

Advanced Digital System Design

Shirshendu Roy

Advanced Digital System Design

A Practical Guide to Verilog Based FPGA and ASIC Implementation



Ane Books
Pvt. Ltd.

 Springer

The Springer logo consists of a stylized chess knight piece, facing left, positioned above a horizontal line.

Shirshendu Roy
Department of Electronics
and Communication Engineering
Dayananda Sagar University
Bengaluru, India

ISBN 978-3-031-41084-0 ISBN 978-3-031-41085-7 (eBook)
<https://doi.org/10.1007/978-3-031-41085-7>

Jointly published with Ane Books Pvt. Ltd.

In addition to this printed edition, there is a local printed edition of this work available via Ane Books in South Asia (India, Pakistan, Sri Lanka, Bangladesh, Nepal and Bhutan) and Africa (all countries in the African subcontinent).

ISBN of the Co-Publisher's edition: 978-81-94891-88-8

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2024

This work is subject to copyright. All rights are solely and exclusively licensed by the Publisher, whether the whole or part of the material is concerned, specifically the rights of reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publishers, the authors, and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publishers nor the authors or the editors give a warranty, expressed or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publishers remain neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Paper in this product is recyclable.

*To My Adorable Daughter
Trijayee.*

Preface

Objective of the Book

In today's world where technology is applied at every application, there has been a huge demand of implementation of signal-, image- or video-processing algorithms. These real-time systems consist of both analog and digital sub-systems. The analog part is mainly responsible for signal acquisition step and the processing part is majorly achieved by digital sub-systems. An optimized implementation of a digital system is very crucial to improve the performance of the overall integrated circuit (IC).

Digital system design is not a new thing to the researchers or to the engineers in the field of VLSI system design. The field of digital system design is divided into two zones, viz., transistor-level design and gate-level architecture design. Over the past few decades many research works, books or online tutorials on both the topics of digital system design are published. In this book, gate-level design of digital systems using Verilog HDL is discussed. The major objective of this book is to cover all the topics which are very important for a gate-level digital system designer.

This book covers some basic topics from digital logic design like basic combinational circuits and sequential circuits. Also covers some advanced topics from digital arithmetic like fast circuit design for addition, multiplication, division and square root operation. Realization of circuits using Verilog HDL is also discussed in this book. Overview on the digital system implementation on Field Programmable Gate Array (FPGA) platform and for Application-Specific Integrated Circuit (ASIC) is covered in this book. Timing and power consumption analysis are two most important things that must be performed to make successful implementation. Thus this book covered these two areas to give readers an overview on timing and power analyses. At the end, few design examples are given in this book which can help readers directly or indirectly. Thus this book can be a perfect manual to the researchers in the field of digital system design.

Organization of the Book

Chapter 1 focusses on the representation of binary numbers. This chapter discusses the representation of binary numbers in One's complement, Two's complement and Signed magnitude number system. Basics of floating point data representation and fixed point data representation is discussed in this chapter. Signed binary number system which is frequently used for performing fast arithmetic operations is also discussed.

Chapter 2 discusses the Verilog HDL which is a very powerful programming language to model the digital systems. In this chapter, concepts about the Verilog HDL are discussed with suitable examples. All the different programming styles are discussed with the help of simple Multiplexer design. The test bench writing technique is also discussed in this chapter.

Basic concepts of combinational circuits are discussed in Chap. 3. All the major combinational circuits are covered in this chapter. Some of the basic circuits are Adder/Subtractor, Multiplexer, De-multiplexer, Encoder and Decoders. In addition to these circuits, design of 16-bit comparator, constant multipliers and code converters is also discussed.

Basic concepts of sequential circuits are discussed in Chap. 4. This chapter initially covers the concepts of different clocked flip-flops and then discusses about the various shift registers. Counter is a very important sequential circuit and this chapter discusses design of a simple synchronous up counter. Then this up counter is converted to a loadable up counter. In addition to the counter design, design of pseudonoise sequence generator and clock division circuits is also discussed.

