Digital System Design
Course Syllabus

This course contributes to the requirements for the Degree of MSc in Computer Science

Title of the Academic Program	Master's Degree Program *Digital intelligent control systems* (delivered in English)
Type of the course	core /mandatory
Course period	2 semesters First semester: from February, the 1st to June, the 1st (18 weeks) Second semester: from October, the 1st to February, the 1st (18 weeks)
Study credits	8 ECTS credits
Duration	288 hours
Language of instruction	English
Academic requirements	• BSc degree in Computer Science or equivalent (transcript of records), • good command of English (examination certificate or another formal document). **Prerequisites:** **To know:** digital electronics, hardware interfaces and peripheral devices; **Possess:** programming

Course Overview

Description

"Digital system design" is a core course.

The course will advance student skills in designing digital systems with Programmable Logic Devices (PLD), such as Field Programmable Gate Arrays (FPGA). FPGAs are applied in a wide range of areas such as digital signal processing, video and image processing, vehicle control, aerospace engineering, military applications and artificial intelligence. FPGA is a flexible, widely used platform for rapid prototyping. Using FPGAs for designing hardware has proved to be a good approach to balance and reduce the time required for implementation and prototyping.

FPGA-based design requires the skills of special design methods and the use of professional CAD systems. The course is highly technological and based on specialized software and hardware. Students learn the Verilog Hardware Description Language and master the FPGA design process using Intel Quartus Prime Design software. The course labs are based on TERASIC evaluation boards with Intel / Altera FPGAs.

The aim of the course is to teach students to develop, test, debug a digital system on FPGA using hardware description language (Verilog) and computer aided design tools.

Course Objectives

- to familiarize students with the architectural features of modern FPGA;
- to acquaint students with the hardware description languages (HDL);
- to teach students to use computer aided tools and techniques for designing FPGA/PLD-based devices.

Learning outcomes

By the end of the course, students ***will know:***

- structural features and applications of the most common FPGA / PLD families;
- basics of hardware description languages (Verilog HDL);
- general principles and approaches to debugging and verification of digital systems;

By the end of the course, students ***will be able to:***

- choose the most appropriate class of FPGA to solve a problem;
- develop algorithms and implement them on FPGA using Verilog hardware description language;
- perform modeling, optimizing and debugging for a FPGA-based module;
- perform synthesis and analysis of testbenches for IP blocks using software and hardware tools.

By the end of the course, students ***will possess***:

- the necessary skills to design FPGA-based embedded devices using computer aided tools and techniques

Course Structure

Learning Activities	**Hours**
Lectures	36
Seminars	36
Assignments	180
Final Exam (including preparation)	36
Total study hours	**288**

Detailed Schedule

Week	Lectures	Seminars/ Assignments	Hours Lec/Lab/HA
Semester 1			
1-2	**Introduction.** Course Overview. Basic information on Application-Specific Integrated Circuit (ASIC), PLD and FPGA.	Getting start. CAD tools for FPGA-based design: Quartus Prime	2/2/6
3-6	**VLSI design flow and CAD.** Hardware description languages. The syntax and semantics of the Verilog hardware description language.	Quartus Prime interface. Scheme input. Full cycle of FPGA project implementation. FPGA programming. FPGA design testing in "FPGA-in-loop" mode	4/4/20
7-12	**HDL Verilog for FPGA-based design.** Synthesized subset of the Verilog language. Using Verilog in FPGA project design. Combinational logic. Sequential logic. Final state mashines.	Describing combinational circuits in Verilog. Describing standard sequential (synchronic) designs (counters, registers, memory units). FSM design	6/6/30
13-14	**HDL code reuse and IP cores.** Parameterized modules. Code reuse. IP-cores. Soft Processors.	Simple Soft Processor design	2/2/8
15-18	**Digital system testing and debugging.** Fundamentals of digital system testing and debugging. Methods of module debugging and verification. Test bench development.	ModelSim interface. Developing and applying a simple testbench. Developing and applying an automatic testbench.	4/4/20
Semester 2			
1-4	**Time constrains and limitations.** Constrains and limitations in digital systems. Metastable. Data transfer between clock domains.	FPGA design time analyzing using Quartus Timing Analyzer	4/2/10
5-8	**FPGA design optimization.** Area, frequency, power consumption. Improving reliability. Parallel, concurrent and pipelined data processing on FPGA.	FPGA design optimization	4/4/16
9-12	**FPGA-based Systems-on-Chip.** Soft-processors. Interface and protocol implementation on the base of FPGA.	Serial interface. Universal Asynchronous Receiver-Transmitter (UART) controller.	4/4/16
13-14	**AI and FPGA.** Artificial NeuroNet (ANN) implementation.	Individual project	6/8/54
15-16	**Using FPGA in DSP and control systems**.		
18	**Conclusion.** State-of-art and the future of the field.		
	36	**36**	**252**
36	**Final Exam**		**36**

