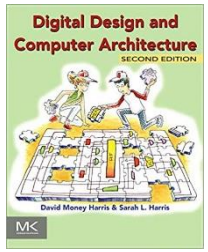
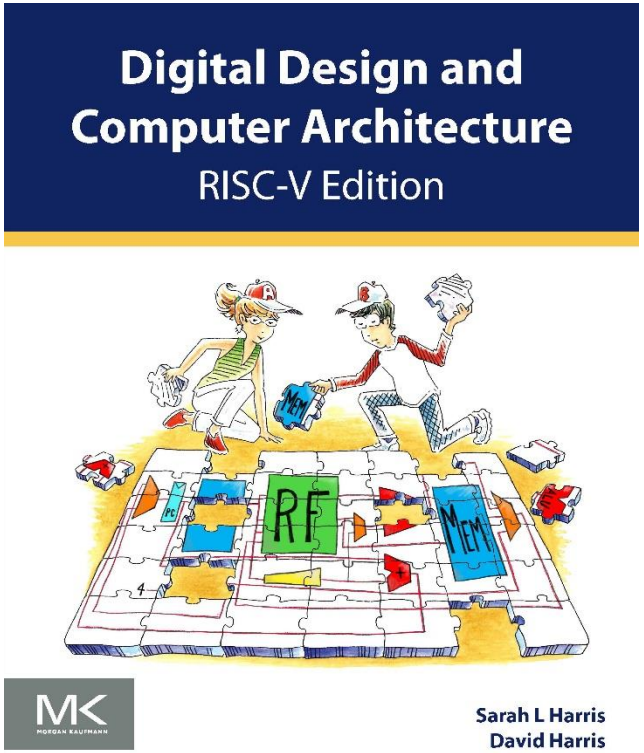
Digital Design & RISC-V Computer Architecture Textbook

UNLV
HOWARD R. HUGHES COLLEGE OF ENGINEERING
Sarah L. Harris
WCAE '21
David Harris
HARVEY MUDD COLLEGE

Overview

- We've adapted our popular Digital Design & Computer Architecture textbook to the RISC-V Architecture
- Prior Books:
  - *Digital Design and Computer Architecture*, 2nd edition (2012)
  - *Digital Design and Computer Architecture: ARM® Edition* (2015)

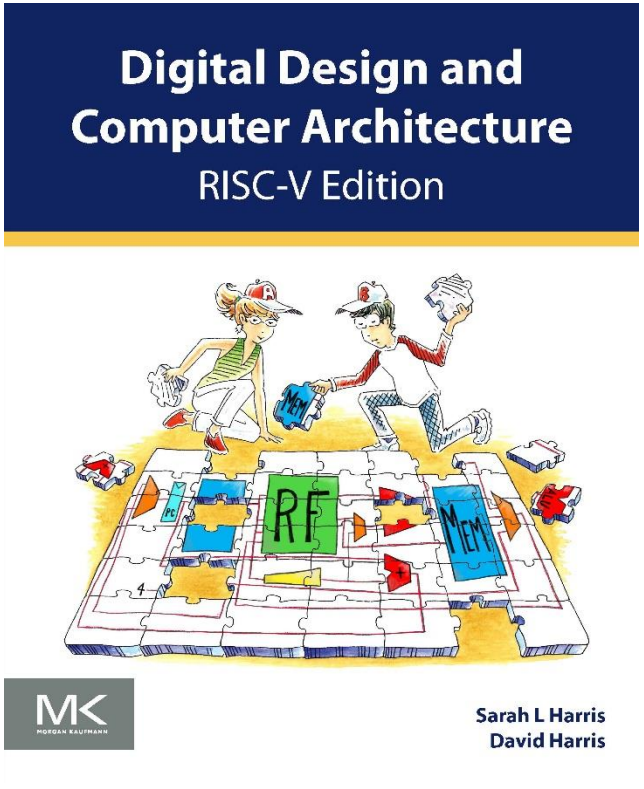




Expected Publication:
Aug 15, 2021

Book Statistics & MOOC

- **Prior textbooks:**
  - Used internationally
  - In US in 2020, used by 10,000 university students
  - Available in 7 languages (English, Chinese, Japanese, Korean, Spanish, Russian, Portuguese)
- **Material may be taught as:**
  - 1-semester course (e.g., at HMC)
  - 2-semester course (e.g., at UNLV)
- **Developed MOOC on EdX:**
  - Digital Design HarveyMuddX
  - Computer Architecture HarveyMuddX



