UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE

DEPARTAMENTO DE ENGENHARIA ELÉTRICA

BACHARELADO EM ENGENHARIA MECATRÔNICA

IMPLEMENTAÇÃO DE PROCESSADOR EM ARQUITETURA VON-NEUMANN COM INSTRUÇÃO EM MÚLTIPLOS OPERANDOS.

NATAL

2017

UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE

DEPARTAMENTO DE ENGENHARIA ELÉTRICA

BACHARELADO EM ENGENHARIA MECATRÔNICA

SISTEMAS DIGITAIS

EMERSON MATHEUS DANTAS CLEMENTINO

IMPLEMENTAÇÃO DE PROCESSADOR EM ARQUITETURA VON-NEUMANN COM INSTRUÇÃO EM MÚLTIPLOS OPERANDOS.

NATAL

2017

**Sumário**

**1. Introdução 03**

**2. Fundamentação Teórica 04**

**3. Implementação 06**

**4. Resultados Obtidos 08**

**5. Conclusão 09**

**6. Referências Bibliográficas 10**

**1. Introdução**

Neste documento será discutida a implementação de um modelo de processador de uso geral, desenvolvido em linguagem VHDL e adotando conceitos de arquitetura e organização semelhantes à Máquina de Von Neumann e conjunto de instruções baseadas em arquitetura RISC.

A máquina de Von Neumann é considerada o modelo precursor dos processadores de uso geral atuais, seu grande diferencial reside na possibilidade de alteração do programa implementado em sua memória, não restringindo a utilização do processador a um uso específico.

Seu conjunto de instruções e relativamente simples, possuindo operações de movimentação de dados, operações lógicas, operações aritméticas, operações de pilha, saltos diretos ou condicionais e chamadas de funções.

A construção da instrução nativa à Máquina de Von Neumann possui uma palavra única, e nesta são inseridos a operação a ser realizada e seus respectivos operandos; para fins acadêmicos este fato foi deixado à parte. Na implementação abaixo documentada, será aplicado o modelo de instrução utilizado por processadores de arquitetura RISC, possuindo o código de operação, e seus respectivos operandos em palavras binárias separadas.

**2. Fundamentação Teórica**

A arquitetura de Von Neumann, ou máquina de Von Neumann, é composta por quatro elementos principais: uma memória de dados, uma unidade lógica e aritmética, uma unidade de controle (composta por bloco de controle banco de registradores) e um dispositivo de entrada e saída de dados controlado pela unidade de controle.

O funcionamento da máquina de Von Neumann se baseia em três estados de controle: aquisição, decodificação e execução. O primeiro estado realiza a captura do(s) operando(s) na memória e os armazena no registrador de instrução. O próximo estado, decodificação, interpreta a instrução armazenada no registrador de instrução e determina a lógica de controle adequada, criando desta forma o caminho de dados a ser realizado. Por último o estado de execução realiza o processamento propriamente dito da informação, realizando sua passagem pelo caminho de dados criado anteriormente.

A memória de dados é o ente de grande armazenamento do processador de Von Neumann. Neste elemento são guardadas instruções, dados pertinentes às instruções e dados provenientes de instruções previamente executadas pelo processador.

A unidade lógica e aritmética realiza uma gama de operações lógicas e matemáticas necessárias ao funcionamento do processador. Este elemento é composto unicamente de uma série de lógicas combinacionais capazes de realizar operações aritméticas com números inteiros em notação binária (adição, subtração, multiplicação e divisão), assim como operações lógicas diretas aos dados inseridos, sendo essas de dois operandos como OR, XOR e AND lógicos; ou operações de operandos únicos tais como complemento A1, complemento A2, incrementos, decrementos e deslocamentos de bits, tendo resposta instantânea.

O banco de registradores constitui-se como o elemento de rápido armazenamento e baixa capacidade de dados do processador. Os registradores internos ao processador podem ser subdivididos em dois tipos principais, os registradores de uso geral e registradores de uso específico.

