Journal of Advances and Scholarly Researches in Allied Education

Vol. VIII, Issue No. XVI, Oct-2014, ISSN 2230-7540

RECONFIGURABLE EMBEDDED SYSTEMS: A SYSTEMATIC DESIGN TOOLS & ARCHITECTURE

AN INTERNATIONALLY INDEXED PEER REVIEWED & REFEREED JOURNAL

www.ignited.in

Journal of Advances and Scholarly Researches in Allied Education
Vol. VIII, Issue No. XVI, October-2014, ISSN 2230-7540

Reconfigurable Embedded Systems: A Systematic Design Tools & Architecture

Nihar Ranjan Panda[1] Prof. Dr. Rajendra Singh[2]

[1]KEISIE International University

[2]Rohella-Hitech Institute of Technology- Bhubaneswar

Abstract – Embedded systems are playing an increasingly important role in control engineering. Despite their popularity, embedded systems are generally subject to resource constraints and it is therefore difficult to build complex control systems on embedded platforms. This approach is therefore based on the IEC 61499 reference model for distributed and reconfigurable automation and uses the service interface function block concept and high level communication patterns to achieve a hardware-independent access to communication services. Digital Control System in the industry has been used in most of the applications based on expensive Programmable Logical Controllers (PLC). These Systems are, in general, highly complex and slow, with an operation cycle around 10ms. In this work, a Reconfigurable Logic Controller (RLC) approach is presented based on a small and low cost Xilinx Virtex-II FPGA architecture, operating as a virtual hardware machine. Reconfigurable Manufacturing Systems hold promise to satisfy the requirements of dynamic and competitive modern manufacturing. Holonic control architecture is often used for the control of such reconfigurable systems. In this research, software agents and IEC 61499 function blocks are evaluated as alternative strategies for implementing holonic control for a modular feeder subsystem of an experimental Reconfigurable Assembly System. The strategies are evaluated through four reconfiguration experiments. The evaluation is based on qualitative and quantitative performance measures. The results show that agent-based control is more suitable in this specific case study.

Keywords: Reconfigurable, Embedded Systems, Design, Tools & Architecture, etc.

----------X----------

INTRODUCTION

The reconfigurable hardware was used to construct high performance computational structures, whereas the microprocessor controlled when and how often a new computational structure was initiated. At the time, the F+V structure computer was intended for scientific computational problems that were out of reach for conventional microprocessors. However, it did not reach into commercial products as reconfigurable hardware was a hardly explored field and as the performance of conventional microprocessors was constantly growing and simply considered sufficient. Today, the field of reconfigurable computing is compelling as a complement to embedded systems that contain numerous application specific integrated circuits (ASICs). Since 1960, reconfigurable architectures have evolved and there is a consensus that they are a promising solution to bridge the gap between ASIC performance and processor flexibility (Guohuan, *et. al.,* Dec. 2009). However, customary architectures such as field programmable gate arrays (FPGAs) cannot match energy and area efficiency of ASICs. It is because FPGAs are built as arrays of interconnected functional blocks operating at bit-level, whereas many signal processing algorithms utilize word-level arithmetic's. Word-level arithmetic's implemented on architectures that have bit-level functional blocks, because an overhead in required hardware area and energy consumed during operation. In order to reduce this weakness, researchers have suggested new types of reconfigurable architectures with coarse-grained functional blocks. The design space for these coarse-grained architectures is neither sufficiently explored nor well understood when it comes to implement their micro-architecture, to integrate them into an embedded systems, and to efficiently map programs to them. Advanced design tools are required to explore design parameters and establish their relationship to performance, area, and power. That modeling methods and computer-aided tools are keys in development and analysis of reconfigurable computers was recognized already by the pioneers (IEC 61499, 2005. IEC 61131, 2003. R. W. Lewis, 2001). Still, much recently presented reconfigurable architecture has been developed without tools for efficient modeling and analysis of the embedded

system. New reconfigurable architectures, to be used in increasingly complex embedded systems, require development of modeling and exploration tools.

REVIEW OF LITERATURE:

Embedded systems are a broad definition of computer systems that are designed for specific tasks. Their architectures are optimized to conform to requirements in their intended application field. In application fields without specific requirements, standard processor systems are used. In contrast, battery operated systems with real-time performance constraints; require customized architectures to achieve the desired performance with constrained power consumption.

