Multiprocessor System On Chip Hardware Design And Tool Integration

Alexander Biedermann presents a generic hardware-based virtualization approach, which may transform an array of any off-the-shelf embedded processors into a multi-processor system with high execution dynamism. Based on this approach, he highlights concepts for the design of energy aware systems, self-healing systems as well as parallelized systems. For the latter, the novel so-called Agile Processing scheme is introduced by the author, which enables a seamless transition between sequential and parallel execution schemes. The design of such virtualizable systems is further aided by introduction of a dedicated design framework, which integrates into existing, commercial workflows. As a result, this book provides comprehensive design flows for the design of embedded multi-processor systems-on-chip.

System on chips designs have evolved from fairly simple unicore, single memory designs to complex heterogeneous multicore SoC architectures consisting of a large number of IP blocks on the same silicon. To meet high computational demands posed by latest consumer electronic devices, most current systems are based on such paradigm, which represents a real revolution in many aspects in computing. The attraction of multicore processing for power reduction is compelling.

By splitting a set of tasks among multiple processor cores, the operating frequency necessary for each core can be reduced, allowing to reduce the voltage on each core. Because dynamic power is proportional to the frequency and to the square of the voltage, we get a big gain, even though we may have more cores running. As more and more cores are integrated into these designs to share the ever increasing processing load, the main challenges lie in efficient memory hierarchy, scalable system interconnect, new programming paradigms, and efficient integration methodology for connecting such heterogeneous cores into a single system capable of leveraging their individual flexibility. Current design methods tend toward mixed HW/SW co-designs targeting multicore systems on-chip for specific applications. To decide on the lowest cost mix of cores, designers must iteratively map the device's functionality to a particular HW/SW partition and target architectures. In addition, to connect the heterogeneous cores, the architecture requires high performance complex communication architectures and efficient communication protocols, such as hierarchical bus, point-to-point connection, or Network-on-Chip. Software development also becomes far more complex due to the difficulties in breaking a single processing task into multiple parts that can be processed separately and then reassembled later. This reflects the fact that certain

processor jobs cannot be easily parallelized to run concurrently on multiple processing cores and that load balancing between processing cores - especially heterogeneous cores - is very difficult.

Over the past decade, the processing speed requirement of embedded systems has steadily increased. Since faster clocking of a single processor can no longer be considered to increase the processing speed of the system (due to overheating and other constraints), the development of multiprocessors on a single chip has stepped up to meet the demand. One approach has been to design and develop a multiprocessing platform to handle a large set of homogeneous applications. However, this development has been slow due to the intractable design space, which results when both the hardware and software are required to be adjustable to meet the needs of the dissimilar applications. A different approach has been to limit the number of targeted applications to be similar in some sense. By limiting the number of targeted applications to a cohesive set, the design space can become manageable. This thesis proposes a framework for a multiprocessing system-on-chip (MPSoC), consisting of a cohesive hardware and software architecture intended specifically for problems that are stream-oriented (e.g., video streaming). The framework allows the hardware and software to be customized to fit a specific application within the cohesive

set, while narrowing the design space to a manageable set of design parameters. In addition, this thesis designs and develops an analytic model, using a discrete-time Markov chain, to measure the performance of an MPSoC framework implementation when the number of concurrent processing elements is varied. Finally, a chaotic simulated annealing algorithm was developed to determine an optimal mapping and scheduling of tasks to processing elements within the MPSoC. Conventional on-chip communication design mostly use ad-hoc approaches that fail to meet the challenges posed by the next-generation MultiCore Systems on-chip (MCSoC) designs. These major challenges include wiring delay, predictability, diverse interconnection architectures, and power dissipation. A Network-on-Chip (NoC) paradigm is emerging as the solution for the problems of interconnecting dozens of cores into a single system on-chip. However, there are many problems associated with the design of such systems. These problems arise from non-scalable global wire delays, failure to achieve global synchronization, and difficulties associated with non-scalable bus-based functional interconnects. The book consists of three parts, with each part being subdivided into four chapters. The first part deals with design and methodology issues. The architectures used in conventional methods of MCSoCs design and custom multiprocessor architectures are not flexible enough to meet

