Contents

1 Introduction **1**
- 1.1 Motivation . . . 1
- 1.2 Embedded System Trends . . . 2
  - 1.2.1 Market Trends . . . 3
  - 1.2.2 Technology Trends . . . 6
- 1.3 Development of Embedded Real-Time Systems . . . 9
  - 1.3.1 Specification and Implementation Phase . . . 11
  - 1.3.2 Integration and Verification Phase . . . 12
- 1.4 Integration Challenges . . . 13
  - 1.4.1 Multicore Integration Challenges . . . 14
- 1.5 Tackling the Integration Challenges . . . 15
  - 1.5.1 Traditional Approaches . . . 16
  - 1.5.2 Time-Orthogonalization of Resources . . . 16
  - 1.5.3 Formal Performance Analysis . . . 18
- 1.6 Summary and Contributions . . . 22
  - 1.6.1 Summary and Problem Statement . . . 22
  - 1.6.2 Contributions . . . 23
- 1.7 Outline . . . 24

2 Compositional Performance Analysis of Systems with Shared Resources **27**
- 2.1 Related Work . . . 28
- 2.2 Modeling of Real-Time Systems . . . 29
  - 2.2.1 System Model . . . 30
  - 2.2.2 Analysis Baseline . . . 31
- 2.3 Analysis Decomposition . . . 32
- 2.4 A Self-Contained Set of Analysis Functions For Multiprocessor Systems 36
  - 2.4.1 Dependencies between Analysis Functions . . . 39
- 2.5 Analysis Composition . . . 40
  - 2.5.1 Finding a Conservative Set of Parameter Values . . . 40
  - 2.5.2 Finding a Consistent Set of Parameter Values . . . 42
  - 2.5.3 Solving Analysis Dependencies with Fixed-Point Theory . . . 45
  - 2.5.4 Conditions for Analysis Functions . . . 46
  - 2.5.5 Speed of Convergence . . . 48
- 2.6 Summary . . . 48

3 Timing Analysis with General Load Event Models 51
3.1 Introduction . . . 51
3.2 Modeling the Timing of Events . . . 52
3.3 Properties of the Generalized Load Event Model . . . 54
3.3.1 General Load Event Models form a Complete Partial Order . . 55
3.3.2 Reconstruction of Incomplete Event Stream Information 56
3.4 Resource Models . . . 57
3.4.1 Task Response Time . . . 58
3.4.2 Continuous Service Bounds . . . 59
3.4.3 State-Based Resource Models . . . 61
3.4.4 The Need for a new Model . . . 62
3.5 Multiple Event Busy Time Model . . . 62
3.5.1 Deriving a Task's Worst-Case Response Time from its Multiple Event Busy Time . . . 64
3.5.2 Application to Static Priority Preemptive Scheduling . . . 64
3.5.3 Application to Time-Driven Scheduling . . . 65
3.5.4 Alternative Methods for Deriving the Busy Time Function . . 66
3.6 Deriving Output Event Models . . . 69
3.6.1 Derivation of Minimum Event Distances . . . 69
3.6.2 Derivation of Maximum Event Distances . . . 75
3.6.3 Monotonicity with Respect to Input Parameters . . . 75
3.7 Experimental Evaluation . . . 78
3.8 Conclusion . . . 80

4 Pipelined Path Latency 81
4.1 Introduction . . . 81
4.1.1 Example . . . 82
4.1.2 Related Work . . . 84
4.2 Recursive Path Latency Computation . . . 85
4.2.1 Definitions . . . 85
4.2.2 Computing the Path Latency . . . 86
4.2.3 Reconciling Worst-Case Latency with Long-Term Throughput 88
4.3 Fork and Join Application Topologies . . . 89
4.3.1 Multiple Outputs . . . 89
4.3.2 Multiple Inputs . . . 90
4.4 Cyclic Dependencies . . . 92
4.4.1 Non-functional Cyclic Dependencies . . . 92
4.4.2 Functional Cycles . . . 93
4.5 Experimental Evaluation . . . 97
4.6 Conclusion . . . 99

5 Shared Resource Request Bounds 101
5.1 Recapitulation of the Analysis Procedure . . . 101
5.2 Related Work . . . 102
5.3 Introduction . . . 105
5.4 Modeling Refinement . . . 106
5.4.1 Extended Task Model . . . 106
5.4.2 Capturing the Timing of Shared Resource Requests . . . 108
5.5 Deriving Bounds on the Shared Resource Requests . . . 109
5.5.1 Remote Operations Initiated by a Single Task Instance . . . 109
5.5.2 Multiple Instances of the Same Task . . . 111
5.5.3 Scheduling Multiple Tasks on the Same Processor . . . 114
5.6 Embedding the Analysis Functions into the Multiprocessor Analysis . 117
5.7 Summary . . . 118

6 Interdependent Scheduling Analysis in the Presence Of Shared Resources 121
6.1 Introduction . . . 121
6.1.1 Analysis Concept and Related Work . . . 121
6.2 Synchronous Shared Resource Requests . . . 125
6.2.1 Static Priority Preemptive Scheduler with Shared Resources . . 126
6.2.2 Discussion on Alternative Handling of Synchronous Shared Resource Operations . . . 128
6.3 Asynchronous Shared Resource Requests . . . 130
6.3.1 Implementation and Scheduling Options . . . 130
6.3.2 Multithreaded Round-Robin . . . 133
6.3.3 Multithreaded Round-Robin with Shared Resources . . . 136
6.3.4 Task Synchronization Through Semaphores . . . 138
6.4 Embedding the Task's Busy Time Analysis into the Multiprocessor Analysis Procedure . . . 140
6.5 Summary . . . 141

7 Latency of Shared Resource Operations 143
7.1 Introduction . . . 143
7.1.1 Capturing the Aggregate Latency Of Shared Resource Operations 143
7.1.2 Intractability of Exact Solutions . . . 144
7.2 Analysis of the Aggregate Request Latency . . . 145
7.2.1 Sum of Worst Cases . . . 145
7.3 Extended Scope Interference Analysis . . . 147
7.3.1 Bounding the Aggregate Busy Time For Work-Conserving Arbiters . . . 147
7.3.2 A Dedicated Analysis for Round-Robin Arbitration . . . 148
7.4 Resource Requests over multiple Hops . . . 149
7.4.1 Analysis Toolbox . . . 150

7.5 Monotonocity 152
7.6 Experimental Illustration and Evaluation 153
7.7 Summary 154

8 Applications **157**
8.1 The Timing of Multiprocessor Systems with Local Instruction Caches 157
8.1.1 Bounding Intrinsic Cache Misses 157
8.1.2 Bounding Preemption-Related Cache Misses 158
8.1.3 Experimental Evaluation 162
8.1.4 Conclusion 164
8.2 Performance Analysis of the StepNP Multiprocessor Platform 165
8.2.1 Platform Architecture 165
8.2.2 Image Processing Application 166
8.2.3 Experiments 166
8.2.4 Summary 170

9 Conclusion **171**

List of Publications **173**

References **178**