2010 12th Biennial Baltic Electronics Conference (BEC2010)
Tallinn, Estonia, October 4-6, 2010

Teaching Digital Systems Using a Unified FPGA Platform

Olli Vainio, Erno Salminen, and Jarmo Takala

*Department of Computer Systems, Tampere University of Technology, P.O. Box 553, 33101 Tampere, Finland,
E-mail: firstname.lastname@tut.fi*

ABSTRACT: This paper presents how the course syllabus for digital and computer systems education at our department has been adapted to utilize the same FPGA-based hardware platform in many laboratory exercises. The reform has been carried out in order to streamline the learning towards core content instead of studying multiple complex design software and hardware environments. Each student is given a package including an FPGA board and related design tools. In addition to the course exercises, the students are encouraged to use it also for thesis projects and hobbies. In this paper, we describe the syllabus, the key courses, and the platform used in education of microelectronic systems. Examples of completed thesis works are given.

1 Introduction

The Department of Computer Systems (DCS) [1] is a part of the Faculty of Computing and Electrical Engineering in Tampere University of Technology (TUT), Finland. The department provides education mainly in the topical areas of design of digital systems, processor and computer architectures, embedded systems, positioning and navigation, and wireless sensor networks. The department has a faculty of five professors and about 80 other employees.

In 2008, a project was launched to modernize and enhance education particularly regarding the laboratory exercise works in the courses. The challenge was that the number of courses is large and the exercises, evolved over many years, were disconnected and in some cases outdated. The objective was to use a common hardware platform in several consecutive courses. This concentrates the education towards core contents and reduces the time needed to study the manuals of diverse platforms and tools. Moreover, students in DCS may borrow a personal unit to implement exercises and hobby projects also outside the classrooms. The aim is to motivate the students by providing experiences of successful tangible working hardware, since pencil-and-paper designs or simulations alone are not sufficient learning experiences.

This paper describes the syllabus of the department, the chosen platform, the key courses, example projects, and the experiences from the reform. Note that the core contents in courses are not specific to the chosen FPGA platform. For example, VHDL, RTL synthesis, IP reuse, and HW/SW co-design apply ASIC design equally well.

2 Syllabus of Computer Systems

The university uses ECTS (European Credit Transfer and accumulation System) where nominal yearly accumulation is 60 cr. This means that the B.Sc. degree, which includes 180 credits, may be completed in three years. The M.Sc. degree, 120 cr., takes two additional years to complete. The structure of teaching offered by DCS is outlined in Fig. 1 and the following description is mainly for the degree program in Information Technology (IT). The common basic studies of the degree program contain 110 ECTS credits including three compulsory DCS courses:

1. Basic Digital Circuits	(4 cr), 1st year
2. Microprocessors	(5 cr), 1st year
3. Computer Architecture I	(5 cr), 2nd year.

The numbers of students in these courses are very large, as they are also taken by students from other degree programs.

The B.Sc. degree must include two modules of subject related studies, each 25 cr. There are 17 modules to choose from. The DCS subject study module includes the following courses:

1. Digital Design	(5 cr), 2nd year
2. Digital Systems Laboratory	(3 cr), 2nd year
3. Implementation of Digital Systems	(8 cr), 3rd year
4. Computer Architecture II	(5 cr), 2nd year
5. Computer Arithmetic	(4 cr), 3rd year.

At the M.Sc. level, at least one advanced study module of 30 cr. is chosen. The 5 advanced modules offered by DCS are listed in Fig. 1. The subject study module may also be taken at the M.Sc. level.

As shown in Fig. 1, the department also has two study modules targeted to foreign students holding a B.Sc. degree on the same or a closely related area [2]. The language of teaching for the international students is English, while the other courses are given either in Finnish or in English. A total of 28 regular courses and few seminars are taught at DCS. Most courses last one semester, that is 14-16 weeks plus an exam. There is a strong emphasis on exercise works, both using computers and theoretical ones done with pen and paper. Computer exercises are closer to everyday work and they also allow automated checking (at least partially).

