

Tandon School of Engineering of New York University
Dept. Electrical & Computer Engineering
ECE-GY 6113 / BE-GY 6403 ---- Digital Signal Processing I,
Fall 2019

Lectures: Thursday 3:20 PM – 5:50 PM, 370 Jay St. 202

Course Description: Discrete and continuous-time linear systems. Z-transforms. Fourier transforms. Sampling. Discrete Fourier transform (DFT). Fast Fourier transform (FFT). Digital filtering. Design of FIR and IIR filters. Windowing. Least squares in signal processing. Minimum-phase and all-pass systems. Matlab programming exercises

Prerequisites: Graduate status. You should have had UG level signals and systems in your BS program. Undergraduate students must have completed EE-UY 3054 Signals and Systems.

Instructor:

Professor Thomas Marzetta, 370 Jay St, Room 902, (646)-997-3028, Email: tom.marzetta@nyu.edu, Homepage: <http://engineering.nyu.edu/people/thomas-marzetta>, Office hours: Wednesday 2:00 – 4:00 PM, Friday 2:00 – 4:00 PM, or appointment by email.

Teaching Assistants:

Hangsong Yan (lead TA), hy942@nyu.edu, office hours: ???

???

???

Text Book:

We will not require a particular textbook. Lecture notes and supplementary materials will be posted on NYU Classes. However, you are highly recommended to have one of the following books for additional reading and more comprehensive treatment of digital signal processing. Other books on DSP are generally fine as well.

- Discrete-Time Signal Processing, third edition, by Oppenheim and Schaffer. Prentice Hall, 2010. ISBN-13: 978-0131988422. [Link](#)
 - Companion Website containing MATLAB based problems, demos, and projects. [Link](#)
- Digital Signal Processing – A Computer Based Approach, 4th edition, by Sanjit K. Mitra, McGraw Hill, ISBN-13: 978-0073380490. [Link](#)

Homework, Exam, and grading Policy: Midterm Exam: 40%, Final Exam: 40%, MATLAB assignments: 10%, Written HW: 10%.

Written homework problems (some include simple MATLAB exercises) will be assigned every week and are due the following week at the lecture time. Late submissions are not accepted except in exceptional cases. Solutions to written HW will be posted on NYU Classes after the HW due date.

MATLAB assignments will be given separately and due dates will be noted (roughly every week as well). You must submit the completed MATLAB assignment directly through NYU Classes.

Students are encouraged to discuss with each other and seek help from the TA and the instructor for completing these assignments but must submit their solutions separately. What is submitted should reflect their own understanding of the material. However, if you worked with others to derive your solutions, you should write the name(s) of the students that you worked with on the top of your hand-in homework, so that you will not be penalized for cheating.

Please note that written assignments may be graded selectively (i.e. not all problems are graded in each assignment and detailed comments may not be provided due to the size of the class). But we will provide solution to all problems. Students should review the provided HW solutions carefully to verify their understanding of the HW problems.

Both exams will be closed-book. Students can bring a cheat sheet (single page, double sided OK). Calculators are not allowed.

Access to MATLAB

Students registered for this course can have MATLAB installed on their laptop computer at the Help-Desk in Rogers Hall 339. Students without prior experience with MATLAB should spend time during the first two weeks of class learning MATLAB. MATLAB tutorials are available at <https://www.mathworks.com/> including the following links: [tutorial in pdf](#), [on-line-tutorial](#)

Homework problems and datasets will be selected from Prof. Selesnick's exercise packets:

- Signals_Exercises.[pdf](#)
- DSP_Exercises.[pdf](#)
- Data_files.[zip](#)

Supplementary materials (including MATLAB demos) will be posted on NYU Classes, and some can also be accessed from [here](#)

Links to resources in previous offerings:

- Professor Selesnick's website for [EL6113](#)

