Analysis of Resource Utilization in FPGA Implementation of an Embedded System Using Soft Core Processor.

M.U. Kharat, Ph.D.[1] and V.S. Rahangadale, M.E.[2]

[1]Department of Information Technology, G.H. Raisoni College of Engineering, Nagpur, India.
[2]Department of Computer Science and Technology, G.H. Raisoni College of Engineering, Nagpur, India.

E-mail: mukharat@rediffmail.com[1]
vijayadsc97@yahoo.co.in[2]

ABSTRACT

Embedded systems are application-specific computers that interact with the physical world. Reconfigurable technologies provide designers the opportunity to diminish the life-cycle in embedded system creation. The basic purpose of using reconfigurable technologies is for solving the problem of obsolescence. New emerging capabilities in Field Programmable Gate Array (FPGA), including improvements in power consumption, time delays, and cost per unit device, are enabling us to incorporate these devices in several designs as reconfigurable embedded processors.

The major objective of this work was to implement an embedded system on FPGA using soft core processor and analyze the effect of on-board memories on resource utilization. An audio system has been selected for implementation.

(Keywords: field programmable gate array, FPGA, reconfigurable embedded processors,)

INTRODUCTION

A new technology has emerged that enables designers to utilize a large FPGA that contains both memory and logic elements along with a intellectual property (IP) processor core to implement a computer along with custom hardware for system-on-a-chip (SOC) applications. This new approach has been termed system–on–a–programmable–chip (SOPC). In the past few years, several commercial RISC processor cores have been introduced. Several commercial processor cores are overviewed which can be used in the rapid prototyping of embedded systems.

Circuit design for FPGAs is typically done using a CAD (Computer Aided Design) tool. Modern CAD tools support design entry using several different methods. As the complexity of the circuits grows, Hardware Description Languages (HDLs) become the only practical choice. HDLs support circuit description using high-level language constructs. Low-level implementation details are handled by the CAD tool automatically, so the designer can focus on the design functionality.

A *soft-core processor* is a microprocessor fully described in software, usually in an HDL, which can be synthesized in programmable hardware, such as FPGAs. A soft-core processor targeting FPGAs is flexible because its parameters can be changed at any time by reprogramming the device. Traditionally, systems have been built using general-purpose processors implemented as Application Specific Integrated Circuits (ASIC), placed on printed circuit boards that may have included FPGAs if flexible user logic was required. Using soft-core processors, such systems can be integrated on a single FPGA chip, assuming that the soft-core processor provides adequate performance.

Continuous research has been going into the development of embedded system using soft core processors. These soft core processors are nowadays very popular because they are

reconfigurable. The importance of reconfigurable technology is presented in [8]. The currently available reconfigurable technology, FPGAs, soft-cores, and their applicability in the new embedded systems are described in [5]. Researchers [6] have implemented the Nios soft-core processor from Altera Corporation using Verilog.

The development cycle of embedded systems includes the hardware/software co-design process. It determines which portions of the overall specification should be mapped into the reconfigurable logic and which is retained on the processor. Recent literature [11] describes the use of a reconfigurable architecture platform for embedded control applications. The importance of FPGA application and SOPC development tools for research work is worked out in [20] and [2].

For this audio system, a Wolfson WM8731 audio CODEC of DE2 board is used, it provides high-quality 24-bit audio. It is presented in [16]. The methodology for interfacing the processor core with the audio system is described in [1].

An embedded system has a diverse set of tasks to perform, and although a very flexible language might be able to handle all of them, instead a variety of problem-domain-specific languages have evolved that are easier to write, analyze, and compile. Research from [16] describes various designing languages for embedded system.

TECHNOLOGY OVERVIEW

Field Programmable Gate Array: FPGA devices are programmable to implement arbitrary user logic. To support this programmability, FPGA devices contain three types of resources: logic blocks, I/O blocks, and programmable interconnection.

