

The Design Warrior's Guide to FPGAs

Devices, Tools and Flows

Clive "Max" Maxfield



AMSTERDAM • BOSTON • HEIDELBERG • LONDON
NEW YORK • OXFORD • PARIS • SAN DIEGO
SAN FRANCISCO • SINGAPORE • SYDNEY • TOKYO

Newnes is an imprint of Elsevier



Contents

Preface	ix
Acknowledgments	xi
Chapter 1	
Introduction	1
What are FPGAs?	1
Why are FPGAs of interest?	1
What can FPGAs be used for?	4
What's in this book?	6
What's not in this book?	7
Who's this book for?	8
Chapter 2	
Fundamental Concepts	9
The key thing about FPGAs.	9
A simple programmable function	9
Fusible link technologies	10
Antifuse technologies	12
Mask-programmed devices	14
PROMs	15
EPROM-based technologies	17
EEPROM-based technologies	19
FLASH-based technologies	20
SRAM-based technologies	21
Summary	22
Chapter 3	
The Origin of FPGAs	25
Related technologies	25
Transistors	26
Integrated circuits	27
SRAMs, DRAMs, and microprocessors.	28
SPLDs and CPLDs	28
ASICs (gate arrays, etc.)	42
FPGAs	49
Chapter 4	
Alternative FPGA Architectures 57	
A word of warning	57
A little background information	57
Antifuse versus SRAM versus	59
Fine-, medium-, and coarse-grained architectures	66
MUX- versus LUT-based logic blocks	68
CLBs versus LABs versus slices.	73
Fast carry chains	77
Embedded RAMs.	78
Embedded multipliers, adders, MACs, etc.	79
Embedded processor cores (hard and soft)	80
Clock trees and clock managers	84
General-purpose I/O	89
Gigabit transceivers	92
Hard IP, soft IP, and firm IP	93
System gates versus real gates	95
FPGA years.	98

Chapter 5**Programming (Configuring)**

an FPGA	99
Weasel words	99
Configuration files, etc.	99
Configuration cells	100
Antifuse-based FPGAs	101
SRAM-based FPGAs	102
Using the configuration port	105
Using the JTAG port	111
Using an embedded processor.	113

Chapter 6**Who Are All the Players? . . . 115**

Introduction.	115
FPGA and FPAA vendors	115
FPNA vendors	116
Full-line EDA vendors	116
FPGA-specialist and independent EDA vendors	117
FPGA design consultants with special tools.	118
Open-source, free, and low-cost design tools.	118

Chapter 7**FPGA Versus ASIC****Design Styles. 121**

Introduction.	121
Coding styles	122
Pipelining and levels of logic	122
Asynchronous design practices	126
Clock considerations	127
Register and latch considerations	129
Resource sharing (time-division multi- plexing)	130
State machine encoding	131
Test methodologies	131

Chapter 8**Schematic-Based Design Flows 133**

In the days of yore.	133
The early days of EDA	134
A simple (early) schematic-driven ASIC flow.	141
A simple (early) schematic-driven FPGA flow	143
Flat versus hierarchical schematics	148
Schematic-driven FPGA design flows today.	151

Chapter 9**HDL-Based Design Flows . . . 153**

Schematic-based flows grind to a halt.	153
The advent of HDL-based flows	153
Graphical design entry lives on.	161
A positive plethora of HDLs	163
Points to ponder.	172

Chapter 10**Silicon Virtual Prototyping****for FPGAs 179**

Just what is an SVP?	179
ASIC-based SVP approaches	180
FPGA-based SVPs	187

Chapter 11**C/C++ etc.-Based Design Flows****. 193**

Problems with traditional HDL-based flows	193
C versus C++ and concurrent versus sequential	196
SystemC-based flows	198
Augmented C/C++-based flows	205
Pure C/C++-based flows	209
Different levels of synthesis abstraction	213

Mixed-language design and verification environments . . . 214

Chapter 12
DSP-Based Design Flows . . . 217
 Introducing DSP 217
 Alternative DSP implementations 218
 FPGA-centric design flows for DSPs. 225
 Mixed DSP and VHDL/Verilog etc. environments . . . 236