the requirements of different application domains and not scalable enough to meet different computation needs and different complexities of various applications. Several chapters of the first part will emphasize on the design techniques and methodologies. The second part covers the most critical part of MCSoCs design — the interconnections. One approach to addressing the design methodologies is to adopt the so-called reusability feature to boost design productivity. In the past years, the primitive design units evolved from transistors to gates, finite state machines, and processor cores. The network-on-chip paradigm offers this attractive property for the future and will be able to close the productivity gap. The last part of this book delves into MCSoCs validations and optimizations. A more qualitative approach of system validation is based on the use of formal techniques for hardware design. The main advantage of formal methods is the possibility to prove the validity of essential design requirements. As formal languages have a mathematical foundation, it is possible to formally extract and verify these desired properties of the complete abstract state space. Online testing techniques for identifying faults that can lead to system failure are also surveyed. Emphasis is given to analytical redundancy-based techniques that have been developed for fault detection and isolation in the

automatic control area.
System-on-Chip Design
Computer System Design
Multiprocessor System Architectures
High-level Estimation and Exploration of Reliability for Multi-Processor System-on-Chip
Multiprocessor System-on-Chip
Memory Controllers for Mixed-Time-Criticality Systems

This book introduces a novel framework for accurately modeling the errors in nanoscale CMOS technology and developing a smooth tool flow at high-level design abstractions to estimate and mitigate the effects of errors. The book presents novel techniques for high-level fault simulation and reliability estimation as well as architecture-level and system-level fault tolerant designs. It also presents a survey of state-of-the-art problems and solutions, offering insights into reliability issues in digital design and their cross-layer countermeasures.

This comprehensive survey of the technology used to design high-performance multiprocessing systems is a valuable reference for design engineers and a hands-on design guide. Intended for hardware and software engineers, it clearly explains the architectural components driving the next generation of multiprocessing and multithreading architectures from Sun Microsystems, Inc. The first book to cover new technology in the area of packet-switched buses and multithreaded application environments integrated in the design of multiprocessor systems. It brings together in one volume a coherent description of the problems and

solutions unique to the design and development of multiprocessor systems. The book reviews design considerations associated with multiprocessor systems and presents practical solutions; provides an indepth study on the Scalable Processor ARChitecture (SPARC); introduces and details multi-level bus architectures including MBus and XBus/XDBus; details components of the SunSoft SunOS two-level multithreaded architecture such as threads, lightweight processes, synchronization, scheduling, preemption and real-time facilities; introduces multithreaded programming with code examples; examines several multiprocessor system implementations by highlighting tightly coupled, shared-memory model architectures and multi-level bus implementations; and provides the latest MBus specification and MBus Design Guide needed for the design of MBus based systems.

This book gives a comprehensive introduction to the design challenges of MPSoC platforms, focusing on early design space exploration. It defines an iterative methodology to increase the abstraction level so that evaluation of design decisions can be performed earlier in the design process. These techniques enable exploration on the system level before undertaking time- and cost-intensive development.

This book offers up a deep understanding of concepts and practices behind the composition of heterogeneous components. After the analysis of existing computation and execution models used for the specification and validation of different sub-systems, the book introduces a systematic approach to build an execution model for

systems composed of heterogeneous components. Mixed continuous/discrete and hardware/software systems are used to illustrate these concepts. The benefit of reading this book is to arrive at a clear vision of the theory and practice of specification and validation of complex modern systems. Numerous examples give designers highly applicable solutions.

