

Soft Core Processors and Embedded Processing: a survey and analysis

Humberto Calderón, Christian Elena and Stamatis Vassiliadis

Computer Engineering Laboratory
Electrical Engineering Dept., EEMCS, TUDelft,
Mekelweg 4, 2628 CD, Delft, The Netherlands
Tel: +31-15-27-86196, Fax: +31-15-27-84898
{H.Calderon, C.Elena, S.Vassiliadis}@ewi.tudelft.nl

Abstract— Reconfigurable technologies provide designers the opportunity to diminish the life-cycle into processor creation. 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. New Electronic-Design-Automation (EDA) tools, allow us to construct rapid prototypes of Systems-on-a-Chip in a very mature way. The design is realized into hardware-software co-design environments, and the use of soft-cores like processor and peripherals reduces drastically the development cycle. Nowadays, the prototyping is done in hardware platforms with several FPGAs, which contains integrated circuits for external connectivity. In this paper, the authors describe the current available reconfigurable technology, focusing on FPGAs, soft-cores and its applicability in the new embedded systems. We evidence in this survey the current potentiality of these technologies, presenting their main qualitative and quantitative characteristics.

Index Terms—Embedded systems, FPGAs, hardware-software co-design, Systems-on-a-Chip, Run time reconfiguration.

I. INTRODUCTION

THE embedded computing born in the earliest age of computers. Whirlwind I, constructed at the Massachusetts Institute of Technology was conceived to operate as an aircraft simulator at the end of the 1940s [1]. Nowadays, several applications use a computer inside of another device, realizing control, communication and multimedia processing tasks; constituting the largest type of computers used in the market [2]. Microprocessor have become smaller in size, cheaper in price and with lower power consumption, therefore, a new era of digital systems is emerging imposed by military, industrial,

communications and entertainment necessities. We inhabit multimedia environments in the work, home and entertainment. Handheld devices, cellular phones, PDAs and play game products [3] are some examples of embedded systems that we use daily.

New standard in multimedia and communications requires flexible and adaptable hardware [4]. New industry requirements diminish the life-cycle of microcontrollers, and several processors became into obsolescence in shorter periods of time. Hardware architects have a big challenge into the fulfillment of new requirements; they need to create processors with adaptable characteristics for such applications, exhibiting high performances and power efficiency.

Flexibility and adaptability are today synonymous of reconfigurable technologies [5]. Improved computer aided design tools arises into the design process. The available Electronic-Design-Automation (EDA) tools [6], proportion the ability to construct rapid prototypes of Systems-on-a-Chip in a very mature way with the help of Hardware Description Language (HDL) [7]. New emerging capabilities in Field Programmable Gate Array (FPGA) technologies, as well as improvements in power consumption, time delays in Configurable Logic Block (CLB) [8], and cost per unit device, enable us to think in the incorporation of these devices into several designs as reconfigurable embedded processors [9].

In this paper, we review the soft-core processors and embedded processing available technologies. For this we give a short summary about reconfigurable technologies in section II. Section III presents an overview of the most used technologies to construct reconfigurable Systems on a Chip. Finally we conclude in section IV.

II. RECONFIGURABLE COMPUTING TECHNOLOGIES

A. Programmable Logic Devices

Programmable logic devices emerge in the 1970s, with logic circuitry without a fixed structure; they were denominated Programmable Logic Devices (PLDs). The main idea behind these chips is based on the fact that logic functions can be realized in sum-of-products form. Therefore, the chips were constructed with an array of AND gates used to build the products, and a set of OR gates to produce the sum. The I.C. created with the aforementioned architecture was called Programmable Array Logic (PAL) [10]. From this time the Programmable devices evolve and the Programmable Logic Arrays (PLA) is a good example of the improvements. PLA devices include Flip-Flops and feedback paths; therefore, it was feasible to construct small sequential systems. PLAs and PALs were limited by the small sizes of its input-output resources [11]. Large systems were constructed utilizing a Complex Programmable Logic Device (CPLD) which internally embeds Logic Blocks similar than PLA or PAL circuits [12]. The architecture of a CPLD device used to create large combinational and sequential systems is schematically depicted in Fig. 1.