Os registradores de uso específico são elementos de utilização exclusiva ao processador, neles são armazenados dados transitórios referentes à rotinas internas do bloco de controle. Entre os principais registradores de uso específico encontram-se o contador de programa, o registrador de instrução, o contador de ponto de pilha e registradores de armazenamento de dados temporários de utilização do processador. O contador de programa possui como função básica o armazenamento e incremento do endereço de memória dos operandos destinados ao processador. Seu valor é quase sempre sequencial e incrementado em operações de captura, com exceção a operações de saltos, nas quais o valor por este registrado é alterado de forma direta. O registrador de instrução armazena as instruções que serão executadas após o estado de decodificação. O contador de ponto de pilha indica o espaço de memória destinado às operações que envolvem a pilha; por último, o registrador auxiliar retém dados temporariamente em operações de movimentação memória-memória.

Os registradores de uso geral são elementos de armazenamento que podem ser diretamente manipulados pelas instruções do processador, diferentemente dos registradores de uso específico. Nestes são armazenados dados provindos da memória principal, pilha de dados, informações advindas de outros registradores assim como dados gerados pela ULA.

O bloco de controle realiza a decodificação da instrução dada ao processador, e posteriormente enviando dados de controle aos elementos supracitados, construindo o caminho de dados adequado à tarefa a ele determinada.

Por último, o dispositivo de entrada e saída realiza a comunicação entre processador e mundo externo, provendo ao processador dados a serem trabalhados ou exportando dados previamente processados.

**3. Implementação**

A implementação da máquina supracitada foi realizada utilizando-se linguagem VHDL, com a finalidade de simular a estrutura de circuito utilizada em um processador real. Em sua construção foram utilizadas construções executadas de forma concorrente assim como construções executadas de maneira procedural, estas, aplicadas para reduzir a complexidade de implementação de certos componentes.

Para a construção do circuito foram utilizados conceitos de projeto RTL, decorrendo assim na necessidade de elementos como registradores, contadores, comparadores e multiplexadores. Para a implantação destes elementos foram utilizados recursos de linguagem procedural, aplicada com maior frequência em dispositivos que necessitam de sincronismo com clock, e recursos de linguagem concorrente para a concepção de dispositivos constituídos por repetição contada de uma determinada lógica (multiplexadores e comparadores). As lógicas aplicadas ao acionamentos dos elementos supracitados foi realizada utilizando-se recursos internos da linguagem, criando, desta forma, sensibilidade aos estados de controle e ao operandos armazenados.

A máquina de estados de baixo nível do processador seguiu o modelo de máquina Mealey. As saídas, destinadas ao acionamento dos elementos de memória, possuem sensibilidade direta ao estado atual da máquina, associado à entradas advindas dos registradores de instrução; aumentando, desta forma, a quantidade de graus de liberdade da saída sem acrescer a quantidade total de estados. A lógica de transição de estados foi construída utilizando recursos de linguagem procedural, procurando reduzir a complexidade da máquina dada a grande quantidade de variáveis à qual o dispositivo possui sensibilidade.

Os elementos do datapath, citados acima, foram divididos em três blocos principais: elementos de controle, elementos de armazenamento e elementos de processamento. Entre os elementos de controle se encontram os registradores e contadores utilizados pela máquina de estados para o controle do processador como um todo. Entre estes encontram-se o contador de programa, os registradores de instrução, o contador de ponto de pilha e um registrador de armazenamento temporário destinado às movimentações memória-memória. Os registradores de instrução foram aplicados neste modelo mediante à forma de armazenamento do programa na RAM, utilizando-se um, dois ou até três operandos por tarefa realizada, exigindo assim, três registradores de instrução

Os elementos de armazenamento são constituídos pelos registradores de uso geral, acomodados de forma coletiva em um banco de registradores, sendo estes manipulados diretamente pelo programador.

Os elementos de processamentos utilizados são somente dois: a unidade lógica e aritmética eu um bloco destinado a operações de shifting em registradores. A ULA aqui implementada é constituída por uma série de lógicas combinacionais destinadas à soma, subtração, incremento, decremento, complemento A1, comparações e operações lógicas a nível e bit(AND, OR e XOR); operações estas selecionadas através de um código binário inserido conjuntamente com os operandos. O bloco de shifting é constituído por uma máquina RTL síncrona que controla a direção e quantidade de shiftings.

A comunicação entre os blocos supracitados é realizada através de multiplexadores. A lógica de seleção inserida nestes é determinada pela máquina de estado de baixo nível, como citado anteriormente, a fim de determinar o caminho de dados adequado à operação determinada pelas informações armazenadas no registrador de instrução. A implementação dos multiplexadores foi realizada com comandos concorrentes da linguagem VHDL.