Expected Publication:
Aug 15, 2021

Background

- **Why RISC-V Architecture?**
  - Open-source
  - Base ISAs: RV32I/E, RV64I, RV128I
  - Extensible
  - Increasingly adopted commercially
  - Supports a range of processors: embedded to high-performance
  - Forward-thinking: for example, supporting wider data
  - Offers commercial, inexpensive development boards

Architecture Comparisons

- **Compared to MIPS, RISC-V:**
  - Maintains **similar instruction formats** (i.e., R-type, I-type, etc.) and **register/instruction names** (i.e., `addi`, `lw`, `s0`, `s1`, `a0`, `etc`.)
  - Gets rid of idiosyncrasies such as the **branch delay slot**, **branch offsets relative to PC+4** instead of PC, and **inconsistent register locations in instruction encodings**
  - Has more **complex immediate encodings** to minimize hardware
  - Is **commercially viable**, whereas MIPS is seldom used in current products
- **Compared to ARM, RISC-V:**
  - Does **not** include **conditional execution** or **complex indexing modes**, which result in added hardware complexity
  - Does offer **16-bit** (compressed) instructions like ARM's 16-bit Thumb instruction set, which are well-suited for low-power, embedded applications
  - **Does not require licensing** – whereas ARM's licensing can be prohibitive and costly

Chapters

Digital Design

- **1:** Fundamentals: Numbers, Logic gates, Transistors, Power
- **2:** Combinational Logic: SOP, POS, Boolean algebra, K-maps, timing
- **3:** Sequential Logic: Latches, Flip-flops, FSMs, timing, parallelism, pipelining
- **4:** Hardware Description Languages (HDLs): SystemVerilog & VHDL
- **5:** Digital Building Blocks: Arithmetic circuits, number systems, memory, logic arrays

Computer Architecture

- **6:** RISC-V Architecture: RV32I, also RVF/D, RVC
- **7:** RISC-V Microarchitecture: Single-cycle, multicycle, & pipelined processors, performance
- **8:** Memory systems: Caches, virtual memory
- **9:** I/O Systems: Memory-mapped I/O, serial interfaces, interrupts, motors, etc.

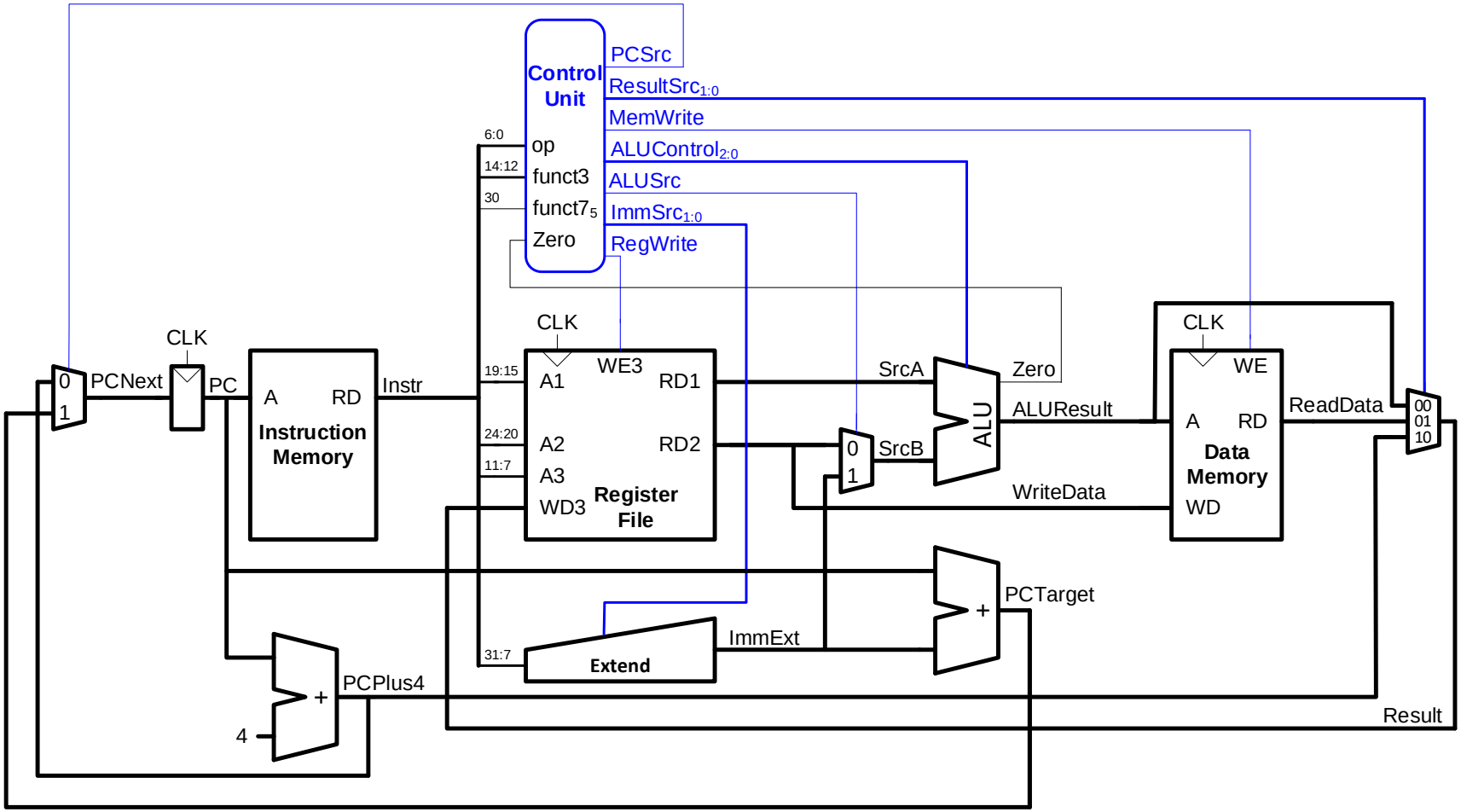
- **Appendices A-C:** Digital System Implementation, RISC-V Instruction Summary, C Programming

