

Contents

Part I Design of Digital Circuits and Systems on the Basis of FPGA

1	FPGA Architectures, Reconfigurable Fabric, Embedded Blocks and Design Tools	3
1.1	Introduction to FPGA	3
1.2	The Basis of FPGA Devices	8
1.2.1	Configurable Logic Blocks of Xilinx FPGAs	8
1.2.2	Logic Elements of Altera FPGAs	12
1.3	Embedded Blocks	13
1.3.1	Embedded Memories	13
1.3.2	Embedded DSP Slices	17
1.4	Clock Distributions and Resets	20
1.5	Design Tools	21
1.6	Implementation and Prototyping	27
1.7	Interaction with FPGA-Based Circuits and Systems	34
	References	40
2	Synthesizable VHDL for FPGA-Based Devices	43
2.1	Introduction to VHDL	43
2.2	Data Types, Objects and Operators	50
2.3	Combinational and Sequential Processes	55
2.3.1	Combinational Processes	56
2.3.2	Sequential Processes	59
2.4	Functions, Procedures, and Blocks	63
2.5	Generics and Generates	70
2.6	Libraries, Packages, and Files	76
2.7	Behavioral Simulation	81
2.8	Prototyping	85
	References	88

3	Design Techniques	89
3.1	Combinational Circuits	89
3.1.1	Encoders	92
3.1.2	Decoders	93
3.1.3	Multiplexers	94
3.1.4	Comparators	95
3.1.5	Arithmetical Circuits	95
3.1.6	Barrel Shifters	96
3.2	Sequential Circuits	97
3.2.1	Registers	97
3.2.2	Shift Registers	98
3.2.3	Counters	98
3.2.4	Arithmetical Circuits with Accumulators	99
3.3	Finite State Machines	100
3.4	Optimization of FPGA-Based Circuits and Systems	104
3.4.1	Highly Parallel Network-Based Solutions	105
3.4.2	Hardware Accelerators	110
3.4.3	Parallel Modular Algorithms Running in Hierarchical FSMs	111
3.5	Design Examples for Parallel Sort	112
3.6	Design Examples for Parallel Search	118
3.7	Design Examples for Parallel Counters	123
3.8	Design Examples for Counting Networks	127
3.9	Design Examples for LUT-Based Hamming Weight Counters/Comparators	130
3.10	Design Examples for Operations Over Vectors	137
	References	141
4	Embedded Blocks and System-Level Design	143
4.1	Using IP Cores	143
4.2	Design with Embedded DSP Slices	153
4.3	Interaction with FPGA	158
4.3.1	Digilent Parallel Port Interface	159
4.3.2	UART Interface	167
4.4	Software/Hardware Co-design and Co-simulation	179
4.4.1	Software-Hardware Co-design with Digilent Parallel Port Interface	181
4.4.2	Software-Hardware Co-design with UART Interface	188
4.5	Programmable Systems-on-Chip	197
	References	202
5	Design Technique Based on Hierarchical and Parallel Specifications	205
5.1	Modular Hierarchical Specifications	205
5.2	Hierarchical Finite State Machines	210
5.2.1	HDL Template for HFSM with Explicit Modules	211

5.2.2	HDL Template for HFSM with Implicit Modules	220
5.3	Synthesis of HFSMs	221
5.3.1	Synthesis of HFSMs with Explicit Modules	222
5.3.2	Synthesis of HFSMs with Implicit Modules	229
5.4	Parallel Specifications and Parallel HFSMs	230
5.5	Hardware Implementations of Software Programs Based on HFSM Models	239
5.6	Using Stacks Based on Embedded or Distributed Memories	242
5.7	Optimization Techniques	245
5.7.1	Execution of Hierarchical Returns	245
5.7.2	Providing Multiple Entry Points to HGSS	246
5.7.3	Fast Stack Unwinding	247
5.8	Practical Applications	247
	References	254
 Part II Methods for Optimization of Finite State Machines for FPGA-Based Circuits and Systems		
6	Hardware Reduction in Logic Circuits of Moore FSM	259
6.1	General Characteristic of Existing Methods	259
6.2	Object Transformation in Moore FSM	266
6.3	Expansion of State Codes for Moore FSM	271
6.4	Synthesis of Moore FSM with Replacement of Logical Conditions	279
	References	283
7	Design of FSMs with Embedded Memory Blocks	285
7.1	Trivial Implementation of Mealy and Moore FSMs	285
7.2	Structural Decomposition of FSMs	291
7.3	Design of Mealy FSM with Encoding of the Collections of Microoperations	295
7.4	Design of Mealy FSM with Encoding of the Fields of Compatible Microoperations	298
7.5	Design of Mealy FSM with Encoding of the Rows of Structure Table	300
7.6	Optimization of BIMF Based on Pseudoequivalent States of Moore FSM	305
	References	310
8	Optimization of FSMs with Embedded Memory Blocks	313
8.1	Trivial Implementation of MP Mealy FSMs	313
8.2	Optimization of LUTer	321
8.3	Optimization of LUTer Based on Pseudoequivalent States	325
8.4	Optimization of LUTer Based on Encoding of Collections of Microoperations	334
	References	341

9 Finite State Machines with Operational Implementation of Transitions	343
9.1 Conception of Operational Implementation of Transitions	343
9.2 Organisation of FSM with Operational Generation of Transitions . .	346
9.3 Example of FSM Design	349
9.4 Structural Representation of Synthesis Process for FSM with OAT .	353
9.4.1 Base Structure of Synthesis Process for FSM with OAT . . .	354
9.4.2 Refinement of Basic Structure of Synthesis Process	355
9.5 Organization of Operational Automaton of Transitions	359
9.5.1 Typical Structure Models of Operational Automata.	359
9.5.2 Organizational Specifics of OAT	360
9.5.3 Organization of Combinational Part of OAT	360
9.6 Synthesis Method for FSM with Supplemented Set of Operations of Transitions	363
9.7 Investigation of Efficiency of FSM with OAT	367
References	373
 Appendix A: VHDL Constructions Used in the Book and Additional Support Materials	 375
 Appendix B: Coding Examples	 403
 Index	 429

Synthesis and Optimization of FPGA-Based Systems

Sklyarov, V.; Skliarova, I.; Barkalov, A.; Titarenko, L.

2014, XIX, 432 p. 235 illus., Hardcover

ISBN: 978-3-319-04707-2