In Chap. 5, memory design problem is discussed. This chapter mainly focusses on realization of memory elements using Verilog HDL. Behavioural HDL coding style is used to model the memory elements. Verilog codes for ROM and RAM are provided in this chapter. In addition to the single port memory elements, dual port ROM and dual port RAM are also modelled in this chapter.

Design of Finite State Machines is very important in designing digital systems. Thus a detailed discussion on the FSM design is given in Chap. 6. Design of Mealy and Moore machine is explained with the help of '1010' sequence detector. Then some of the applications are discussed where FSM design style is used. Various FSM state minimization techniques are also discussed in this chapter using a design problem.

Various architectures for addition operation are discussed in Chap. 7. This chapter mainly focusses on fast addition techniques but also discusses some other addition techniques. The different techniques which are discussed here are Carry Look-Ahead, Carry Skip, Conditional Sum, Carry Increment and Carry Bypass. Multi-operand addition techniques like Carry Save Adders are also discussed here.

Chapter 8 focusses on various architectures for multiplication operation and these architectures can be sequential or parallel. The array multipliers for both signed and unsigned operands are discussed. Like previous chapter, this chapter also focusses mainly on fast multiplication techniques like Booth multiplier. But, other important

multiplier design aspects like VEDIC multiplication techniques are also discussed here. Along with the multiplication, techniques to efficiently compute square of a number are also discussed in this chapter.

Chapter 9 discusses various division algorithms like restoring and non-restoring algorithm with proper example. Implementation of these algorithms is discussed here. Basic principle of SRT division algorithm is also given here with some examples. Some iterative algorithms for division operation are also explained here. Along with the division operation, computation of modulus operation without division operation is discussed in this chapter.

Square root and square root reciprocal are also very important arithmetic operations in implementing digital systems. Thus in Chap. 10, various algorithms and architectures to compute square root and square root reciprocal are discussed. Sequential algorithms, restoring and non-restoring algorithm also can be applied to compute square root. Likewise SRT algorithm is also applicable for square root with minor modifications. Some iterative algorithms are also explained to compute square root and square root reciprocal.

CORDIC algorithm is a very promising algorithm to compute various arithmetic operations and some other functions. Thus in Chap. 11, CORDIC theory and its architectures are explained. Two architectures for CORDIC are possible, serial and parallel. Both the architectures are discussed in detail. This chapter also provides a brief survey on different CORDIC architectures which are reported in recent publications.

Till this chapter fixed point data point is used to implement the digital systems. But floating point representation is another technique to represent the real numbers. Floating point data format is useful if high accuracy is desired. Thus in Chap. 12 floating point architectures are discussed to compute addition/subtraction, multiplication, division and square root with proper examples.

Timing analysis or more specifically static timing analysis is an important step to verify that a digital IC will work satisfactorily after fabrication or not. Thus Chap. 13 focusses on explaining different timing definitions and important concepts of static timing analysis. These topics are discussed here so that readers can carefully plan their design for desired maximum frequency at strict area constraint.

Digital systems can be implemented on FPGA platform or can be designed for ASIC as an IC. Chapter 14 covers a detailed discussion on the FPGA and ASIC implementation steps. First a detailed theory on the FPGA device is discussed and then the FPGA implementation steps are explained using XILINX EDA tool. A brief theory on the ASIC implementation using the standard cells with help of CADENCE EDA tool is covered.

Power consumption is a very important design metric to analyse the design performance. Thus Chap. 15 focusses on various techniques to achieve low power consumption. Dynamic power consumption can be reduced at every level of abstraction. Dynamic power consumption reduction using both algorithmic and architectural techniques is discussed here.

Example of some digital systems is given in Chap. 16 to give the readers idea about designing their own systems. First, implementation of digital filters (FIR and IIR) is

described using various topologies. Comparative study of the performances of the different FIR and IIR filter structures is also given. Two algorithms are implemented on FPGA which are K-means algorithm and spatial Median filtering algorithm. In addition to this, various sorting structures and architectures for matrix multiplication are discussed. At last, Verilog codes are provided to interface SPI protocol-based external ICs (DAC, ADC) or computers and micro-controllers using UART protocol with the FPGA device.