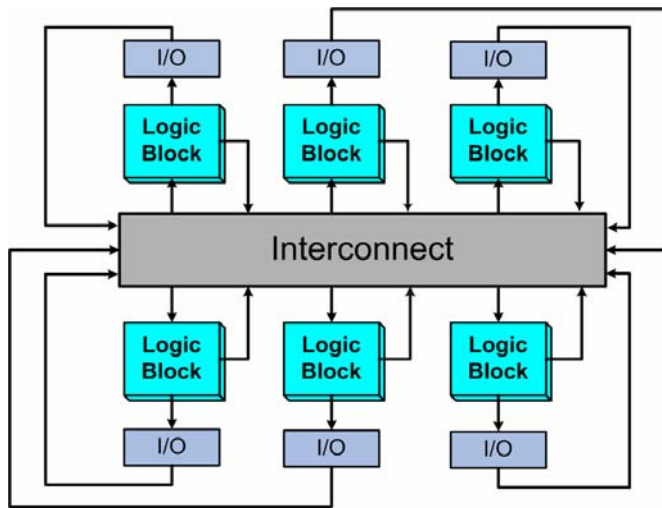


Fig. 1. CPLD Architecture.

The implementation of large and complex circuits is achieved using a Field-Programmable Gate Array (FPGA) [13]. This architecture is quite different to CPLD mentioned above, 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 fig. 2. 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 [14]. As seen in fig 2, these programmable blocks are embedded in a sea of routing channels which contain wires and programmable switches

used to set up the desired functionalities connecting the logic blocks.

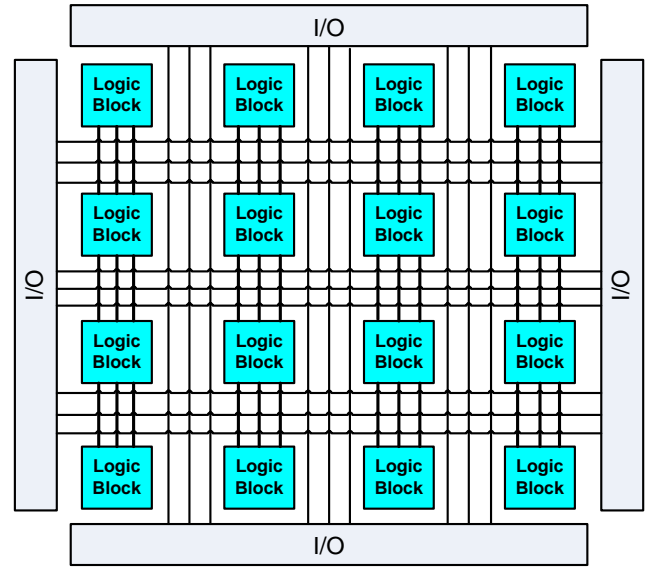


Fig. 2. FPGA Architecture.

B. Reconfigurable Computing

Reconfigurable Computer [15] systems compute with a support hardware elaborated with spatial connection established in the application field (post-fabrication), therefore, Reconfigurable Computing (RC) has the capacity to transform a hardware platform imposed and controlled by the software. This technology was introduced four decades ago, but recently the last decade [16] witnesses the evolution and growth of this important field in the computer engineering. The catalyst of this development comes with the improved performance of the Field Programmable Logic, it usually assembles a general purpose core and Reconfigurable Functional Unit (RFU) used to accelerate the hard demand Kernels [17]. It is well known that large portions of software application nearly the 80 %, consumes only a 20 % of the processor time, and hard demand kernels uses the 80 % of the processor time (i.e. Sum of Absolute Differences processing [18]), degrading in this way the total throughput of the system. A solution to the aforementioned dilemma occurs with the use of reconfigurable computing.

Reconfigurable processors have associated the concept of Virtual Hardware, in which any application believes that has a sized and optimized engine to run on it, therefore, it is possible to establish the hardware on demand paradigm, accelerating the Kernels with specialized and sized hardware. This rationality is presented in Fig 3; the problematic kernels run over reconfigurable functional units.

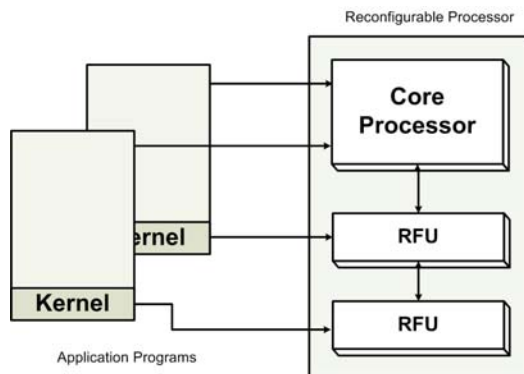


Fig. 3. Reconfigurable processor - functionality.