1. DESIGN TOOLS:

With the advent of Field Programmable Logic Devices (FPGA in particular) it became possible to design and implement digital systems without the need for technological steps dealing with silicon. Developing digital devices on the basis of high capacity FPGAs requires using numerous tools, such as FPGA-based boards for rapid prototyping, computer-aided design (CAD) systems, libraries, IP (intellectual property) cores, etc. The study suggests and describes such tools targeted to the design of heterogeneous embedded systems, whose specifications may change continuously (Christensen, 2012. IEC 61131, 2000). The developed tools include: **1.** Core Xilinx Spartan 3 FPGA-based prototyping board with an external USB interface. The latter is a removable block, which might be replaced if necessary; **2.** Extension boards for connecting typical peripheral equipment and solving application-specific problems, such as implementation of interfaces, communication with standard devices (such as memories), etc.; **3.** Software tools for configuring the FPGA, debugging hardware projects, interactions with the board and providing the required experiments **4.** Hardware/software tools for data compression and decompression that enable the volume of data transmitted from/to FPGA to be reduced significantly; **5.** Hardware/software support for reconfiguration of FPGA-based circuits through reloading reconfiguration bit streams, which have to be preliminary stored in a flash memory available on the board; **6.** Hardware/software tools for executing operations over binary/ternary vectors and matrices **7.** VHDL templates for interactions with typical peripheral devices, such as an USB controller, a VGA monitor, a static RAM, a keyboard, a mouse, an LCD panel, etc. **8.** VHDL templates and IP cores for application specific circuits.

Extendable prototyping board - The basic architecture of the board (see **fig. 1**) has been selected in such a way that allows satisfying the objectives and enabling many design problems from the scope of embedded systems to be solved.

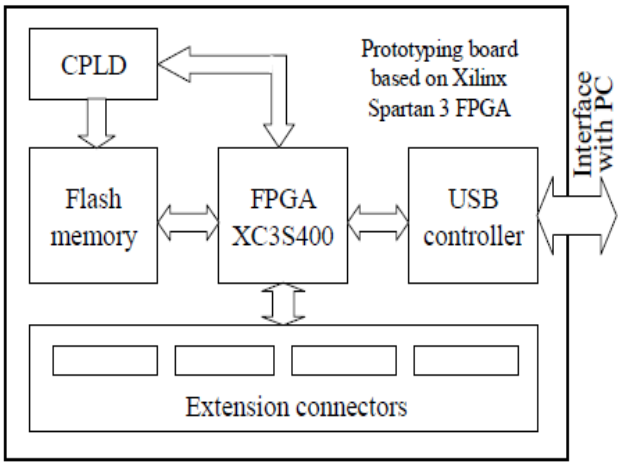

Figure 2.1. Basic architecture of the kernel prototyping board.

2. RECONFIGURABLE ARCHITECTURES:

Reconfigurable architectures (RAs) are devices that contain programmable functional blocks and programmable interconnects between functional blocks, as illustrated in **Figure 2.** Spatial distribution of functional blocks in conjunction with a flexible interconnect of them, allows exploiting various forms of parallelism inherent in the application. In comparison with the programmability provided by instruction set architectures, the programmability provided by RAs allows substantial changes to the data path itself (OPC UA, 2010). Hence, as with dedicated architectures, RAs can implement application-specific computing structures, but without sacrificing flexibility. That property makes reconfigurable architectures a promising solution to bridge the gap between the programmability of instruction set architectures and the performance of dedicated architectures.

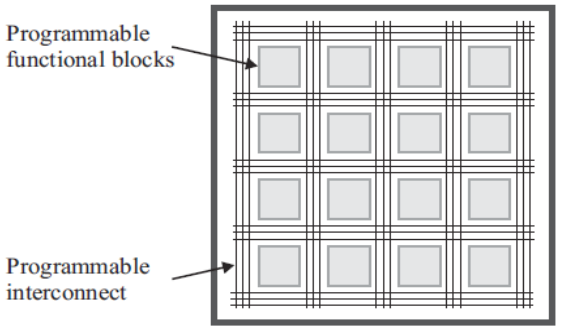

Figure 2- A reconfigurable architecture is built from programmable functional blocks and interconnect.

