Rogues Gallery - Reconfigurable Computing - Spring 2025

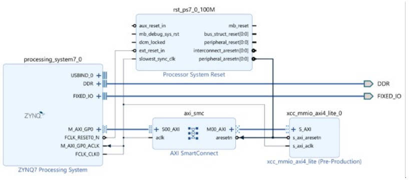
Andrew Burgess, Patricio Cortez, Theodore Halpern, Kent Hepfinger, Yuanda Liu, Max Zhu

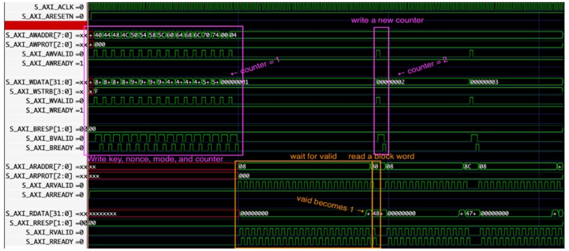
Semester Goals

- Verification of existing verilog implementations ChaCha20 and XChaCha20 algorithms via Testbench files
- Successful containerization of testing procedures
- Deployment/synthesis onto Rogue's Gallery FPGA hardware

CPU Integration Progress

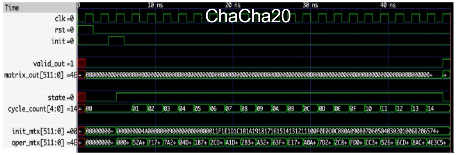
- Created and Verified MMIO State Machine that translates memory accesses into block function commands.
- Created and Verified AXI4-Lite bus control that translates AXI4 memory accesses into formats our MMIO FSM can understand.

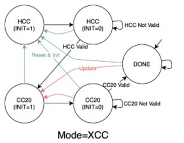


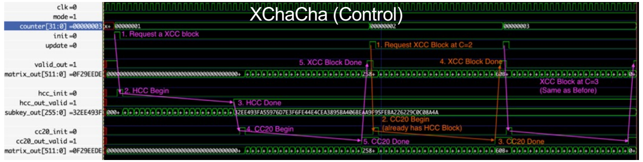


Design Verification Progress

- Fixed ChaCha20 block function and verified it with publicly available test vectors.
- Implemented HChaCha block function and XChaCha control block function, and verified them against known test vectors.
- Made 11 tests for block function functional correctness
- Made 4 tests for correctness of control circuits
- Aesthetic and readability changes including the adoption of concise and readable packed arrays and other features from newer version of Verilog or SystemVerilog.

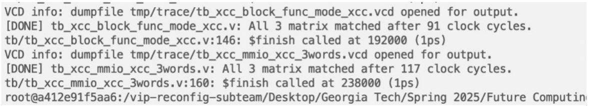






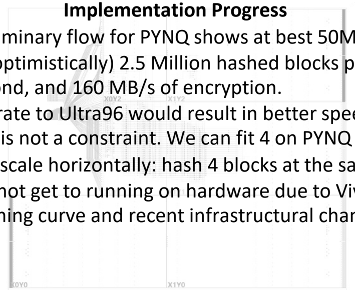
Containerization Progress

- Successful running of testbenches through the containerization of iverilog and vvp in both Docker and Apptainer.
- Automation of testing via Makefile.



```
VCD info: dumpfile tmp/trace/tb_xcc_block_func_mode_xcc.vcd opened for output.
[DONE] tb_xcc_block_func_mode_xcc.v: All 3 matrix matched after 91 clock cycles.
tb/tb_xcc_block_func_mode_xcc.v:146: $finish called at 192000 (1ps)
VCD info: dumpfile tmp/trace/tb_xcc_mmio_xcc_3words.vcd opened for output.
[DONE] tb_xcc_mmio_xcc_3words.v: All 3 matrix matched after 117 clock cycles.
tb/tb_xcc_mmio_xcc_3words.v:160: $finish called at 238000 (1ps)
root@a412e91f5aa6:/vip-reconfig-subteam/Desktop/Georgia Tech/Spring 2025/Future Computin
```

Implementation Progress



- Preliminary flow for PYNQ shows at best 50Mhz clock, or (optimistically) 2.5 Million hashed blocks per second, and 160 MB/s of encryption.
- Migrate to Ultra96 would result in better speed.
- Size is not a constraint. We can fit 4 on PYNQ fabric.
- Can scale horizontally: hash 4 blocks at the same time.
- Did not get to running on hardware due to Vivado learning curve and recent infrastructural changes.

Challenges and Lessons Learned

Challenges

- Repo from the previous semester was disorganized and needed fixing
- Having to rewrite the Verilog implementations
- Couldn't access any development VMs for a bit

Lessons Learned

- Verilog is now cleaner and more readable
- Learned how to properly containerize the design
- Proper documentation: New members should no longer have to start from scratch

Moving Forward

- Run on hardware. Easy now as implementation is mostly done.
- Speedup the existing design with faster busses, better FSM, and DSP optimized block functions.
- Improve onboarding & documentation, especially for students new to Verilog and FPGA.
- Formal Verification of the FSMs.