

PROJETO E IMPLEMENTAÇÃO DE UM PROCESSADOR EM FPGA PARA ENSINO DE ARQUITETURA DE COMPUTADORES

Vinicius L. NASCIMENTO¹ Heber R. MOREIRA² Tiago G. BOTELHO³

RESUMO

O ensino prático de hardware requer componentes caros e estudos complexos, tornando o desenvolvimento de material caro e muitas vezes inviável. Tecnologias de desenvolvimento de hardware como FPGAs (Field Programmable Gate Arrays) permitem desenvolvimento prático e barato de hardware, por meio de ambientes de modelagem e simulação. Um processador pode ser desenvolvido utilizando essas tecnologias e carregado em um FPGA, efetivamente implementando o processador em hardware real, capaz de executar programas em uma linguagem assembly própria.

Palavras-chave:

Hardware, Educação, Assembly.

1. INTRODUÇÃO

Contrastado ao ensino de software, amplamente apoiado por ferramentas que exigem pouco ou nenhum recurso, o ensino de hardware por meios práticos requer materiais caros e de desenvolvimento complexo e demorado, inviabilizando tal abordagem. FPGAs fornecem uma alternativa acessível ao desenvolvimento de hardware por meio de modelagem de circuitos que definem funções lógicas após a produção do chip (XILINX, 2018), utilizando uma linguagem de alto nível. O processo é apoiado por um ambiente de desenvolvimento e simulação, enquanto placas baseadas em FPGA proveem recursos úteis como memórias e displays.

Objeto central de estudo na disciplina de arquitetura e organização de computadores, processadores são geralmente inacessíveis e com detalhes de implementação proprietários. Essas dificuldades geralmente tornam simulações a única ferramenta de ensino, conseqüentemente afastando estudantes de hardware real. Tais fatores tornam o desenvolvimento de um processador de hardware acessível desejável na prática educacional.

A arquitetura desenvolvida contém componentes que a aproximam da arquitetura MIPS apresentada por Patterson e Hennessy (2012), como registradores de entrada da Unidade Lógica e

¹Bolsista PIBIC/Fapemig, IFSULDEMINAS – Campus Muzambinho. E-mail: 12151002633@muz.ifsuldeminas.edu.br.

²Coorientador, IFSULDEMINAS – Campus Muzambinho. E-mail: heber.moreira@muz.ifsuldeminas.edu.br.

³Orientador, IFSULDEMINAS – Campus Muzambinho. E-mail: tiago.botelho@muz.ifsuldeminas.edu.br.

Aritmética (ALU), Contador de Programa (PC) e registradores de memória. A proximidade à uma arquitetura conhecida permite que o estudo do processador desenvolvido resulte em uma melhor compreensão do funcionamento de processadores comerciais.

Um sistema básico de entrada e saída que utilize dos recursos disponíveis na placa de desenvolvimento é indispensável para a interatividade e usabilidade do projeto. Uma abordagem simples e compatível com a arquitetura desenvolvida, mas também viável e utilizada comercialmente, é um sistema de mapeamento de entrada e saída por meio de memória (*Memory Mapped IO*). A presença de um sistema de entrada e saída permite a leitura de dados do usuário e a exibição de resultados de processamento.

A programação do processador é feita por meio de carregamento direto de dados na memória, utilizando arquivos de inicialização de memória. Essa abordagem permite que o processador seja usado com seu potencial completo, mas apenas por meio de código binário. O desenvolvimento pode ser apoiado por meio de um *assembler* que implemente uma linguagem própria a ser traduzida para código binário, chamada linguagem *assembly* (PATTERSON, HENNESSY, 2012).

2. MATERIAL E MÉTODOS

O desenvolvimento de circuitos para implementação em FPGA utiliza linguagens de descrição de hardware, sendo a *Very High Speed Integrated Circuit Hardware Description Language* (VHDL) uma notação formal para o desenvolvimento de sistemas eletrônicos (IEEE, 2000), escolhida no desenvolvimento do projeto devido a sua clareza e facilidade de desenvolvimento, aliados à sua velocidade. A legibilidade da linguagem adotada se torna desejável por facilitar a compreensão do circuito implementado no caso de interesse em desenvolvimento futuro, além da facilidade e velocidade de desenvolvimento.

A modelagem do circuito que implementa o processador utiliza uma abordagem bottom-up, permitindo que componentes simples do circuito sejam desenvolvidos e propriamente testados para em seguida serem utilizados na modelagem de componentes mais complexos. Cada componente é estruturado por meio da definição dos seus sinais de entrada e saída, seguida da definição do comportamento do circuito. Componentes individuais são interligados para a implementação de componentes mais complexos e ao final do processo o circuito completo do processador pode ser montado.

O assembler desenvolvido pode ser executado por linha de comando e é capaz de receber um programa de entrada escrito na linguagem própria do processador em um arquivo de texto e produzir um arquivo de inicialização de memória formatado para uso no ambiente de desenvolvimento para carregamento de memória.

Um diagrama da arquitetura, exposto na figura 1, é apresentado como material de apoio para apresentação do funcionamento do circuito, incluindo o sistema de mapeamento de memória utilizado para entrada e saída.

4. CONCLUSÕES

O resultado final do projeto foi um processador funcional, programável com sua linguagem própria, capaz de executar programas interativos em hardware utilizando uma arquitetura de fácil compreensão. O desenvolvimento buscou um resultado similar à arquitetura e linguagem apresentadas na disciplina de arquitetura e organização de computadores. A ferramenta pode ser estendida por meio da implementação de comunicação serial com um computador externo para programação, sistemas mais robustos de entrada e saída e otimizações na arquitetura visando performance.

AGRADECIMENTOS

Ao órgão de fomento PIBIC/Fapemig pelo financiamento da pesquisa e ao IFSULDEMINAS - Campus Muzambinho pela disponibilidade das instalações.

REFERÊNCIAS

ALTERA. **Intel Quartus Prime Software**. Disponível em <<https://www.altera.com/products/design-software/fpga-design/quartus-prime/overview.html>>. Acesso em 24/07/2018.

IEEE. **Standard VHDL Language Reference Manual**. 2000.

PATTERSON, D. A.; HENNESSY, J. L. **Organização e Projeto de Computadores: A Interface Hardware/Software**. 3ª Edição. Editora Campus, 2005.

XILINX. **FPGA (definition)**. Disponível em <https://www.xilinx.com/support/documentation/sw_manuals/help/iseguide/mergedProjects/fpga_editor/html/fe_d_fpga.htm>. Acesso em 24/07/2018.