Verilog HDL is very popular in modelling the digital systems but has some limitations when verification of such systems comes into the picture. Thus system Verilog develops. Nowadays, system Verilog is mostly used and industry standard, which combines the features of C++ and Verilog. Basics of system Verilog is discussed in Chap. 17. This chapter highlights the major features of system Verilog and the differences from Verilog HDL.

Many advanced technologies are established to program the FPGAs. One such advancement is the idea to integrate the whole system on a single chip. In order to do this, many modern FPGAs are accommodating a dedicated processor. Partial re-configuration is another advanced feature of modern FPGAs. Thus in Chap. 18, these modern techniques of FPGA implementation are discussed.

Bengaluru, India

Shirshendu Roy

Acknowledgements

I would like to place on record my gratitude and deep obligation to the professors of IEST Shibpur and NIT Rourkela as this book is a result of their teachings and guidance. Specifically, I like to thank Dr. Ayan Banerjee, IEST Shibpur (Department of ETC) and Dr. Debiprasad P. Acharya, NIT Rourkela (Department of ECE) to inspire me to pursue research in the field of digital system design. Also, I like to thank Prof. Ayas K. Swain whose teachings helped me writing this book.

I am indebted to my fellow researches and friends who have helped me by giving inspiration, moral support and encouragement to complete the book. Specifically I would like to thank S. Aloka Patra, Ardhendu Sarkar, Sandeep Gajendra and Jayanta Panigrahi to help me in finalizing the contents and preparing the manuscript.

I give immeasurable thanks to my family members for their patience, understanding and encouragement during the preparation of this book. They all kept me going and this book would not have been possible without their support.

Contents

1	Binary Number System	1
1.1	Introduction	1
1.2	Binary Number System	1
1.3	Representation of Numbers	2
1.3.1	Signed Magnitude Representation	2
1.3.2	One's Complement Representation	3
1.3.3	Two's Complement Representation	4
1.4	Binary Representation of Real Numbers	6
1.4.1	Fixed Point Data Format	6
1.5	Floating Point Data Format	7
1.6	Signed Number System	9
1.6.1	Binary SD Number System	9
1.6.2	SD Representation to Two's Complement Representation	12
1.7	Conclusion	13
2	Basics of Verilog HDL	15
2.1	Introduction	15
2.2	Verilog Expressions	16
2.2.1	Verilog Operands	16
2.2.2	Verilog Operators	16
2.2.3	Concatenation and Replication	16
2.3	Data Flow Modelling	18
2.4	Behavioural Modelling	20
2.4.1	Initial Statement	20
2.4.2	Always Statement	21
2.4.3	Timing Control	21
2.4.4	Procedural Assignment	24

2.5	Structural Modelling	26
2.5.1	Gate-Level Modelling	26
2.5.2	Hierarchical Modelling	27
2.6	Mixed Modelling	28
2.7	Verilog Function	29
2.8	Verilog Task	30
2.9	File Handling	30
2.9.1	Reading from a Text File	31
2.9.2	Writing into a Text File	31
2.10	Test Bench Writing	32
2.11	Frequently Asked Questions	33
2.12	Conclusion	38
3	Basic Combinational Circuits	39
3.1	Introduction	39
3.2	Addition	39
3.3	Subtraction	41
3.4	Parallel Binary Adder	42
3.5	Controlled Adder/Subtractor	43
3.6	Multiplexers	44
3.7	De-Multiplexers	44
3.8	Decoders	45
3.9	Encoders	45
3.10	Majority Voter Circuit	46
3.11	Data Conversion Between Binary and Gray Code	47
3.12	Conversion Between Binary and BCD Code	48
3.12.1	Binary to BCD Conversion	49
3.12.2	BCD to Binary Conversion	51
3.13	Parity Generators/Checkers	52
3.14	Comparators	53
3.15	Constant Multipliers	55
3.16	Frequently Asked Questions	57
3.17	Conclusion	60
4	Basic Sequential Circuits	61
4.1	Introduction	61
4.2	Different Flip-Flops	61
4.2.1	SR Flip-Flop	62
4.2.2	JK Flip-Flop	63
4.2.3	D Flip-Flop	65
4.2.4	T Flip-Flop	67
4.2.5	Master-Slave D Flip-Flop	68
4.3	Shift Registers	68
4.3.1	Serial In Serial Out	69
4.3.2	Serial In Parallel Out	69