System-on-Chip

Proceedings of the ICCEE 2019, Kuala Lumpur, Malaysia

Architectures

Embedded Software for SoC

Applications

MULTICORE SYSTEMS ON-CHIP

The end of dramatic exponential growth in single-processor performance marks the end of the dominance of the single microprocessor in computing. The era of sequential computing must give way to a new era in which parallelism is at the forefront. Although important scientific and engineering challenges lie ahead, this is an opportune time for innovation in programming systems and computing architectures. We have already begun to see diversity in computer designs to optimize for such considerations as power and throughput. The next generation of discoveries is likely to require advances at both the hardware and software levels of computing systems. There is no guarantee that we can make parallel computing as common and easy to use as yesterday's sequential single-processor computer systems, but unless we aggressively pursue efforts suggested by the recommendations in this book, it will be

"game over" for growth in computing performance. If parallel programming and related software efforts fail to become widespread, the development of exciting new applications that drive the computer industry will stall; if such innovation stalls, many other parts of the economy will follow suit. The Future of Computing Performance describes the factors that have led to the future limitations on growth for single processors that are based on complementary metal oxide semiconductor (CMOS) technology. It explores challenges inherent in parallel computing and architecture, including ever-increasing power consumption and the escalated requirements for heat dissipation. The book delineates a research, practice, and education agenda to help overcome these challenges. The Future of Computing Performance will guide researchers, manufacturers, and information technology professionals in the right direction for sustainable growth in computer performance, so that we may all enjoy the next level of benefits to society.

This Open Access book introduces readers to many new techniques for enhancing and optimizing reliability in embedded systems, which have emerged particularly within the last five years. This book introduces the most prominent reliability concerns from today's points of view and roughly recapitulates the progress in the community so far. Unlike other books that focus on a single abstraction level such circuit level or system level alone, the focus of this book is to deal with the different reliability challenges across different levels starting from the physical level all the way to the system level (cross-layer approaches). The book aims at demonstrating how new

hardware/software co-design solution can be proposed to ef-fectively mitigate reliability degradation such as transistor aging, processor variation, temperature effects, soft errors, etc. Provides readers with latest insights into novel, cross-layer methods and models with respect to dependability of embedded systems; Describes cross-layer approaches that can leverage reliability through techniques that are pro-actively designed with respect to techniques at other layers; Explains run-time adaptation and concepts/means of self-organization, in order to achieve error resiliency in complex, future many core systems.

For Electrical Engineering and Computer Engineering courses that cover the design and technology of very large scale integrated (VLSI) circuits and systems. May also be used as a VLSI reference for professional VLSI design engineers, VLSI design managers, and VLSI CAD engineers. Modern VSLI Design provides a comprehensive "bottom-up" guide to the design of VSLI systems, from the physical design of circuits through system architecture with focus on the latest solution for system-on-chip (SOC) design. Because VSLI system designers face a variety of challenges that include high performance, interconnect delays, low power, low cost, and fast design turnaround time, successful designers must understand the entire design process. The Third Edition also provides a much more thorough discussion of hardware description languages, with introduction to both Verilog and VHDL. For that reason, this book presents the entire VSLI design process in a single volume.

This book presents the proceedings of ICCEE 2019, held in Kuala Lumpur, Malaysia, on 29th–30th April 2019. It includes the latest advances in electrical engineering and electronics from leading experts around the globe.
Multi-Core Embedded Systems
Integrating Heterogeneous Components
Dependable Embedded Systems
Chip Multiprocessor Architecture
Architectures, Methodologies and Trade-offs
VLSI-SoC: Research Trends in VLSI and Systems on Chip

Covers the significant embedded computingtechnologies—highlighting their applications in wirelesscommunication and computing power An embedded system is a computer system designed for specificcontrol functions within a larger system—often with real-timecomputing constraints. It is embedded as part of a complete deviceoften including hardware and mechanical parts. Presented in threeparts, Embedded Systems: Hardware, Design, andImplementation provides readers with an immersive introductionto this rapidly growing segment of the computer industry. Acknowledging the fact that embedded systems control many oftoday's most common devices such as smart phones, PC tablets, aswell as hardware embedded in cars, TVs, and even refrigerators andheating systems, the book starts with a basic introduction toembedded computing systems. It hones in on system-on-a-chip (SoC),multiprocessor

system-on-chip (MPSoC), and network-on-chip (NoC).It then covers on-chip integration of software and custom hardwareaccelerators, as well as fabric flexibility, custom architectures,and the multiple I/O standards that facilitate PCB integration. Next, it focuses on the technologies associated with embeddedcomputing systems, going over the basics of field-programmable gatearray (FPGA), digital signal processing (DSP) andapplication-specific integrated circuit (ASIC) technology,architectural support for on-chip integration of customaccelerators with processors, and O/S support for thesesystems. Finally, it offers full details on architecture, testability,and computer-aided design (CAD) support for embedded systems, softprocessors, heterogeneous resources, and on-chip storage beforeconcluding with coverage of software support—in particular,O/S Linux. Embedded Systems: Hardware, Design, and Implementation isan ideal book for design engineers looking to optimize and reducethe size and cost of embedded system products and increase theirreliability and performance.

