THE IMAGINATION UNIVERSITY PROGRAMME

RVfpga-SoC
Getting Started Guide

Acknowledgments

AUTHORS
Prof. Sarah Harris
Prof. Daniel Chaver
Zubair Kakakhel
M. Hamza Liaqat

ADVISER
Prof. David Patterson

CONTRIBUTORS
Robert Owen
Olof Kindgren
Prof. Luis Piñuel
Ivan Kravets
Valerii Koval
Ted Marena
Prof. Roy Kravitz

ASSOCIATES
Prof. Daniel León
Prof. José Ignacio Gómez
Prof. Katzalin Olcoz
Prof. Alberto del Barrio
Prof. Fernando Castro
Prof. Manuel Prieto
Prof. Ataur Patwary
Prof. Christian Tenllado
Prof. Francisco Tirado

Prof. Román Hermida
Cathal McCabe
Dan Hugo
Braden Harwood
Prof. David Burnett
Gage Elerding
Prof. Brian Cruickshank
Deepen Parmar
Thong Doan

Oliver Rew
Niko Nikolay
Guanyang He
Chuck Faber
Brett Thornhill
Daniel Hale
Joshua Blazek
Tiffani Shilts

Sponsors and Supporters

AUTHORS
- Prof. Sarah Harris (https://www.linkedin.com/in/sarah-harris-12720697/)
- Prof. Daniel Chaver (https://www.linkedin.com/in/daniel-chaver-a5056a156/)
- Zubair Kakakhel (https://www.linkedin.com/in/zubairlk/)
- M. Hamza Liaqat (https://www.linkedin.com/in/muhammad-hamza-liaqat-ab73a0195/)

ADVISER
- Prof. David Patterson (https://www.linkedin.com/in/dave-patterson-408225/)

CONTRIBUTORS
- Robert Owen (https://www.linkedin.com/in/robert-owen-4335931/)
- Olof Kindgren (https://www.linkedin.com/in/olofkindgren/)
- Prof. Luis Piñuel (https://www.linkedin.com/in/lpinuel/)
- Ivan Kravets (https://www.linkedin.com/in/ivankravets/)
- Valerii Koval (https://www.linkedin.com/in/valeros/)
- Ted Marena (https://www.linkedin.com/in/tedmarena/)
- Prof. Roy Kravitz (https://www.linkedin.com/in/roy-kravitz-4725963/)

ASSOCIATES
- Prof. José Ignacio Gómez (https://www.linkedin.com/in/jos%C3%A9-ignacio-gomez-182b981/)
- Prof. Christian Tenllado (https://www.linkedin.com/in/christian-tenllado-31578659/)
- Prof. Daniel León (www.linkedin.com/in/danileon-ufv)
- Prof. Katzalin Olcoz (https://www.linkedin.com/in/katzalin-olcoz-herrero-5724b0200/)
- Prof. Alberto del Barrio (https://www.linkedin.com/in/alberto-antonio-del-barrio-garc%C3%ADa-1a85586a/)
- Prof. Fernando Castro (https://www.linkedin.com/in/fernando-castro-5993103a/)
- Prof. Manuel Prieto (https://www.linkedin.com/in/manuel-prieto-matias-02470b8b/)
- Prof. Francisco Tirado (https://www.linkedin.com/in/francisco-tirado-fern%C3%A1ndez-40a45570/)
- Prof. Román Hermida (https://www.linkedin.com/in/roman-hermida-correa-a4175645/)
- Cathal McCabe (https://www.linkedin.com/in/cathalmccabe/)
- Dan Hugo (https://www.linkedin.com/in/danhugo/)
- Braden Harwood (https://www.linkedin.com/in/braden-harwood/)
- David Burnett (https://www.linkedin.com/in/david-burnett-3b03778/)
- Gage Elerding (https://www.linkedin.com/in/gage-elerding-052b16106/)
- Brian Cruickshank (https://www.linkedin.com/in/bcruiksh/)
- Deepen Parmar (https://www.linkedin.com/in/deepen-parmar/)
- Thong Doan (https://www.linkedin.com/in/thong-doan/)
- Oliver Rew (https://www.linkedin.com/in/oliver-rew/)
- Niko Nikolay (https://www.linkedin.com/in/roy-kravitz-4725963/)
- Guanyang He (https://www.linkedin.com/in/guanyang-he-5775ba109/)
- Prof. Ataur Patwary (https://www.linkedin.com/in/ataurpatwary/)
- Chuck Faber (https://www.linkedin.com/in/chuck-faber-3b502339/)
- Brett Thornhill (https://www.linkedin.com/in/brett-d-thornhill/)

- Daniel Hale (https://www.linkedin.com/in/daniel-lee-hale/)
- Joshua Blazek (https://www.linkedin.com/in/joshuablazek/)
- Tiffani Shilts (https://linkedin.com/in/tiffanishilts/)