4.3.3	Parallel In Serial Out	70
4.3.4	Parallel In Parallel Out	71
4.4	Sequence Generator	72
4.5	Pseudo Noise Sequence Generator	73
4.6	Synchronous Counter Design	75
4.7	Loadable Counter	77
4.7.1	Loadable Up Counter	78
4.7.2	Loadable Down Counter	78
4.8	Even and Odd Counter	79
4.9	Shift Register Counters	80
4.10	Phase Generation Block	82
4.11	Clock Divider Circuits	82
4.11.1	Clock Division by Power of 2	83
4.11.2	Clock Division by 3	84
4.11.3	Clock Division by 6	85
4.11.4	Programmable Clock Divider Circuit	86
4.12	Frequently Asked Questions	86
4.13	Conclusion	88
5	Memory Design	89
5.1	Introduction	89
5.2	Controlled Register	89
5.3	Read Only Memory	90
5.3.1	Single Port ROM	90
5.3.2	Dual Port ROM (DPRAM)	92
5.4	Random Access Memory (RAM)	93
5.4.1	Single Port RAM (SPRAM)	93
5.4.2	Dual Port RAM (DPRAM)	94
5.5	Memory Initialization	97
5.6	Implementing Bigger Memory Element Using Smaller Memory Elements	97
5.7	Implementation of Memory Elements	98
5.8	Conclusion	100
6	Finite State Machines	101
6.1	Introduction	101
6.2	FSM Types	101
6.3	Sequence Detector Using Mealy Machine	103
6.4	Sequence Detector Using Moore Machine	107
6.5	Comparison of Mealy and Moore Machine	111
6.6	FSM-Based Serial Adder Design	111
6.7	FSM-Based Vending Machine Design	113
6.8	State Minimization Techniques	115
6.9	Row Equivalence Method	115
6.10	Implication Chart Method	116
6.11	State Partition Method	119

6.12	Performance of State Minimization Techniques	120
6.13	Verilog Modelling of FSM-Based Systems	120
6.14	Frequently Asked Questions	123
6.15	Conclusion	126
7	Design of Adder Circuits	127
7.1	Introduction	127
7.2	Ripple Carry Adder	127
7.3	Carry Look-Ahead Adder	128
7.3.1	Higher Bit Adders Using CLA	130
7.3.2	Prefix Tree Adders	132
7.4	Manchester Carry Chain Module (MCC)	136
7.5	Carry Skip Adder	137
7.6	Carry Increment Adder	137
7.7	Carry Select Adder	137
7.8	Conditional Sum Adder	138
7.9	Ling Adders	139
7.10	Hybrid Adders	140
7.11	Multi-operand Addition	141
7.11.1	Carry Save Addition	141
7.11.2	Tree of Carry Save Adders	142
7.12	BCD Addition	142
7.13	Conclusion	144
8	Design of Multiplier Circuits	145
8.1	Introduction	145
8.2	Sequential Multiplication	145
8.3	Array Multipliers	146
8.4	Partial Product Generation and Reduction	149
8.4.1	Booth's Multiplication	149
8.4.2	Radix-4 Booth's Algorithm	150
8.4.3	Canonical Recoding	154
8.4.4	An Alternate 2-bit at-a-time Multiplication Algorithm	154
8.4.5	Implementing Larger Multipliers Using Smaller Ones	156
8.5	Accumulation of Partial Products	156
8.5.1	Accumulation of Partial Products for Unsigned Numbers	157
8.5.2	Accumulation of Partial Products for Signed Numbers	159
8.5.3	Alternative Techniques for Partial Product Accumulation	162
8.6	Wallace and Dedda Multiplier Design	163
8.7	Multiplication Using Look-Up Tables	167
8.8	Dedicated Square Block	168

