Course management Information

EE-8217: RECONFIGURABLE COMPUTING SYSTEMS

1. Objectives

This course is designed to offer an introduction in the theory and engineering design principles of the modern Reconfigurable Computing Systems (RCS) – one of the most rapidly growing sectors of the high-performance computer technology. The emphasis is in understanding of the concepts of architecture reconfigurability, programmable logic devices and adaptation of the RCS architecture to the task algorithm and data structure. The course covers hardware basics of the modern RCS – fine and coarse-grained programmable logic devices: Field Programmable Gate Arrays (FPGA) and Coarse-Grained Reconfigurable Arrays (CGRA). The overview of RCS architectures and areas of their application also is provided. Languages and compilers for the RSC are other aspects to be covered in this course. Course gives brief description of RCS application in DSP, Video / Image Processing and Supercomputing applications. Then the RCS development, virtual components design and system integration processes will be discussed including high and low level synthesis, simulation and verification stages.

2. Course Outline

WEEK	LECTURE	PROJECTS
1	**Introduction**: Course content & organization, Course text and marking scheme. Web site	
2	**Definition of RCS**: the Reconfigurable Computing Systems (RCS) and RCS classification. **RCS Architectures:** Fine-grain and Coarse-grain architectures. RCS-supercomputers, Massively parallel FPGA array, Stream processing RCS, Hybrid RCS and Embedded RCS architectures.	**Project 1:** Announcement of projects, project format and report organization
3	**RCS component basis:** Field Programmable Gate Array (FPGA) and Coarse-Grained Reconfigurable Arrays (CGRA) architectures. Modern FPGA architectures: DSP functional blocks, embedded processors, block-RAM, Giga-bit serializers, etc.	**Project 1:** Selection of project topic
4	**Languages and Compilers for RCS:** Algorithmic languages for RCS, Hardware Description Languages (HDL). High-level of Compilation and low-level design flow	**Project 1:** Literature research of 5 papers
5	**RCS configuration:** Application segmentation and RCS resources partitioning: spatial and temporal. Multiplexing vs. run-time reconfiguration. Spatial and temporal reconfiguration mechanisms in RCS	**Project 1:** Analysis of the state-of-the-art.
6	**RCS Applications:** DSP, Video/Image processing Cryptography and Supercomputing applications. Functional restoration over reconfiguration in RCS.	**Project 1:** Submission of the Project report

WEEK	LECTURE	PROJECTS
8	**RCS Development:** Application and specification analysis. Concept of High-level synthesis of RCS architecture in multi-parametric design space.	**Project 2:** Selection of project topic
9	**RCS development:** Intellectual Property (IP) cores. Concept of Virtual Hardware Component (VHC) and Application Specific Virtual Processor (ASVP). Spatial and temporal partitioning of FPGA resources	**Project 2:** Development of the block-diagram, symbol and HDL source code (Version 1)
10	**RCS implementation:** Virtual components (VHC) design process: High-level synthesis of VHC and optimization, VHC data-path and control unit design, Low-level synthesis, simulation and verification of VHC.	**Project 2:** Simulation and design verification. Corrections in the HDL-source code (Version 2).
11	**RCS implementation:** Component integration in space and in time domain Determination of reconfiguration scheme and associated loading mechanisms (special & temporal partitioning) for the RCS	**Project 2:** Hardware emulation on the FPGA platform. Debugging the design and first run of the system.
12	**RCS verification and debugging** Synthesis, Simulation and Verification tools. Virtual instruments & performance analyzers.	**Project 2:** Demonstration of the complete system
13	• **Course review & Final Examination**	**Project 2:** Technical report submission

3. Course Text

1. Maya B. Gokhale and Paul S. Graham, "Reconfigurable Computing. Accelerating computation with Field Programmable Gate Arrays", Springer, 2005, ISBN-10 0-387-26105-2
2. Reference text: Clive Maxfield, "The Design Warrior's Guide to FPGAs", ELSEVIER, ISBN 0-7506-7604-3

4. Marking Scheme

Project 1 = 25 %
Project 2 = 25 %
Final Exam = 50 %

All of the required course specific written reports will be assigned not only on their technical or academic merit, but also on the communication skills of the author as exhibited through these reports.

Instructor: Dr. Lev Kirischian, P.Eng.
Office: ENG 432
Telephone: (416) 979-5000 ext. 6076
E-mail: lkirsch@ee.ryerson.ca
Web: www.ee.ryerson.ca/~lkirisch