Chapter 13
Embedded Processor-Based Design Flows 239
 Introduction. 239
 Hard versus soft cores 241
 Partitioning a design into its hardware and software components 245
 Hardware versus software views of the world 247
 Using an FPGA as its own development environment . . . 249
 Improving visibility in the design 250
 A few coverification alternatives 251
 A rather cunning design environment 257

Chapter 14
Modular and Incremental Design 259
 Handling things as one big chunk 259
 Partitioning things into smaller chunks 261
 There's always another way. 264

Chapter 15
High-Speed Design and Other PCB Considerations 267
 Before we start. 267
 We were all so much younger then 267
 The times they are a-changing 269
 Other things to think about . . . 272

Chapter 16
Observing Internal Nodes in an FPGA 277
 Lack of visibility. 277
 Multiplexing as a solution 278
 Special debugging circuitry 280
 Virtual logic analyzers. 280
 VirtualWires 282

Chapter 17
Intellectual Property 287
 Sources of IP 287
 Handcrafted IP 287
 IP core generators 290
 Miscellaneous stuff 291

Chapter 18
Migrating ASIC Designs to FPGAs and Vice Versa 293
 Alternative design scenarios . . . 293

Chapter 19
Simulation, Synthesis, Verification, etc. Design Tools 299
 Introduction. 299
 Simulation (cycle-based, event-driven, etc.) 299
 Synthesis (logic/HDL versus physically aware) 314
 Timing analysis (static versus dynamic). 319
 Verification in general 322
 Formal verification 326

Miscellaneous	338	Introduction.	381
Chapter 20		Algorithmic evaluation	383
Choosing the Right Device	343	picoChip's picoArray technology	384
So many choices	343	QuickSilver's ACM technology	388
If only there were a tool.	343	It's silicon, Jim, but not	
Technology	345	as we know it!.	395
Basic resources and packaging	346	Chapter 24	
General-purpose I/O interfaces	347	Independent Design Tools	397
Embedded multipliers,		Introduction.	397
RAMs, etc.	348	ParaCore Architect	397
Embedded processor cores.	348	The Confluence system	
Gigabit I/O capabilities	349	design language	401
IP availability	349	Do you have a tool?	406
Speed grades.	350	Chapter 25	
On a happier note.	351	Creating an Open-Source-Based	
Chapter 21		Design Flow	407
Gigabit Transceivers	353	How to start an FPGA design	
Introduction.	353	shop for next to nothing	407
Differential pairs	354	The development platform: Linux	
Multiple standards	357	407
8-bit/10-bit encoding, etc.	358	The verification environment	411
Delving into the transceiver		Formal verification	413
blocks	361	Access to common IP components	
Ganging multiple transceiver		416
blocks together	362	Synthesis and implementation	
Configurable stuff	364	tools	417
Clock recovery, jitter, and		FPGA development boards	418
eye diagrams.	367	Miscellaneous stuff	418
Chapter 22		Chapter 26	
Reconfigurable Computing.	373	Future FPGA Developments	419
Dynamically reconfigurable logic	373	Be afraid, be very afraid	419
Dynamically reconfigurable		Next-generation architectures	
interconnect	373	and technologies	420
Reconfigurable computing	374	Don't forget the design tools	426
Chapter 23		Expect the unexpected	427
Field-Programmable		Appendix A:	
Node Arrays	381	Signal Integrity 101	429
		Before we start.	429

Capacitive and inductive coupling (crosstalk)	430
Chip-level effects	431
Board-level effects.	438
Appendix B:	
Deep-Submicron Delay Effects 101	443
Introduction.	443
The evolution of delay specifications	443
A potpourri of definitions.	445
Alternative interconnect models	449
DSM delay effects.	452
Summary	464
Appendix C:	
Linear Feedback Shift Registers 101	465
The Ouroboros	465
Many-to-one implementations .	465
More taps than you know what to do with	468
Seeding an LFSR	470
FIFO applications	472
Modifying LFSRs to sequence 2 ⁿ values.	474
Accessing the previous value . .	475
Encryption and decryption applications	476
Cyclic redundancy check applications	477
Data compression applications .	479
Built-in self-test applications . .	480
Pseudorandom-number-generation applications	482
Last but not least	482
Glossary.	485
About the Author	525
Index	527