Most FPGAs contain logic blocks that consist of a lookup table (LUT) and a flip-flop. A LUT can be programmed to implement any logic function of its inputs. A LUT with n inputs is called an *n-LUT*. An n-LUT is internally implemented as a set of 2-to-1 multiplexers, functioning as a 2n-to-1 multiplexer. Multiplexer inputs are programmable, while select lines are used for inputs of the implemented function.

The implementation of large and complex circuits is achieved using a FPGA [8]. This architecture is quite different to CPLD. FPGA do not contain AND neither OR arrays, instead of that, FPGAs are constructed with three main components: logic blocks, I/O blocks, and interconnection wires as is presented in Figure 1. The Logic blocks are built with a small number of inputs and one output; internally the required functions are implemented with the use of Look Up Tables (LUT), memory elements (FFs), and special arithmetic logic support.

As shown in Figure 1, these programmable blocks are embedded in a see of routing channels which contain wires and programmable switches used to set up the desire functionalities connecting the logic blocks.

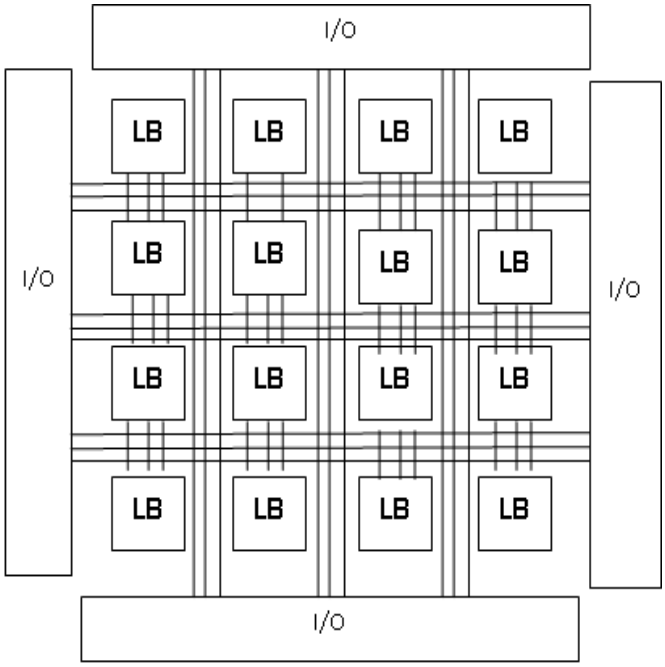

Figure 1: FPGA Architecture.

Soft Core Processor: Presently, reconfigurable platforms, follow the parading, which combines several reconfigurable fabrics with a general-purpose processor. The loosely-coupled architectures enable the efficient use of the RFU

as a coprocessor, exploiting the instruction level parallelism of multimedia and communications specific applications. Today, we are witnessesing the emergance of many commercially embedded processors, as well as supporting and development tools; some of the principal products available are: Altera Nios/NiosII [13], Xilinx MicroBlaze [12], Lattice Mico32 [9], and ESA Leon [10]. They offer memory and logic elements with several Intellectual Property (IP) peripherals for the rapid development of System-on-Programmable-Chip (SoPC).

Furthermore, reconfigurable systems on a chip became a reality with softcore processor, which are a microprocessor fully described in software, usually in a VHDL, and capable to be synthesized in programmable hardware. After analyzing all the four soft core processors, the basic difference found is that the programmable interconnects are different.

NIOS II: Altera's NIOS II [18] has a load-store RISC architecture, in which many architectural parameters can be customized at design time. The user can decide between 16 or 32 bits of width in the data path, register file sizes, as well as cache size, and customize instructions for the performing of user-defined operation in the speeding-up customized hardware.

Those functionalities are supported by the builder development tools and when using the Nios II Integrated Development Environment (IDE) [18] it is possible to build, run, and debug software of several platforms. A general representation of the NIOS II system is presented in Figure 2.

Altera also introduced a SOPC builder [7] for the rapid creation and easy evaluation of embedded systems. The integration of off-the-shelf intellectual property (IP) as well as reusable custom components is realized in a friendly way, diminishing the required time to set up a SoC and enabling to construct and designs in hours instead of weeks.

