FPGAs - Fabric, Design and Architecture

Nanditha Rao

Agenda

- Module 1
  - FPGA Introduction
  - Design of a 4-bit counter through Vivado on Basys3 FPGA
  - Use of Virtual Input/Output (VIO)
- Module 2
  - OpenFPGA Introduction (https://openfpga.readthedocs.io/en/master/overview/motivation/)
  - VTR Flow (https://docs.verilogtorouting.org/en/latest/quickstart/)
  - Demos:
    - VPR flow: tseng on Earch
    - VTR flow: 4 bit counter on Earch
    - Timing, area, power and post implementation simulation
    - Basys3 vs VTR results

Agenda

- Module 3:
  - Explain RISC-V RVMyth processor code
  - Run it till bitstream on Basys3
- Module 4:
  - Skywater OpenSource FPGA (SOFA): 4-bit counter on SOFA
  - Area, timing and post implementation
- Module 5:
  - Skywater OpenSource FPGA (SOFA): RVMyth on SOFA (https://github.com/lnis-uofu/SOFA)
  - Area, timing and post implementation

Collaterals

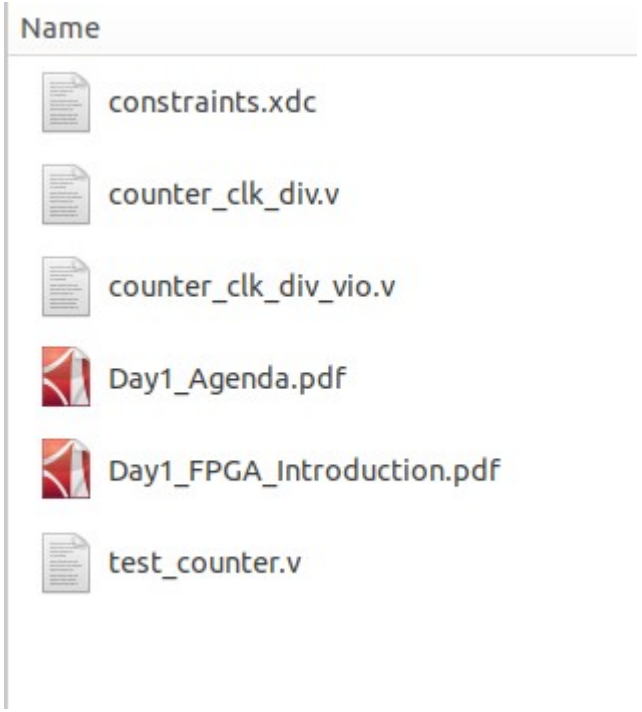
- Needed for the workshop
  - git clone https://github.com/nandithaec/fpga_workshop_collaterals.git

    Or
  - gh repo clone nandithaec/fpga_workshop_collaterals

Module 1: Counter in Vivado

Design of a 4-bit counter through Vivado on Basys3 FPGA: counter_clk_div.v and test_counter.v with xdc

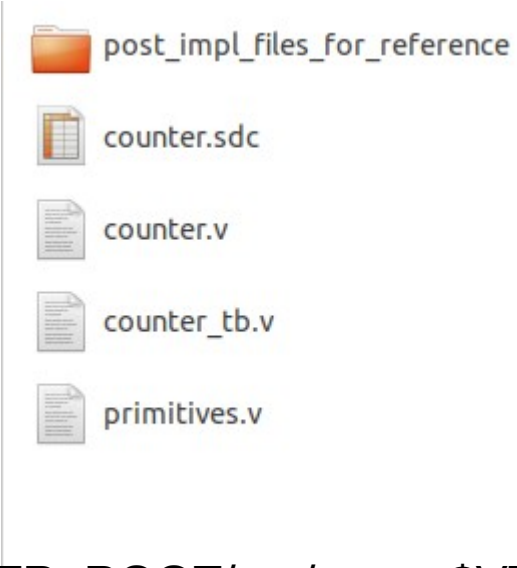
- Use of Virtual Input/Output (VIO): counter_clk_div_vio.v. You need to still create the VIO IP yourself in Vivado