Techniques for Optimizing Multiprocessor Implementations of Signal Processing Applications An indispensable component of the information age, signal processing is embedded in a variety of consumer devices, including cell phones and digital television, as well as in communication infrastructure,

such as media servers and cellular base stations. Multiple programmable processors, along with custom hardware running in parallel, are needed to achieve the computation throughput required of such applications. Reviews important research in key areas related to the multiprocessor implementation of multimedia systems Embedded Multiprocessors: Scheduling and Synchronization, Second Edition presents architectures and design methodologies for parallel systems in embedded digital signal processing (DSP) applications. It discusses application modeling techniques for multimedia systems, the incorporation of interprocessor communication costs into multiprocessor scheduling decisions, and a modeling methodology (the synchronization graph) for multiprocessor system performance analysis. The book also applies the synchronization graph model to develop hardware and software optimizations that can significantly reduce the interprocessor communication overhead of a given schedule. Chronicles recent activity dealing with single-chip multiprocessors and dataflow models This edition updates the background material on existing embedded multiprocessors, including single-chip multiprocessors. It also summarizes the new research on dataflow models for signal processing that has been carried out since the publication of the first edition. Harness the power of multiprocessors This book explores the optimization of interprocessor

communication and synchronization in embedded multiprocessor systems. It shows you how to design multiprocessor computer systems that are streamlined for multimedia applications.

A Multi-Processor System-on-Chip (MPSoC) is the key component for complex applications. These applications put huge pressure on memory, communication devices and computing units. This book, presented in two volumes – Architectures and Applications – therefore celebrates the 20th anniversary of MPSoC, an interdisciplinary forum that focuses on multi-core and multi-processor hardware and software systems. It is this interdisciplinarity which has led to MPSoC bringing together experts in these fields from around the world, over the last two decades. Multi-Processor System-on-Chip 2 covers application-specific MPSoC design, including compilers and architecture exploration. This second volume describes optimization methods, tools to optimize and port specific applications on MPSoC architectures. Details on compilation, power consumption and wireless communication are also presented, as well as examples of modeling frameworks and CAD tools. Explanations of specific platforms for automotive and real-time computing are also included.

The aggressive evolution of the semiconductor industry - smaller process geometries, higher densities, and greater chip complexity - has provided

design engineers the means to create complex, high-performance System-on-a-Chip (SoC) designs. Such SoC designs typically have more than one processor and huge (tens of Mega Bytes) amount of memory, all on the same chip. Dealing with the global on-chip memory allocation/deallocation in a dynamic yet deterministic way is an important issue for upcoming billion transistor multiprocessor SoC designs. To achieve this, we propose a memory management hierarchy we call Two-Level Memory Management. To implement this memory management scheme which presents a shift in the way designers look at on-chip dynamic memory allocation - we present the System-on-a-Chip Dynamic Memory Management Unit (SoCDMMU) for allocation of the global on-chip memory, which we refer to as Level Two memory management (Level One is the management of memory allocated to a particular on-chip Processing Element, e.g., an operating system's management of memory allocated to a particular processor). In this way, processing elements (heterogeneous or non-heterogeneous hardware or software) in an SoC can request and be granted portions of the global memory in a fast and deterministic time. A new tool is introduced to generate a custom optimized version of the SoCDMMU hardware. Also, a real-time operating system is modified support the new proposed SoCDMMU. We show an example where shared memory multiprocessor SoC that employs the Two-

Level Memory Management and utilizes the SoCDMMU has an overall average speedup in application transition time as well as normal execution time.

