Course Information

INSTRUCTOR	Daniel Llamocca
CONTACT INFO	email: llamocca@oakland.edu
OFFICE HOURS	Tuesday 2:00 to 4:00 pm @ Room EC-438, or by appointment
LECTURES	Tuesday/Thursday 7:30 pm - 9:17 pm @ Room EC-281
LABORATORY	Friday: 1:00 to 3:00 pm @ Room EC-562

COURSE CATALOG DESCRIPTION: ECE 495/595 – Reconfigurable Computing (4 credits)

Analysis and design of reconfigurable computer architectures. Advanced topics in computer arithmetic. Hardware/Software co-design using Programmable System-on-Chip (ARM processor + FPGA fabric). Introduction to Self-Reconfigurable Architectures. With laboratory and design project. Offered: Fall. Prerequisite(s): ECE 378 and major standing.

COURSE WEBPAGE

- The course material will be hosted on Moodle (moodle.oakland.edu). Grades will be periodically posted via this system.
- As a backup resource, the material will also be posted at: www.secs.oakland.edu/~llamocca/Fall2016_ece495.html

TEXTBOOK:

- There is no required textbook. Students are encouraged to use the extra references.

EXTRA REFERENCES:

- Louise H. Crockett, Ross A. Elliot, Martin A. Enderwitz, Robert W. Stewart, *The Zynq Book: Embedded Processing with the ARM® Cortex®-A9 on the Xilinx® Zynq®-7000 All Programmable SoC*, 1st ed., 2014.
  - ✓ Free download: http://www.zynqbook.com
- Louise H. Crockett, Ross A. Elliot, Martin A. Enderwitz, Robert W. Stewart, *The Zynq Book Tutorials*, v 1.2, 2014.
  - ✓ Free download (including tutorial files): http://www.zynqbook.com
- Bryan Mealy, Fabrizio Tappero, Free Range VHDL, Free Range Factory, 2013
  - ✓ Free download: http://www.freerangefactory.org/dl/free_range_vhdl.pdf
- S. Brown, Z. Vranesic, *Fundamentals of Digital Logic with VHDL Design*, 3rd ed., McGraw Hill, 2009
- Peter J. Ashenden, The Designer's Guide to VHDL, 3rd ed., Elsevier, 2008.
- B. Parhami, Computer Arithmetic: Algorithms and Hardware Designs, 2nd ed., Oxford University Press, Inc., 2009.

COURSE OBJECTIVES

1. Design custom architectures using fixed-point and floating-point arithmetic
2. Describe how to unfold a sequential algorithm to turn it into a fully pipelined architecture.
3. Learn advanced coding and testbench techniques in Hardware Description Language.
4. Design an embedded system using FPGA fabric and an embedded ARM® microprocessor.
5. Describe the process of Dynamic Partial Reconfiguration on an All-Programmable System-on-Chip device.
6. Design high-performance application-specific reconfigurable systems.
7. Work in a team environment to design a reconfigurable systems and communicate the results in a written report and an oral presentation.

GRADING SCHEME:

Homeworks:	20%	**Midterm Exam**:	20% (October 13th, 7:30-9:17 pm)
Laboratory:	30%	**Final Project:**	30% (December 13th, 7:00-10:00 pm)