The .bit can be posted on shared drive and we can help program

commands_for_workshop.txt has all necessary commands

Module 2: OpenFPGA and VTR

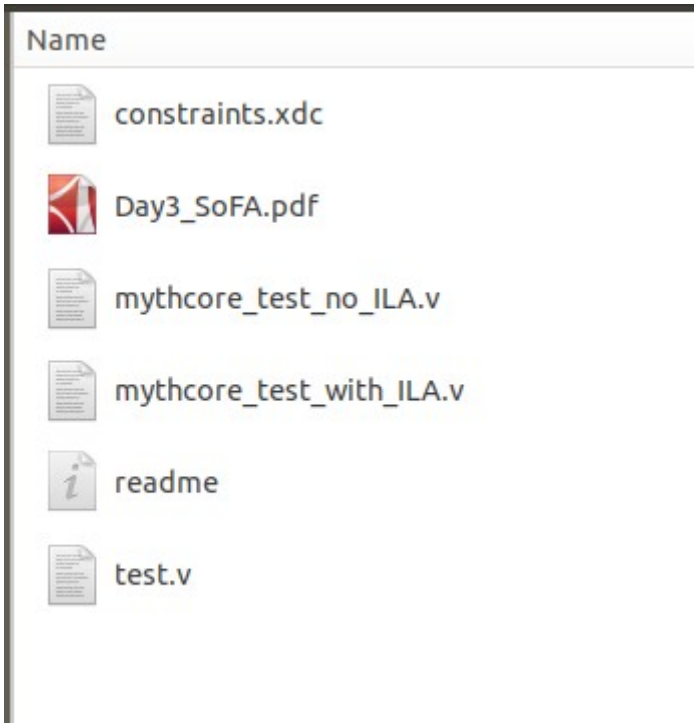


Use the files provided. The primitives.v that you have in your directory needs to be replaced with the one here
Place the sdc file in your folder and provide the path to it

$VTR_ROOT/vpr/vpr $VTR_ROOT/vtr_flow/arch/timing/EArch.xml < >/osfpga/vtr_work/quickstart/counter_earch/counter.pre-vpr.blif --route_chan_width 100 --sdc_file < >/osfpga/vtr_github/vtr-verilog-to-routing/doc/src/quickstart/counter.sdc

Module 3: RVMyth on Vivado

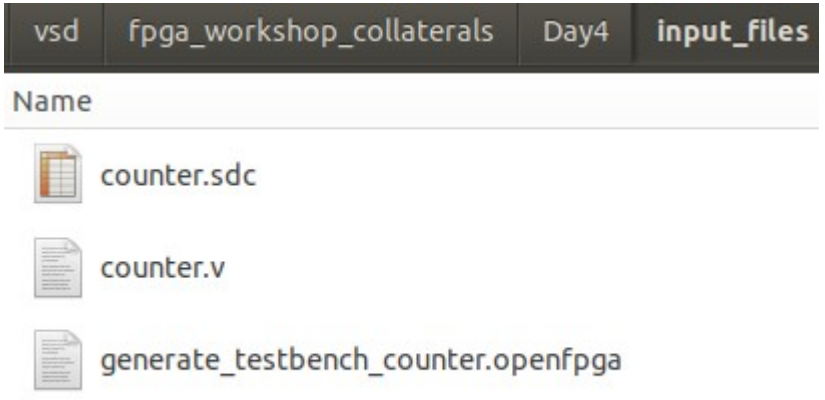
- Explain RISC-V RVMyth processor code
  - Run it till bitstream on Basys3



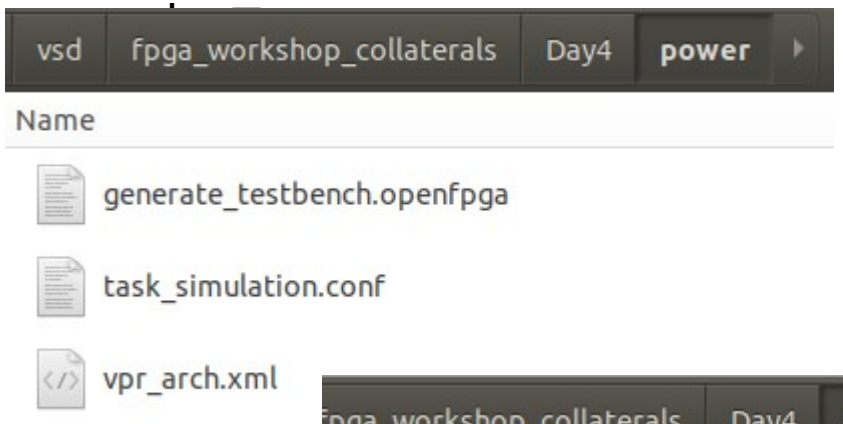
Use of ILA: mythcore_test_with_ILA. You need to still create the ILA IP yourself in Vivado

The .bit can be posted on shared drive and we can help program

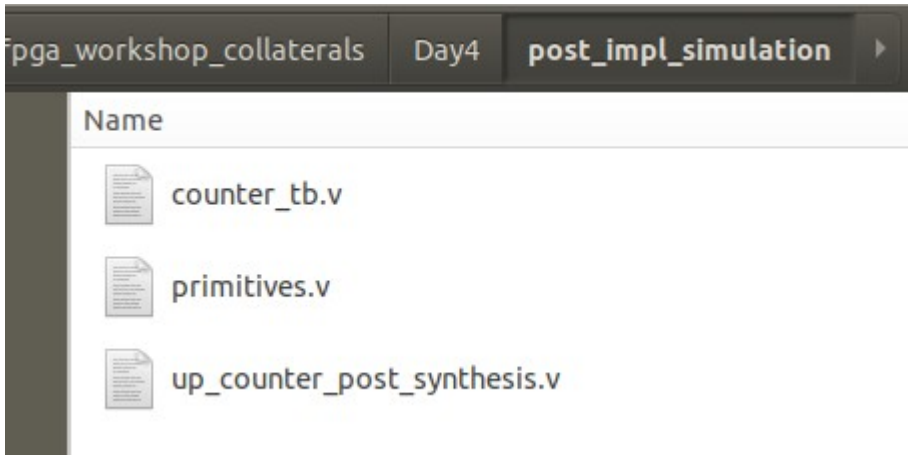
Module 4: counter on SOFA



Use the generate_testbench.openfpga provided here.
Create the sdc and change the path accordingly

