

Rutgers University, Department of Electrical and Computer Engineering
ABET COURSE SYLLABUS
COURSE: 14:332:346

Course Catalog Description:	14:332:346 – Digital Signal Processing (3) Introduction to digital signal processing, sampling and quantization, A/D and D/A converters, discrete time systems, convolution, z-transforms, transfer functions, digital filter realizations, fast Fourier transforms, filter design, and digital audio applications.
Pre-Requisite Courses:	14:332:345, 01:640:244
Co-Requisite Courses:	14:332:348
Pre-Requisite by Topic:	<ol style="list-style-type: none">1. Complex numbers and trigonometry2. Differential and integral calculus3. Linear time-invariant systems4. Convolution and transfer functions5. Laplace transforms and z-transforms6. Difference equations
Textbook & Materials:	S.J. Orfanidis, <i>Introduction to Signal Processing</i> , Prentice-Hall, 1996, or equivalent.
References:	<i>MatLab: Student Version</i> , Current Edition, The MathWorks, Inc..
Overall Educational Objective:	To introduce the basic principles, methods, and applications of digital signal processing, emphasizing its algorithmic, computational, and programming aspects.
Course Learning Outcomes:	<p>A student who successfully fulfills the course requirements will have demonstrated:</p> <ol style="list-style-type: none">1. Understanding of the two key DSP concepts of sampling and quantization, and the practical issues involved in sampling, aliasing, and analog reconstruction of signals, and in choosing and defining specifications for anti-aliasing prefilters and anti-image postfilters.2. Understanding of the quantization process and some practical implementations of A/D and D/A converters, such as the conversion algorithm for bipolar two's complement successive approximation converters.3. Understanding of basic discrete-time systems concepts, such as linearity, time-invariance, impulse response, convolution, FIR and IIR filters, causality, stability, z-transforms, transfer functions, frequency response, time constants, transient and steady-state response.4. Understanding of how to implement digital filters in software and hardware, using block processing methods based on convolution, or real-time sample-by-sample processing methods based on block diagram realizations that are implemented with linear or circular delay-line

buffers.

5. Ability to translate a filter's transfer function into block-diagram realizations, such as direct, canonical, transposed, and cascade forms. And conversely, the ability to start with a given block diagram, determine its transfer function, and translate it into a real-time processing algorithm implementable in software or hardware.
6. Understanding of various digital filter design methods meeting prescribed specifications, such as pole/zero placement or bilinear transformation methods, and appreciating design tradeoffs between the specifications and filter order, time constant, and pole locations.
7. Understanding of the discrete Fourier transform and the fast Fourier transform and their use in spectral analysis, data compression, and fast convolution. Understanding of the tradeoffs between frequency resolution and signal duration and the use of windows for reducing frequency leakage. Ability to perform short FFTs by hand.

How Course Outcomes are Assessed:

Two Mid-Term Exams (60 %)

Final Exam (40 %)

N = none S = Supportive H = highly related

Outcome	Level	Proficiency assessed by
(a) an ability to apply knowledge of Mathematics, science, and engineering	H	HW Problems, Exams
(b) an ability to design and conduct experiments and interpret data	S	Design Problems in HW, Exams, and in the DSP lab course 332:348
(c) an ability to design a system, component or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability	S	Digital filter design examples meeting prescribed specifications
(d) an ability to function as part of a multi-disciplinary team	N	
(e) an ability to identify, formulate, and solve ECE problems	H	HW Problems, Exams
(f) an understanding of professional and ethical responsibility	N	
(g) an ability to communicate in written and oral form	S	HW Problems, exams
(h) the broad education necessary to understand the impact of electrical and computer engineering solutions in a global, economic, environmental, and societal context	N	
(i) a recognition of the need for, and an ability to engage in life-long learning	S	Home-work, emphasized during lectures
(j) a knowledge of contemporary issues	N	
(k) an ability to use the techniques, skills, and modern engineering tools necessary for electrical and computer engineering practice	H	HW Problems, Exams, Software and DSP hardware lab experience
Basic disciplines in Electrical Engineering	H	HW Problems, Exams
Depth in Electrical Engineering	S	HW Problems, Exams, Lab
Basic disciplines in Computer Engineering	H	Programming DSP algorithms in C, MATLAB, and assembly language for DSP chips
Depth in Computer Engineering	S	Software and hardware programming
Laboratory equipment and software tools	H	Analog Devices DSP-2181 digital signal processor. Programming in C, MATLAB, and DSP software development environment
Variety of instruction formats	S	Lecture, office hour discussions

Topics Covered week by week:

Week 1: Sampling Theorem; sampling and aliasing of sinusoids; practical antialiasing prefilters.

Week 2: Analog reconstructors; anti-image postfilters; quantization; A/D and D/A converters; data

formats for DSP chips.

Week 3: Review of discrete-time systems; linearity and time-invariance; stability and causality; convolution.

Week 4: Block processing convolution methods; Sample by sample processing methods; FIR filters.

Week 5: EXAM 1; Software and hardware implementations of FIR filters; linear and circular delay-line buffers.

Week 6: Review of z-transforms; inverse z-transform methods; transfer functions.

Week 7: Transfer function analysis of FIR and IIR filters; frequency response; pole/zero filter designs; transient and steady-state response; time constants; z-domain characterization of stability and causality.

Week 8: Digital filter realization forms: direct, canonical, transposed, and cascade of second-order sections; hardware and software implementation of IIR filters using circular delay-line buffers.

Week 9: EXAM 2; Noise reduction and signal enhancement applications.

Week 10: DSP applications in digital audio, such as multitap delays and reverberation effects.

Week 11: Discrete Fourier transform; spectral analysis; frequency resolution and windowing; Hamming windows.

Week 12: Fast Fourier transform; radix-2 decimation-in-time FFT algorithm.

Week 13: FIR digital filter design; course evaluations.

Week 14: IIR digital filter design;

Week 15: Review and Final Examination.

Computer usage: DSP algorithm programming in C, MATLAB, and Assembly Language.

Laboratory Experiences: 14:332:348 Digital Signal Processing Laboratory

Design Experiences: HW problems in designing digital filters using various techniques. In conjunction with 332:348, designing and programming real-time audio signal processing algorithms on DSP hardware.

Independent Learning Experiences: 1. Home-Work, 2. MATLAB programming, 3. Testing (Quizzes, Exams)

Contribution to the Professional Component:

(a) College-level mathematics and basic sciences: 0.5 credit hours

(b) Engineering Topics (Science and/or Design): 2.5 credit hours

(c) General Education: 0 credit hours

Total credits: 3

Prepared by: S. J. ORFANIDIS

Date: July 2007