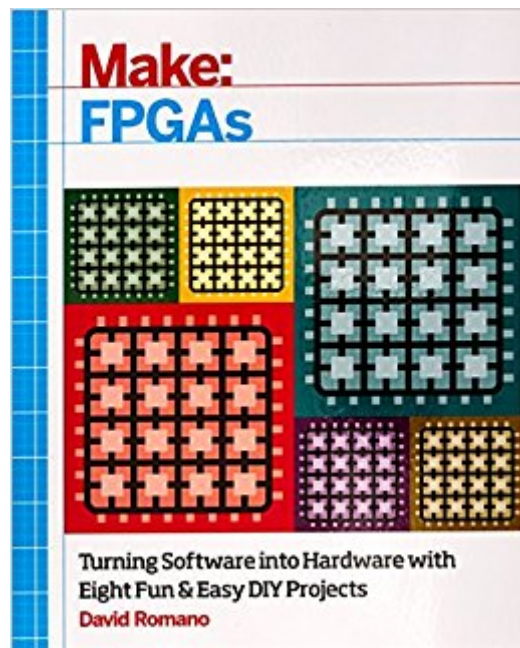The book was found

Make: FPGAs: Turning Software Into Hardware With Eight Fun And Easy DIY Projects

Synopsis

What if you could use software to design hardware? Not just any hardware--imagine specifying the behavior of a complex parallel computer, sending it to a chip, and having it run on that chip--all without any manufacturing? With Field-Programmable Gate Arrays (FPGAs), you can design such a machine with your mouse and keyboard. When you deploy it to the FPGA, it immediately takes on the behavior that you defined. Want to create something that behaves like a display driver integrated circuit? How about a CPU with an instruction set you dreamed up? Or your very own Bitcoin miner You can do all this with FPGAs. Because you're not writing programs--rather, you're designing a chip whose sole purpose is to do what you tell it--it's faster than anything you can do in code. With Make: FPGAs, you'll learn how to break down problems into something that can be solved on an FPGA, design the logic that will run on your FPGA, and hook up electronic components to create finished projects.

Book Information

Series: Make

Paperback: 256 pages

Publisher: Maker Media, Inc; 1 edition (March 18, 2016)

Language: English

ISBN-10: 145718785X

ISBN-13: 978-1457187858

Product Dimensions: 7.5 x 0.5 x 9.2 inches

Shipping Weight: 1.1 pounds (View shipping rates and policies)

Average Customer Review: 3.6 out of 5 stars   10 customer reviews

Best Sellers Rank: #160,310 in Books (See Top 100 in Books)   #14 inÃ Â Books > Engineering & Transportation > Engineering > Electrical & Electronics > Circuits > Logic   #36 inÃ Â Books > Engineering & Transportation > Engineering > Electrical & Electronics > Electronics > Semiconductors   #41 inÃ Â Books > Engineering & Transportation > Engineering > Electrical & Electronics > Circuits > Design

Customer Reviews

David Romano founded Tri-Tech Pathways Inc. to bring STEM education to students with a real-world industry perspective. He is a proven technical leader whose engineering career has spanned over twenty-five years and multiple High-Tech companies, including Raytheon, Motorola, HP, Intel and two start-up companies. He is also the co-author of multiple technology patents and

he is currently the president and CEO of Tri-Tech Pathways Inc. David is currently pursuing a doctorate degree in education as part of Intel's Tuition-for-Teaching grant. He holds a BS in Electrical Engineering and a MA in Theology. He is actively involved in STEM advisory and teaching roles. He is also active member of the International Society for Technology in Education (ISTE) and the Computer Science Teachers Association (CSTA).

Don't expect to learn how to write RTL hardware descriptions from this book. Do expect to have a complete guide to using the Xilinx ISE development environment and several different Xilinx FPGA based boards. All firmware that you build and load into hardware is already written. The book is simply a guide to compiling and loading this firmware into FPGA devices. As such it is complete and accurate. Easy to read and understand.

This book uses multiple boards for illustration, and does not go beyond the examples which come with the boards. I promptly return the book. If you want to learn Verilog, I would recommend "Advanced Chip Design, Practical Examples in Verilog" instead. Or even free articles like this one: [...]. For FPGA boards, this Xilinx board comes with necessary (English) docs: XC6SLX9 Starter Board,Xilinx Spartan 6 FPGA (find it on flea bay).

Love it

Not surprisingly, the FPGAs featured are not the latest and best, but the writing is good.

Very Pleased! Thank You!