3. DYNAMIC RECONFIGURABILITY IN EMBEDDED SYSTEM:

Many emerging applications in the fields of (mobile) communications, computing, and consumer electronics require flexible and evolvable functionalities after the system deployment. In order to offer better computing

capabilities, high-performance commercial reconfigurable architectures provide ample reconfigurable logic and, often, have also integrated a number of fixed components, including digital signal processing (DSP) and microprocessor cores, custom hardware, and distributed memory modules (R. Brennan, *et. al.,* 2002. Khalgui, *et. al.,* Aug. 2011). Such reconfigurable architectures, integrated with distributed memory modules, exhibit superior computing capabilities, storage capacities, and flexibility over traditional FPGAs. The successful deployment of these novel embedded systems to the market requires the identification, formalization, and implementation of concepts, methods and tools for their design, considering, together with the traditional requirements, the possibility of dynamically reconfiguring themselves at runtime, over a set of predefined behaviors.

4. CONVERGENCE OF RECONFIGURABLE AND EMBEDDED STYLES:

Embedded systems comprise complete devices ranging from portable such as video cameras and set-top boxes, to industrial controllers. They are designed to perform dedicated specific tasks with real-time processing constraints (Vall´ee, *et. al.,* Aug 2011). The large diffusion of embedded systems raises very demanding requirements in terms of computing performance, power budgeting and functional flexibility. By looking closely at those requirements, we will see the convergence of reconfigurable and embedded trends.

CONCLUSION:

Embedded partial dynamic reconfiguration, due to its internal nature, can be used to realize embedded systems without involving any other device. The successful deployment of such complex and reconfigurable embedded systems to the market requires the identification, formalization, and implementation of concepts, methods, and tools for embedded software design that are able to ease the development of software components and the implementation of the system architecture. In this study we presented an overview on the comprehensive work that has been done in the area of reconfigurable embedded systems, describing both the architectural and the methodological aspects of such systems. Reconfigurable on chip communication platforms are now part of multi-core architectures. Performance improvements can still be achieved working at the bit-stream level with FPGAs or at the instruction level in the case of reconfigurable processors. System level evaluation of reconfigurable platforms is necessary to predict performance, and to find and cure bottlenecks prior to fabrication. A set of flexible transaction level models were developed to allow complete embedded systems with instruction set processors, buses, and memories to be evaluated together with the reconfigurable architecture.

REFERENCES:

L. Guohuan, Z. Hao, and Z. Wei, (Dec. 2009). “Research on designing method of CAN bus and Modbus protocol conversion interface,” in BioMedical Information Engineering, 2009. FBIE 2009. International Conference on Future, pp. 180–182.

IEC 61499 (2005).: Function blocks, Part 1-4, International Electrotechnical Commission Std. IEC 61 499, [Online]. Available: www.iec.ch

IEC 61131 (2003).: Programmable controllers, Part 3: Programming languages, International Electrotechnical Commission Std. IEC 61 131 [Online]. Available: www.iec.ch

R. W. Lewis, (2001). Modeling control systems using IEC 61499. IEE Publishing, 2001, no. ISBN: 0 85296 796 9.

J. H. Christensen. (Access Date: May 2012) HOLOBLOC.com - Function Block-Based, Holonic Systems Technology. [Online]. Available: www.holobloc.com

IEC 61131 (2000): Programmable controllers, Part 5: Communications, International Electrotechnical Commission Std. IEC 61 131 [Online]. Available: www.iec.ch

OPC UA (2010). Information Model for IEC 61131-3, PLCopen & OPC Foundation Std. 1.00, 2010. [Online]. Available: www.plcopen.org

R. Brennan, M. Fletcher, and D. Norrie, (Aug. 2002) “An agent-based approach to reconfiguration of real-time distributed control systems,” Robotics and Automation, IEEE Transactions on, vol. 18, no. 4, pp. 444–451.

M. Khalgui, O. Mosbahi, Z. Li, and H.-M. Hanisch, (Aug. 2011). “Reconfiguration of Distributed Embedded-Control Systems,” Mechatronics, IEEE/ASME Transactions on, vol. 16, no. 4, pp. 684–694.

M. Vall´ee, M. Merdan, W. Lepuschitz, and G. Koppensteiner, (Aug. 2011). “Decentralized Reconfiguration of a Flexible Transportation System,” Industrial Informatics, IEEE Transactions on, vol. 7, no. 3.