Reconfigurable technologies have demonstrated a great flexibility and a good performance in order to replace the traditional solutions in high demand tasks [19], offering spatial and temporal parallelism characteristics [20] and also the inherently bit level parallelism [21]. Table I summarizes the principal characteristics of RC compared to traditional solutions.

Table I

Advantages and drawbacks of Reconfigurable Processors

	Power	Performance	Flexibility	Time to Market
General Purpose Processor	Medium	Low	Medium	Low
ASIC	Low	High	Low	High
Re-Configurable Processor	High	Medium	High	Low

Nevertheless, the main drawback of the RC is the necessary configuration time of the new hardware functionality. Several studies show the importance of the run time reconfiguration [22][23][24], and some researchers look for the hiding of the configuration latency time. One of this works proposes the use of matched common components for his use in different tasks, sharing in this way the same hardware [25] and diminishing the overall configuration time. Another approach proposes work with different contexts [26]; the dynamic of this solution is based on switching the context on demand; this solution consumes a less time compared with the configuring of the FPGAs. A similar approach [27] stores different configurations in the internal memory of the programmable logic, and has the capability to change context in a single cycle. The previously presented solutions constitutes the first approaches in order to hide the configuration time and fulfill the goal, nevertheless they suffer a memory overuse.

As mentioned before, reconfigurable computing can be used to speed up the processing with a CPU and customizable RFUs as accelerators. Reconfigurable processors [28][29][30][31] usually include a number of reconfigurable logic devices, local memory and a bus

interface to a host processor with a loosely-coupled and closely-coupled architectures.

III. SYSTEMS ON A CHIP (SoC)

Nowadays, reconfigurable platforms, follows the parading presented in Fig 3, 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 [32]. Today, we are witnesses of the emerging of many commercially embedded processors, as well as supporting and development tools; some of principal products available are: Altera Nios/NiosII [33], Tensilica Xtensa V/LX [34], and Xilinx MicroBlaze [35]. 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 soft-core processor, which are a microprocessor fully described in software, usually in a VHDL, and capable to be synthesized in programmable hardware. In this section, there is presented two consolidates development environments for the creation of Reconfigurable SoC (RSoC), widely used in the academia.

A. NIOS

Altera's NIOS II has a load-store RISC architecture, in which many architectural parameters can be customized at design time [36]. The user can decide between 16 or 32 bits of width in datapath, register file sizes; as well as cache size and custom instructions for the performing of user-defined operation in the speeding-up customized hardware. Those functionalities are supported by the builder development tools, and using the Nios II Integrated Development Environment (IDE) [36] is possible to build, run, and debug software of several platforms. Fig.4 presents the most recently introduced platform, Nios II-Stratix [37].

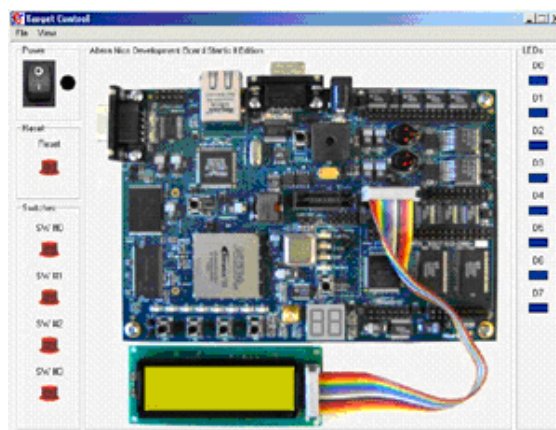


Fig. 4 Stratix Development board

Altera also introduces a SOPC builder [38], for the rapidly creation and easily evaluation of embedded systems. The integration 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. Fig. 5 presents the Stratix II FPGA with some of the customizable peripherals and external memories, as an example of their applicability.