978-1-4244-7359-5/10/$26.00 ©2010 IEEE

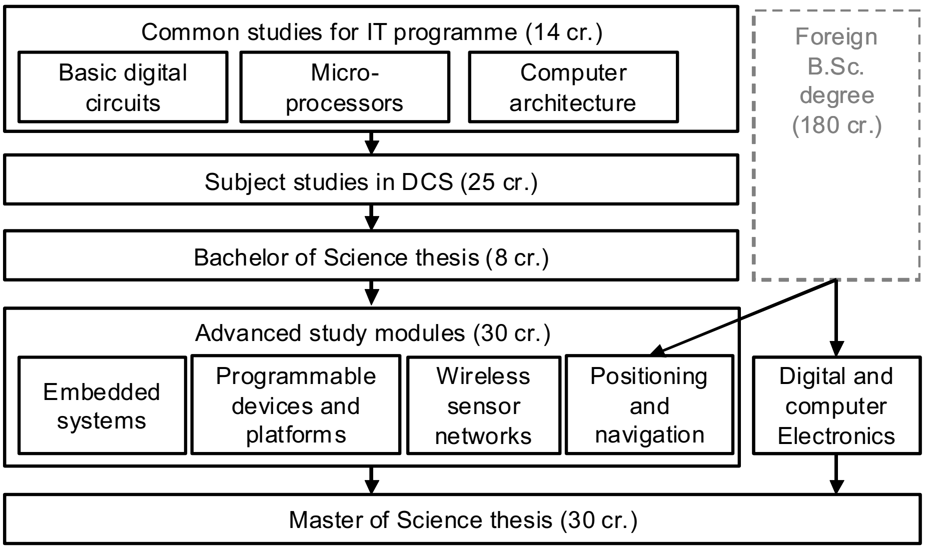
Common studies for IT programme (14 cr.)
Basic digital circuits
Micro-processors
Computer architecture
Foreign B.Sc. degree (180 cr.)
Subject studies in DCS (25 cr.)
Bachelor of Science thesis (8 cr.)
Advanced study modules (30 cr.)
Embedded systems
Programmable devices and platforms
Wireless sensor networks
Positioning and navigation
Digital and computer Electronics
Master of Science thesis (30 cr.)

Fig. 1. Teaching provided by the Department of Computer Systems.

3 Chosen FPGA Platform

The platform must have enough capacity to facilitate actual meaningful applications. The chosen hardware platform is the Altera DE2 development and education board [3] that is part of Altera's university program. It is based on the Altera Cyclone II EP2C35 FPGA which features over 30 000 logic elements (LE), 480 kb RAM, embedded multipliers and Phase-locked-loops (PLLs). FPGA is large enough to host e.g. several synthesizable soft-core processors allowing one to build own multiprocessor-system-on-chip. Operating frequency is usually around 100 MHz.

The development board includes, e.g., external memories: 8 MByte SDRAM, 512 kByte SRAM, 4 MByte Flash. The user interface uses switches, buttons, LEDs, LCD and 7-segment displays; whereas the other interfaces include RS-232, infrared, video/audio in/out, PS/2, Ethernet, and USB. Expansion headers allow 76 general-purpose signal pins. Design entry can be carried out in a PC environment using VHDL description or schematics. The package includes Altera's Quartus II synthesis tool. The FPGA is configured either through the USB interface or by storing the configuration in the flash memory, which enables stand-alone operation. The functionality of the implementation can be inspected by a synthesizable logic analyzer called SignalTap.

4 Key Courses

This section summarizes the contents and learning outcomes of the key courses of the DCS department that utilize the common hardware platform or the related design tools. The first 5 belong to B.Sc. studies and the rest to M.Sc. studies. The exercises are mandatory in all these courses; either the students can select certain tasks from a list or all of them must be completed. Bonus points can be earned for doing little extra work or doing the work exceptionally well. Bonuses help getting a better grade and motivate the students to dig deeper into the problems at hand.

In addition to the courses mentioned in this paper, the department offers several others that do not utilize the FPGA board. Those courses are either theoretical by definition or focus on other kinds of environments, such as microcontrollers or signal processors.

