

FPGA (Field Programmable Gate Arrays) – Based Systems-On-a-Programmable-Chip (SOPC) Development for Educational Purposes

Valentina Rankovska

Abstract – An approach to studying Systems-On-a-Programmable-Chip (SOPC) based on Field-Programmable Gate Arrays and microprocessor cores is presented, considering the complexity of the circuits and design software. Altera's soft microprocessor core – Nios II, design software Quartus II and SOPC Builder and development board DE2-70 are used.

Keywords – Systems-On-a-Programmable-Chip (SOPC) design, Field Programmable Gate Arrays - FPGA, Microprocessor cores, Nios II, Education.

I. INTRODUCTION

The innovative component base in the area of the digital and microprocessor circuits - Field Programmable Gate Arrays (FPGA), intelligent peripherals (controllers, sensors, actuators, etc.), modern interface standards, and the development technology and tools, affect the embedded systems development. Their application areas continuously expand which leads to more complex architecture and increasing requirements to their features and parameters.

At the same time the design approaches, stages and tools develop – they renovate and become more complex, upgrading the traditional ones. On the other hand the incomplete compatibility of hardware and software development tools sometimes leads to errors which are difficult to be found, especially when using several software tools simultaneously.

The FPGA datasheets and development tools literature extends from several hundreds to more than 1000 pages.

All mentioned above makes difficult studying them by the students at the bachelor's degree at the Technical University.

In the present paper an approach for studying development stages of Systems-On-a-Programmable-Chip (SOPC)/ embedded microprocessor systems based on FPGA and microprocessor cores is suggested.

II. SYSTEMS-ON-A-PROGRAMMABLE-CHIP (SOPC)/ EMBEDDED SYSTEMS IN FPGA. MICROPROCESSOR CORES.

Using microprocessor (Central Processor Unit -CPU) cores in FPGA makes possible the implementation of embedded

Valentina Rankovska is with the Faculty of Electrotechnics and Electronics at Technical University of Gabrovo, 4 Hadji Dimitar str., Gabrovo 5300, Bulgaria, E-mail: rankovska@tugab.bg.

systems with considerable advantages in the cases when they include a large amount of digital circuits or when it is necessary to use many peripheral modules, some of which missing in the traditional microcontrollers.

There are two types of SOPCs according to the architecture of the FPGA – FPGA consisting only of an array of configurable logic blocks and FPGA including configurable logic blocks and hardware implemented blocks – microprocessor(s) and peripherals. Except Altera there are many partners supplying Intellectual Property (IP) cores for software implementation of peripherals – memory controllers, transceivers, arithmetic blocks, signal processing, protocol interfaces, etc.

CPU cores can be classified as follows [1]: according to implementation – *hardware and software*; according to architecture – *based on traditional architecture and unique*; according to the possibilities for reconfiguring in real time – *with hard implemented and reconfigurable architecture*; *commercial and open-cores*, etc.

The study of architecture, features, resources and application of the FPGA in the Microprocessor Circuits and Embedded Systems laboratory in the Department of Electronics of the Technical University – Gabrovo is based on Altera's FPGAs. That is why Altera's soft-microprocessor Nios II is used in the initial study of embedded systems. After achieving some experience the designed systems are expanded with additional peripheral modules and the next stage have to be design of a unique CPU core.

There are many CPU cores for Altera's FPGAs. The features of some of them (recommended on their site) are presented in Table I. The core which occupies the least logical area is Nios II/e. Another advantage is that it is free with software Quartus II v. 9.1 and the following and that is why it is used in the design in section IV.

Nios II is a general purpose RISC processor with 32-bit instructions, address and data busses; 32 general purpose registers; 32 interrupt sources, etc.