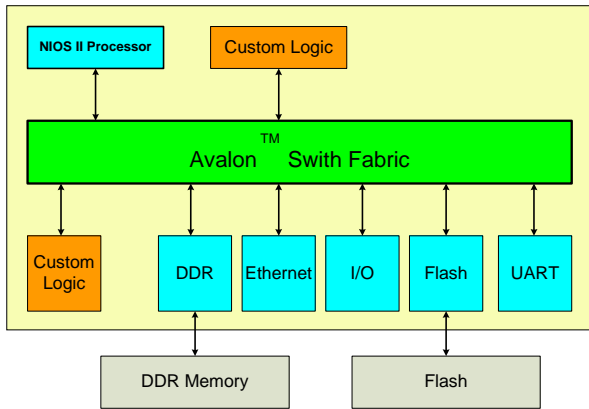


Fig. 5 NIOS II embedded processor solution

B. MicroBlaze

Xilinx’s MicroBlaze is a 32-bits RISC reconfigurable soft-processor, and like Stratix II can be customized with different peripheral and memory configurations. This processor has a three-stage pipeline with variable length instruction latencies, typically ranging from one to three cycles [35]. The tool used to accomplish the design is denominated Xilinx Platform Studio and with this friendly environment we are able to create a MicroBlaze based system instantiating and configuring cores from the provided libraries [39].

MicroBlaze was constructed around Harvard memory architecture. The 2 Local Memory Busses (LMB) are used to connect the instruction and data memories. The sizes of this memory as well as the number of peripheral used in a particular design are defined by the user. Additionally the On-Chip-Peripheral Bus [40] is used to alleviate systems performance bottlenecks and is designed to support low-performance/speed peripherals such as UART, GPIO, USB, external bus controllers. A MicroBlaze system is presented in Fig. 6 as a good example of this technology.

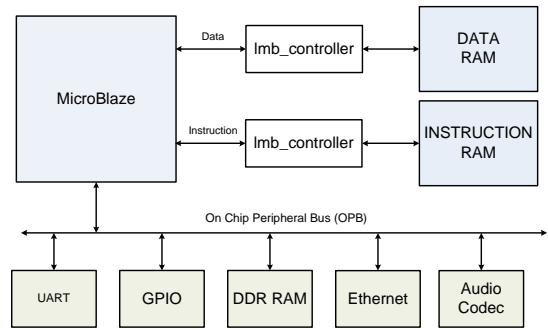


Fig. 6 MicroBlaze based system

With the use of EDK Base System Builder (BSB) [41], a small embedded system was constructed using the XUP Virtex-II PRO (V2-Pro) development system, presented in following figure 7.

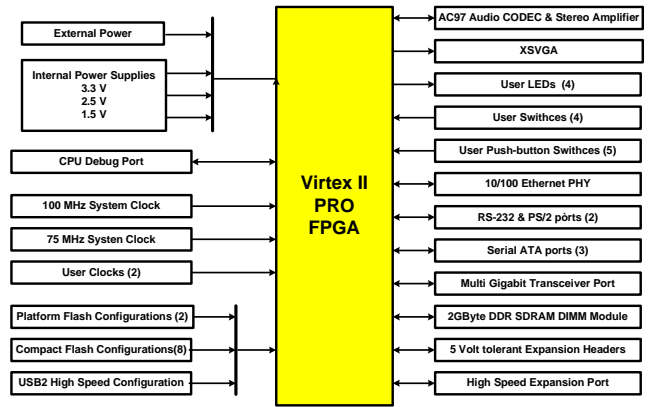


Fig. 7 XUP Virtex II PRO Platform.

The system was constructed in half an hour, instantiating the following components:

1. MicroBlaze processor.
2. On-Chip H/W debug module.
3. Internal BRAM 8 KB
4. RS232 UART
5. GPIO (leds, pushbuttons and DIPSWs).

over a 2vp30ff896-7 FPGA device. The synthesis report gives the following numbers presented in table II:

Table II.
System on a Chip – MicroBlaze in a Virtex II PRO

Resource Type	Used	Available	Percent
Slices	934	13696	6%
Slices FF	769	27392	2%
4 input LUTs	1284	27392	4%

C. Virtex-II PRO evolution.

The Virtex-II Pro is a high performance platform FPGA. The FPGA can hold up to: 24 multi-gigabit transceivers, 4 PowerPCs, 10,008 Kb of BRAM as well as different embedded hardware for the arithmetic processing.

The Virtex II PRO family evolution is representative of the improvement of other FPGA families. The number of resources surpasses the FPGA devices used as glue logic in the 1980s, becoming at the beginning of the 2000s in a complete System Logic with embedded: microprocessor, blocks of RAM, multipliers and registers as is presented on table III [42].