IMPLEMENTATION METHODOLOGY

The objective of implementing this system is to play an audio file which is present in on-board memory and controlled by NIOS II soft core processor. Many commercial media/audio players use a large external storage device, such as an SD card or CF card, to store music or video files. In this project Altera's DE2-70 board [14] is used which provides the hardware and software needed for on board memory access and professional audio performance so that it is possible to design advanced multimedia products using this DE2 board.

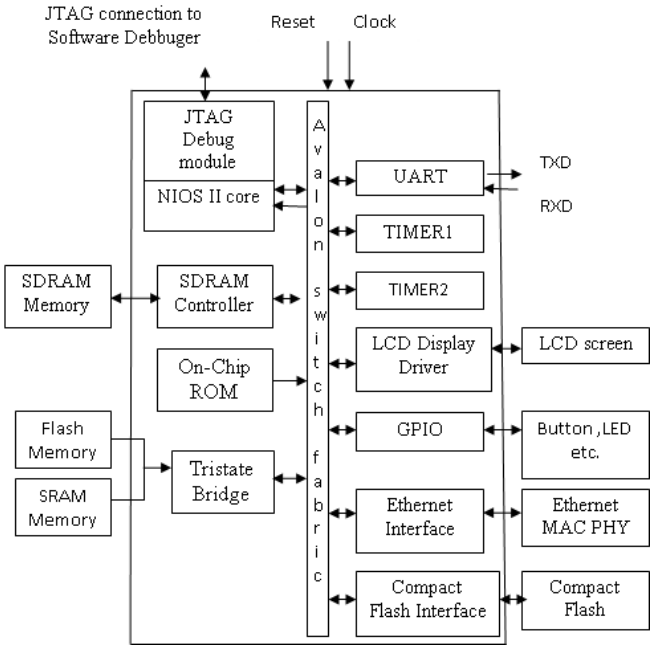

Figure 2: NIOS II Processor System.

The NIOS II processor is used to read the music data stored in the memory and a Wolfson WM8731 audio CODEC [15] is used to play the audio signal. The Audio DAC Controller is available as an IP core in the library of SOPC builder which can be implemented in FPGA. Schematic representation of this audio system is shown in Figure 3.

EXPERIMENTAL FRAME WORK

Tools: For the implementation of the hardware and software parts of the experimental system the following tools were used:

1. Altera's QUARTUS II software [17] and SOPC builder for Configuring and Synthesizing system along with NIOS II processors.

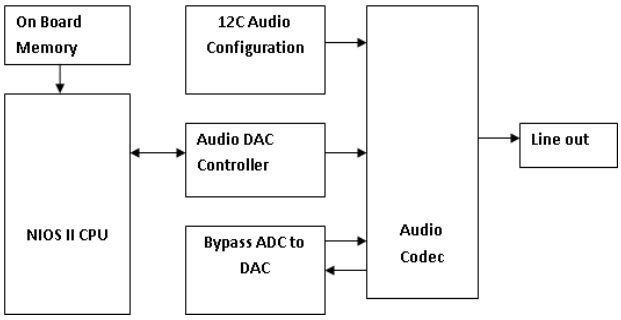

Figure 3: Audio System Using NIOS II and Audio Codec.

2. Writing software application for system, compiling and debugging using NIOS II integrated development environment (IDE).

The hardware platform used for implementation is Cyclone II FPGA Chip available with Altera's DE2 Development Board. The hardware which is utilized for system implementation from the board are Altera Cyclone® II 2C70 FPGA device, Altera Serial Configuration Device - EPCS16, USB Blaster (on board) for programming and user API control.

Both JTAG and Active Serial (AS) programming modes are supported, 2-Mbyte SSRAM, one 32-Mbyte SDRAM, 8-Mbyte Flash memory, SD Card socket, 4 pushbutton switches, 18 toggle switches, 18 red user LEDs, 9 green user LEDs, 50-MHz oscillator for clock source, and a 24-bit CD-quality audio CODEC with line-out.

