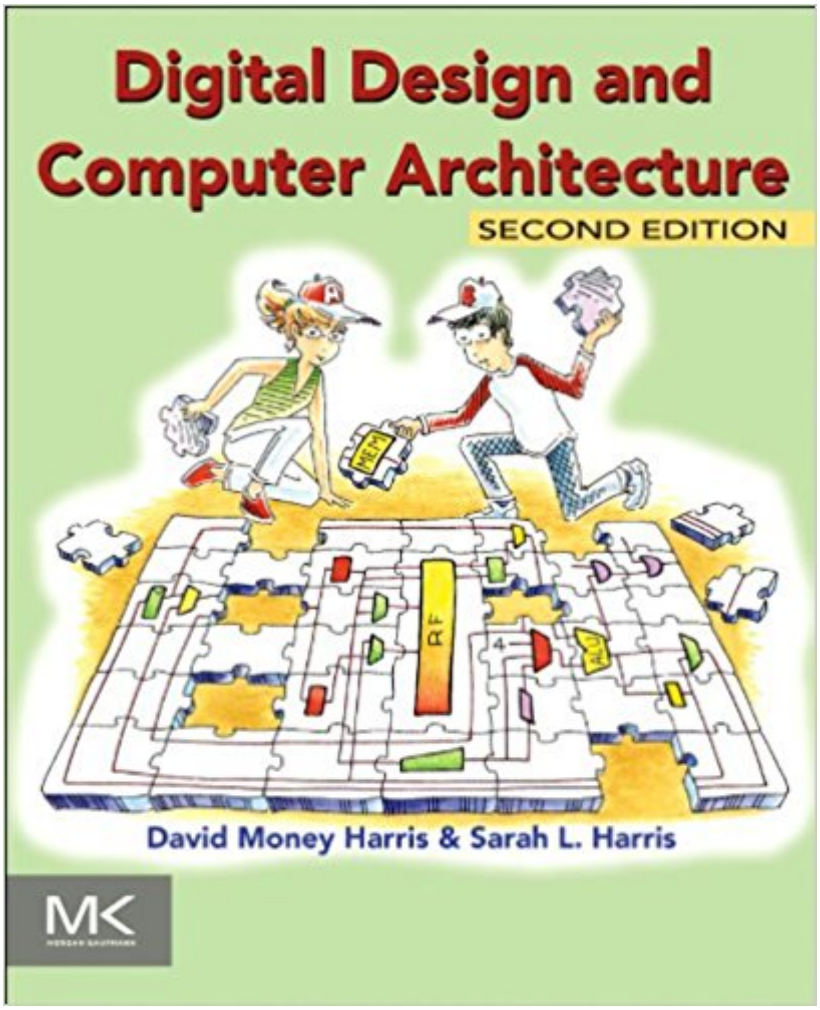
Digital Design And Computer Architecture



Synopsis

Digital Design and Computer Architecture takes a unique and modern approach to digital design. Beginning with digital logic gates and progressing to the design of combinational and sequential circuits, Harris and Harris use these fundamental building blocks as the basis for what follows: the design of an actual MIPS processor. SystemVerilog and VHDL are integrated throughout the text in examples illustrating the methods and techniques for CAD-based circuit design. By the end of this book, readers will be able to build their own microprocessor and will have a top-to-bottom understanding of how it works. Harris and Harris have combined an engaging and humorous writing style with an updated and hands-on approach to digital design. This second edition has been updated with new content on I/O systems in the context of general purpose processors found in a PC as well as microcontrollers found almost everywhere. The new edition provides practical examples of how to interface with peripherals using RS232, SPI, motor control, interrupts, wireless, and analog-to-digital conversion. High-level descriptions of I/O interfaces found in PCs include USB, SDRAM, WiFi, PCI Express, and others. In addition to expanded and updated material throughout, SystemVerilog is now featured in the programming and code examples (replacing Verilog), alongside VHDL. This new edition also provides additional exercises and a new appendix on C programming to strengthen the connection between programming and processor architecture. SECOND Edition Features Covers the fundamentals of digital logic design and reinforces logic concepts through the design of a MIPS microprocessor.Features side-by-side examples of the two most prominent Hardware Description Languages (HDLs)Ã¢â ¬â •SystemVerilog and VHDLÃ¢â ¬â •which illustrate and compare the ways each can be used in the design of digital systems.Includes examples throughout the text that enhance the readerÃ¢â ¬â„¢s understanding and retention of key concepts and techniques.Companion Web site includes links to CAD tools for FPGA design from Altera and Mentor Graphics, lecture slides, laboratory projects, and solutions to exercises.