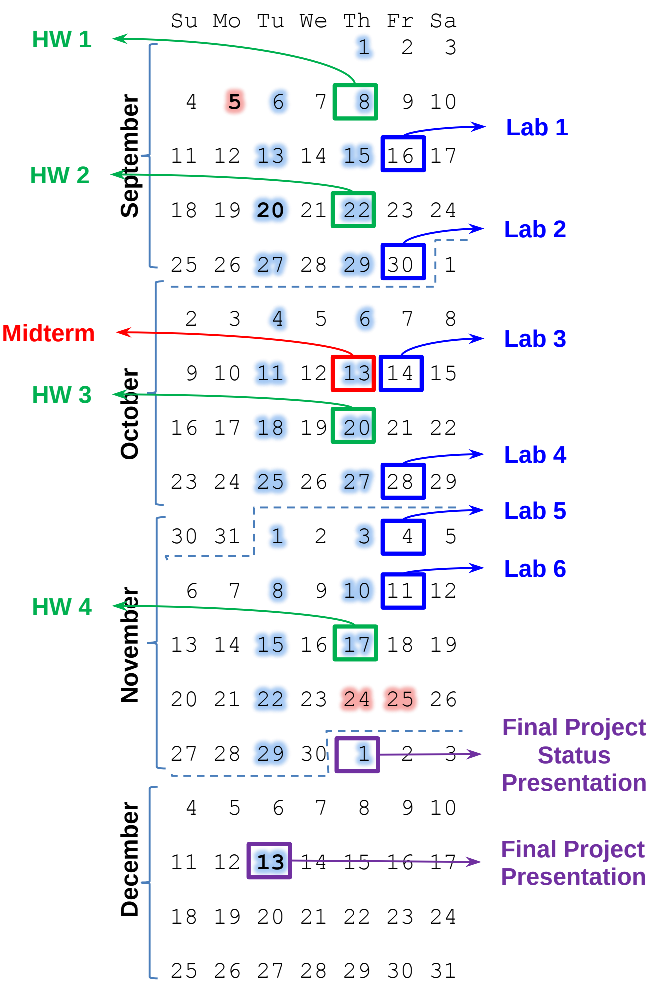
- **Homeworks:** Homework assignments are meant to strengthen your conceptual understanding of the topics. Completing homework assignments is a key component of this course as it will help students master the course material and prepare them for the examinations.
  Homeworks will be posted according to the schedule (green rectangles). Students have one week to turn in the completed assignments in class. Groups of two (2) are allowed. Late submissions are NOT accepted.

- **Midterm Exam:** Closed-books, closed-notes, in-class exam. Students are not allowed to take the exams neither before nor after the exam date. Make-up exams are given *only* under extreme circumstances (such as a medical emergency).

- **Laboratory:** This important component of the class will reinforce your understanding of the topics. There will be six (6) labs throughout the semester. Groups of two (2) are allowed.
  The instructor will be present during the regularly scheduled laboratory times. Students can work during those times or at any other time and place.
  Students have one week to complete the lab assignments and have them checked off by the instructor.

- **Final Project:** Students will work in groups in a Final Project. Each group will prepare an oral presentation and submit a final research paper for conference presentation. A Status Presentation is also required and it amounts to 25% of the Final Project grade.

Schedule



	Su	Mo	Tu	We	Th	Fr	Sa
September					1	2	3
	4	5	6	7	8	9	10
	11	12	13	14	15	16	17
	18	19	20	21	22	23	24
	25	26	27	28	29	30	1
October	2	3	4	5	6	7	8
	9	10	11	12	13	14	15
	16	17	18	19	20	21	22
	23	24	25	26	27	28	29
	30	31	1	2	3	4	5
November	6	7	8	9	10	11	12
	13	14	15	16	17	18	19
	20	21	22	23	24	25	26
	27	28	29	30	1	~~2~~	~~3~~
December	4	5	6	7	8	9	10
	11	12	13	~~14~~	~~15~~	~~16~~	~~17~~
	18	19	20	21	22	23	24
	25	26	27	28	29	30	31

GRADE ASSIGNMENT:

90-100	3.6 to 4.0 (A)
80-89	3.0 to 3.5 (B)
60-79	2.0 to 2.9 (C)
50-59	1.0 to 1.9 (D)
49 and below	0.0 (no credit)

LABORATORY MATERIALS

- **Hardware**: ZYBO board ($125.00)
  ✓ To order the board: http://store.digilentinc.com/zybo-zynq-7000-arm-fpga-soc-trainer-board/
  Select the Academic Version

- **Software**:
  ✓ Vivado Webpack, SDK, MATLAB®
  To download:
  http://www.xilinx.com/support/download/index.html/content/xilinx/en/downloadNav/vivado-design-tools.html