Table III
Virtex II PRO Family

	2VP2 16X22	2VP30 80X46	2VP50 88X70	2VP100 120X94	2VP125 136X106
Logic Cells	3,168	30,816	53,136	99,216	125,136
BRAM(Kbits)	216	2,448	4,176	7,992	10,008
Processors	0	2	2	2	4
Transceivers	4	8	16	20	24
DCMs	4	8	8	12	12
Multipliers	12	36	232	444	556

Nowadays, the specialization comes to scenery with the Domain optimized System Logic.

D. Domain-Optimized System Logic

It has been reported that Virtex-4 FPGA family dramatically reduce power consumption. The power-per-CLB has been dividing by 2 (50 % optimization), the improvement in static power is around a 40 % and the dynamic power reaches also the 50 %. Furthermore, they announce that some functions embedded in the Virtex-4 FPGA have been cut between 80 and 95 % [43]; those reductions are presented with respect to Virtex-II Pro devices, without compromising performance. Furthermore, improvements in several functionalities in terms of time delay are reported with the introduction of new speed grades [44]. Table IV presents these new characteristics.

Table IV.
Virtex-4 Performance by speed grade

	Virtex- 4 -10	Virtex- 4 -11	Virtex- 4 -12
Multiplier (9x9 or 19x18)	400 Mhz	450 Mhz	500 Mhz
16 Tap FIR filter	400 Mhz	450 Mhz	500 Mhz
16Kx32 bit RAM	527 Mhz	610 Mhz	643 Mhz
16Kx64 bit RAM	311 Mhz	311 Mhz	335 Mhz
4Kx144 bit RAM	400 Mhz	450 Mhz	500 Mhz
64 bit Adder	205 Mhz	221 Mhz	245 Mhz
48 bit magnitude compare	335 Mhz	364 Mhz	401 Mhz
High speed Serial I/O	6.25 Gb/s	6.25 Gb/s	8.25 Gb/s

Regarding the speed up into Virtex-4 platforms, it can be noted that speed grade has been incremented respect to the -

7 grade offered by the Virtex II PRO. This speed up is notable when we consider a simple addition. Nowadays, it is possible to compute a 64 bit addition at 245 MHz. This numbers opens new sceneries for the processing of DSP systems.

Finally, the new VIRTEX 4 comes with the Application Specific Modular Block architecture (ASMBL). This technology, arranges into a FPGA different functionalities. The feature options includes: logic, DPS oriented blocks, memory, processing, high speed I/O, hard IP and mixed signal. Table V presents three representative Virtex-4 Platforms, oriented for the processing of different applications (see [45]).

Table V.
Three Virtex - 4 Platforms

Resource	LX	FX	SX
Logic	14-200K LCs	12-140K LCs	23-55K LCs
Memory	0.9-6Mb	0.6-10Mb	2.3-5.7Mb
DCMs	4-12	4-20	4-8
DSP slices	32-96	32-192	128-512
SelectIO	240-960	240-896	320-640
RocketIO	N/A	0-24 Channels	N/A
PowerPC	N/A	1 or 3 Cores	N/A
Ethernet MAC	N/A	2 to 4 Cores	N/A

IV. CONCLUSION

FPGA devices have changed dramatically since Xilinx introduced the technology in 1985. New development tools, and reconfigurable hardware, are ready for fulfil new industry requirements, catalyzing the development of RSoC. The highest levels of abstraction achieved with the new EDA tools, provide the opportunity of diminish the life cycle into the processor creation. New and improved FPGAs in terms of power consumption, less time delays, and even less cost per unit device will increases the development of the new reconfigurable embedded systems. When an improved partial reconfiguration tool arises, the reconfigurable systems will be ready to: make in the-field hardware upgrades, hardware updates, runtime reconfiguration and work with adaptive hardware algorithms; therefore, the improvement in terms of device count and power consumptions achieved with partial reconfiguration will make the reconfigurable technology, the leading technology in the embedded world.