4.1 Basic Digital Circuits (4 cr)

This course is taken in the very beginning of the freshman year. The objective is to introduce the basics of combinational and sequential logic circuits and binary arithmetic. The principle of hierarchical design is also introduced. After completing the course, the student can design a simple finite state machine starting from a written specification and finishing in a gate-level implementation. He/she can also carry out basic arithmetic operations on binary numbers and identify the corresponding hardware components.

The exercises in this course are mostly pencil-and-paper design tasks in class and as homework. Bonus points are given for implementing a scrolling text on 7-segment displays. We are currently preparing a simple block diagram based exercise to better demonstrate the capabilities of the board. Personal FPGA packages cannot be given due to the large number of students.

4.2 Digital Design (5 cr)

This is a second-year course with the objective to advance skills in digital design and implement a functional system in hardware. The student will be able to identify and use different forms of design description and choose a suitable method for different parts of a digital system. He will learn to use the main features of EDA (Electronic Design Automation) tools for schematic design and simulation. The testbenches are provided by the assistants.

The exercises consist of theoretical problems as well as computer exercises. During the modernization, we increased the number of FPGA exercises and tied them together to implement a pocket calculator on DE2. This is the first course with compulsory exposure to the FPGA board and where the students may borrow one for themselves. So far, about 100 DE2 boards have been borrowed. Students may select the exercise tasks they wish to implement from a large pool. Half of the tasks must be completed and bonuses are earned by doing more.

4.3 Implementation of Digital Systems (8 cr)

This course continues from Digital Design and introduces VHDL language as a design entry method. The students learn how to transform a system specification into hardware description language (HDL), verify the functionality with their own testbenches, and synthesize the description for FPGA.

They are taught good practices in VHDL coding style and will understand the correspondence between the textual description and hardware implementation. The design work is to create a simple audio synthesizer. The work is partitioned and balanced into weekly exercises. Bonuses are earned by implementing fancier features, such as more generic components or more thorough testbenches.

4.4 Digital Design for FPGA (6 cr)

This course is given to international students and combines elements from courses 4.2 and 4.3 mentioned above. It covers the design flow of digital circuits, including schematic capture, basics of VHDL, verification, and FPGA implementation (a pocket calculator).

4.5 Computer Arithmetic I (4 cr)

The course is targeted to second or third year students. Its objective is to introduce the common number systems in computers and methods to implement arithmetic operations. The student learns how to carry out basic arithmetic operations (addition, subtraction, multiplication, division) at algorithm level, basic performance analysis, and the basic speed enhancement techniques. The exercises include theoretical homework problems discussed in class as well as implementation tasks for the FPGA board, e.g., multiplier modules.

4.6 ASIC Design I (5 cr)

The ASIC (Application Specific Integrated Circuit) technologies and design flow are discussed. The student learns how to carry out the front-end part of an ASIC design project, i.e., the design steps from circuit specification to a synthesized netlist. Knowledge of VHDL is required.

The exercises cover system partitioning, writing specifications, documentation, VHDL descriptions, test benches, and verification and synthesis of the descriptions. There is more freedom than in earlier courses since students specify the system themselves. The FPGA board is used for demonstrations of a digital stopwatch clock.

4.7 SoC Design (5 cr)

The objective is to familiarize the students with the design of complex digital systems, with the focus on SoC (System-on-Chip) and embedded systems. The most important concepts and phases in system design are introduced, especially software-hardware codesign, Intellectual Property (IP) reuse, and platform-based design.

Quartus synthesis software includes the SOPC Builder tool that helps composing the full system from various IP blocks. In the exercise project, a video compression system is implemented on FPGA. First it is implemented purely with software running on a synthesizable RISC (Reduced Instruction Set Computer) processor called Nios II. After profiling, the encoder is accelerated by integrating a dedicated hardware component for discrete cosine transform. Students do not write VHDL but integrate and parameterize IP components. Students can earn bonuses for good documentation, code clarity, and obtaining certain encoding speed. Moreover, the group with the fastest implementation earns extra points. This has stirred friendly competition among the most active students and, even better, they have come up with some optimizations that the assistants did not think of.