8.9	Architectures Based on VEDIC Arithmetic	170
8.9.1	VEDIC Multiplier	170
8.9.2	VEDIC Square Block	171
8.9.3	VEDIC Cube Block	172
8.10	Conclusion	175
9	Division and Modulus Operation	177
9.1	Introduction	177
9.2	Sequential Division Methods	177
9.2.1	Restoring Division	178
9.2.2	Unsigned Array Divider	180
9.2.3	Non-restoring Division	181
9.2.4	Conversion from Signed Binary to Two's Complement	184
9.3	Fast Division Algorithms	185
9.3.1	SRT Division	185
9.3.2	SRT Algorithm Properties	186
9.4	Iterative Division Algorithms	187
9.4.1	Goldschmidt Division	187
9.4.2	Newton–Raphson Division	187
9.5	Computation of Modulus	188
9.6	Conclusion	191
10	Square Root and its Reciprocal	193
10.1	Introduction	193
10.2	Slow Square Root Computation Methods	193
10.2.1	Restoring Algorithm	194
10.2.2	Non-restoring Algorithm	195
10.3	Iterative Algorithms for Square Root and its Reciprocal	197
10.3.1	Goldschmidt Algorithm	197
10.3.2	Newton–Raphson Iteration	198
10.3.3	Halley's Method	199
10.3.4	Bakhshali Method	199
10.3.5	Two Variable Iterative Method	199
10.4	Fast SRT Algorithm for Square Root	200
10.5	Taylor Series Expansion Method	200
10.5.1	Theory	200
10.5.2	Implementation	202
10.6	Function Evaluation by Bipartite Table Method	203
10.7	Conclusion	205
11	CORDIC Algorithm	207
11.1	Introduction	207
11.2	Theoretical Background	207
11.3	Vectoring Mode	212
11.3.1	Computation of Sine and Cosine	213

11.4	Linear Mode	214
11.4.1	Multiplication	215
11.4.2	Division	215
11.5	Hyperbolic Mode	215
11.5.1	Square Root Computation	216
11.6	CORDIC Algorithm Using Redundant Number System	217
11.6.1	Redundant Radix-2-Based CORDIC Algorithm	217
11.6.2	Redundant Radix-4-Based CORDIC Algorithm	219
11.7	Example of CORDIC Iteration	219
11.8	Implementation of CORDIC Algorithms	219
11.8.1	Parallel Architecture	220
11.8.2	Serial Architecture	220
11.8.3	Improved CORDIC Architectures	222
11.9	Application	225
11.10	Conclusion	225
12	Floating Point Architectures	227
12.1	Introduction	227
12.2	Floating Point Representation	228
12.3	Fixed Point to Floating Point Conversion	230
12.4	Leading Zero Counter	231
12.5	Floating Point Addition	233
12.6	Floating Point Multiplication	236
12.7	Floating Point Division	238
12.8	Floating Point Comparison	239
12.9	Floating Point Square Root	240
12.10	Floating Point to Fixed Point Conversion	242
12.11	Conclusion	243
13	Timing Analysis	245
13.1	Introduction	245
13.2	Timing Definitions	246
13.2.1	Slew of Waveform	246
13.2.2	Clock Jitter	246
13.2.3	Clock Latency	247
13.2.4	Launching and Capturing Flip-Flop	248
13.2.5	Clock Skew	248
13.2.6	Clock Uncertainty	249
13.2.7	Clock-to-Q Delay	249
13.2.8	Combinational Logic Timing	250
13.2.9	Min and Max Timing Paths	250
13.2.10	Clock Domains	251
13.2.11	Setup Time	251
13.2.12	Hold Time	251

