CMPE-660 Reconfigurable Computing

Fall 2023

Class instructor:

Marcin Lukowiak, mxleec@rit.edu,

Office: **GLE (09)-3439**

Phone: **(585) 475-2808**

Course Description:

The objective of this course is to present the foundations of reconfigurable computing methodologies from both hardware and software perspectives. Topics may include: VHDL modeling of combinational and sequential logic for synthesis and efficient hardware implementation, architectures of modern field programmable gate arrays (FPGAs), hardware-software co-design with embedded processors, hardware optimization techniques, system level integration under operating system, and dynamic reconfiguration. Hands-on exercises, in which students will acquire a solid capability of Xilinx CAD tools and FPGA devices, are required. All practical assignments include the whole FPGA design flow: system design, modeling and model verification, hardware verification.

Prerequisites:

CMPE–260 Digital System Design II (design of digital systems with VHDL), basic programming in Matlab and C

Classes – studio format:

Monday :	5.00pm - 6.15pm	James E Gleason Hall (GLE)-3159
Wednesday:	5.00pm - 6.15pm	James E Gleason Hall (GLE)-3159

CE support staff:

CE lab manager; **Sean Cain –** srceec@rit.edu;
Phone: (585) 475-5056, Office: (GLE)-3411

EE lab manager; **Vincent Antonicelli –** contact him for access to the lab; vaaeee@rit.edu;
Phone: (585) 475-2121, Office: (GLE) -3245

TAs:

Long Lam ll5530@rit.edu

Jacob Lapierta jdl4988@rit.edu

Office hours:

M, W :	6.15pm - 7.00pm;	Office: GLE (09)-3439

Textbook: No textbook is required for this course, on-line materials will be posted on *mycourses*.

References:

VHDL for Logic Synthesis, Andrew Rushton, Wiley 2011, ISBN-13: 978-0470688472

VHDL for Engineers, Kenneth L. Short, Prentice Hall 2008, ISBN-13: 978-0131424784

The Design Warrior's Guide to FPGAs: Devices, Tools and Flows, C. "Max" Maxfield, Newnes, 2004, ISBN- 13: 978-0750676045

Online reference to VHDL syntax, http://www.vhdl-online.de/courses/system_design/start

Generative AI:

While the world of Generative AI such as GPT, keeps evolving, in this class we adopt a no-use policy for GPT in all graded course assignments.

Grade Weighting:

Tutorials	:	12.5%
Homework	:	10.0%
Labs	:	50.0%
Exams	:	20.0%
Paper presentation	:	7.5%
Sum	**:**	**100.0%**

	Quality points
A	4.000
A-	3.667
B+	3.333
B	3.000
B-	2.667
C+	2.333
C	2.000
C-	1.667
D	1.000
F	0.000

Homework: Homework assignments will be posted on myCourses. Each homework will be submitted as a single file PDF format to its myCourses assignment folder. Late submissions will incur a penalty of 10% per day.

Tutorials: Tutorial assignments will be posted on myCourses. Results of each tutorial will need to be demonstrated to TAs. Proof of completing each tutorial in the form of screenshots and sign-offs will be submitted as a single file PDF format to its myCourses assignment. Late submissions will incur a penalty of 10% per day.

Lab: Lab assignments will be posted on the myCourses. Results of each lab will need to be demonstrated to TAs. Proof of completing each lab exercise in the form of screenshots and sign-offs will be required together with a 2-3 pages report in length (not including figures and tables). More detailed submission guidelines will be posted on myCourses. Late lab demos and reports will be accepted with penalty of 5% per day and 15% per week.

Students are responsible to observe announcements sent via Email addresses used in http://mycourses.rit.edu and online grading/early alert system.