Fundamentals, Advanced Features, and Applications in Industrial Electronics

Embedded Systems

Design and Implementation of Instruction Set Extension Identification for a Multiprocessor System-on-chip Hardware/software Co-design Toolchain

Multi-Processor System-on-Chip 1

Game Over or Next Level?

Global Specification and Validation of Embedded Systems

This book discusses the design and performance analysis of SDRAM controllers that cater to both real-time and best-effort applications, i.e. mixed-time-criticality memory controllers. The authors describe the state of the art, and then focus on an architecture template for reconfigurable memory controllers that addresses effectively the quickly evolving set of SDRAM standards, in terms of worst-case timing and power analysis, as well as implementation. A prototype implementation of the controller in SystemC and synthesizable VHDL for an FPGA development board are used as a proof of concept of the architecture template.

This book outlines a set of issues that are critical

to all of parallel architecture--communication latency, communication bandwidth, and coordination of cooperative work (across modern designs). It describes the set of techniques available in hardware and in software to address each issues and explore how the various techniques interact.

Modern embedded systems come with contradictory design constraints. On one hand, these systems often target mass production and battery-based devices, and therefore should be cheap and power efficient. On the other hand, they still need to show high (sometimes real-time) performance, and often support multiple applications and standards which requires high programmability. This wide spectrum of design requirements leads to complex heterogeneous System-on-Chip (SoC) architectures -- consisting of several types of processors from fully programmable microprocessors to configurable processing cores and customized hardware components, integrated on a single chip. This study targets such multiprocessor embedded systems and strives to develop algorithms, methods, and tools to deal with a number of fundamental problems which are encountered by the system designers during the early design stages.

This book contains extended and revised

versions of the best papers presented during the fourteenth IFIP TC 10/WG 10.5 International Conference on Very Large Scale Integration. This conference provides a forum to exchange ideas and show industrial and academic research results in microelectronics design. The current trend toward increasing chip integration and technology process advancements brings about stimulating new challenges both at the physical and system-design levels.

Design Concepts for a Virtualizable Embedded MPSoC Architecture

Integrated System-Level Modeling of Network-on-Chip enabled Multi-Processor Platforms

Architecture, On-Chip Network, Design

Fourteenth International Conference on Very Large Scale Integration of System on Chip (VLSI-SoC2006), October 16-18, 2006, Nice, France

System-level Modelling and Design Space Exploration for Multiprocessor Embedded System-on-chip Architectures

Hardware, Design and Implementation

Master's Thesis from the year 2007 in the subject Computer Science - Applied, grade: 1.0, Technical University of Munich (Institute for Informatics), 82 entries in the bibliography, language: English, abstract: Multicore systems are dominating the processor market; they enable the increase in

computing power of a single chip in proportion to the Moore's law-driven increase in number of transistors. A similar evolution is observed in the system-on-chip (SoC) market through the emergence of multi-processor SoC (MPSoC) designs. Nevertheless, MPSoCs introduce some challenges to the system architects concerning the efficient design of memory hierarchies and system interconnects while maintaining the low power and cost constraints. In this master thesis, I try to address some of these challenges: namely, non-cache coherent DMA transfers in MPSoCs, low instruction cache utilization by OS codes, and factors governing the system throughput in MPSoC designs. These issues are investigated using the empirical and simulation approaches. Empirical studies are conducted on the Danube platform. Danube is a commercial MPSoC platform that is based on two 32-bit MIPS cores and developed by Infineon Technologies AG for deployment in access network processing equipments such as integrated access devices, customer premises equipments, and home gateways. Simulation-based studies are conducted on a system based on the ARM MPCore architecture. Achievements include the

successful implementation and testing of novel hardware and software solutions for improving the performance of non-cache coherent DMA transfers in MPSoCs. Several techniques for reducing the instruction cache miss rate are investigated and applied. Finally, a qualitative analysis of the impact of instruction reuse, number of cores, and memory bandwidth on the system throughput in MPSoC systems is presented.