IMPLEMENTATION OF SYSTEM

For investigation and development of the system, an initially small system named 'niosII_system.qpf' is created with the Quartus II software and tested on DE2-70 board. It is needed to use the SOPC builder tool to implement a design with the Nios II processor core. It generates the Nios II processor system by configuring and adding the desired peripherals. The hardware structure of the system is shown in Figure 4.

All of the components which are configured are listed in the window of Figure 5. The base addresses and IRQs are assigned with the 'auto-assign' option. Instantiation of the system components is followed by pin assignment.

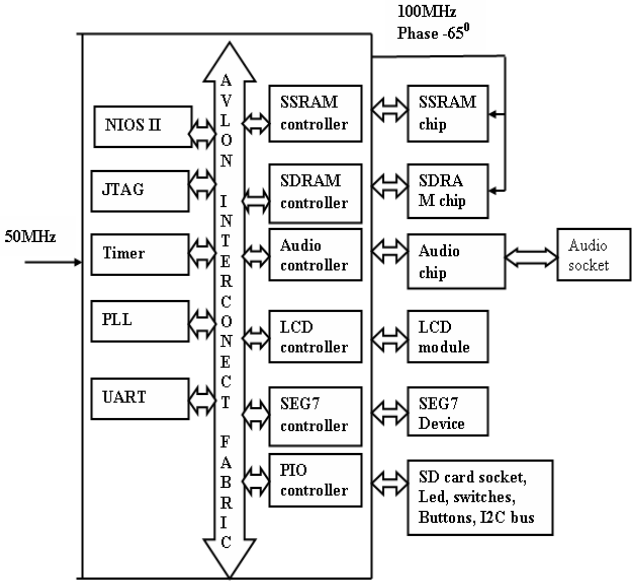

Figure 4: Hardware Structure.

Every peripheral component of DE2-70 board is connected to Cyclone® II FPGA chip. After pin assignment is completed, the system is recompiled and compilation report is generated.

Use	Module Name	Description
☑	cpu	Nios II Processor - Altera Corporation
☑	onchip_mem	On-Chip Memory (RAM or ROM)
☑	jtag_uart	JTAG UART
☑	pio_green_led	PIO (Parallel I/O)
☑	pio_red_led	PIO (Parallel I/O)
☑	pio_button	PIO (Parallel I/O)
☑	pio_switch	PIO (Parallel I/O)
☑	lcd	Character LCD (16x2, Optrex 16207)
☑	timer	Interval timer
☑	sd_clk	PIO (Parallel I/O)
☑	sd_cmd	PIO (Parallel I/O)
☑	sd_dat	PIO (Parallel I/O)
☑	sd_dat3	PIO (Parallel I/O)
☑	pll	PLL (Phase-Locked Loop)
☑	i2c_sclk	PIO (Parallel I/O)
☑	i2c_sdat	PIO (Parallel I/O)
☑	sdram_u1	SDRAM Controller
☑	uart	UART (RS-232 serial port)
☑	sdram_u2	SDRAM Controller
☑	ssram	Cypress CY7C1380C SSRAM
☑	tristate_bridge_ssram	Avalon Tristate Bridge
☑	cfi_flash	Flash Memory (Common Flash Interface)
☑	tristate_bridge_flash	Avalon Tristate Bridge
☑	SEG7	No legacy module
☑	AUDIO	No legacy module

Figure 5: SOPC Window Showing System Content.

Software stack of system is shown in Figure 6. SD 1-bit mode implements the SD 1-bit mode protocol for reading raw data from the SD card. The FAT16 block implements FAT16 file system for reading wave files that stored in the SD card.

In this block, only the read function is implemented. The WAVE Lib block implements WAVE file decoding function for receiving audio signal from wave files. The I2C block implements I2C protocol for configuring audio chip. The SEG7 block implements displaying function for display elapsed playing time.

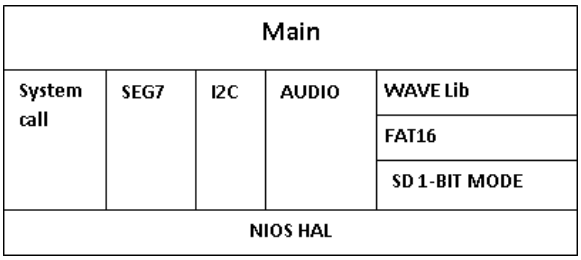

Figure 6: Software Stack of System.