Tentative Topics

	Class topics	Lab	Due
Week 1 8/28-9/1	Course introduction Digital systems design flow with VHDL	**Tutorial1:** Introduction to Xilinx Vivado environment and Digilent Nexys4 DDR board, VHDL review, RTL schematics, testbenches	
Week 2 9/4-9/8	No classes on Monday – Labor Day Modeling of combinational logic for simulation and synthesis **HW#1**	**Tutorial2:** Device view	**Tutorial1**
Week 3 9/11-9/15	Modeling of sequential logic for simulation and synthesis **HW#2**	**Tutorial3:** Sequential circuits; state machines	**HW#1** **Tutorial2**
Week 4 9/18-9/22	Design and modeling of state machines; handshaking protocol **HW#3**	**Lab1.** USB/PS2 - state machines, datapath, crossing clock domains; ILA	**HW#2** **Tutorial3**
Week 5 9/25-9/29	Synchronizing across clock domains FPGA technology	**Tutorial4:** Understanding AXI4-Light	**HW#3**
Week 6 10/2-10/6	Coding style review Term exam 1: VHDL, modeling of combinational and sequential circuits		**Tutorial4** **Exam1**
Week 7 10/9-10/13	No classes on Monday and Tuesday - Fall break UART design; FIFO models; AXI	**Lab2.** USB/UART, data transmission between PC (Matlab) and FPGA, FIFO buffering, AXI	**Lab1**
Week 8 10/16-10/20	File IO, subprograms, writing advanced test benches, modeling delays		
Week 9 10/23-10/27	Advanced VHDL concepts, modeling for specific optimization goal	**Lab3.** Interfacing memory, data transmission between PC (Matlab), FPGA with FIFO buffering and memory. **Tutorial 5:** Memory tutorial	**Lab2**
Week 10 10/30-11/3	Modeling for specific optimization goal Coding style review		
Week 11 10/6-11/10	Median filter	**Lab4**. Modeling and implementation of median filter for image restoration	**Lab3** **Tutorial 5**
Week 12 11/13-11/17	Timing analysis, Synthesis optimizations, pipelining, and retiming		**Exam2**

	HW#4 Term exam 2: Advanced VHDL concepts, synchronization		
Week 13 11/20-11/24	Introduction to hardware-software co-design using FPGAs and embedded processors; Microblaze example Research paper selection No classes on Wednesday, Thursday and Friday – Thanksgiving Holiday	**Lab5.** Microblaze, profiling, median filter in C and as a hardware accelerator	**HW#4** **Research paper selection**
Week 14 11/27-12/3	Design of a custom hardware accelerator		**Lab4**
Week 15 12/6-12/10	Pipelining and retiming		
Week 16 12/11	Term exam 3: Timing analysis, design optimizations, tools		**Lab5** **Exam3**
Exams week 12/18 7pm-9.30pm	Presentations on research papers		

Academic Honesty: Although students are strongly encouraged to talk with each other and with the instructor to learn the course material, each student must individually complete work submitted for grading, including quizzes, exams, graded homework, and lab exercises. No grading credit is earned for work not completed independently and completely by the individual student. Copying assignments (including programs where various changes are made to make them "different") will not be tolerated; all students involved in such copying will receive a grade of zero for copied assignments, regardless of who copied from whom.

While the world of Generative AI such as GPT, keeps evolving, in this class we adopt a no-use policy for GPT in all graded course assignments.

RIT syllabus required policies

- **Academic integrity statement**: All conduct in this course is governed by the KGCOE Academic Honesty Policy, RIT Honor Code (P03.0), and RIT Student Academic Integrity Policy (D08.0).
- **Academic adjustments statement**: RIT is committed to providing academic adjustments to students with disabilities. If you would like to request academic adjustments such as testing modifications due to a disability, please contact the Disability Services Office (DSO). Contact information for the DSO and information about how to request adjustments can be found at `http://www.rit.edu/dso`. After you receive academic adjustment approval, it is imperative that you contact the instructor as early as possible to work out whatever arrangement is necessary.
- **Title IX statement**: Title IX violations are taken very seriously at RIT. RIT is committed to investigate complaints of sexual discrimination, sexual harassment, sexual assault, and other sexual misconduct, and to ensure that appropriate action is taken to stop the behavior, prevent its recurrence, and remedy its effects. Title IX rights and resources at RIT can be found at `http://www.rit.edu/fa/compliance/content/title-ix` .