

Contents

1	Introduction to Multicore Systems On-Chip	1
1.1	The Multicore Revolution	1
1.1.1	Moore's Law	2
1.1.2	On-Chip Interconnection Schemes	2
1.1.3	Parallelism and Performance	4
1.1.4	Parallel Hardware Architectures	6
1.1.5	The Need for Multicore Computing	8
1.1.6	Multicore SoCs Potential Applications	8
1.2	Multicore SoC Basics	10
1.2.1	Programmability Support	10
1.2.2	Software Organization	12
1.2.3	Programming Multicore Systems	12
1.2.4	Multicore Implementations	13
1.3	Multicore SoCs Design Challenges	15
1.3.1	Cache Coherence	15
1.3.2	Power and Temperature	16
1.3.3	Multi-Threading and Memory Management	16
1.3.4	Interconnection Networks	17
1.4	Conclusion	17
2	Multicore SoCs Design Methods	19
2.1	Introduction	19
2.2	Design Space Exploration	20
2.3	Parallel Software Development Phase	22
2.3.1	Compiler-Based Schemes	22
2.3.2	Language Extensions Schemes	23
2.3.3	Language Extensions with APIs	24
2.3.4	Model-Based Schemes	24
2.4	Generic Architecture Template for Real Multicore SoC Design	25
2.4.1	Target Multicore SoC Platform	25
2.4.2	Design Method	26
2.4.3	QueueCore Architecture	28

- 2.4.4 Performance Analysis 32
- 2.5 Conclusion. 35
- 3 Multicore SoC Organization 37**
 - 3.1 Introduction 37
 - 3.1.1 Heterogeneous MCSoC 38
 - 3.1.2 Homogeneous MCSoC 41
 - 3.1.3 Multicore SoC Applications 42
 - 3.1.4 Applications Mapping 43
 - 3.2 MCSoC Building Blocks 44
 - 3.2.1 Processor Core 46
 - 3.2.2 Memory. 46
 - 3.2.3 Cache 48
 - 3.2.4 Communication Protocols 48
 - 3.2.5 Intellectual Property Cores. 50
 - 3.2.6 IP Cores with Multiple Clock Domains. 51
 - 3.2.7 Selection of IP Cores 52
 - 3.3 MCSoC Memory Hierarchy 53
 - 3.3.1 Types on On-Chip Memory 54
 - 3.3.2 Scratchpad Memory 56
 - 3.3.3 Off-Chip Memory. 56
 - 3.3.4 Memory Power Reduction in SoC Designs 57
 - 3.4 Memory Consistency in Multicore Systems 59
 - 3.4.1 Cache Coherence Problem 59
 - 3.4.2 Cache Coherence Protocols 61
 - 3.5 Conclusion. 63
- 4 2D Network-on-Chip 65**
 - 4.1 Introduction 65
 - 4.2 2D NoC Architecture 68
 - 4.2.1 Topology and Routing. 68
 - 4.2.2 Switching 68
 - 4.2.3 Flow Control 69
 - 4.2.4 Arbiter 69
 - 4.2.5 Network Interface 69
 - 4.3 2D NoC Hardware Design Details 70
 - 4.3.1 Topology Design 70
 - 4.3.2 Pipeline Design 71
 - 4.3.3 Arbiter Design 74
 - 4.3.4 Crossbar Design 75
 - 4.3.5 Network Interface 76
 - 4.3.6 Limitations of Regular Mesh Topology 77
 - 4.3.7 SPL Insertion Algorithm 78
 - 4.3.8 Putting it all Together 83

- 4.4 Evaluation 83
 - 4.4.1 Environments and Parameters. 83
 - 4.4.2 Dimension Reversal and Hotspot Simulation Results 84
 - 4.4.3 JPEG Encoder Simulation Results 84
- 4.5 Conclusion. 88
- 5 3D Network-on-Chip 89**
 - 5.1 Introduction 89
 - 5.1.1 Why 3D-NoCs? 90
 - 5.1.2 3D-NoC Versus 2D-NoC. 92
 - 5.1.3 Router Architectures 93
 - 5.1.4 Routing Algorithms. 93
 - 5.2 Topology Design 95
 - 5.3 Switching Policy 97
 - 5.3.1 Flit Format Design 98
 - 5.4 3D-NoC Router Architecture Design. 99
 - 5.4.1 Input-Port Module Design 101
 - 5.4.2 Semi-Adaptive Look-Ahead Routing. 102
 - 5.4.3 Switch Allocator Design 104
 - 5.4.4 Stall-Go Flow Control Architecture. 105
 - 5.4.5 Matrix-Arbitrer Scheduling Architecture. 106
 - 5.4.6 Crossbar Design 109
 - 5.5 Network Interface Architecture 109
 - 5.6 3D-ONoC Architecture Design Evaluation. 113
 - 5.6.1 JPEG Encoder on 3D-ONoC 113
 - 5.6.2 Matrix Multiplication on 3D-ONoC 114
 - 5.6.3 Evaluation Results 119
 - 5.6.4 Performance Analysis Evaluation 120
 - 5.7 Conclusion. 125
- 6 Network Interface Architecture and Design for 2D/3D NoCs 127**
 - 6.1 Introduction 127
 - 6.2 Network Interface Basics. 128
 - 6.2.1 Source Routing Network Interface 128
 - 6.2.2 Distributed Routing Network Interface 129
 - 6.3 Overview of OASIS NoC Architecture 129
 - 6.4 Architecture, and Design Decision for Distributed Routing NI 130
 - 6.4.1 Network Size Decision 131
 - 6.4.2 Packet Size Decision. 132
 - 6.4.3 Buffer Size Decision. 132
 - 6.4.4 Communication Protocol and Flow Control Decisions. 132