The Audio block implements audio FIFO checking function and audio signal sending/receiving function. The audio chip is configured before sending audio signal to the audio chip. The main program uses I2C protocol to configure the audio chip working in master mode, the audio interface as I2C with 16-bits per channel, and sampling rate according to the wave file content. In the audio playing loop, the main program reads 512-byte audio data from the SD card, and then writes the data to DAC FIFO in the Audio Controller. Before writing the data to the FIFO, the program have to make sure the FIFO is not full. The status of the SD music player is obtain from the 2x16-LCD module, the 7 segment display and the LEDs. The top and bottom row of the LCD module display the filename of the music that is playing on the board and the value of music volume, respectively. The 7 segments display shows how long the music file has been played. The LED indicates the audio signal strength.

ANALYSIS AND RESULTS

DE2-70 board contains on-board memories SDRAM, SRAM, and FLASH for storing application code. The audio system has been implemented by using each kind of on board memory and its effect on resource utilization was analyzed. Table 1 shows the summary of resource utilization of each system. Table 2 shows the comparison between the percentage of resources are utilized by all the three systems. Figure 7 shows the comparison for the number of logic elements utilized by SDRAM, SRAM, and FLASH systems, where SDRAM systems requires highest no. of logic elements and FLASH system requires less no. of logic elements.

Table 1. Resource Utilization Summary.

Resources	Total	SDRAM System	SRAM System	FLASH System
Logic Elements	68,416	4850	4708	4637
Combinational Functions	68,416	4095	3819	3913
Dedicated Logic Registers	68,416	3118	3194	3033
Pins	622	198	214	192

Table2: Resource Utilization Percentage.

Resources	SDRAM System	SRAM System	FLASH System
Logic Elements	7%	7%	7%
Combinational Functions	6%	6%	6%
Dedicated Logic Registers	5%	5%	4%
Pins	32%	34%	31%

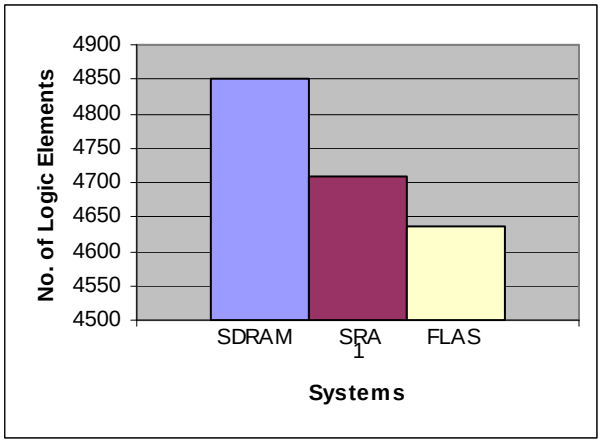

Figure 7: No. of Logic Elements Comparison.

Figure 8 prominently indicates that the combinational function utilization is highest in case of SDRAM system and lowest in case of SRAM systems. The highest noumber of Dedicated Logic registers are utilized by SRAM system and smallest number of dedicated logic

registers are utilized by FLASH systems. This is clearly depicted in Figure 10.

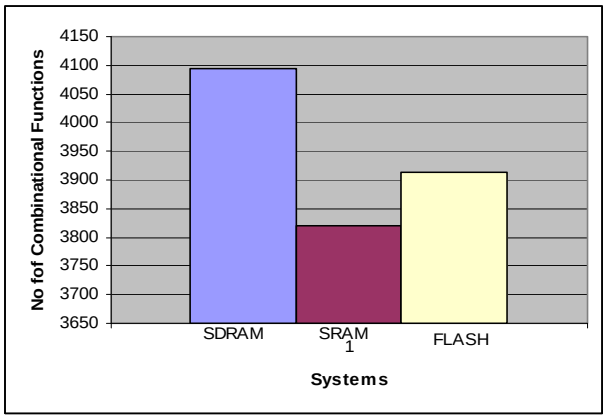

Figure 8: Number of Combinational Function Comparison.