Tentative Course Schedule

- Week 1 (Sept 5): Introduction and LTI systems
 - Why digital signal processing
 - Discrete time signals. Examples.
 - Discrete time systems. System properties.
 - Linear and time invariant systems: Impulse response. Convolution. BIBO stability test.
 - MATLAB function: *conv*
 - MATLAB assignment 1: plotting signals, convolution
 - Reading: Oppenheim Sec. 2.0-2.4, Lecture note by Prof. Voltz (lect1_voltz.pdf)
- Week 2 (Sept 12): Z-transform
 - Z-transform: region of convergence (ROC), right-sided and left-sided signals.
 - Inverse Z-transform: partial fraction expansion.
 - MATLAB functions: *roots*, *poly*, *residue*.
 - MATLAB assignment 2: filtering of EKG signals
 - Reading : Oppenheim Sec. 3.0-3.4, Lecture note by Prof. Voltz (lect2_Voltz_Ztransform.pdf)
- Week 3 (Sept 19): Z-transform and convolution, inverse systems
 - Convolution theorem. Convolution using Z-transform and inverse Z-transform.
 - Transfer function of LTI systems.
 - Difference equations: basic examples, from difference equations to transfer functions using Z-transform;
 - MATLAB function: *filter*.
 - Inverse LTI systems (deblurring) (stable non-causal inverse systems).
 - Complex poles.
 - MATLAB demo: *complex poles*, *inverse system*.
 - MATLAB assignment 3: inverse filtering of EKG signals
 - Reading: Oppenheim Sec. 3.5, 5.2. Lecture note by Prof. Voltz (lect2_Voltz_Ztransform.pdf)
- Week 4 (Sept 26): Frequency Response
 - DC gain.
 - Frequency response: complex exponential function as the eigen function of the LTI system, meaning of frequency response, the periodicity of frequency response, response of LTI system to sinusoidal inputs. Evaluation of frequency response.

- Transfer function / frequency response / unit circle: relationship between time domain and frequency domain; relationship between poles-zeros and frequency response
- MATLAB functions: *freqz*, *polyval*
- MATLAB demo: *frequency response*
- MATLAB assignment 4: frequency response
- Reading: Oppenheim Sec. 2.6, 5.1, 5.3, 5.4. Lecture note by Prof. Voltz (lect3_FreqResponse.pdf)
- Week 5 (Oct 3): Discrete Time Fourier Transform (DTFT)
 - Definition and relation with Z-transform
 - Impulse response using inverse DTFT
 - Ideal low-pass filter (LPF) \leftrightarrow Sinc function
 - N-point moving average filter: roots of unity, digital sinc function, roots of unity equally spaced on unit circle
 - Properties of DTFT
 - Symmetry for real valued signals
 - Meaning of digital frequency, real frequency given sampling rate
 - Example spectra of signals (e.g. voiced and unvoiced speech)
 - Spectrogram
 - MATLAB assignment 5: speech recording and computing spectra using fft and spectrogram
 - Reading: Oppenheim Sec. 2.7-2.9. Lecture note by Prof. Voltz (Voltz_DTFT.pdf)
- Week 6 (Oct 10). Special systems
 - Simple system design: low order systems, specified dc gain and nulls and frequency response at specified frequencies
 - Differentiator
 - Notch filters: motivation (removing 60 Hz hum, baseline drift correction), second-order recursive filter, tonal noise suppression, baseline drift correction, MATLAB demo: *notch filter demo*
 - Phase delay of systems: delay of sinusoid signals through LTI systems, MATLAB demo: *filter delay demo*
 - MATLAB assignment 6: baseline drift correction in EKG signals using dc notch filters, tonal noise suppression using 4th order filter
 - Reading: Selesnick's note and examples (posted at NYU Classes)
- Week 7 (Oct 17): DFT
 - DFT: definition, DFT as a special case of orthogonal transform, properties, connection with DTFT
 - FFT algorithm (brief review). MATLAB function: *fft*, *ifft*, *fftshift*
 - Circular convolution and Convolution theorem: relation to linear convolution, using FFT to compute linear convolution
 - MATLAB assignment 7: Using FFT to compute circular and linear convolution, using FFT to plot signal spectrum.

