RISC-V Processor
Team RISC-V Business

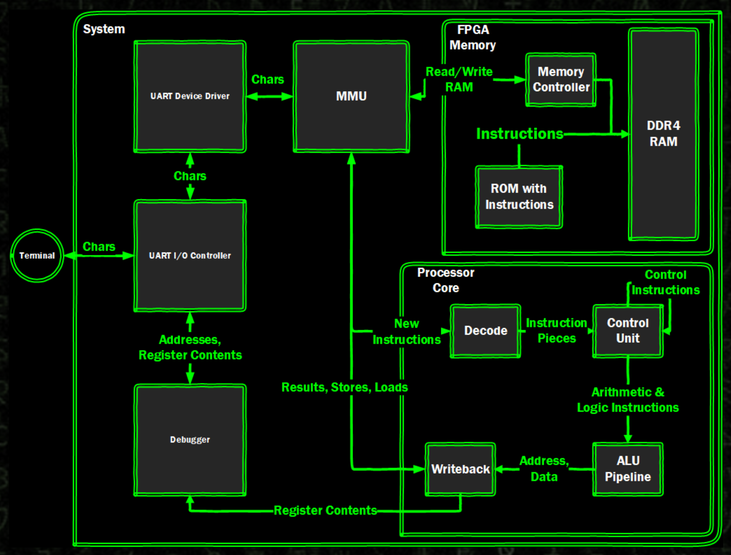
Cesar Avalos, Jacob Fustos, Stephen Longofono

Premise

l33t8ox:~$ cat premise.txt

The premise of the project was to create a simple computer system that implements the RISC-V instruction set architecture. Building on the basic architecture presented by the RISC-V foundation, we built the project in VHDL, targeting the Nexys4 FPGA hardware. The programs are stored in ROM and pulled into RAM for execution when the system is powered on.

Block Diagram



Goals

l33t8ox:~$ cat goals.txt

Develop a simple core to support arithmetic instructions and basic branching, loads, stores

Develop support for interrupts, privileged instructions

Develop an MMU with RAM and ROM controllers
Adapt the Berkeley Bootloader to bootstrap our code into RAM

Develop peripheral drivers to interface a thin client for I/O

.

Results

l33t8ox:~$ cat wins.txt

Implemented the core components

Implemented an MMU and debugger unit to interface the core

Implemented peripheral and hardware drivers to bring the system together

Ran simple programs demonstrating the operation of the system