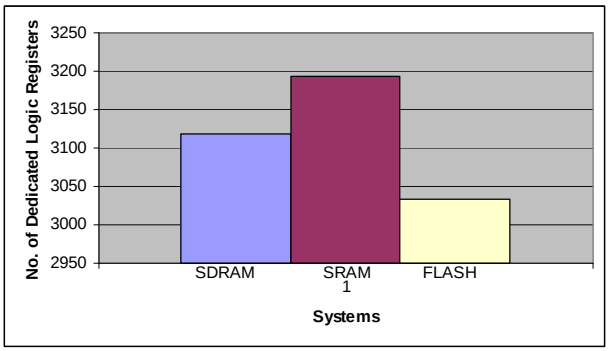

Figure 9: Number of Dedicated Logic Registers Comparison.

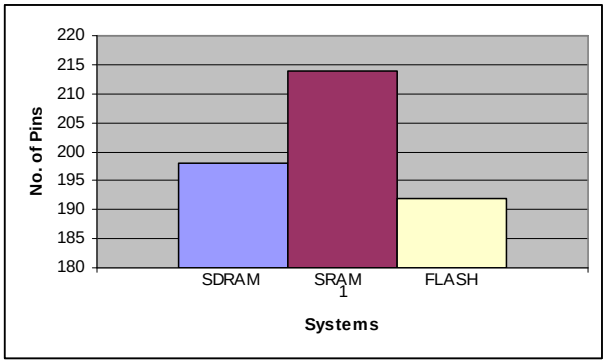

Figure 10: Number of Pins Comparison.

CONCLUSION

Overall, FPGA implementation of the system is time and cost effective. QUARTUS II software simplifies complex system designing. The NIOS II processor is able to receive, manage, and send digital signals to other components of the board and the interfacing processor core is easier to implement because of SOPC builder tool. For analysis and synthesis of the configured system, a compilation report has been generated which gives details about resource utilization, state machine, source assignments, parameter setting by entity instance and timing analysis, which helped in comparing features of embedded systems.

The effect of on-board memories on implementation of embedded system design was observed. Resource utilization summary indicates that SDRAM system requires large numbers of logic elements and combinational function as compared to other two systems but the dedicated logic registers and pin utilization is more in the SRAM system. In DE2-70 board 2MByte SDRAM is organized as 512KX36 bits, 32Mbytes of SDRAM is organized as 4MX 16bits X4 banks whereas 8Mbytes Flash memory supports both byte and word mode access. This memory organization is responsible for increased pins and dedicated logic registers utilization in SRAM system. For platform based implementation of large embedded system, it is necessary to select on-board memories carefully for optimizing resource utilization.

REFERENCES

1. Papakonstantinou, A., Y. Kifle, G. Lucas, and D. Chen. 2008. “MP3 Decoding on FPGA: A Case Study for Floating Point Acceleration”. *Proceedings of Reconfigurable Systems*. Summer Institute: Urbana, IL.

2. Koch, A. and U. Golze. 1993. “FPGA Applications in Education and Research”. Technical University Braunschweig.

3. Etiemble, D., S. Bouaziz, and L. Lacassagne. 2005. “Customizing 16-bit Floating Point Instructions on a NIOS II Processor for FPGA Image and Media Processing”. *Proceeding of Embedded System for Real Time Multimedia, 2005, 3rd Workshop*. 22-23 Sept 2005. 61-66.

4. Altera. 2008. DE2_70 User_manual. Altera: San Jose, CA.

5. Plavec, F. 2004. "Soft-Core Processor Design" Masters Thesis, University of Toronto: Toronto, Canada.

6. http://www.xilinx.com/ise/embedded/edk_docs.htm

7. Calderon, H., C. Elena, and S. Vassiliadis. 2006. "Soft Core Processors and Embedded Processing: A Survey and Analysis". *Proceeding of Microelectronics, ICM'06 International Conference*. 16-19 Dec 2006.170-173.

