Course Approval Form

For approval of new courses and deletions or modifications to an existing course.

registrar.gmu.edu/facultystaff/curriculum

Action Requested:

- [x] Create new course
- [ ] Delete existing course
- [ ] Modify existing course (check all that apply)
  - [ ] Title
  - [ ] Credits
  - [ ] Repeat Status
  - [ ] Grade Type
  - [ ] Prereq/coreq
  - [ ] Schedule Type
  - [ ] Restrictions
  - [ ] Other:

Course Level:

- [ ] Undergraduate
- [x] Graduate

College/School: VSE (Volgenau School of Engineering) **Department:** ECE

Submitted by: Jeremy Allnutt **Ext:** 3-6097 **Email:** jallnutt@gmu.edu

Subject Code: ECE **Number:** 740

(Do not list multiple codes or numbers. Each course proposal must have a separate form.)

Effective Term:

- [x] Fall
- [ ] Spring
- [ ] Summer

Year 2014

Title:

Current: DSP Hardware Architectures

Banner (30 characters max including spaces):

New:

Credits: (check one)

- [3] Fixed
- [ ] Variable

X to

Repeat Status: (check one)

- [X] Not Repeatable (NR)
- [ ] Repeatable within degree (RD)
- [ ] Repeatable within term (RT)

Maximum credits allowed:

Grade Mode: (check one)

- [X] Regular (A, B, C, etc.)
- [ ] Satisfactory/No Credit
- [ ] Special (A, B C, etc. +IP)

Schedule Type Code(s): (check all that apply)

- [X] Lecture (LEC)
- [ ] Lab (LAB)
- [ ] Recitation (RCT)
- [ ] Internship (INT)
- [ ] Independent Study (IND)
- [ ] Seminar (SEM)
- [ ] Studio (STU)

Prerequisite(s):

ECE 535 and ECE 545 or equivalents or permission of instructor

Corequisite(s):

Instructional Mode:

- [X] 100% face-to-face
- [ ] Hybrid: ≤ 50% electronically delivered
- [ ] 100% electronically delivered

Special Instructions: (list restrictions for major, college, or degree; hard-coding; etc.)

Are there equivalent course(s)?

- [ ] Yes
- [X] No

If yes, please list

Catalog Copy for NEW Courses Only (Consult University Catalog for models)

Description (No more than 60 words, use verb phrases and present tense)	**Notes** (List additional information for the course)
Addresses topics that include high-level DSP optimizations, such as pipelining, unfolding, and parallel processing; common DSP structures such as FFTs, filters, direct digital frequency synthesizers, and correlators; modeling of DSP algorithms in MATLAB and conversion of MATLAB models into fixed-point VHDL blocks; platform implementation issues: hardware vs. software, FPGA vs. ASIC, power, area, throughput, and applications of DSP hardware.	
Indicate number of contact hours: Hours of Lecture or Seminar per week: 3 Hours of Lab or Studio:	
When Offered: (check all that apply) [x] Fall [ ] Summer [x] Spring	

Approval Signatures

Andre Manitius — Department Approval — 10/31/13 — Date

College/School Approval — Date

If this course includes subject matter currently dealt with by any other units, the originating department must circulate this proposal for review by those units and obtain the necessary signatures prior to submission. Failure to do so will delay action on this proposal.

Unit Name	Unit Approval Name	Unit Approver's Signature	Date

For Graduate Courses Only

Graduate Council Member — Provost Office — Graduate Council Approval Date

For Registrar Office's Use Only: Banner_____Catalog_____

revised 2/2/10

SCHOOL PROPOSAL TO THE GRADUATE COUNCIL
BY
SCHOOL OF INFORMATION TECHNOLOGY AND ENGINEERING

1. CATALOG DESCRIPTION

a) ECE 740 Digital Signal Processing Hardware Architectures
b) Prerequisites: ECE 535 and ECE 545 or equivalents or permission of instructor
c) Catalog Description:

Addresses topics that include high-level DSP optimizations, such as pipelining, unfolding, and parallel processing; common DSP structures such as FFTs, filters, direct digital frequency synthesizers, and correlators; modeling of DSP algorithms in MATLAB and conversion of MATLAB models into fixed-point VHDL blocks; platform implementation issues: hardware vs. software, FPGA vs. ASIC, power, area, throughput, and applications of DSP hardware.

2. JUSTIFICATION

(a) Course Objectives:
At the conclusion of this course, the student will have a strong foundation in the area of hardware architectures for digital signal and image processing, with application to communications, sensing, multimedia, and robotics. The student will get familiar with the corresponding design flow, languages, and tools required for efficient implementation of digital signal processing algorithms in Field Programmable Gate Arrays (FPGAs). Finally, the student will also get acquainted with the most recent developments in the field through the literature survey covering most recent journals and conference papers related to efficient implementation of digital signal processing algorithms in hardware.

(b) Course Necessity:
DSP hardware implementations have become ubiquitous in modern society. This course is intended to meet the needs of graduate students in MS EE, MS CpE, or PhD ECE desiring a treatment of DSP optimization and implementation techniques. In particular, this course is highly recommended for students pursuing the Digital Signal Processing specialization area within MS CpE. The course has been taught two times under the course number ECE 699, and has been popular with students.