Details a real-world product that applies a cutting-edge multi-core architecture Increasingly demanding modern applications—such as those used in telecommunications networking and real-time processing of audio, video, and multimedia streams—require multiple processors to achieve computational performance at the rate of a few giga-operations per second. This necessity for speed and manageable power consumption makes it likely that the next generation of embedded processing systems will include hundreds of cores, while being increasingly programmable, blending processors and configurable hardware in a power-efficient manner. Multi-Core Embedded Systems presents a variety of perspectives that elucidate the technical challenges associated with such increased

integration of homogeneous (processors) and heterogeneous multiple cores. It offers an analysis that industry engineers and professionals will need to understand the physical details of both software and hardware in embedded architectures, as well as their limitations and potential for future growth. Discusses the available programming models spread across different abstraction levels The book begins with an overview of the evolution of multiprocessor architectures for embedded applications and discusses techniques for autonomous power management of system-level parameters. It addresses the use of existing open-source (and free) tools originating from several application domains—such as traffic modeling, graph theory, parallel computing and network simulation. In addition, the authors cover other important topics associated with multi-core embedded systems, such as: Architectures and interconnects Embedded design methodologies Mapping of applications
The purpose of this book is to evaluate strategies for future system design in multiprocessor system-on-chip (MPSoC) architectures. Both hardware design and integration of new development tools will be

discussed. Novel trends in MPSoC design, combined with reconfigurable architectures are a main topic of concern. The main emphasis is on architectures, design-flow, tool-development, applications and system design.

The first book to survey this emerging field in digital system design.

Advances in Electronics Engineering

Microprocessor Architecture

Techniques to Improve Throughput and Latency

Embedded Multiprocessors

FPGAs

Multicore Systems On-Chip: Practical Software/Hardware Design

Chip multiprocessors - also called multi-core microprocessors or CMPs for short - are now the only way to build high-performance microprocessors, for a variety of reasons. Large uniprocessors are no longer scaling in performance, because it is only possible to extract a limited amount of parallelism from a typical instruction stream using conventional superscalar instruction issue techniques. In addition, one cannot simply ratchet up the clock speed on today's processors, or the power dissipation will become prohibitive in all but water-cooled systems. After a discussion of the basic pros and cons of CMPs when they are compared with conventional uniprocessors, this book examines how CMPs can best be designed to handle two radically different kinds of workloads that are likely to be used with a CMP: highly parallel, throughput-sensitive applications at one end of the spectrum, and less parallel, latency-sensitive applications at the other. Throughput-sensitive

applications, such as server workloads that handle many independent transactions at once, require careful balancing of all parts of a CMP that can limit throughput, such as the individual cores, on-chip cache memory, and off-chip memory interfaces. Several studies and example systems, such as the Sun Niagara, that examine the necessary tradeoffs are presented here. In contrast, latency-sensitive applications - many desktop applications fall into this category - require a focus on reducing inter-core communication latency and applying techniques to help programmers divide their programs into multiple threads as easily as possible. This book discusses many techniques that can be used in CMPs to simplify parallel programming, with an emphasis on research directions proposed at Stanford University. To illustrate the advantages possible with a CMP using a couple of solid examples, extra focus is given to thread-level speculation (TLS), a way to automatically break up nominally sequential applications into parallel threads on a CMP, and transactional memory. This model can greatly simplify manual parallel programming by using hardware - instead of conventional software locks - to enforce atomic code execution of blocks of instructions, a technique that makes parallel coding much less error-prone. Book jacket.

This title serves as an introduction ans reference for the field, with the papers that have shaped the hardware/software co-design since its inception in the early 90s.