REFERENCES

- [1] John P. Hayes, Computer Architecture and Organization, McGraw-Hill Company, ISBN 0-07-027366-9, 1988, pp 1-88.
- [2] Wayne Wolf, Computers as Components: Principles of Embedded Computing Systems Design, Morgan Kaufmann Publishers, 2001, ISBN 1-55860-541-X
- [3] Ramchan Woo, Sungdae Choi, Ju-Ho Sohn, Seong-Jun Song, Young-Don Bae and Hoi-Jun Yoo, A Low-Power and High-Performance 2D/3D Graphics Accelerator for Mobile Multimedia Applications, HotChips 2003; <http://ssl.kaist.ac.kr/ramp/>

- [4] Matthew Eldridge; Designing Graphics Architectures Around Scalability and Communication, Ph.D. dissertation, Stanford University, June 2001.
- [5] Lew, A.; Halverson, R., Jr.; A FCCM for dataflow (spreadsheet) programs, FPGAs for Custom Computing Machines, 1995. Proceedings. IEEE Symposium on, 19-21 April 1995, Page(s): 2 -10
- [6] Krupnova, H.; Saucier, G.; FPGA technology snapshot: current devices and design tools, Rapid System Prototyping, 2000. RSP 2000. Proceedings. 11th International Workshop on, 21-23 June 2000, Page(s): 200 -205
- [7] Zeninalabedin Navabi, *VHDL Analysis and Modelling of Digital Systems*, MCGrawHill, 1993, ISBN 0-07-046472-3
- [8] Howard Bogrow, The Continued Evolution of Re-Configurable FPGAs for Aerospace and Defense Strategic Applications, 2005 MAPLD International Conference, September 7-9, 2005
- [9] Stamatis Vassiliadis, Stephan Wong and Sorin Cotozana, "The MOLEN μ -coded processor", Proceedings of the 11th International Conference on Field-Programmable Logic and Applications 2001 (FPL2001), Belfast, Northern Ireland, UK, August 2001
- [10] Stephen Brown and Zvonko Vranesic, *Fundamental of Digital Logic with VHDL Design*, MCGrawHill, 2000, ISBN 0-07-012591-0
- [11] Richard C. Dorf, *Electrical Engineering Handbook*, CRC Press, 1993, ISBN 0-8493-0185-8.
- [12] John F. Wakerly, *Digital Design: Principles and Practices*, Prentice Hall, 2001, ISBN 0-13-089896.
- [13] David A Patterson and John L. Hennessy, *Computer Organization and Design*, Morgan Kaufmann, Section B77, 2005, ISBN 1-55860-604-1
- [14] Xilinx, Virtex II Pro Platform FPGA Handbook, Xilinx Inc. UG012 (v1.0) January 2002.USA.
- [15] Stamatis Vassiliadis, Stephan Wong, Georgi Gaydadjiev, Koen Bertels, Georgi Kuzmanov and Elena Moscu Panainte, *The MOLEN Polymorphic Processor*, IEEE Transactions on Computers, Nov. 2004, Vol. 53, No. 11, pp. 1363-1375 .
- [16] Reiner Hartenstein, A Decade of Reconfigurable Computing: a Visionary Retrospective, Proceedings of the Design, Automation, and Test in Europe (DATE '01), March 2001
- [17] Humberto Calderón and Stamatis Vassiliadis, "Reconfigurable Universal SAD-Multiplier Array", Computing Frontiers 2005, Ischia, Italy, May 2005.
- [18] Humberto Calderón and Stamatis Vassiliadis, "Reconfigurable Multiple Operation Array", SAMOS'05 Conference Proceedings, SAMOS, Greece, July18-20 2005.
- [19] Villasenor, J.; Hutchings, B.; The flexibility of configurable computing, Signal Processing Magazine, IEEE, Volume: 15 Issue: 5, Sept. 1998, Page(s): 67 – 84
- [20] A. DeHon. Reconfigurable Architectures for General-Purpose Computing, Ph.D. Dissertation, Massachusetts Institute of Technology, October 1996.
- [21] Bernardo Kastrup, Automatic Synthesis of Reconfigurable Instruction Set Accelerators, Promoters: prof. dr. ing. J.A.G. Jess, prof.dr.ir. J.L. van Meerbergen. TU/e, ISBN 90-74445-50-0, 22 May 2001, pp. 1-129.