Book Information

File Size: 19879 KB

Print Length: 712 pages

Publisher: Morgan Kaufmann; 2 edition (August 24, 2012)

Publication Date: August 24, 2012

Sold by:Ã Â Digital Services LLC

Language: English

ASIN: B00HEHG7W2

Text-to-Speech: Not enabled

X-Ray: Not Enabled

Word Wise: Not Enabled

Lending: Not Enabled

Enhanced Typesetting: Not Enabled

Best Sellers Rank: #126,335 Paid in Kindle Store (See Top 100 Paid in Kindle Store) #16 inÃ Â Books > Computers & Technology > Hardware & DIY > Microprocessors & System Design > Microprocessor Design #24 inÃ Â Kindle Store > Kindle eBooks > Computers & Technology > Computer Science > Computer Engineering #25 inÃ Â Kindle Store > Kindle eBooks > Engineering & Transportation > Engineering > Electrical & Electronics > Circuits

Customer Reviews

I bought this book after I was 6 chapters deep into Fundamentals of Logic Design 7th Ed. This book is the one you want to learn from. It explains things concisely, provides motivation and distinction between concepts, while remaining making you laugh. It will not provide you with everything you will need for an introductory course in Digital Logic Design, but I firmly believe that this book, along with Wikipedia (or the use of a search engine for diagrams) will do a much better job than that disgusting FLD 7th Ed.

I’m a veteran software engineer, but new to FPGAs. I am not an electrical engineer, and only had a little exposure to digital logic in college.This book is a great introduction to FPGAs and HDL. The writing style and comic illustrations make it very approachable - indeed, fun to read. The implementation of a MIPS processor seems MUCH more thorough than I’ve seen elsewhere.I’m still working through it, and still struggling with some details. The difference between blocking and non-blocking assignment wasn’t made as clear as I’d like. I found this was better addressed in Pong Chu’s FPGA Prototyping By Verilog Examples. I recommend that book as well; the two are very complimentary.

This is one of very few (arguably the only one) texts that combines and integrates digital design with actual architecture-- high and detail level. For the new (2nd) 2013 edition, Harris and Harris still teach simpler/ elegant systems that beginning Engineers and hobbyists love like MIPS and PIC 32, however they also add very recent and modern design and implementation solutions including

parallel and multicore processors, the x86, multithreading, out of order and superscalar operations and branch prediction, to name a few. These topics are not only state of the art, but normally covered in grad rather than undergrad courses.The thorny issues of parallel programming start at the assembly level, and it is astonishing and refreshing that these authors integrate methods as high level as embedded C and as basic as the digital circuits that implement assembly, and then relate them to considerations like temperature, memory, component sharing of workloads (the GPU often doubles as a CAS implementer or APU in these days where "math coprocessors" have been eliminated), etc.Every Engineer and hobbyist knows that getting a serious shot at a patent means implementation beyond simulation. That is where this new edition really shines. Other texts are out of date in a few months-- Harris and Harris give web and manufacturer resources that are available NOW (we checked), from design to finished boards. The authors also assume that after you spent your entire budget on this book you will appreciate cheap, open source solutions to getting to that million dollar patent. They don’t disappoint-- the "lab" includes cheapware and freeware in the form of IDEs/SDKs like Quartus II, MPLAB and Synplify, then take your favorite HDL (Verilog OR VHDL) and move from IDE output to code. Finally, the authors give altera alternatives in boards like the DE2 that are specifically designed to execute educational, developmental and student code-- as well as hobbyists!A REALLY cool feature if you’re getting into this as a career-- each chapter has sample interview questions for your next job. Like good programming books, the authors CARE that you get that job and include examples of what you’ll be asked, with great answers on their support websites.All in all, a GREAT update to their first trend setting text, and a hands on manual on "how to" build your own chipset. If you’re an OOP person you might be shocked that they cover C so much, but you’ve got to realize that "high level" at the circuit to assembly level is STILL C, and not so much Ada, Python, C# or Java (yet). Some other reviews around the web and in previous editions zinged them about this, but those reviewers aren’t in the real world-- even for the most modern 2013 luxury autos with 60+ embedded chips, when designers go beyond assembly, they still default to C. Just because it’s not OOP doesn’t mean it’s dead! If your own design prefers Python, or you’re a JAVA junkie, fret not-- there are plenty of libraries that will handshake with assembly since embedded is the wave of the future, and this text is just as relevant. Eiffel even has a plug in that you can run on Visual Studio, and "lunch" off of your C# SDK to debug a second language-- although, granted, they are both OOP.NOTE FOR EDUCATORS: If you’re a Junior College ID or exec/ dept. head, you might consider using this book as the basis for a year long course on circuit design to either prepare your grads for an AS/AA in electronics, or as a step to the EE. Once the grad gets into the real world of multi core, they will quickly find that "it’s about the