A Multi-Processor System-on-Chip (MPSoC) is the key component for complex applications. These applications put huge pressure on memory, communication devices and computing units. This book, presented in two volumes – Architectures and Applications – therefore celebrates the 20th anniversary of MPSoC, an interdisciplinary forum that focuses on multi-core and multi-processor hardware and software systems. It is this interdisciplinarity which has led to MPSoC bringing together experts in these fields from around the world, over the last two decades. Multi-Processor System-on-Chip 1 covers the key

components of MPSoC: processors, memory, interconnect and interfaces. It describes advance features of these components and technologies to build efficient MPSoC architectures. All the main components are detailed: use of memory and their technology, communication support and consistency, and specific processor architectures for general purposes or for dedicated applications. Current multimedia and telecom applications require complex, heterogeneous multiprocessor system on chip (MPSoC) architectures with specific communication infrastructure in order to achieve the required performance. Heterogeneous MPSoC includes different types of processing units (DSP, microcontroller, ASIP) and different communication schemes (fast links, non standard memory organization and access). Programming an MPSoC requires the generation of efficient software running on MPSoC from a high level environment, by using the characteristics of the architecture. This task is known to be tedious and error prone, because it requires a combination of high level programming environments with low level software design. This book gives an overview of concepts related to embedded software design for MPSoC. It details a full software design approach, allowing systematic, high-level mapping of software applications on heterogeneous MPSoC. This approach is based on gradual refinement of hardware/software interfaces and simulation models allowing to validate the software at different abstraction levels. This book combines Simulink for high level programming and SystemC for the low level software development. This approach is illustrated with multiple examples of application software and MPSoC architectures that can be used for deep understanding of software design for MPSoC.

Readings in Hardware/software Co-design

Multiprocessor Systems-on-chips

Embedded Multiprocessor System-on-Chip for Access Network Processing

Design Space Exploration

Enabling Virtualization in Embedded Multi-Processor Systems

Scheduling and Synchronization, Second Edition

Intelligent readers who want to build their own embedded computer systems-- installed in everything from cell phones to cars to handheld organizers to refrigerators-- will find this book to be the most in-depth, practical, and up-to-date guide on the market. Designing Embedded Hardware carefully steers between the practical and philosophical aspects, so developers can both create their own devices and gadgets and customize and extend off-the-shelf systems. There are hundreds of books to choose from if you need to learn programming, but only a few are available if you want to learn to create hardware. Designing Embedded Hardware provides software and hardware engineers with no prior experience in embedded systems with the necessary conceptual and design building blocks to understand the architectures of embedded systems. Written to provide the depth of coverage and real-world examples developers need, Designing Embedded Hardware also provides a road-map to the pitfalls and traps to avoid in designing embedded systems. Designing Embedded Hardware covers such essential topics as: The principles of developing computer hardware Core hardware designs Assembly language concepts Parallel I/O Analog-digital conversion Timers (internal and external) UART Serial Peripheral Interface Inter-Integrated Circuit Bus Controller Area Network (CAN) Data Converter Interface (DCI) Low-power operation This invaluable and eminently useful book gives you the practical tools and skills to develop, build, and program your own application-specific computers.

From basic architecture, interconnection, and parallelization to power optimization, this book provides a comprehensive description of emerging multicore systems-on-chip (MCSoCs) hardware and software design. Highlighting both fundamentals and advanced software and hardware design,

it can serve as a primary textbook for advanced courses in MCSoCs design and embedded systems. The first three chapters introduce MCSoCs architectures, present design challenges and conventional design methods, and describe in detail the main building blocks of MCSoCs. Chapters 4, 5, and 6 discuss fundamental and advanced on-chip interconnection network technologies for multi and many core SoCs, enabling readers to understand the microarchitectures for on-chip routers and network interfaces that are essential in the context of latency, area, and power constraints. With the rise of multicore and many-core systems, concurrency is becoming a major issue in the daily life of a programmer. Thus, compiler and software development tools are critical in helping programmers create high-performance software. Programmers should make sure that their parallelized program codes will not cause race condition, memory-access deadlocks, or other faults that may crash their entire systems. As such, Chapter 7 describes a novel parallelizing compiler design for high-performance computing. Chapter 8 provides a detailed investigation of power reduction techniques for MCSoCs at component and network levels. It discusses energy conservation in general hardware design, and also in embedded multicore system components, such as CPUs, disks, displays and memories. Lastly, Chapter 9 presents a real embedded MCSoCs system design targeted for health monitoring in the elderly.