RISC-V Processors

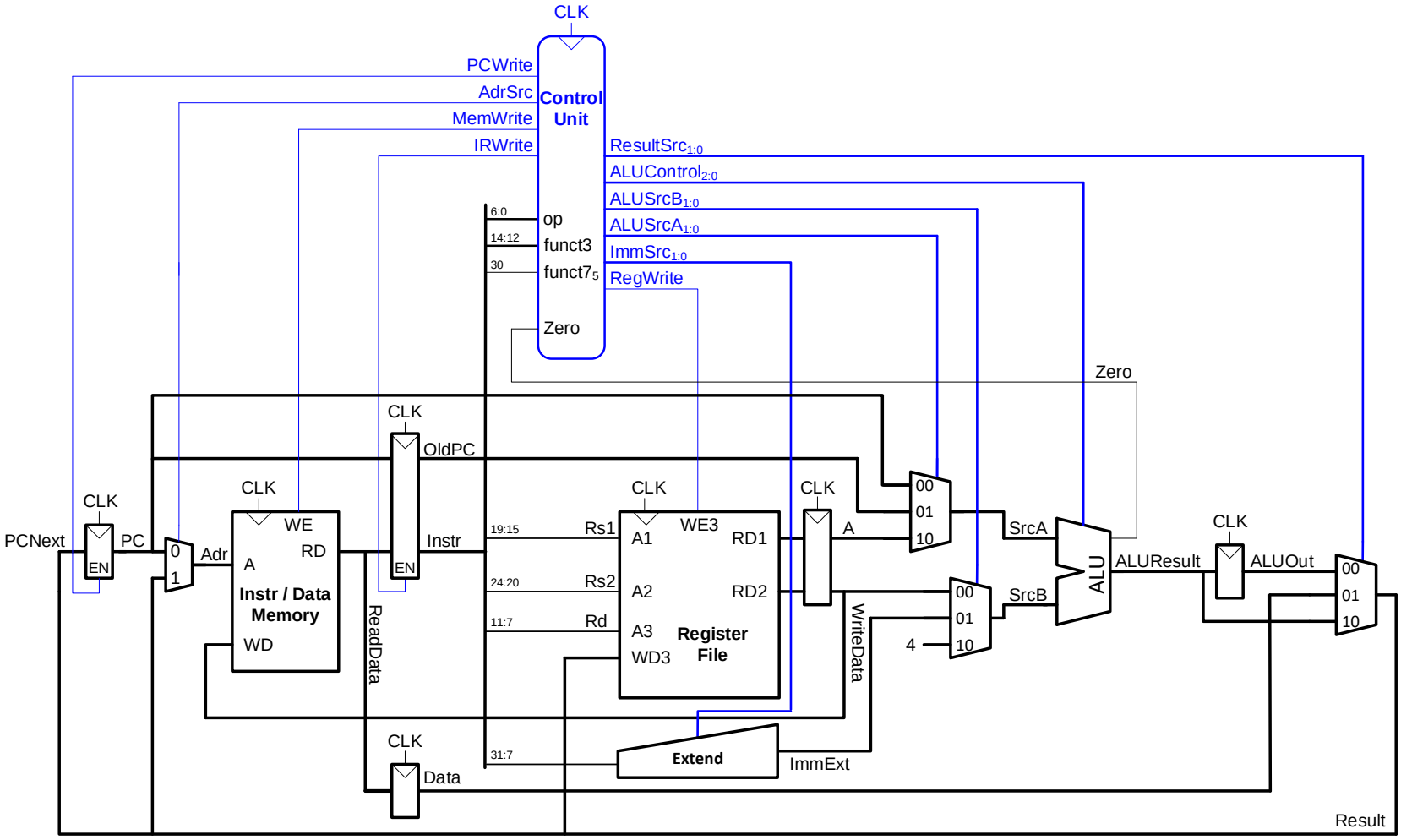
- **Single-Cycle**
- **Multicycle**
- **Pipelined**

