ECE 497: Special Project
Weekly Report
Week 04

Alexander Lukens Karl Hallsby

Illinois Institute of Technology

February 18, 2021

What We Did

- Attempted to flash default chip to Alex's FPGA.
- Successfully passed USB connection to FPGA through to VirtualBox VM, interfaced with FPGA in Vivado
- Conducted "Hello World" project on FPGA to ensure that bitstream was being sent to FPGA correctly.
- Used FPGA Prototyping Flow in Chipyard to generate a bitstream for the Arty FPGA board. Strangely, the default "example" project for the Arty did not pass all timing constraints.
  - Will require additional investigation
- When creating bitstream in Chipyard, Vivado runs several tests on the design and produces detailed reports in a Chipyard folder.

What We Learned

- The repository is incredibly complicated
- **VERY** deep directory nesting (Partly due to Scala/Java project directory conventions).
- Putting the generated chip on an FPGA seems to be much more difficult than originally thought.
- Generating a non-default chip can be very easy or very hard.
  - Some of the options that must be overridden to ensure a different chip is built and simulated/benchmarked are not easy to understand or find.

Next Steps

- Continue trying to write the default chip out to Alex's FPGA and test.
- Using Alex's discovery in Vivado, collect gate counts of components within the default chip.
- Practice generating other non-default chips to understand all the options used when generating a new chip.
- Hopefully, start defining a new, custom, chip using what we know, and building a *very* small proof-of-concept.

References

Alon Amid et al. "Chipyard: Integrated Design, Simulation, and Implementation Framework for Custom SoCs." In: *IEEE Micro* 40.4 (2020), pp. 10–21. ISSN: 1937-4143. DOI: 10.1109/MM.2020.2996616.