Table 1. RVfpga Terms

Name	Description
Courses	
RVfpga	A course that shows how to use RVfpgaNexys and RVfpgaSim, RISC-V system-on-chips (SoCs), to run programs and extend the system by adding peripherals (RVfpga Labs 1-10), and explore the core and memory system by running simulations, measuring performance, adding instructions, and modifying the memory system (RVfpga Labs 11-20). Throughout the course, users are also shown how to use the RISC-V toolchain (compilers and debuggers) and simulators, the Verilator HDL simulator, and Western Digital's Whisper instruction set simulator (ISS).
RVfpga-SoC	A course that shows how to build a subset of SweRVolfX SoC from scratch using building blocks such as the SweRV core, memories, and peripherals. The course also shows how to load the Zephyr real-time operating system (RTOS) onto SweRVolf and run programs including Tensorflow Lite's hello world example on top of the operating system.
Cores and SoCs	
SweRV EH1 Core	Open-source commercial RISC-V core developed by Western Digital (https://github.com/chipsalliance/Cores-SweRV).
SweRV EH1 Core Complex	SweRV EH1 core with added memory (ICCM, DCCM, and instruction cache), programmable interrupt controller (PIC), bus interfaces, and debug unit (https://github.com/chipsalliance/Cores-SweRV).
SweRVolfX	The System on Chip that we use in the RVfpga course. It is an extension of SweRVolf. **SweRVolf** (https://github.com/chipsalliance/Cores-SweRVolf): An open-source SoC built around the SweRV EH1 Core Complex. It adds a boot ROM, UART interface, system controller, interconnect (AXI Interconnect, Wishbone Interconnect, and AXI-to-Wishbone bridge), and an SPI controller. **SweRVolfX**: It adds four new peripherals to SweRVolf: a GPIO, a PTC, an additional SPI, and a controller for the 8 Digit 7-Segment Displays.
RVfpgaNexys	The SweRVolfX SoC targeted to the Nexys A7 board and its peripherals. It adds a DDR2 interface, CDC (clock domain crossing) unit, BSCAN logic (for the JTAG interface), and clock generator. RVfpgaNexys is the same as SweRVolf Nexys (https://github.com/chipsalliance/Cores-SweRVolf), except that the latter is based on SweRVolf.
RVfpgaSim	The SweRVolfX SoC with a testbench wrapper and AXI memory intended for simulation. RVfpgaSim is the same as SweRVolf Sim (https://github.com/chipsalliance/Cores-SweRVolf), except that the latter is based on SweRVolf.

RVfpga-SoC Labs Overview

This RVfpga-SoC course shows how to build a RISC-V SoC from scratch using provided building blocks and a visual block-based design approach. The building blocks include the SweRV EH 1 CPU core, Interconnect, Boot-ROM, System controller, and GPIO controller. The SoC created by the user using the block design approach is a subset of SweRVolfX. Subsequent labs show how to run programs on the SoC, compare the block design SoC with SweRVolf made using FuseSoC, run Zephyr real-time operating system on SweRVolf, and then run a Tensorflow Lite Hello-World example on SweRVolf.

RVfpga-SoC labs have been built with the following platform:
- Operating System: Ubuntu 18.04 LTS
  - Labs 1 and 2 can be easily run on Windows 10. Labs 3, 4, and 5 use some packages that run in a Linux environment only. Windows 10 users can run the simulation parts of the labs using Windows Subsystem for Linux.
- Hardware Target (optional): Nexys A7-100T board (or Nexys 4 DDR board)
- Full system simulator: Verilator

Before starting RVfpga-SoC Labs, you must have already completed the RVfpga-SoC Installation Guide. The Installation Guide has been divided into instructions needed for each Lab. The structure of the Installation guide is as follows:
- **Installation for Lab 1:** Installation of Vivado 2019.2 Web Pack, Cable drivers, and Digilent board files.
- **Installation for Lab 2:** Installation of Visual Studio Code (VScode), PlatformIO, Verilator version 4.106, and GTKWave.
- **Installation for Lab 3:** Installation of FuseSoC, and OpenOCD.
- **Installation for Lab 4:** Installation of Zephyr dependencies, west, CMake, PuTTY, and Zephyr SDK version 0.12.4.

If you have already completed the RVfpga course, you will have already installed much of this software.

Make sure that you have copied the ***RVfpgaSoC*** folder that you downloaded from Imagination's University Programme to your machine. We will refer to the directory's absolute path to place folder RVfpgaSoC as [*RVfpgaSoCPath*]. Preferably place the **RVfpgaSoC** folder in your home directory. i.e : `/home/<username>/RVfpgaSoC`

The following labs are provided:
- **Lab 1**: Introduction to RVfpga-SoC
- **Lab 2**: Running Software on RVfpga-SoC
- **Lab 3**: Introduction to SweRVolf and FuseSoC
- **Lab 4**: Running Zephyr on SweRVolf
- **Lab 5**: Running Tensorflow Lite on SweRVolf

These labs show how to create an SoC from a core and other building blocks (Lab 1), how to target it to an FPGA and run programs on the newly created SoC (Labs 2), how to use a FuseSoC-based SoC (SweRVolf) for SweRV EH1 (Labs 3), how to add a real-time operating system (RTOS) to SweRVolf (Lab 4). and how to run Tensorflow Lite's Hello World example on SweRVolf (Lab 5).

The organization of the RVfpgaSoC/Labs/ folder is as follows:
- **LabInstructions:** Instructions for each lab.
- **LabProjects:** The folder where you will create projects.

- o Lab1: Directory for Lab 1 Vivado Project.
- o SweRVolf: Directory for Labs 3, 4 and 5
- **LabResources:** Resources to be used while completing the labs.
- **LabSolutions:** Solution of labs.
Instructors should remove this folder before distributing RVfpgaSoC to students.