- **Subset of instructions:**
  - `add`, `sub`, `and`, `or`, `slt`, `lw`, `sw`, `beq`, `jal`, I-type ALU

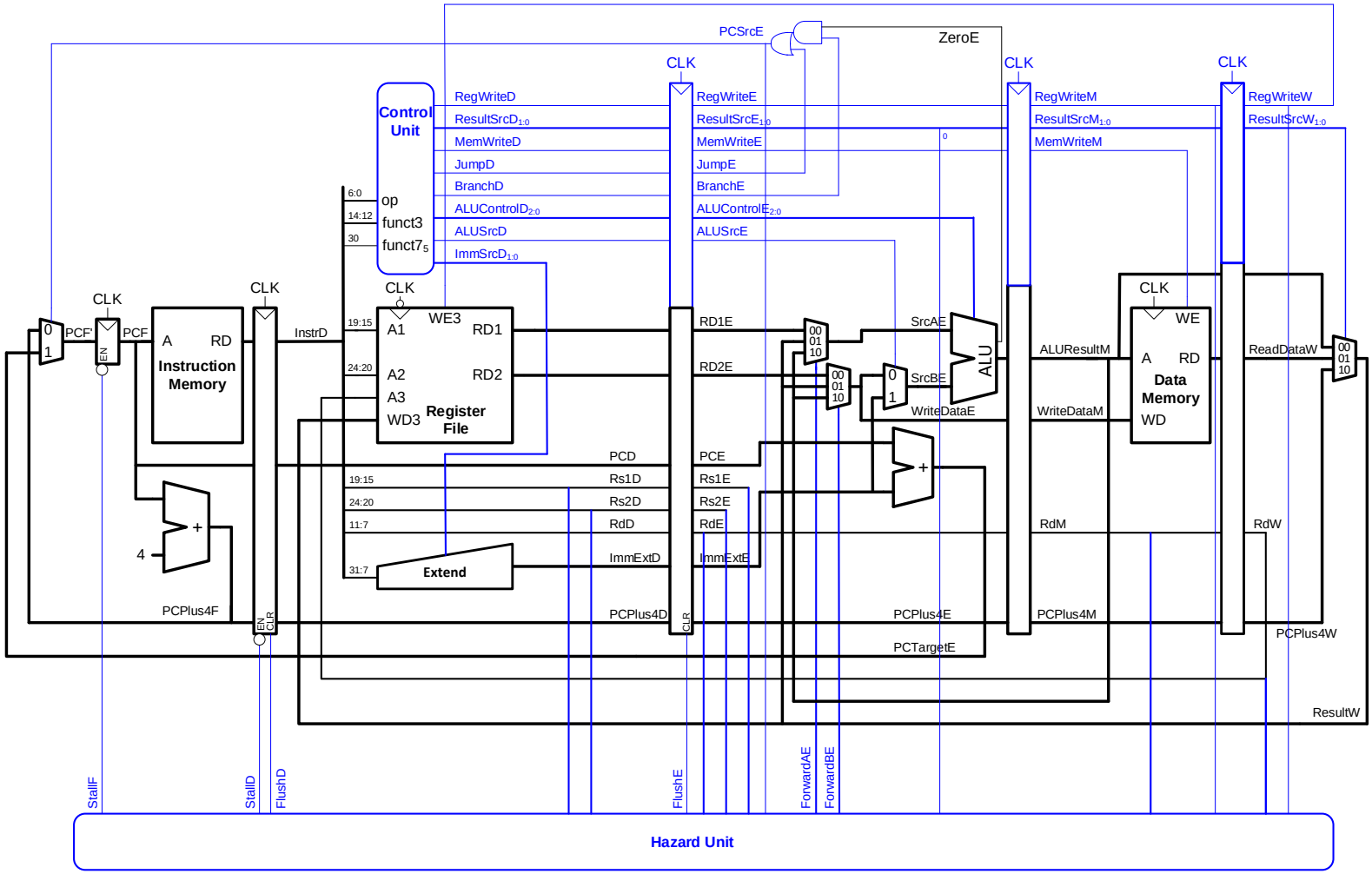
Single-Cycle RISC-V Processor



Multicycle RISC-V Processor



Pipelined RISC-V Processor



Labs: Software & Hardware

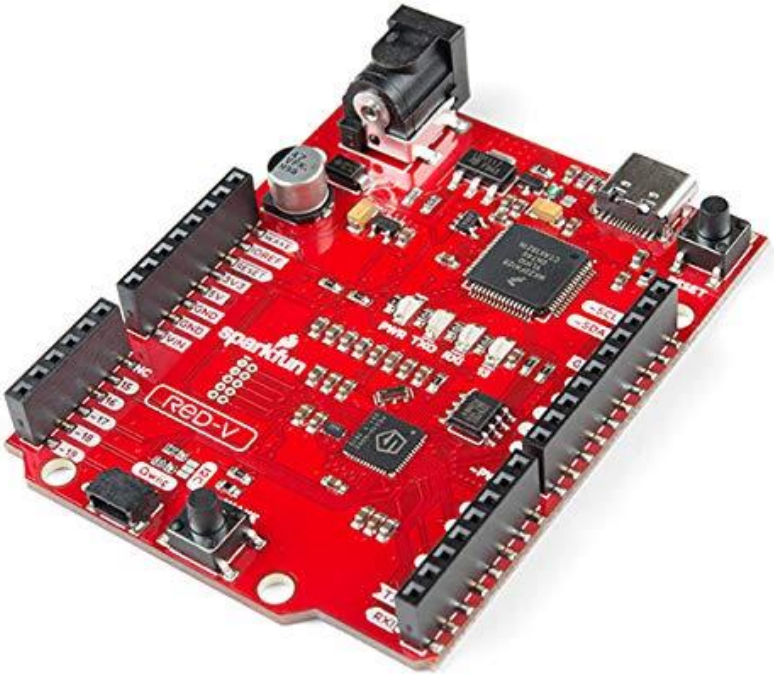
Sofware	Link
Quartus Lite / Web Edition	https://fpgasoftware.intel.com
ModelSim Intel FPGA/Student Edition	https://fpgasoftware.intel.com/?product=modelsim_ae#tabs-2
Visual Studio Code (VS Code)	https://code.visualstudio.com/download
PlatformIO	Extension within VS Code

