## Contents

Preface v
Acknowledgments xi
List of Figures xix
List of Tables xxv

1. INTRODUCTION  1
   1.1 System-Design Challenges  1
   1.2 Abstraction Levels  3
      1.2.1 Y-Chart  3
      1.2.2 Processor-Level Behavioral Model  5
      1.2.3 Processor-level structural model  7
      1.2.4 Processor-level synthesis  10
      1.2.5 System-Level Behavioral Model  13
      1.2.6 System Structural Model  14
      1.2.7 System Synthesis  14
   1.3 System Design Methodology  18
      1.3.1 Missing semantics  20
      1.3.2 Model Algebra  21
   1.4 System-Level Models  23
   1.5 Platform Design  27
   1.6 System Design Tools  29
   1.7 Summary  32

2. SYSTEM DESIGN METHODOLOGIES  35
   2.1 Bottom-up Methodology  35
   2.2 Top-down Methodology  37
   2.3 Meet-in-the-middle Methodology  38
2.4 Platform Methodology 40
2.5 FPGA Methodology 43
2.6 System-level Synthesis 44
2.7 Processor Synthesis 45
2.8 Summary 47

3. MODELING 49

<table>
<thead>
<tr>
<th>Section</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1 Models of Computation</td>
<td>50</td>
</tr>
<tr>
<td>3.1.1 Process-Based Models</td>
<td>52</td>
</tr>
<tr>
<td>3.1.2 State-Based Models</td>
<td>58</td>
</tr>
<tr>
<td>3.2 System Design Languages</td>
<td>65</td>
</tr>
<tr>
<td>3.2.1 Netlists and Schematics</td>
<td>66</td>
</tr>
<tr>
<td>3.2.2 Hardware-Description Languages</td>
<td>66</td>
</tr>
<tr>
<td>3.2.3 System-Level Design Languages</td>
<td>68</td>
</tr>
<tr>
<td>3.3 System Modeling</td>
<td>68</td>
</tr>
<tr>
<td>3.3.1 Design Process</td>
<td>69</td>
</tr>
<tr>
<td>3.3.2 Abstraction Levels</td>
<td>71</td>
</tr>
<tr>
<td>3.4 Processor Modeling</td>
<td>72</td>
</tr>
<tr>
<td>3.4.1 Application Layer</td>
<td>73</td>
</tr>
<tr>
<td>3.4.2 Operating System Layer</td>
<td>75</td>
</tr>
<tr>
<td>3.4.3 Hardware Abstraction Layer</td>
<td>78</td>
</tr>
<tr>
<td>3.4.4 Hardware Layer</td>
<td>80</td>
</tr>
<tr>
<td>3.5 Communication Modeling</td>
<td>83</td>
</tr>
<tr>
<td>3.5.1 Application Layer</td>
<td>84</td>
</tr>
<tr>
<td>3.5.2 Presentation Layer</td>
<td>88</td>
</tr>
<tr>
<td>3.5.3 Session Layer</td>
<td>90</td>
</tr>
<tr>
<td>3.5.4 Network Layer</td>
<td>92</td>
</tr>
<tr>
<td>3.5.5 Transport Layer</td>
<td>93</td>
</tr>
<tr>
<td>3.5.6 Link Layer</td>
<td>94</td>
</tr>
<tr>
<td>3.5.7 Stream Layer</td>
<td>98</td>
</tr>
<tr>
<td>3.5.8 Media Access Layer</td>
<td>99</td>
</tr>
<tr>
<td>3.5.9 Protocol and Physical Layers</td>
<td>100</td>
</tr>
<tr>
<td>3.6 System Models</td>
<td>102</td>
</tr>
<tr>
<td>3.6.1 Specification Model</td>
<td>103</td>
</tr>
<tr>
<td>3.6.2 Network TLM</td>
<td>104</td>
</tr>
<tr>
<td>3.6.3 Protocol TLM</td>
<td>106</td>
</tr>
<tr>
<td>3.6.4 Bus Cycle-Accurate Model (BCAM)</td>
<td>107</td>
</tr>
<tr>
<td>3.6.5 Cycle-Accurate Model (CAM)</td>
<td>108</td>
</tr>
</tbody>
</table>
3.7 Summary