This book has a very straightforward goal, and it meets that goal well. "Make: FPGAs" does not attempt to turn you into an expert in using software and Field Programmable Gate Arrays (FPGAs) to create special-function hardware. Instead, it presents "eight interesting FPGA projects that will help you develop some of the skills you will need to really begin exploring this exciting world of turning software into hardware through FPGA technology." David Romano, the author, shows several consumer-affordable ($29.95 to $200) FPGA development boards and then explains the design flow necessary to work with them effectively. The projects range from a simple frequency divider, to a Bitcoin miner, and a software-defined radio (SDR). The software for the projects, meanwhile, can be downloaded from GitHub. The book takes a clear, step-by-step approach to

each project and offers many illustrations, screenshots and photographs.(My thanks to O'Reilly Media for providing a review copy.)

Typical Make: publication. Overly general, a little good info. Left me wanting more depth.

This book is one of two beginner books that I would highly recommend. The other is Simon Monk'sÃ Â Programming FPGAs: Getting Started with Verilog. In fact, if you can afford it, I say by both. Both books are very similar. Both use Verilog as the HDL of choice. Both have projects that are worthy of following along with. Both Require minimal cost in a development board unless you want to complete all projects in the book(s). Both used Xilinx WEB ISE. Simon's book discusses the VGA video aspect of a couple of the FPGA development boards. Simon's book does a slightly better job of explaining step-by-step, David's book covers more in depth details of Xilinx Web ISE. David's book covers SDR in on project. I started out with the Zynq 7000 as a target FPGA platform but decided that the Spartan 3 and Spartan 6 would be good to understand as well due to the number of projects, books, and kits available. In the ended I purchased 6 different FPGA development boards ranging in price from about $30 to $900. Clearly I have the motivation to get totally absorbed in this subject and come from a PLA/PLD/Bitslice (geez did I just date myself?) background. Both books coupled with the Zynq BookÃ Â The Zynq Book: Embedded Processing with the Arm Cortex-A9 on the Xilinx Zynq-7000 All Programmable SocÃ Â and the associated Zynq lab bookÃ Â The Zynq Book Tutorials for Zybo and ZedBoardÃ Â are serving me well.

Make: FPGAs: Turning Software into Hardware with Eight Fun and Easy DIY Projects DIY: 365 Days of DIY: A Collection of DIY, DIY Household Hacks, DIY Cleaning and Organizing, DIY Projects, and More DIY Tips to Make Your Life Easier (With Over 45 DIY Christmas Gift Ideas) DIY Wood Pallet Projects: 23 Creative Wood Pallet Projects That Are Easy To Make And Sell! (DIY Household Hacks, DIY Projects, Woodworking) DIY For Men: Woodworking, Ham Radio, Blacksmithing, Homemade Weapons and Even DIY Internet Connection: (DIY Projects For Home, Woodworking, How To Build A Shed, Blacksmith, DIY Ideas, Natural Crafts) Make: Arduino Bots and Gadgets: Six Embedded Projects with Open Source Hardware and Software (Learning by Discovery) DIY Household Hacks for Beginners: DIY Hacks For Cleaning And Organizing, Increased Productivity, Declutter your Home (DIY Home Improvements, DIY Household ... And Organizing, Increase Productivity) The Hardware Hacker: Adventures in Making and Breaking Hardware Software Engineering: The Current Practice (Chapman & Hall/CRC Innovations in

Software Engineering and Software Development Series) Gifts In Jars: 88 Easy To Make DIY Gifts In Jars (Gifts in Mason Jars - Jar Gifts - Recipes - DIY Projects) Computer Organization and Design MIPS Edition, Fifth Edition: The Hardware/Software Interface (The Morgan Kaufmann Series in Computer Architecture and Design) Computer Organization and Design, Fourth Edition: The Hardware/Software Interface (The Morgan Kaufmann Series in Computer Architecture and Design) Specifying Systems: The TLA+ Language and Tools for Hardware and Software Engineers Getting Started with 3D Printing: A Hands-on Guide to the Hardware, Software, and Services Behind the New Manufacturing Revolution The Architecture of Computer Hardware, Systems Software, and Networking: An Information Technology Approach The HCS12 / 9S12: An Introduction to Software and Hardware Interfacing IEC 61511-1 Ed. 1.0 b:2003, Functional safety - Safety instrumented systems for the process industry sector - Part 1: Framework, definitions, system, hardware and software requirements AVR Programming: Learning to Write Software for Hardware The Architecture of Computer Hardware, Systems Software, & Networking: An Information Technology Approach PIC Microcontroller: An Introduction to Software & Hardware Interfacing Holiday Crafts: Easy and Fun, DIY Gifts and DÃƒÂ©cor Ideas for Christmas (Holidays & DIY Gifts)