Radiation-Hardened-By-Construction Microcontroller

PICO Chipathon Design Review

ADEPT Laboratory

Janani Aravind, Phillip Gibson, Joshua Govan, Tyler Houston
Dawood Rauf, Kristian Rosa, Cebron Williams, Ahmed Yiwere, Dr. Daniel Limbrick

Department of Electrical and Computer Engineering
North Carolina Agricultural and Technical State University

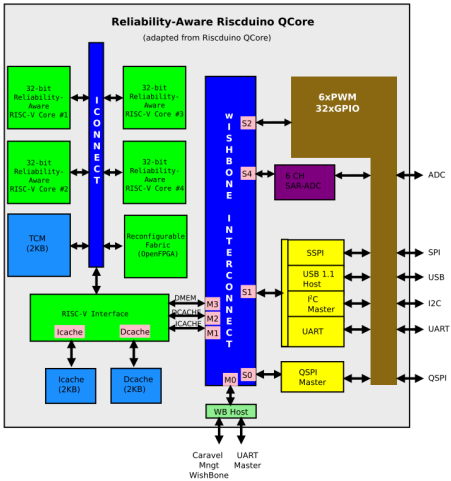
September 30, 2022

Design Goal: Reliability-Aware IC Design

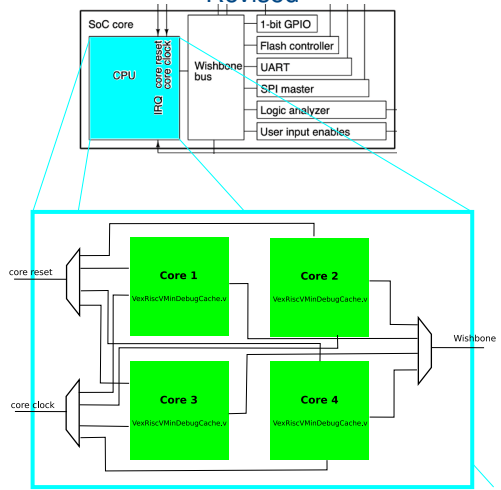
- **ICs are susceptible to radiation-induced soft errors**
- **Rad-hard ICs are costly and underperform compared to commercial ICs**
- **Question: How can ICs be designed to meet both reliability and performance demands?**
- **Original plan**
  1. Find critical nodes
  2. Find gaps in the design
  3. Identify most efficient hardening strategy

Design Overview

Original



Revised



Summary of Design Modifications

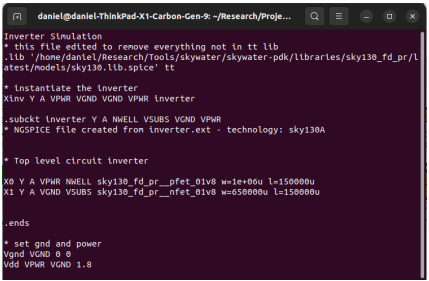
- **Replaced quad-core Riscduino design w/ single-core VexRiscv w/ 4 multiplexed cores**
  - VexRiscv is proven to work since it is used as part of the management SoC
  - VexRiscv soft core works well on FPGA
  - Quad-core functionality would make post-silicon fault injection testing harder
- **Removed OpenFPGA fabric**
  - ran out of time
  - always considered a bonus addition to the research
- **Changed from finding critical nodes to harden to random node replacement**
  - Quicker
  - Can still prove that selective node hardening can be done without increasing chip area/delay

Design Sources

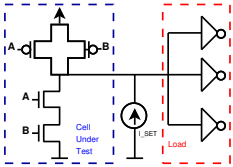
- **“junga_soc_MPW5”**
  - implementation of single-core vexRISCV
  - also contains two SRAM blocks
- **Management SoC Core**
  - contains VexRiscv, SRAM, connections to logic analyzer and wishbone interface
  - more than we need (would require more duplication of peripherals that we are not testing)
- **“SEL_SET”**
  - Author: Janani Aravind
  - implements 8 4-bit ALUs with different layouts for radiation testing
  - uses tri-state buffers to turn on/off ALUs (same approach for our design)

Fault Injection with ngSPICE

- **Characterize standard cell library for expected transient pulse width**
- **Methodology in place (FreePDK45)**
- **Results on sky130_fd_pr pending**



SPICE Modeling



Results for FreePDK45

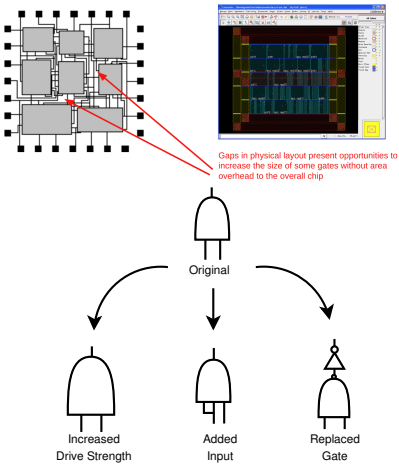
Cell	Worst-case Pulse Width (ps)
INV	464
NAND2	475
NOR2	643
AND2	631
OR2	811
XOR2	808
XNOR2	639

Testbench Applications

- **We are not changing functionality of Vexriscv**
- **Using existing testbenches from**
  - caravel: user_proj_example and caravel_mgmt_soc_lite (tests-caravel)
  - From VexRiscv Github: simple LED light program that we ran on DE-10 Lite FPGA
- **Also, in the process of performing bit-flip analysis on RTL with XOR fault model**

Design of Each Vexriscv

- **Each will have the same nodes hardened**
- **Each will have the same gaps exploited**
- **Differences**
  - Vexriscv 1 - Unmitigated
  - Vexriscv 2 - Larger drive strength cell used
  - Vexriscv 3 - Multiple input cell used
  - Vexriscv 4 - Replaced logic used



Area Estimate and Pin List

- **Estimated Area: 4 mm^2**
  - Based the area on the following:
    - die area of user_proj macro for “junga_soc_mpw5” = 1.0049374208 mm^2
    - die area of user_project_wrapper macro for “junga_soc_mpw5” = 10.2784 mm^2
    - multiplied the die area of user_proj (Vexriscv + 2 SRAMs) by 4
- **Pin list**
  - Logic analyzer signal used to switch between Vexrisc5 macros
  - Can we keep all pins internal and run off of logic analyzer signals?
  - Need help

Caravel Sharing

- **Pending confirmation of area and pin list**

Timeline

- **Finalize Design - Done**
- **Dummy Precheck - Done**
- **SET Characterization of Skywater 130-nm library - In Progress**
- **Caravel Sharing - In Progress**
- **Design of each VexRiscv core - In progress**
- **GDSII of Final Design - Not started**

Useful links

- **GitHub page of project: https://github.com/ADEPT-Group/RHBC_RISC_V.git**
- **Efabless page of project: https://platform.efabless.com/projects/928**
- **Lab website: https://sites.google.com/view/adept-laboratory**

- **Efabless page of “junga_soc_mpw5” project: https://platform.efabless.com/projects/780**
- **Efabless page of “SEL_SET” project: https://platform.efabless.com/projects/1065**