4. SYSTEM SYNTHESIS
4.1 System Design Trends
4.2 TLM Based Design
4.3 Automatic TLM Generation
   4.3.1 Application Modeling
   4.3.2 Platform Definition
   4.3.3 Application to Platform Mapping
   4.3.4 TLM Based Performance Estimation
   4.3.5 TLM Semantics
4.4 Automatic Mapping
   4.4.1 GSM Encoder Application
   4.4.2 Application Profiling
   4.4.3 Load Balancing Algorithm
   4.4.4 Longest Processing Time Algorithm
4.5 Platform Synthesis
   4.5.1 Component data models
   4.5.2 Platform Generation Algorithm
   4.5.3 Cycle Accurate Model Generation
   4.5.4 Summary

5. SOFTWARE SYNTHESIS
5.1 Preliminaries
   5.1.1 Target Languages for Embedded Systems
   5.1.2 RTOS
5.2 Software Synthesis Overview
   5.2.1 Example Input TLM
   5.2.2 Target Architecture
5.3 Code Generation
5.4 Multi-Task Synthesis
   5.4.1 RTOS-based Multi-Tasking
   5.4.2 Interrupt-based Multi-Tasking
5.5 Internal Communication
5.6 External Communication
   5.6.1 Data Formatting
   5.6.2 Packetization
   5.6.3 Synchronization
   5.6.4 Media Access Control
5.7 Startup Code 193
5.8 Binary Image Generation 194
5.9 Execution 195
5.10 Summary 196

6. HARDWARE SYNTHESIS 199
6.1 RTL Architecture 201
6.2 Input Models 204
   6.2.1 C-code specification 204
   6.2.2 Control-Data Flow Graph specification 205
   6.2.3 Finite State Machine with Data specification 207
   6.2.4 RTL specification 208
   6.2.5 HDL specification 209
6.3 Estimation and Optimization 211
6.4 Register Sharing 216
6.5 Functional Unit Sharing 220
6.6 Connection Sharing 224
6.7 Register Merging 227
6.8 Chaining and Multi-Cycling 229
6.9 Functional-Unit Pipelining 232
6.10 Datapath Pipelining 235
6.11 Control and Datapath Pipelining 237
6.12 Scheduling 240
   6.12.1 RC scheduling 243
   6.12.2 TC scheduling 244
6.13 Interface Synthesis 248
6.14 Summary 253

7. VERIFICATION 255
7.1 Simulation Based Methods 257
   7.1.1 Stimulus Optimization 260
   7.1.2 Monitor Optimization 262
   7.1.3 SpeedUp Techniques 263
   7.1.4 Modeling Techniques 264
7.2 Formal Verification Methods 265
   7.2.1 Logic Equivalence Checking 266
   7.2.2 FSM Equivalence Checking 268
7.2.3 Model Checking 270
7.2.4 Theorem Proving 273
7.2.5 Drawbacks of Formal Verification 275
7.2.6 Improvements to Formal Verification Methods 275
7.2.7 Semi-formal Methods: Symbolic Simulation 276

7.3 Comparative Analysis of Verification Methods 276

7.4 System Level Verification 278
  7.4.1 Formal Modeling 280
  7.4.2 Model Algebra 282
  7.4.3 Verification by Correct Refinement 283

7.5 Summary 285

8. EMBEDDED DESIGN PRACTICE 287

8.1 System Level Design Tools 287
  8.1.1 Academic Tools 289
  8.1.2 Commercial Tools 296
  8.1.3 Outlook 299

8.2 Embedded Software Design Tools 300
  8.2.1 Academic Tools 301
  8.2.2 Commercial Tools 303
  8.2.3 Outlook 305

8.3 Hardware Design Tools 306
  8.3.1 Academic Tools 308
  8.3.2 Commercial Tools 314
  8.3.3 Outlook 319

8.4 Case Study 319
  8.4.1 Embedded System Environment 320
  8.4.2 Design Driver: MP3 Decoder 324
  8.4.3 Results 327

8.5 Summary 333

References 335

Index 349
Embedded System Design
Modeling, Synthesis and Verification
Gajski, D.D.; Abdi, S.; Gerstlauer, A.; Schirner, G.
2009, XXV, 352 p., Hardcover
ISBN: 978-1-4419-0503-1