- [22] Jose O. Cadenas, Graham M. Megson and Toomas P Plaks. Quantitative evaluation of three reconfiguration strategies on FPGAs: A case study. In HPC-Asia 2000. Proc. of the Fourth Int. Conf. on High-Performance Computing in the Asia-Pacific Region, 14-17 May, 2000, Beijing, China. Vol. I, pages 337- 342. IEEE Computer Society Press, 2000.
- [23] Heron, J.P., Woods, R., Sezer, S., and Turner, R. H., Development of a Run-Time Reconfiguration System with low reconfiguration overhead, the Journal of VLSI Signal Processing, Special issue on Re-configurable Computing, vol. 28 (1/2), pp 97-113, May 2001.
- [24] Wirthlin, M.J.; Hutchings, B.L.; Improving functional density using run-time circuit reconfiguration [FPGAs] Very Large Scale Integration (VLSI) Systems, IEEE Transactions on, Volume: 6 Issue: 2, June 1998 Page(s): 247 -256
- [25] Michael Flynn, Technology trends and adaptive Computing, FPL 2001, LNCS 2147, pp 1-5, 2001
- [26] Scalera, S.M.; Vazquez, J.R.; The design and implementation of a context switching FPGA, FPGAs for Custom Computing Machines, 1998. Proceedings. IEEE Symposium on, 15-17 April 1998 Page(s): 78 -85
- [27] S. Trimmerger, D. Carberry, A. Johnson, J. Wong Xilinx Inc., A time-multiplexed FPGA, 5 IEEE Symposium on FPGA-Based Custom Computing Machines (FCCM '97), April 16-18 , 1997, Napa Valley, CA
- [28] James Stephen Soe Mien Wong, Microcoded Reconfigurable Embedded Processors, PhD Thesis, TU Delft 2002, ISBN 90-9016380-8
- [29] H. Singh et al. *MorphoSys: An Integrated Re-Configurable Architecture*, The application of information Technologies (computer Science) to mission Systems, NATO Research and technology organizations, pp19/1-11, Monterrey, April 1998.
- [30] E. Mirsky and A Dehon. *MATRIX: A reconfigurable Computing Architecture with Configurable Instruction Distribution and Deployable Resources*, Proc. IEEE Symp. on FPGAs for Custom Computing Machines, pp 157-188, Los Alamitos, April 1996.
- [31] C. Ebelin et al, *RaPID: Reconfigurable Pipelined Datapath*, Proc. of Field Programmable logic and Applications (Lecture Notes in Computer Science 1142Springer Verlag), pp. 126-135, Darmstadt, 1996.
- [32] Xtreme DSP Design Considerations User Guide, (UG073.pdf), Xilinx 2005, <http://www.xilinx.com/>
- [33] Nios II Processor Reference Handbook (ver 5.0, Jul 2005), <http://www.altera.com/literature/lit-nio2.jsp>
- [34] Xtensa LX Configurable Processor Core, Xtensa_LX_Final.pdf, http://www.tensilica.com/products/lits_docs.htm
- [35] MicroBlaze Processor Reference Manual, Xilinx 2005, EDK 7.1i, http://www.xilinx.com/ise/embedded/edk_docs.htm
- [36] Nios II Integrated Development Environment <http://www.altera.com/literature/lit-index.html>
- [37] Nios II Development Kit, Stratix II Edition, ALTERA 2005, <http://www.altera.com/products/devkits/altera/kit-niosii-2S30.html>
- [38] SOPC Builder Applications ALTERA 2005, <http://www.altera.com/products/software/products/sopc/sopc-index.html>
- [39] Embedded Tool Reference Manual, Xilinx 2005, EDK 7.1i, http://www.xilinx.com/ise/embedded/edk_docs.htm
- [40] Processor IP Reference Manual, Xilinx 2005, EDK 7.1i, http://www.xilinx.com/ise/embedded/edk_docs.htm
- [41] Platform Studio Users Guide, Xilinx 2005, EDK 7.1i, http://www.xilinx.com/ise/embedded/edk_docs.htm
- [42] Virtex-II PRO Platform FPGA Handbook, UG012 (V2.0), October 14, 2002.
- [43] Matt Klein, The Virtex-4 Power Play, Xcell Journal, Spring 2005.
- [44] WP218 – Achieving Breakthrough Performance in Virtex-4 FPGAs, Xilinx 2005- White Paper, <http://www.xilinx.com/>
- [45] Xilinx Virtex-4 User Guide, (UG070.pdf), Xilinx 2005, <http://www.xilinx.com/bvdocs/userguides/ug070.pdf>