III. TECHNOLOGY SYSTEM-ON-A-PROGRAMMABLE-CHIP (SOPC) IN FPGA. DEVELOPMENT TOOLS

Design technology for SOPC with CPU cores follows in general the development stages for arbitrary devices and systems in FPGA [2], [3]. Altera's Quartus II supplies all the stages of the hardware design of the project – creating a project, entering the design, simulation and configuring the FPGA. There are two variants of Quartus II – free and subscription edition. The free edition is used in the laboratory

TABLE I
FEATURES OF CPU CORES FOR FPGA OF ALTERA

Processor Category Features	Cost- and Power-Sensitive Processors			Real-Time Processors		Applications Processors	
	ARM® CortexTM- M1	V1 ColdFire	Nios® II Economy	Nios II Standard	Nios II Fast	MP32	ARM Cortex-A9 MPCore
Maximum performance efficiency (MIPS per MHz)	-	0.93	0.15	0.64	1.13	1.15	2.5
Maximum performance (MIPS/l) at MHz) Cyclone® SoC	80 at 100	84 at 90	30 at 175	90 at 145	195 at 175	140 at 145	4,000 at 800
16-/32-bit instruction set support	16 and 32	16, 32, and 48	32	32	32	32	32, 16-bit Thumb/Thumb2
Level 1 instruction cache	-	-	-	Configurable	Configurable	Configurable	32 KB
Level 1 data cache	-	-	-	-	Configurable	64 KB	32 KB
Level 2 cache	-	-	-	-	-	-	512 KB
Memory management unit	-	-	-	-	Yes	Yes	Yes
Floating-point unit	-	-	-	Floating-point custom instruction	Floating-point custom instruction	-	Double precision
Vector interrupt controller	Yes	-	-	Yes	Yes	-	-
Tightly coupled memory	Up to 64K	-	-	Configurable	Configurable	-	-
Custom instruction interface	-	-	Up to 256	Up to 256	Up to 256	-	-
Equivalent LE	2,500	6,800	600	1,200	1,800 – 3,200	5,500	HPS

because it has nearly full functionality for the supplied devices and is completely suitable for learning purposes.

Depending on the type of the CPU core there are some more stages in the design technology (Fig. 1).

When using a software CPU core it can be entered into the project in several ways:

- When there is an available HDL model (designed or ready) it is added to the rest part of the project using a text editor.
- When a firmware core (such as Nios II) is used together with the hardware design the CPU core is configured by the means of special software – for instance Nios II Eclipse Platform for Nios II.

There is a particular module in Quartus II for generating SOPCs - **SOPC Builder**, which makes easy embedding and configuring a processor core in the design [5]. It is a powerful design tool allowing defining and generating a whole system in a single chip (SOPC) for quite shorter time in comparison to the traditional design approaches. SOPC Builder generates many files defining the system hardware, memory map, simulation model, additional data, etc.

SOPC Builder can be used for creating systems including Nios II processor, some other processor or without any processor.

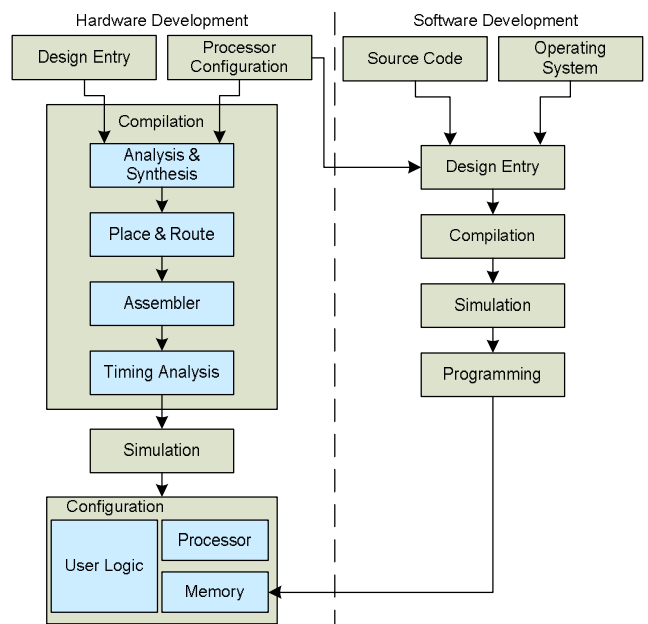


Fig. 1. Design flow for SOPC with FPGA

SOPC Builder includes a component library with processor cores, peripherals (timers, etc.), serial interfaces, and various controllers (off-chip memory, etc.). It is possible to use components (IP cores) of Altera's IP partners and also user components.

CPU core configuring means to define capabilities and parameters such as instruction set (the architecture of the processor), data width, address space, required peripherals, and their features and so on. In general the differences in the variants reduce to the following [4]: *inclusion or exclusion of a feature* – for example hardware multiplier; *more or less of a feature* – for example the volume of the instruction cache memory; *hardware implementation or software emulation* – for example interrupt controller.