(c) Relationship to Existing Courses:
This course is intended as a follow up to the existing ECE535 Digital Signal Processing course and the ECE 545 Digital System Design with VHDL course. The course is not a prerequisite for other courses in the program, but provides technical foundations and motivation for further study in more specialized courses in the area of efficient implementations of digital signal processing algorithms, including ECE 798 Research Project, and ECE 799 Master's Thesis.

3. APPROVAL HISTORY

ECE Department Date: October 31st, 2013

VSE Graduate Committee Date:

VSE Dean Date:

4. SCHEDULING

Once per year, starting in Spring 2014.
Proposed Instructors: Dr. Cohen, Dr. Gaj, and other suitably qualified faculty.

5. COURSE OUTLINE

(a) Syllabus

Week 1: ***Course overview***

Introduction to the course. Overview of Digital Signal Processing (DSP) hardware architectures, fixed-point quantization, iteration bound, and direct digital frequency synthesizer (DDFS).

Week 2: ***Pipelining and Parallel Processing***

Cutset. Pipelining of FIR digital filters. Parallel processing. Pipelining and parallel processing for maximum throughput and for low power.

Week 3: ***Retiming***

Definition and properties. Solving system of inequalities. Retiming techniques.

Week 4: ***Unfolding***

Basic algorithm, properties, analysis of critical path, retiming, applications.

Week 5: ***Folding***

Basic transformation. Register minimization techniques. Folding of multi-rate systems.

Week 6: ***Systolic Arrays***

Design methodology. FIR systolic arrays. Selection of scheduling vector. Matrix-matrix multiplication and 2D systolic array design.

Week 7: ***Redundant Arithmetic and Numerical Strength Reduction***

Redundant number representations. Carry-free radix-2 addition and subtraction. Hybrid radix-4 addition. Radix-2 hybrid redundant multiplication architectures. Data format conversion. Redundant to nonredundant conversion. Subexpression elimination. Multiple constant multiplication. Subexpression sharing in digital filters. Additive and multiplicative number splitting.

Week 8: Midterm Exam

Week 9: Fast Convolution

Cook-Toom algorithm. Winograd algorithm. Iterated convolution. Cyclic convolution. Design of fast convolution algorithm by inspection.

Week 10: ***Algorithmic Strength Reduction***

Parallel FIR filters. Discrete Cosine Transform (DCT) and Inverse DCT. Parallel architectures for rank-order filters.

Week 11: ***Pipelined and Parallel Recursive and Adaptive Filters***

Pipeline interleaving in digital filters. Pipelining in 1^{st}-order IIR digital filters. The difference between scattered look-ahead and clustered look-ahead techniques. Pipelined adaptive digital filters.

Week 12: ***Scaling and Roundoff Noise***

State variable description of digital filters. Scaling and roundoff noise computations. Roundoff noise in pipelined IIR filters. Roundoff noise computation using state variable description. Slow-down, retiming, and pipelining.

Week 13: ***Bit Level Arithmetic. Review for Final Exam***

Bit-serial multipliers. Bit-serial filter design and implementation. Canonic signed-digit arithmetic. Distributed arithmetic. Review for Final Exam.

Week 14: ***Project Presentations***

(b) Required Reading and Reference Material

Primary Textbook:
Keshab K. Parhi, VLSI Digital Signal Processing Systems: Design and Implementation, Wiley, 1999, ISBN: 978-0471241867.

Supplementary Reading:

1. Shoab Ahmed Khan, Digital Design of Signal Processing Systems: A Practical Approach, Wiley, 2011, ISBN: 978-0470741832.
2. Roger Woods, John Mcallister, Richard Turner, Ying Yi, Gaye Lightbody, FPGA-based Implementation of Signal Processing Systems, Wiley, 2008, ISBN: 978-0470030097.
3. Li Tan and Jean Jiang, Digital Signal Processing, Second Edition: Fundamentals and Applications, Academic Press, 2013, ISBN: 978-0124158931.

Additionally, projects will be based on the most recent articles from the following

- journals: IEEE Transactions on Signal Processing, IEEE Transactions on Image Processing, IEEE Transactions on Very Large Scale Integration (VLSI) Systems, IEEE Transactions on Computers, Proceedings of the IEEE, Digital Signal Processing (by Elsevier), Journal of Signal Processing Systems, Circuits, Systems and Signal Processing – CSSP (by Springer); and
- conferences: IEEE International Conference on Acoustics, Speech, & Signal Processing - ICASSP, Asilomar Conference on Signals, Systems, and Computers, Conference on Design and Architectures for Signal and Image Processing – DASIP, International Conference on Information Sciences, Signal Processing and their Application – ISSPA, International Symposium on Field-Programmable Gate Arrays – FPGA, International Conference on Field-Programmable Technology – FPL.

(c) Student Evaluation Criteria

Homework:	20%
Project or mini-projects:	30%
Midterm:	20%
Final:	30%

Weekly homework problems will be assigned to help students absorb the theoretical material and to develop problem solving skills. A project involving DSP design, implementation, and optimization will be assigned to give students more hands-on experience with the concepts learned in the course. The project will end with an oral presentation and a submission of a comprehensive report and project deliverables, including VHDL source codes.