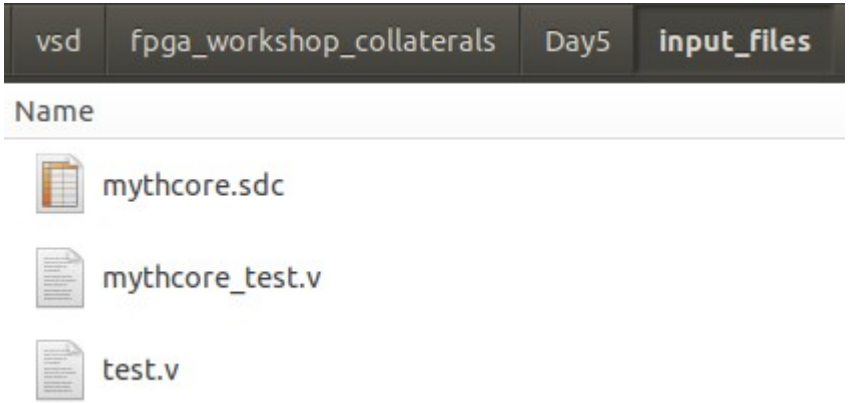


Use the conf, generate_testbench.openfpga and xml provided here. You should place them in your FPGA1212_QLSOFA_HD_task folder

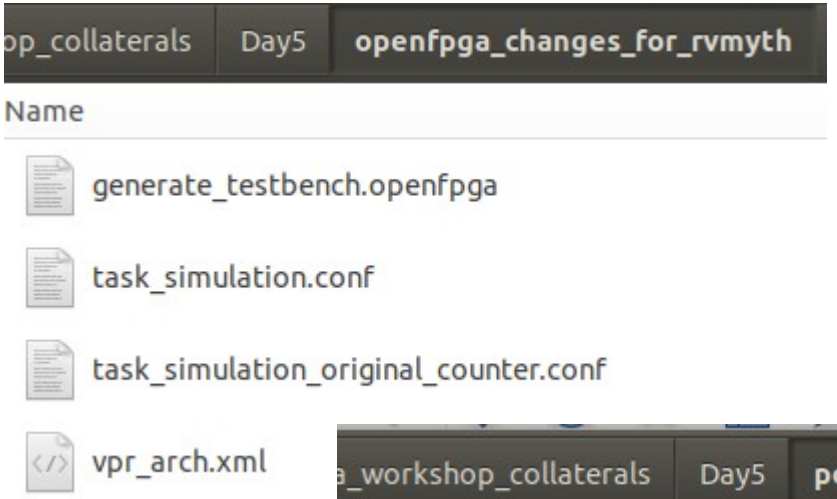


Use the primitives.v shown here

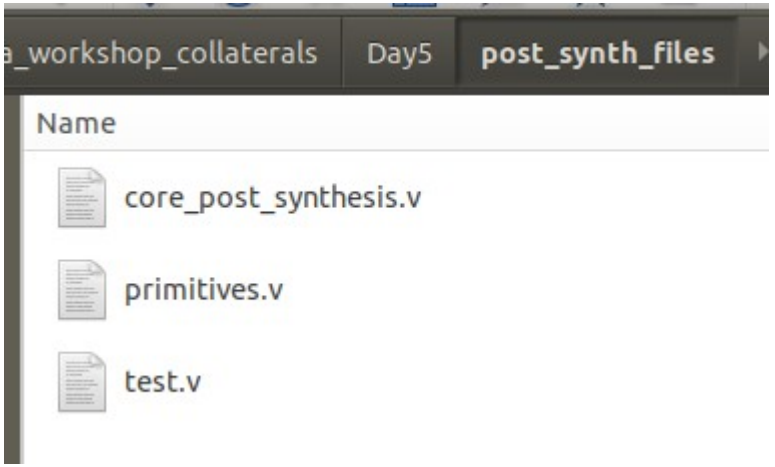
Module 5: RVMyth on SOFA



Use the verilog provided here



Use the conf, generate_testbench.openfpga and xml provided here. You should place them in your FPGA1212_QLSOFA_HD_task folder



Use the primitives.v shown here
The core verilog is for your reference

Acknowledgements

- Dr. Xifan Tang, OpenFPGA and Chief Engineer RapidSilicon