memory, stupid" that causes most performance challenges-- on board being heat and size costly, off board being time and speed costly, with cpu "work arounds" suprisingly more common than memory innovations-- a PERFECT field for that new patent.Many colleges are getting into "game programming" curricula because they offer an applied exposure to math, OOP, etc. This book gives you a non-herd alternative for your school-- with labs that ROCK. I'm an ID at ClassPros, and the schools at which we set up circuit design courses have even used the strategy to partner with name brand 4 year colleges in continuing on to the EE for the brightest students. GET THIS BOOK, and then think about how magnificently it would fit in such a curriculum!5 Stars-- a great start to getting that award-winning, financially rewarding patent on your new chip design, OR introducing a sim lab oriented, fun-project, high STEM curriculum item to your school-- go for it!

I used this book as a supplement for my computer architecture class. It does a great job of walking the student through everything from the basic construction of a transistor, to a complete microarchitecture, to programming.The writing style is precise, the knowledge contained in the book is thorough, and the exercises and examples are challenging. This is a beast of a textbook.Comparing it to the main textbook we used in class (Lobur, 2002), I think the writing from Harris & Harris is better and the examples are clearer. They also cover the material differently. Harris & Harris spends a lot more time on the physical aspects of digital circuit design, while Lobur spends more time on various other higher-level concepts such as compressing data for transmission, magnetic data storage, etc.

I took a digital design class and hated the book. The material is dry in nature, so there's not much you can do to make an enticing design/architecture book. I'm in computer architecture now, and my school is using this book for the first time. I freaking love it. Something about it makes me want to read the whole thing. The art really takes away any intimidation, and the lessons are very concise. Some might feel that there is not enough info in here, but I say "that's what teachers are for!" Anyways, water spilled on my copy and I almost cried. I think I'm going to keep it forever.

I find this book teaches the subject very well. Viewing on a Kindle is not easy as the lessons taught discuss diagrams and graphics. the graphics and diagrams don't display on the same page, doesn't matter if I view on a computer, iPad or Kindle. To fully grasp the concepts one must view the graphics on another device such as a computer or iPad while reading on another device. I am glad the publishers allowed us to have copies on multiple devices. After being in my class for two-weeks,

I am ordering a physical book. I am glad I have the Kindle version as it allows searching, just wish they included community notes!

Covers plenty of material, but the trade-off is that the early chapters on digital logic lack depth. I used another book -- Fundamentals of Logic Design -- to fill in some gaps. Nevertheless, the solutions PDF for this book, available for free, is very extensive and extremely helpful.

College book, however very interesting on its own. If you're interested in this, look up various courses on line from universities to practice.

Series in Computer Architecture and Design) See MIPS Run, Second Edition (The Morgan Kaufmann Series in Computer Architecture and Design) Digital Integrated Circuit Design (The Oxford Series in Electrical and Computer Engineering)