Course Instructor and Tutor, contact information

Natalia J. Sirotinina,

Ph.D. in Engineering, Associate Professor, Computer Science Dept., Head, Laboratory of Microprocessor Systems
School of Space and Information Technologies
Siberian Federal University

e-mail: nsirotinina@sfu-kras.ru
Google Scholar: https://scholar.google.ru/citations?user=qPHGN3cAAAAJ&hl=ru
Additional information is available at:
http://structure.sfu-kras.ru/node/2043#main

Assessment

Assessment strategy	Points, max	Evaluation criteria
Tests	20	Test questions
Lab works	20	Practical questions
Individual Project	30	FPGA design code, report on the project, presenting the project
Final exam	30	2 questions that require preparatory reading and knowledge of the concepts explained

Grading policy for final assessment is:

- A (excellent work) 91–100 points
- B (above average) 81–90 points
- C (average) 71–80 points
- D (below average) 50–70 points
- F (failed) < 50 points

Attendance Policy

The course is designed to use e-learning and distance learning technologies.

The course can be implemented in two versions: classroom lessons or distance learning.

If the course is implemented as classroom lessons: students are expected to attend classes regularly. In case of missing an in-lab activity a student should perform additional work submitted to the instructor within a week after a class was missed.

If the course is implemented in a distance format:

1. It is recommended to attend online lectures.

If a student skips an online lecture, he or she must pass the e-course element "Lecture with test questions" on the relevant topic.

2. Timely submissions of work reports are anticipated.

If the work is not completed on time, you must contact the teacher through the e-course message indicating the reason for the delay and the estimated deadline. No more than 3 postponements are allowed.

3. Final exam is held in the format of videoconferences. It is mandatory and can only be rescheduled for good reason.

Web page of the course

Course materials and required reading materials are available on the webpage of the course **Digital System Design**, SibFU E-learning portal, www.e.sfu-kras.ru . You must be logged in to access this course.

Core reading

Books

1. Monk, S. (2016). Programming FPGAs: Getting Started with Verilog. McGraw Hill Professional.
2. FPGAs For Dummies, 2nd Intel Special Edition Published by John Wiley & Sons, Inc
3. Harris, S., & Harris, D. (2012). Digital design and computer architecture Morgan Kaufmann.
4. Andina, J. J. R., De la Torre Arnanz, E., & Valdes, M. D. (2017). FPGAs: fundamentals, advanced features, and applications in industrial electronics. CRC Press.
5. Kilts, S. (2007). Advanced FPGA design: architecture, implementation, and optimization. John Wiley & Sons.

Documentation

1. Intel Quartus Prime Software Download and Installation Quick Start Guide
   Link: https://fpgasoftware.intel.com/static/quick_start_guide/quick_start_guide_20.1_en.pdf
2. Intel Quartus Prime Standard Edition User Guide: Getting Started
   Link: https://www.intel.com/content/www/us/en/programmable/documentation/yoq1529444104707.html
3. DE2-115 User manual Link: https://www.intel.com/content/dam/altera-www/global/en_US/portal/dsn/42/doc-us-dsnbk-42-1404062209-de2-115-user-manual.pdf
4. IEEE standard Verilog hardware description language - IEEE Std 1364-2001
5. ModelSim-Altera Software Simulation User Guide
   Link: https://www.intel.co.jp/content/dam/altera-www/global/ja_JP/pdfs/literature/ug/ug_gs_msa_qii.pdf
6. Intel Quartus Prime Pro Edition User Guide: Timing Analyzer
   Link: https://www.intel.com/content/www/us/en/programmable/documentation/psq1513989797346.html
7. Intel oneAPI DPC++ FPGA Optimization Guide file
   Link: https://software.intel.com/content/dam/develop/external/us/en/documents/oneapi-dpcpp-fpga-optimization-guide.pdf
8. Intel Introduction to Deep Learning Workbench Link
   https://docs.openvinotoolkit.org/latest/workbench_docs_Workbench_DG_Introduction.html

Facilities, Equipment and Software

Software:

- The Intel® Quartus® Prime Lite Edition. Free, no license required.

Equipment: evaluation boards

- Evaluating board TERASIC DE2-115 – Cyclone IV (EP4CE115F29C7N),
- Evaluating board TERASIC DE-1-SoC- Cyclone IV (5CSEMA5F31C6N).