13.2.13	Slack	252
13.2.14	Required Time and Arrival Time	253
13.2.15	Timing Paths	253
13.3	Timing Checks	253
13.3.1	Setup Timing Check	253
13.3.2	Hold Timing Check	254
13.4	Timing Checks for Different Timing Paths	254
13.4.1	Setup Check for Flip-Flop to Flip-Flop Timing Path	255
13.4.2	Setup and Hold Check for Input to Flip-Flop Timing Path	257
13.4.3	Setup Check for Flip-Flop to Output Timing Path ...	258
13.4.4	Setup Check for Input to Output Timing Path	258
13.4.5	Multicycle Paths	259
13.4.6	False Paths	260
13.4.7	Half Cycle Paths	260
13.5	Asynchronous Checks	261
13.5.1	Recovery Timing Check	261
13.5.2	Removal Timing Check	262
13.6	Maximum Frequency Computation	262
13.7	Maximum Allowable Skew	263
13.8	Frequently Asked Questions	266
13.9	Conclusion	268
14	Digital System Implementation	269
14.1	Introduction	269
14.2	FPGA Implementation	270
14.2.1	Internal Structure of FPGA	270
14.2.2	FPGA Implementation Using XILINX EDA Tool	276
14.2.3	Design Verification	279
14.2.4	FPGA Editor	280
14.3	ASIC Implementation	280
14.3.1	Simulation and Synthesis	281
14.3.2	Placement and Routing	283
14.4	Frequently Asked Questions	292
14.5	Conclusion	295
15	Low-Power Digital System Design	297
15.1	Introduction	297
15.2	Different Types of Power Consumption	297
15.2.1	Switching Power	298
15.2.2	Short Circuit Power	301
15.2.3	Leakage Power	301
15.2.4	Static Power	301

15.3	Architecture-Driven Voltage Scaling	302
15.3.1	Serial Architecture	302
15.3.2	Parallel Architecture	303
15.3.3	Pipeline Architecture	304
15.4	Algorithmic Optimization	304
15.4.1	Minimizing the Hardware Complexity	305
15.4.2	Selection of Data Representation Techniques	306
15.5	Architectural Optimization	307
15.5.1	Choice of Data Representation Techniques	307
15.5.2	Ordering of Input Signals	308
15.5.3	Reducing Glitch Activity	308
15.5.4	Choice of Topology	309
15.5.5	Logic Level Power Down	309
15.5.6	Synchronous Versus Asynchronous	309
15.5.7	Loop Unrolling	310
15.5.8	Operation Reduction	311
15.5.9	Substitution of Operation	313
15.5.10	Re-timing	314
15.5.11	Wordlength Reduction	316
15.5.12	Resource Sharing	316
15.6	Frequently Asked Questions	317
15.7	Conclusion	319
16	Digital System Design Examples	321
16.1	FPGA Implementation FIR Filters	322
16.1.1	FIR Low-Pass Filter	323
16.1.2	Advanced DSP Blocks	324
16.1.3	Different Filter Structures	325
16.1.4	Performance Estimation	330
16.1.5	Conclusion	332
16.1.6	Top Module for FIR Filter in Transposed Direct Form	332
16.2	FPGA Implementation of IIR Filters	333
16.2.1	IIR Low-Pass Filter	334
16.2.2	Different IIR Filter Structures	335
16.2.3	Pipeline Implementation of IIR Filters	338
16.2.4	Performance Estimation	342
16.2.5	Conclusion	344
16.3	FPGA Implementation of K-Means Algorithm	345
16.3.1	K-Means Algorithm	346
16.3.2	Example of K-Means Algorithm	347
16.3.3	Proposed Architecture	348
16.3.4	Design Performance	351
16.3.5	Conclusion	352

