Pipelined CPU - Progress Report

Cody Cziesler and Nick Desaulniers

As of May 5, 2011, the entire design was written and compiled in Verilog hardware description language. The pipelined CPU consists of five stages: Instruction Fetch, Instruction Decode, Execute, Memory, and Write Back. In addition to all of the stages containing the necessary logic, memories were created using the Xilinx Core Generator from the Xilinx ISE Project Navigator. By using this tool, the memories were able to be preloaded with a test program. Our own ISA was written and is used in our test program.

Our Verilog code compiles, and simulates. Currently, we are working on creating a test program using our ISA to preload instruction memory, that way we can simulate our design and verify that it functions correctly.

Once we can verify that our CPU Behaves properly, we want to verify that the stages are in fact pipelined in simulation. We also want to be able to program our FPGA using the on board Platform Flash PROM, that way we can store our configuration in a non-volatile method.

We are also working on ways to demo with the FPGA. To accomplish this, we plan on creating a simple test program, and showing a pass/fail output led on the Xilinx Spartan-3E FPGA. Also, we have been researching how to keep a design in the memory of the FPGA so that it does not have to be reprogrammed as soon as the FPGA is turned on; in other words, the design will remain in memory. This will be useful when demoing the board so that it can be quick and easy.

Overall, the project is looking good. Everything is coming together nicely and it will be done early next week. By next Thursday, a presentation will be ready to show the inner workings such as the Verilog source code and a few block diagrams. Also, we are working on a demo to show that our design actually performs like a CPU should.