* Software is free

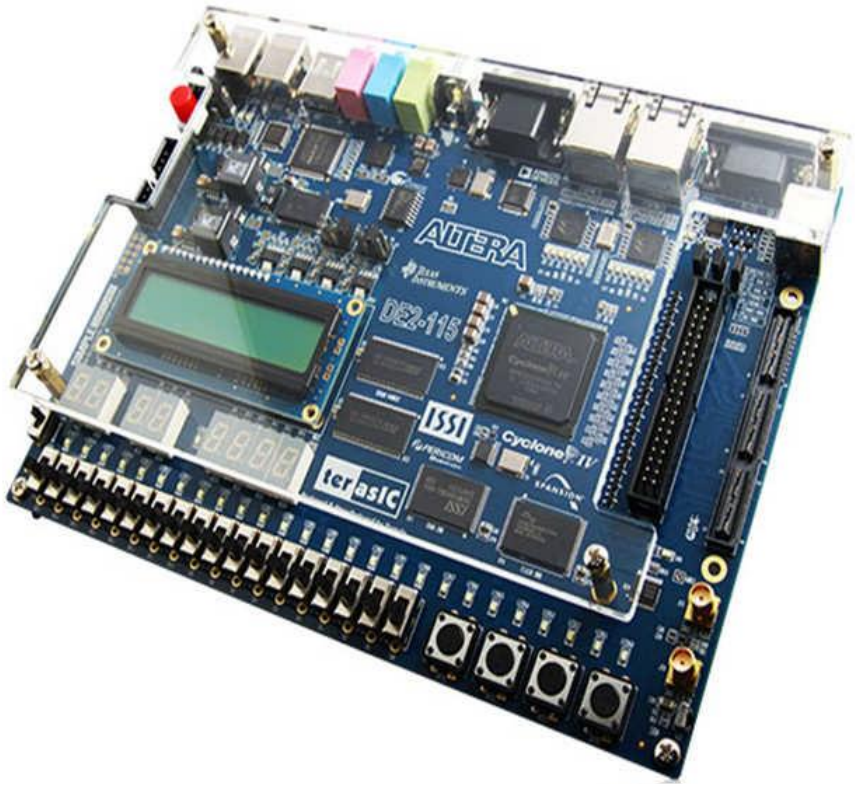
Software	Link	Cost
DE2-115 Board	http://de2-115.terasic.com	$309
RED-V RedBoard	https://www.sparkfun.com/products/15594	$40

* Hardware is optional, but recommended

Lab Hardware



RED-V RedBoard
(contains SiFive's FE310-G002 SoC)



Intel-Altera DE2-115 board
(contains Cyclone IVE FPGA)

Labs

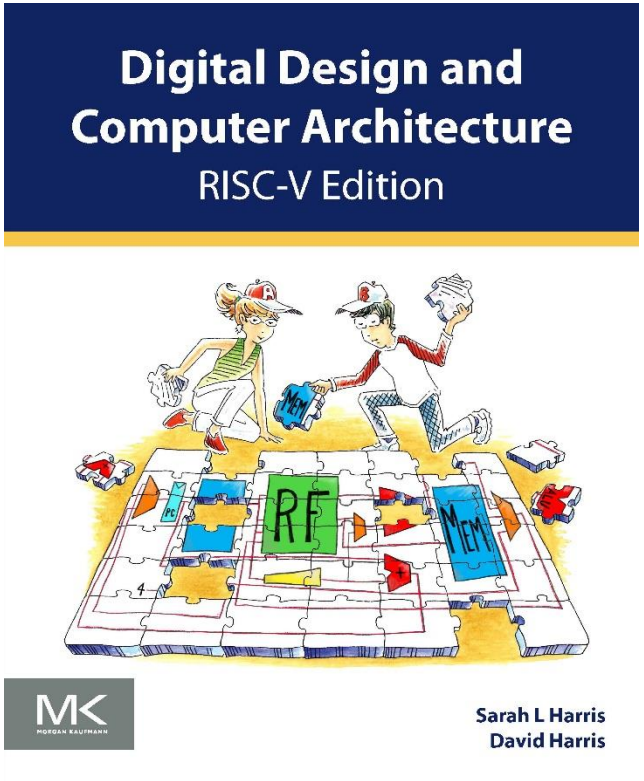
#	Topic	Design Method
1	1-Bit Full Adder	Schematic
2	7-Segment Display	Schematic
3	Finite State Machine: Adventure Game	Schematic
4	Finite State Machine: Thunderbird Turn Signal	SystemVerilog
5	32-Bit ALU and Testbench	SystemVerilog
6	Matrix Multiplication	C Programming
7	Simon Says Game with LEDs & Switches	C Programming
8	Single-Cycle Processor	SystemVerilog
9	Multicycle Datapath	SystemVerilog
10	Multicycle Control	SystemVerilog

Conclusions

- We have written a **textbook** and companion material that starts with digital design and builds up to designing several RISC-V processors.
- We have found that:
  - Teaching digital design and computer architecture **together** enhances understanding of both.
  - Teaching the **RISC-V processor** is both easy to understand and commercially relevant.
  - Teaching processors from **top to bottom** (from transistors up to the software program running on them) empowers students.

Thank You

- We'd like to acknowledge the contributions of:
  - Josh Brake
  - Numerous reviewers including Dani Chaver Martinez, Roy Kravitz, Angel Solis, Andrew Waterman, and others



Expected Publication:
Aug 15, 2021