Together with the hardware design it is necessary to **design the software** for the CPU core. Further more – in contrast to the traditional microprocessors, when a unique CPU cores will be used it is necessary to make development tools for them – compilers, simulators, in-circuit debuggers, etc.

Altera supports the University education by a special program including discounts for development tools, special software for creating, simulation, loading and real time test of the software for Nios II – *Altera Monitor Program*, many tutorials, example applications, on-line courses, etc.

Downloading the program/ data memory

Downloading the binary file (and data, if any) is possible in several ways depending on the type of the memory used – if on-chip memory is used its initialization is a part of the FPGA configuring. However it is not a large amount and commonly off-chip memory is used. In that case the code and data downloading is made using a special interface and the on-chip memory can be used for a bootloader.

IV. DESIGN OF EMBEDDED MICROPROCESSOR SYSTEM WITH MINIMUM FEATURES FOR EDUCATIONAL PURPOSES

SOPC Builder and the development board DE2-70 of Terasic Technologies Inc. are used for the designed SOPC system. DE2-70 includes FPGA Cyclone II various peripherals, LEDs, switches, LCD display, various types of off-chip memory, IrDA transceiver, 10-bit video-DAC, 10/100 Ethernet PHY/MAC, etc. Therefore DE2-70 has enough resources to be used in the engineering education at an initial learning stage without necessity of additional peripheral modules and devices.

An example system with Nios II implemented in DE2-70 is shown in Fig. 2. The processor Nios II and the necessary interfaces for connection with the other components on the development board are implemented in FPGA Cyclone II. These blocks are interfaced by interconnections and logic, called *Avalon Switch Fabric*. The on-chip memory in FPGA Cyclone could be used as a program memory for short programs for Nios II processor. The access to the SRAM, SSRAM, SDRAM and Flash memories on DE2-70 can be implemented by the means of appropriate interfaces. The parallel and serial interfaces allow implementing the desired input-output ports. A special JTAG UART interface is used to

connect by USB to a personal computer (PC). Together with appropriate software it is called USB-Blaster. Another block, called JTAG debug module, allows the PC to control the Nios II system.

It is possible to download program memory in that way, to test and debug the software by various operations - running and stopping the execution, using breakpoints, collecting trace

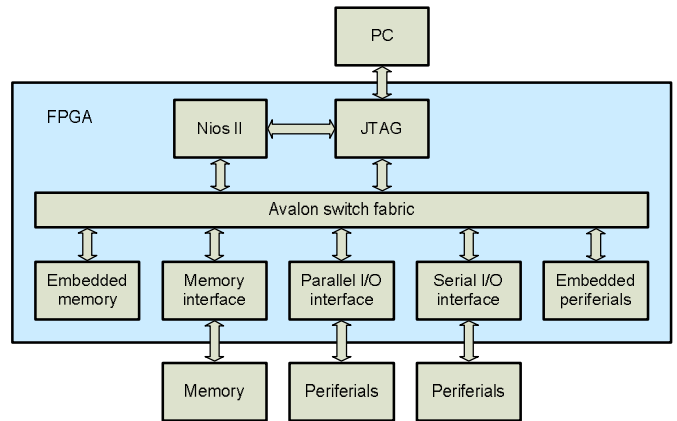


Fig. 2. SOPC with Nios II and DE2-70

data, etc.

All the components in the Nios II systems are modeled in hardware description languages (HDLs). That is why it is necessary for the designer to know some HDL (which needs time and efforts) or to use SOPC Builder for its implementation, by simply choosing and configuring the components needed according to the application requirements.