8. Ignacio, J. 2006. "Interfacing a processor in FPGA to an audio system", Master Thesis in Electronics systems at Linkoping Institute of Technology.

9. Compton, K. and S. Hauck. 2002. "Reconfigurable Computing: A Survey of Systems and Software". *ACM Computing Surveys*. 34(2) June 2002:171–210.

10. Lattice. 2008. *Mico32processor Manual*. http://www.latticesemi.com/dynamic/index.cfm?fuseaction=view_documents&document_type=35&sloc=01-01-08-11-48&source=sidebar .

11. Leon3. 2008. *32 Bit Processor Core Handbook.* http://www.gaisler.com/cms/index.php?option=com_content&task=view&id=115&Itemid=32.

12. Baleani, M., F. Gennari, Y. Jiang, Y. Patel, R.K. Brayton, and A. Sangiovanni-Vincentelli. 2002. "HW/SW Partitioning and Code Generation of Embedded Control Applications on a Reconfigurable Architecture Platform". *Proceeding of Tenth International Symposium on Hardware/Software Codesign*. 151-156.

13. MicroBlaze. 2005. *Processor Reference Manual*. Xilinx: EDK 7.1i

14. Altera. 2008. "Nios II Processor Reference Handbook" http://www.altera.com/literature/hb/nios2/n2cpu_nii5v1.pdf. Altera: San Jose, CA.

15. Chandraiah, P. and R. Domer. 2005. "Specification and Design of a MP3 Audio Decoder". Technical Report CECS-05-04. May 5, 2005. Center for Embedded Computer Systems, University of California, Irvine.

16. Altera. *Quartus II Version 6.1 Handbook*. http://www.altera.com/literature/hb/qts/qts_qii5v4.pdf . Altera: San Jose, CA.

17. Ray, C., C. Cheung, K.P. Pun, S.C.L. Yuen, K.H. Tsoi, and P.H.W. Leong. 2003. "An FPGA-Based Re-Configurable 24-bit 96kHz Sigma-Delta Audio DAC". *Proceedings. 2003 IEEE International Conference*. 15-17 Dec. 2003.110 – 117.

18. Kim, S., J.K. Lee, T.W. You, K.A. Kim, and Y. Choi 2002. "HAT: A High-quality Audio Conferencing Tool using MP3 Codec". *INET 2002*. Washington, DC. June 2002.

19. Edwards, S.A. 2004. "Design Languages for Embedded Systems". Technical Report CUCS–009–03. Columbia University: New York, NY.

20. Altera. 2008. "The Nios II IDE Software". http://www.altera.com/products/ip/processors/nios2/tools/ide/ni2-ide.html Altera: San Jose, CA.

21. Hall, T.S. 2004. "System-on-a-Programmable-Chip Development Platforms in the Classroom". *IEEE Transactions on Education*. 47(4):502 – 507.

22. Altera. 2008. "WM8731L Data Sheet". Altera: San Jose, CA.

ABOUT THE AUTHORS

Dr. M. U. Kharat, BE, MS, Ph.D. was educated at Amravati University. Presently he is working at the G.H. Raisoni College of Engineering and Technology, Nagpur, India as Head of the Information Technology Department and Dean of Research and Development. He has presented papers at national and international conferences and also published papers in national and international journals on various aspects of computer engineering and networks. He has worked in various capacities in academic institutions at the level of Professor, Head of Computer Engineering Department, and Principal. His areas of interest include digital signal processing, computer networks, and the Internet.

Ms. V. S. Rahangadale has completed her B.E. in electronics engineering from the National Institute of Technology, Nanded University, Nanded, India and her M.Eng. in embedded systems and computing from G.H. Raisoni College of Engineering, Nagpur University. Her research area is designing embedded systems, and modulation techniques. She has worked on number of challenging projects.

SUGGESTED CITATION

Kharat, M.U. and V.S. Rahangadale. 2009. "Analysis of Resource Utilization in FPGA

Implementation of an Embedded System using Soft Core Processor". *Pacific Journal of Science and Technology.* 10(1):272-279.


Pacific Journal of Science and Technology