Simulation of computer architectures has made rapid progress recently. The primary application areas are hardware/software performance estimation and optimization as well as functional and timing verification. Recent, innovative technologies such as retargetable simulator generation, dynamic binary translation, or sampling simulation have enabled widespread use of

processor and system-on-chip (SoC) simulation tools in the semiconductor and embedded system industries. Simultaneously, processor and SoC simulation is still a very active research area, e.g. what amounts to higher simulation speed, flexibility, and accuracy/speed trade-offs. This book presents and discusses the principle technologies and state-of-the-art in high-level hardware architecture simulation, both at the processor and the system-on-chip level.

Integrated System-Level Modeling of Network-on-Chip Enabled Multi-Processor Platforms first gives a comprehensive update on recent developments in the area of SoC platforms and ESL design methodologies. The main contribution is the rigorous definition of a framework for modeling at the timing approximate level of abstraction. Subsequently this book presents a set of tools for the creation and exploration of timing approximate SoC platform models.

Processor and System-on-Chip Simulation

Hardware Design and Tool Integration

Designing Embedded Hardware

A Multiprocessng System-on-chip Framework Targeting Stream-oriented Applications

Pipelined Multiprocessor System-on-Chip for Multimedia

Parallel Computer Architecture

Field Programmable Gate Arrays (FPGAs) are currently recognized as the most suitable platform for the implementation of complex digital systems targeting an increasing number of industrial electronics applications. They cover a huge variety of application areas, such as: aerospace, food industry, art, industrial automation, automotive, biomedicine, process control, military, logistics, power electronics, chemistry, sensor networks, robotics,

ultrasound, security, and artificial vision. This book first presents the basic architectures of the devices to familiarize the reader with the fundamentals of FPGAs before identifying and discussing new resources that extend the ability of the devices to solve problems in new application domains. Design methodologies are discussed and application examples are included for some of these domains, e.g., mechatronics, robotics, and power systems.

The next generation of computer system designers will be less concerned about details of processors and memories, and more concerned about the elements of a system tailored to particular applications. These designers will have a fundamental knowledge of processors and other elements in the system, but the success of their design will depend on the skills in making system-level tradeoffs that optimize the cost, performance and other attributes to meet application requirements. This book provides a new treatment of computer system design, particularly for System-on-Chip (SOC), which addresses the issues mentioned above. It begins with a global introduction, from the high-level view to the lowest common denominator (the chip itself), then moves on to the three main building blocks of an SOC (processor, memory, and interconnect). Next is an overview of what makes SOC unique (its customization ability and the applications that drive it). The final chapter presents future challenges for system design and SOC possibilities.

This book describes the architecture of microprocessors from simple in-order short pipeline designs to out-of-order superscalars.

This book describes analytical models and estimation methods to enhance performance estimation of pipelined

multiprocessor systems-on-chip (MPSoCs). A framework is introduced for both design-time and run-time optimizations. For design space exploration, several algorithms are presented to minimize the area footprint of a pipelined MPSoC under a latency or a throughput constraint. A novel adaptive pipelined MPSoC architecture is described, where idle processors are transitioned into low-power states at run-time to reduce energy consumption. Multi-mode pipelined MPSoCs are introduced, where multiple pipelined MPSoCs optimized separately are merged into a single pipelined MPSoC, enabling further reduction of the area footprint by sharing the processors and communication buffers. Readers will benefit from the authors' combined use of analytical models, estimation methods and exploration algorithms and will be enabled to explore billions of design points in a few minutes.

Advanced Multicore Systems-On-Chip
Simulink and System C Case Studies
Dynamic Memory Management for Embedded Real-time Multiprocessor System-on-a-chip
Multiprocessor Systems on Chip
Modern VLSI Design
The Future of Computing Performance

This title covers all software-related aspects of SoC design, from embedded and application-domain specific operating systems to system architecture for future SoC. It will give embedded software designers invaluable insights into the

constraints imposed by the use of embedded software in an SoC context.

Embedded Software Design and Programming of Multiprocessor System-on-Chip

Multi-Processor System-on-Chip 2

From Simple Pipelines to Chip Multiprocessors

A Hardware/software Approach