16.4	Matrix Multiplication	352
16.4.1	Matrix Multiplication by Scalar–Vector Multiplication	353
16.4.2	Matrix Multiplication by Vector–Vector Multiplication	354
16.4.3	Systolic Array for Matrix Multiplication	355
16.5	Sorting Architectures	359
16.5.1	Parallel Sorting Architecture 1	359
16.5.2	Parallel Sorting Architecture 2	359
16.5.3	Serial Sorting Architecture	360
16.5.4	Sorting Processor Design	361
16.6	Median Filter for Image De-noising	363
16.6.1	Median Filter	363
16.6.2	FPGA Implementation of Median Filter	365
16.7	FPGA Implementation of 8-Point FFT	367
16.7.1	Data Path for 8-Point FFT Processor	368
16.7.2	Control Path for 8-Point FFT Processor	370
16.8	Interfacing ADC Chips with FPGA Using SPI Protocol	371
16.9	Interfacing DAC Chips with FPGA Using SPI Protocol	378
16.10	Interfacing External Devices with FPGA Using UART	382
16.11	Conclusion	388
17	Basics of System Verilog	391
17.1	Introduction	391
17.2	Language Elements	391
17.2.1	Logic Literal Values	391
17.2.2	Basic Data Types	392
17.2.3	User Defined Data-Types	393
17.2.4	Enumeration Data Type	393
17.2.5	Arrays	394
17.2.6	Dynamic Arrays	395
17.2.7	Associative Array	396
17.2.8	Queues	396
17.2.9	Events	397
17.2.10	String Methods	397
17.3	Composite Data Types	398
17.3.1	Structures	398
17.3.2	Unions	400
17.3.3	Classes	401
17.4	Expressions	402
17.4.1	Parameters and Constants	402
17.4.2	Variables	403
17.4.3	Operators	404
17.4.4	Set Membership Operator	405
17.4.5	Static Cast Operator	405

17.4.6	Dynamic Casting	406
17.4.7	Type Operator	407
17.4.8	Concatenation of String Data Type	407
17.4.9	Streaming Operators	407
17.5	Behavioural Modelling	408
17.5.1	Procedural Constructs	408
17.5.2	Loop Statements	410
17.5.3	Case Statement	413
17.5.4	If Statement	414
17.5.5	Final Statement	415
17.5.6	Disable Statement	416
17.5.7	Event Control	417
17.5.8	Continuous Assignment	417
17.5.9	Parallel Blocks	418
17.5.10	Process Control	419
17.6	Structural Modelling	420
17.6.1	Module Prototype	420
17.7	Summary	423
18	Advanced FPGA Implementation Techniques	425
18.1	Introduction	425
18.2	System-On-Chip Implementation	425
18.2.1	Implementations Using SoC FPGAs	427
18.2.2	AXI Protocol	430
18.2.3	AXI Protocol Features	431
18.3	Partial Re-configuration (PR)	432
18.3.1	Dynamic PR	432
18.3.2	Advantages of DPR	432
18.3.3	DPR Techniques	433
18.3.4	DPR Terminology	434
18.3.5	DPR Tools	436
18.3.6	DPR Flow	436
18.3.7	Communication Between Reconfigurable Modules	437
18.4	Conclusion	441
	References	443
	Index	447

About the Author

Dr. Shirshendu Roy has completed Bachelor of Engineering (B.E.) degree in Electronics and Tele-Communication Engineering in 2010 and Master of Engineering (M.E.) in Digital Systems and Instrumentation in 2016 from Indian Institute of Engineering Science and Technology (IIST), Shibpur, Howrah, India. He has four years of valuable industrial experience as Control and Instrumentation engineer at MAHAN Captive Power Plant for Hindalco Industries Limited (Aditya Birla Group). He has completed Ph.D. degree from National Institute of Technology (NIT), Rourkela, Odisha, India in VLSI signal processing. Previously he worked in Gandhi Institute of GIET University, Odisha as assistant professor. Currently he is working in Dayananda Sagar University, Bengaluru as assistant professor.

He has published many international journals with the publishing houses like IEEE and IET. He has also authored and published many tutorials on the online platform in the field of Digital System Design. His current research interest includes compressed sensing, FPGA based implementation of algorithms (signal, image or video processing algorithms, machine learning algorithms, artificial neural networks, etc.), low-power architecture design, ASIC, application of FPGA for IoT applications.