6.4.5	Packet Format Decision	132
6.4.6	Flit-Level Decision	133
6.4.7	Summary of all Decisions	136
6.5	Distributed Routing Network Interface Design	136
6.5.1	Core-to-Router (C2R) Buffer	137
6.5.2	Flitizer Module Architecture	137
6.5.3	Core-to-Router (C2R) Controller	138
6.5.4	Router-to-Core (R2C) Buffer	139
6.5.5	Deflitizer Module Architecture	139
6.5.6	Router-to-Core (R2C) Controller	140
6.6	Evaluation	140
6.6.1	RTL and Gate Level Simulation	140
6.6.2	Hardware Prototyping	143
6.6.3	Hardware Complexity	152
6.7	Conclusion	152
7	Parallelizing Compiler for Single and Multicore Computing	153
7.1	Instruction Level Parallelism	153
7.2	Parallel Queue Compiler	155
7.2.1	Queue Processor Overview	155
7.2.2	Compiling for 1-Offset QueueCore Instruction Set	156
7.3	Parallelizing Compiler Framework	158
7.3.1	1-Offset P-Code Generation Phase	159
7.3.2	Offset Calculation Phase	163
7.3.3	Instruction Scheduling Phase	164
7.3.4	Natural Instruction Level Parallelism Extraction: Statement Merging Transformation	165
7.3.5	Assembly Generation Phase	167
7.4	Parallelizing Compiler Development Results	169
7.4.1	Queue Compiler Evaluation	169
7.4.2	Comparison of Generated QueueCore Code with Optimized RISC Code	171
7.5	Conclusion	173
8	Power Optimization Techniques for Multicore SoCs	175
8.1	Introduction	175
8.2	Power Aware Technological-Level Design Optimizations	177
8.2.1	Factors Affecting CMOS Power Consumption	177
8.2.2	Reducing Voltage and Frequency	178
8.2.3	Reducing Capacitance	179
8.3	Power Aware Logic-Level Design Optimizations	180
8.3.1	Clock Gating	180
8.3.2	Logic Encoding	181

8.3.3	Data Guarding	182
8.4	Power-Aware System Level Design Optimizations	183
8.4.1	Hardware System Architecture Power Consumption Optimizations	183
8.4.2	Operating System Power Consumption Optimization	186
8.4.3	Application, Compilation Techniques and Algorithm	188
8.4.4	Energy Reduction in Network Protocols	189
8.5	Conclusion.	193
9	Soft-Core Processor for Low-Power Embedded Multicore SoCs.	195
9.1	Introduction	195
9.2	Produced Order Queue Computing Overview.	197
9.3	QC-2 Core Architecture	199
9.3.1	Instruction Set Design Considerations	199
9.3.2	Instruction Pipeline Structure	200
9.3.3	Dynamic Operands Addresses Calculation	202
9.3.4	QC-2 FPA Organization	203
9.3.5	Circular Queue-Register Structure.	206
9.4	Synthesis of the QC-2 Core	207
9.4.1	Design Approach	207
9.5	Results and Discussions	209
9.5.1	Execution Speedup and Code Analysis	209
9.5.2	Synthesis Results	210
9.5.3	Speed and Power Consumption Comparison with Synthesizable CPU Cores	212
9.6	Conclusion.	213
10	Dual-Execution Processor Architecture for Embedded Computing	215
10.1	Introduction	215
10.2	System Architecture	217
10.2.1	Pipeline Structure	219
10.2.2	Fetch Unit	219
10.2.3	Decode Unit.	219
10.2.4	Dynamic Switching Mechanism	221
10.2.5	Calculation of Produced and Consumed Data.	221
10.2.6	Queue-Stack Computation Unit	222
10.2.7	Sources-Results Computing Mechanism	224
10.2.8	Issue Unit	226
10.2.9	Execution Unit	227
10.2.10	Shared Storage Mechanism	229

10.2.11	Covop Instruction Execution Mechanism	229
10.2.12	Interrupt Handling Mechanism	229
10.3	Sub-Routine Call Handling Mechanism	233
10.4	Hardware Design and Evaluation Results	236
10.4.1	DEP System Pipeline Control	237
10.4.2	Hardware Design Result	238
10.4.3	Comparison Results	241
10.5	Conclusions	242
11	Case Study: Deign of Embedded Multicore SoC for Biomedical Applications	243
11.1	Introduction	243
11.1.1	Electrocardiography and Heart Diseases	244
11.2	Digital Signal Processing	246
11.2.1	Analog and Digital Signals	246
11.2.2	Signal Processing	246
11.2.3	Analog to Digital Conversion	247
11.3	Period-Peak Detection Algorithm	248
11.3.1	Period Detection	248
11.3.2	Peaks Detection	249
11.4	Multicore SoC Architecture and Hardware Design	250
11.4.1	Signal Reading	251
11.4.2	Filtering	253
11.4.3	Data Processing	255
11.4.4	Processor Core	256
11.5	Real-Time Interaction Interface Development	257
11.5.1	Data Capturing	257
11.5.2	Data Display and Analysis	259
11.6	Design Results	259
11.6.1	Hardware Complexity	259
11.6.2	Performance Evaluation	260
11.7	Conclusion	261
	References	263



<http://www.springer.com/978-94-91216-91-6>

Multicore Systems On-Chip: Practical Software/Hardware
Design

Ben Abdallah, A.

2013, XXVI, 273 p. 196 illus., 79 illus. in color.,

Hardcover

ISBN: 978-94-91216-91-6

A product of Atlantis Press