4.8 SoC Platforms (5 cr)

This course continues from SoC Design and teaches to identify various layers of a SoC: application, software platform and hardware platform. For each layer, typical components and interfaces are covered, as well as factors affecting real-time behavior. The student learns to utilize the resources of modern SoC platform chips for the design and implementation of an application with real-time requirements using a real-time operating system.

The exercises use a platform that consists of two Nios II processors connected through a shared bus. An application program - a reaction test game - is programmed such that the functionality is divided between the two processors and RTOS threads. Students must meet (soft) real-time requirements and measure various metrics, such as communication delays and utilization of the CPUs.

4.9 Project Work and B.Sc. seminar (5-8 cr)

This course is compulsory for the students doing the M.Sc. thesis in DCS. The students learn how to plan, carry out and document a project as a member of a team. They also learn how to communicate and negotiate with different interest groups, participate in project meetings and give presentations. Status reports are given as seminar presentations. Several projects and B.Sc. theses have utilized the DE2 board, for example, a WAV audio player and a synthesizable microcontroller.

5 Case Studies

In this section, we give examples of thesis works that have been carried out using the DE2 platform.

5.1 Recursive FIR Filter Structures

A new approach to piecewise-polynomial approximation and recursive implementation structures for linear-phase Finite Impulse Response (FIR) filters have recently been proposed. In a M.Sc. thesis [4], new structures were developed for all four types of linear-phase FIR filters using the DE2 platform and VHDL. Narrowband low-pass filters and narrowband differentiators were used as design examples to demonstrate the functionality and efficiency of the implementations.

5.2 Action Game Doom

Doom is a classic action game, originally developed for IBM PC in 1993 by Id Software. Source codes - around 70000 lines of C - have been published in 1999 under the GNU GPL license. To run the game, DE2 board acts as a simple personal computer and it has natively all necessary interfaces for a keyboard, video and audio outputs. A functional implementation of Doom in DE2 was demonstrated in a B.Sc. thesis [5].

The system is described with the SOPC builder. The B.Sc. work implemented some blocks and used many from the library, for example Nios II. The CPU is placed into the FPGA and run at 100 MHz and, hence, there is no need for an external processor. The performance was adequate, reaching over 30 frames per second display refresh rate.

Moreover, a boot loader program was developed and permanently stored in the flash memory, enabling stand-alone operation of the system. Programs can be downloaded from portable SD (Secure Digital) memory devices that utilize the FAT file system.

6 Conclusions

Nine courses offered by the Department utilize the same FPGA based platform and 2-3 courses more are planned. Table I summarizes the courses that mostly utilize the common FPGA platform (actual course numbering differs from the notation used in this paper).

Table I
Key courses on digital design.

Course	Prereq.	FPGA content	Exercise focus
4.1	-	Demo	Pen and paper
4.2	4.2	Medium	Pen, EDA
4.3	4.1, 4.2	Large	VHDL
4.4	-	Medium	EDA, VHDL
4.5	4.1, 4.2	Small	Pen, schema
4.6	4.3	Large	VHDL
4.7	4.3	Large	HW IP blocks + C
4.8	4.7	Large	HW IP blocks + C
4.9	-	Optional	Case-dependent

A unified platform is beneficial since students do not need to learn a new environment for every course taken. The students have been excited to implement their designs in practice and to interface various peripherals in the board. This is reflected in the number of thesis topics proposed by the students themselves and clearly increased motivation during the exercises when students see and hear their own design running. Positive comments have been received also from the local companies.

Acknowledgement

The curriculum development work was supported by the Technology Industries of Finland Centennial Foundation. We thank all those who have contributed to the project.

References

[1] Department of Computer Systems, TUT, web page, http://www.tkt.cs.tut.fi/

[2] International Master's degree programme in Information technology, TUT, web page http://www.cs.tut.fi/uniprogramme/

[3] Altera Corporation, web page, http://www.altera.com/

[4] T. Taurén, *Implementation and Testing of Recursive Polynomial FIR Filters*, M.Sc. thesis, Tampere University of Technology, 2009 (in Finnish).

[5] J. Kulmala and J. Järvinen, *Doom on the FPGA Development Board,* B.Sc. thesis, Tampere University of Technology, 2009 (in Finnish).