Abbreviations

AB	Average Block
ADC	Analog-to-Digital Converter
ALU	Arithmetic Logic Unit
AMBA	Arm Advanced Micro-controller Bus Architecture
ASIC	Application-Specific Integrated Circuit
ATPG	Automatic Test Pattern Generator
AXI	Advanced Extensible Interface
BCD	Binary Coded Decimal
BIC	Bus Inversion Coding
BN	Basic Network
BNS	Binary Number System
BRAM	Block RAM
CB	Cluster Block
CIA	Carry Increment Adder
CLA	Carry Look-Ahead
CLB	Configurable Logic Block
CMOS	Complementary Metal-Oxide Semiconductor
CORDIC	Co-Ordinate Rotation DIgital Computer
CPA	Carry Propagate Adder
CPF	Common Power Format
CPU	Central Processing Unit
CSA	Carry Save Adder
CTS	Clock Tree Synthesis
DAC	Digital-to-Analog Converter
DCO	Digital Controlled Oscillator
DDR	Double Data Rate
DFT	Design For Testability
DIT	Decimation In Time
DPR	Dynamic Partial Re-configuration
DPRAM	Dual Port Random Access Memory
DPROM	Dual Port Read-Only Memory

DRC	Design Rule Checks
DSP	Digital Signal Processing
DTS	Dynamic Timing Simulation
EDC	Euclidean Distance Calculator
ERC	Electrical Rule Checks
FA	Full Adder
FFT	Fast Fourier Transform
FIR	Finite Impulse Response
FPGA	Field Programmable Gate Array
FS	Full Subtractor
GE	Gaussian Elimination
GPIO	General-Purpose Input/Output
GPU	General Processing Unit
HA	Half-Adder
HDL	Hardware Description Language
HS	Half-Subtractor
IC	Integrated Circuit
ICAP	Internal Configuration Access Port
IIR	Infinite Impulse Response
ILA	Inline Logic Analyzer
IOB	Input/Output Block
IP	Intellectual Property
LEC	Logic Equivalence Check
LEF	Library Exchange Format
LFSR	Linear Feedback Shift Register
LIB	LIBerty timing models
LPF	Low-Pass Filter
LSB	Least Significant Bit
LUT	Look-Up Table
LVS	Layout Vs. Schematic
LZC	Leading Zero Counter
MAC	Multiply-ACcumulate
MCAP	Media Configuration Access Port
MCC	Manchester Carry Chain
MCF	Modified Cholesky Factorization
MFB	Minimum Finder Block
MMMC	Multi-mode Multi-corner
MSB	Most Significant Bit
NAN	Not A Number
NCD	Native Circuit Description
NGC	Native Generic Circuit
NGD	Native Generic Database
NRE	Non-recurring Engineering
OS	Occupied Slices
OTP	One Time Programmable

PAR	Placement And Routing
PCAP	Processor Configuration Access Port
PCF	Physical Constraints File
PG	Phase Generation
PIPO	Parallel Input Parallel Output
PISO	Parallel Input Serial Output
PL	Programmable Logic
PLL	Phase Locked Loop
PN	Pseudonoise
PR	Partial Re-configuration
PRM	Partial Re-configuration Modules
PS	Processing System
PSM	Programmable Switching Block
QRD	QR Decomposition
RCA	Ripple Carry Adder
RMSE	Root Mean Squared Error
RTL	Register Transfer Logic
SAIF	Switching Activity Interchange Format
SB	Sub-block
SD	Signed Digit
SDC	Synopsys Design Constraints
SDF	Standard Delay Format
SEU	Single Event Upsets
SI	Signal Integrity
SIPO	Serial Input Parallel Output
SISO	Serial Input Serial Output
SoC	System-on-Chip
SPI	Serial-to-Parallel Interface
SPR	Static Partial Re-configuration
SPRAM	Single Port Random Access Memory
SPROM	Single Port Read-Only Memory
SRAM	Static RAM
STA	Static Timing Analysis
TDP	Time-Driven Placement
TNS	Total Negative Slack
UART	Universal Asynchronous Receiver and Transmitter
UCF	User Constraints File
UUT	Unit Under Test
VCD	Value Change Dump
WNS	Worst Negative Slack
XST	Xilinx Synthesis Technology