- Reading: Selesnick's note (posted at NYU Classes). Optional: Oppenheim Secs. 8.5, 8.6, 8.8.
- Week 8 (Oct 24): **Midterm exam**
- Week 9 (Oct 31): DTFT/spectrum evaluation using DFT
 - Using FFT to evaluate DTFT, zero padding, understanding relation between frequency and DFT freq. index.
 - Frequency leakage
 - Frequency measurement using windows and DFT
 - Short-time FT (STFT): influence of parameters (frame size, hop, DFT length), MATLAB function: *Spectrogram*
 - MATLAB assignment 8: Filtering a noisy speech signal.
 - Reading: Selesnick's notes and Voltz's note (posted at NYU Classes)
- Nov 4: **Last day to withdraw from a class with "W" grade**
- Week 10 (Nov 7): FIR Filter Design
 - Linear phase FIR filters: why linear phase? The four types, properties
 - MATLAB demo: *phase distortion demo*
 - FIR filter design based on the minimax criterion (Parks-McClellan method). MATLAB functions: *firpm*, *firpmord*, *firpm demo*
 - FIR filter design using windows: effect of truncation and windowing, standard windows (Hamming, Hann, Blackman, Kaiser). MATLAB functions: *Hamming*, *Kaiser*, *kaiserord*, *fir1*
 - MATLAB assignment 9: Filtering a noisy speech signal using FIR filters designed using different methods
 - Readings: Selesnick's notes. Optional: Oppenheim Sec. 7.5-7.8, Sec. 10.1-10.3
- Week 11 (Nov 14) : Recursive (IIR) filter design
 - Classic analog filters: Butterworth, Cheby1, Cheby2, elliptic; MATLAB functions: *buttap*, *cheb1ap*, *cheb2ap*, *ellipap*
 - Conversion of analog to digital filters: Bilinear transformation (BLT), MATLAB function: *bilinear*;
 - Filter transformations: LPF-> HPF, LPF->BPF, LPF->BSF, etc.
 - MATLAB demo: *recursive filters demo*; MATLAB functions: *butter*, *cheby1*, *cheby2*, *ellip*
 - MATLAB assignment 10: filtering of speech using IIR filters designed using different methods
 - Reading: Selesnick's note and demo. Optional: Oppenheim Sec. 7.1-7.4
- Week 12 (Nov. 21)
 - Effect of non-linear phase on speech and images.
 - Zero-phase filtering: forward-backward filtering, MATLAB function: *filtfilt*

- Minimum phase systems: definition, minimum delay property, Factorization of $H(z)$ into the product of minimum phase and all-pass
- Spectral factorization.
- MATLAB assignment 11: filtering of image using IIR and FIR filters (linear phase and minimum phase), and using forward-backward filtering
- Reading: Lecture notes and demo in NYU Classes. Optional: Oppenheim Sec. 5.5, 5.6.
- Week 13 (Dec 5)
 - Review of sampling theorem: Frequency domain representation, Reconstruction of a bandlimited signal from its samples
 - Discrete time processing of continuous time signals
 - Downsample a digital signal: Decimation filter design
 - Upsample a digital signal: Interpolation filter design
 - MATLAB assignment 12: Sampling rate conversion using different filters
 - Reading: Oppenheim Chap. 4.
- Week 14 (Dec 12) Least squares methods
 - Convolution as a matrix product
 - Overdetermined and underdetermined linear equations
 - Using regularization (minimum energy, smoothness, etc.)
 - Constrained least squares
 - Least squares smoothing as an LTI filter
 - Least squares deconvolution as an LTI filter
 - Missing sample interpolation
 - Other example applications
 - MATLAB assignment 13: Least squares
 - Reading: Selesnick_LeastSquares.pdf
- Week 15 (Dec 19): **Final exam**

Last updated: 7/03/2019, Thomas Marzetta