The design flow in brief is the following:

o **Creating project in Quartus II Web Edition**

The project creation is a stage, which the students had been acquired in advance. At this stage the name of top-level file and entity, the FPGA family and device, and other data are entered.

o **Configuring and generation of the Nios II system in SOPC Builder**

SOPC Builder library components are used in our case: processor Nios II/e, on-chip memory, two parallel interfaces – input and output and JTAG interface for connection with the PC to configure the FPGA and design debugging. (Fig. 3)

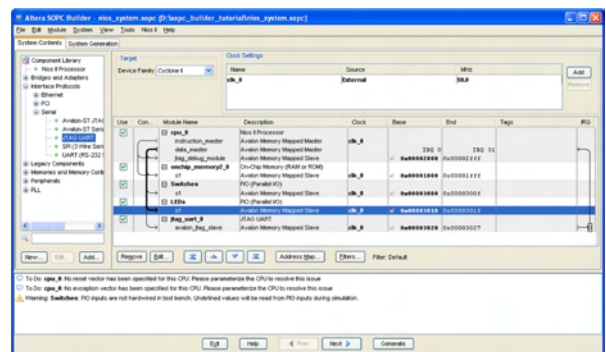


Fig. 3. Nios II system in the SOPC Builder

- Including the SOPC system into a Quartus II project
- SOPC Builder generates corresponding .vhd files for the Quartus II project. They have to be integrated into the top-level .vhd file created using the text editor in Quartus II.
- Assigning pins, compilation the design and configuring the FPGA
- CPU core software design and testing the whole system

The software for the CPU can be written either in Assembler, or in a high level language. Assembler is used in our case to test the functionality of the designed system. Its operation is simple as the aim at this stage of learning is not to study the language – the state of eight switches (SW7-SW0) on DE2-70 is read and it is output on eight LEDs (LEDG7-LED0). *Altera Monitor Program* is used to develop the software, to download it into the memory assigned to the processor and to run the program (Fig. 4 and Fig. 5).

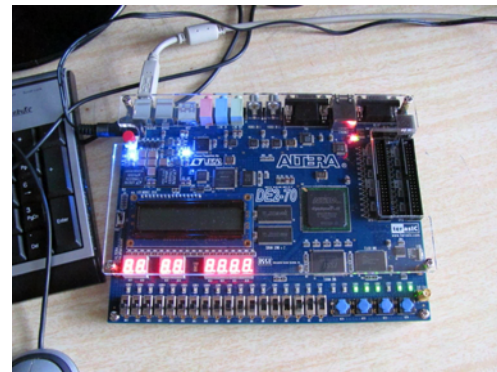


Fig. 5. Testing the Nios II system with the application software

An approach to learning the initial steps in designing FPGA-based SOPCs/ Embedded systems is presented in the paper. These steps are an obligatory stage in studying the complex software development environments, prior to expand the features and architecture of the embedded systems.

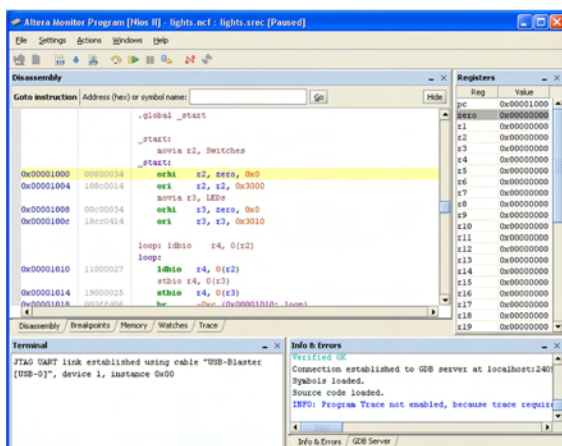


Fig. 4. Nios II software in Altera Monitor Program

V. CONCLUSION

ACKNOWLEDGEMENT

The present work is supported by the Science Research Fund at the Ministry of Education, Youth and Research.

REFERENCES

- [1] V. Rankovska, „Microprocessor cores for complex programmable logic arrays”, Unitech’11, Conference Proceedings, vol. I, pp. I-186 – I-191, Nov. 2011. (in Bulgarian)
- [2] H. Karailiev, V. Rankovska, „Synthesizing Sine Wave Signals Based on Direct Digital Synthesis Using Field Programmable Gate Arrays”, ICEST 2007, Conference Proceedings, pp. 637-640, Ohrid, Macedonia, 2007.
- [3] J. Hamblin, T. Hall, “Using System-on-a-Programmable-Chip Technology to Design Embedded Systems”, 33rd International Symposium on Computer Architecture, vol. 13, no. 3, pp. 2395-2503, Sept. 2006.
- [4] Nios II Processor Reference Handbook, Altera Corp., May, 2011.
- [5] Quartus II Handbook Version 9.1 Altera Corp., May, 2009.