OUTLINE OF TOPICS

Computer Arithmetic		▪ Unsigned and signed numbers: binary representation ▪ Fixed-point arithmetic: addition, subtraction, multiplication, division ▪ Floating point arithmetic: addition, subtraction, multiplication, division ▪ Dual fixed point arithmetic: addition, subtraction
Special-Purpose Arithmetic Circuits and Techniques		▪ Arithmetic units for fixed-point, floating-point and dual fixed-point. ▪ Multi-operand Addition: iterative (accumulator) and adder tree. ▪ CORDIC Algorithm: circular, linear, and hyperbolic. Special functions: exp, ln, sqrt ▪ Square Root: Iterative version ▪ LUT approach: Pixel processor example (gamma correction, contrast stretching) ▪ Distributed Arithmetic: FIR Filter, DCT
Advanced Coding in VHDL		▪ Custom-defined data types, arrays ▪ Parameterization: for-generate, if-generate, std_logic_2d ▪ Reading/writing text files for synthesis and simulation
Pipelining and unfolding		▪ Multiplier: iterative vs array ▪ Divider: iterative vs array ▪ Square root: iterative and pipelined ▪ CORDIC: iterative and pipelined
Embedded System in a SoC	Hardware	Zynq architecture: FPGA fabric + ARM® microprocessor AXI bus: AXI4, AXI4-Lite, and AXI4-Stream Interfaces Interface development for AXI4
	Software	Introduction to SDK ARM processor SD card
		▪ Hardware/software co-design ▪ Custom IP and driver generator for AXI in Vivado ▪ Writing software applications in SDK ▪ FPGA features: FIFOs, MMCMs, Dual-port RAMs ▪ Case example: Pixel processor ▪ Case example: Matrix multiplication (with constant) ▪ Case example: 2D DCT
Dynamic Partial Reconfiguration		▪ Introduction to Self-Reconfigurable Systems ▪ Static vs dynamic reconfiguration ▪ DPR requirements: reconfiguration controller, generating and downloading bitstreams ▪ Case example: pixel processor ▪ Time and memory overhead ▪ Dynamic Frequency Control
Applications		▪ Image processing: Pixel Processor ▪ DSP: Audio filter ▪ Video Compression: Transform and Quantization for HEVC ▪ Communications

CLASS POLICIES

- **Academic conduct policy**: All members of the academic community at Oakland University are expected to practice and uphold standards of academic integrity and honesty. Academic integrity means representing oneself and one's work honestly. Misrepresentation is cheating since it means students are claiming credit for ideas or work not actually theirs and are thereby seeking a grade that is not actually learned. Academic dishonesty will be dealt with seriously and appropriately. Academic dishonesty includes, but it is not limited to cheating on examinations, plagiarizing the works of others, cheating on lab reports, unauthorized collaboration in assignments, hindering the academic work of other students.

- **Special Considerations**: Students with disabilities who may require special consideration should make an appointment with campus Disability Support Services, 106 North Foundation Hall, phone 248 370-3266. Students should also bring their needs to the attention of the instructor as soon as possible. For academic help, such as study and reading skills, contact the Academic Skills/Tutoring Center, 103 North Foundation Hall, phone 248 370-4215.

- **Add/Drops**: The university policy will be explicitly followed. It is the student's responsibility to be aware of deadline dates for dropping courses.

- **Attendance**: It is assumed that the students are aware of and understand the university attendance policy. Attendance is mandatory and maybe monitored. Students are responsible for all material covered in classes that they miss. There will no excuses for being late to exams.

- **Athlete Excused Absences**: Students shall inform the instructor of dates they will miss class due to an excused absence prior to the date of that anticipated absence. For activities such as athletic competitions whose schedules are known prior to the start of a term, students must provide their instructors during the first week of each term a written schedule showing days they expect to miss classes. For other university excused absences, students must provide the instructor at the earliest possible the dates that they will miss.

- **Special Circumstances:** The instructor should be notified as early as possible regarding any special conditions or circumstances which may affect a student's performance during the course timeframe (e.g., medical emergencies, family circumstances).

- **Cellphones**: A ringing cellphone going off during a lecture is disruptive to other students as well as the instructor. Students are strongly advised to set their cellphones to vibrate (not ringing) and leave the classroom discretely to answer the phone.