explain and interpret the Gibbs phenomenon for windows based FIR filter design.
- Determine and implement the appropriate type of design method (e.g. analog prototype with bilinear transformations, direct digital design, Yulewalk, etc.) to design an IIR filter for a given set of filter specifications.

RANDOM SIGNAL ANALYSIS AND SPECTRAL ESTIMATION

- Explain different type of probability distributions to characterize noise in digital signals. Explain the meaning, properties and limitations of additive white Gaussian noise.
- Compute different statistical quantities (mean, variance, etc.) of random signals.
- Define, implement and interpret the autocorrelation, autocovariance, cross correlation and cross covariance of (a) signal(s).
- Obtain frequency domain representation of stochastic signals and systems. Define, explain, implement and interpret power spectrum / power spectral density of a signal.
- Explain why simple Fourier transform is not appropriate for obtaining the frequency domain representation of a stochastic signal.
- Determine and implement the appropriate type of spectrum estimation method (periodogram, Welch's method, etc.) for a given stochastic signal.

TIME-FREQUENCY REPRESENTATION

- Explain the meaning of stationarity.
- Explain why Fourier based methods are insufficient in obtaining the time-frequency representation of (nonstationary) signals.
- Implement and interpret short time Fourier transform, how STFT obtains the time-frequency representation, its resolution properties and its shortcomings.
- Describe the Heisenberg’s principle / inequality
- Implement and interpret continuous wavelet transform and how it addressed the shortcomings of the STFT.
- Implement and interpret discrete wavelet transform
- Determine and implement the appropriate form of DWT for different types of applications, such as compression, signal denoising, discontinuity detection.
MATLAB & SIMULINK

- Implement any of the algorithms, approaches, or methods discussed in class in Matlab and/or Simulink.
- See the additional document on submission of Matlab/Simulink code for assignments or exams.

ADDITIONAL TOPICS

- Instructional objectives of additional topics, if time permits to cover them, will be provided when necessary.