A memória de dados utilizada foi implementada em modelo RAM, possuindo duas entradas de leitura e escrita, uma destinada ao uso interno do processador e a outra simulando um dispositivo de IO. Foi-se utilizado o modelo padrão criado pela interface Quartus da Altera.

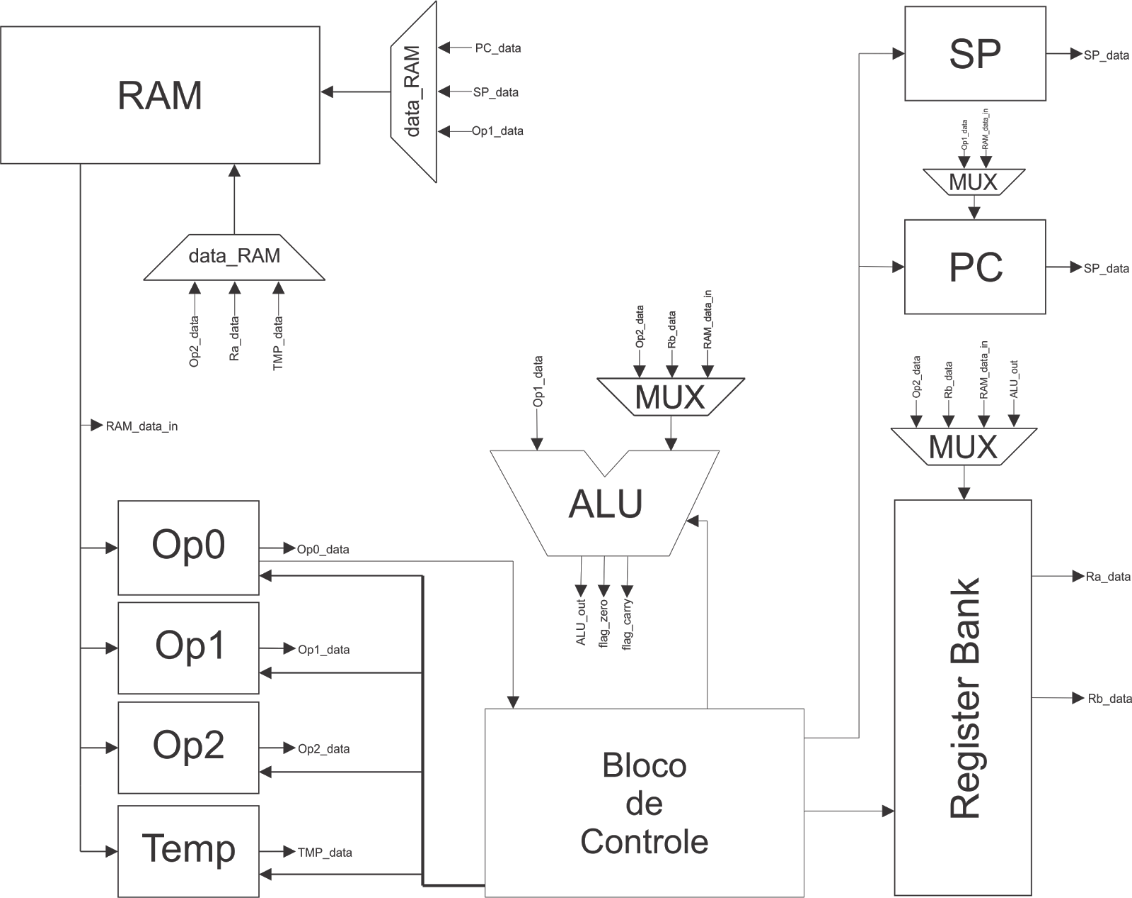


Figura 01: Esquema em blocos do processador implementado

**4. Resultados Obtidos**

A implementação do processador foi de certa forma extensa dada à verbosidade da linguagem. O teste das funções foi realizando através da execução do código na plataforma Vsim da altera, utilizando um false clock para a máquina de estados e o negado deste sinal destinado à memória RAM, e um arquivo de carregamento de memória com um programa teste destinado à execução de todas as funções exigidas.

Após testes exaustivos em simulador o código foi posteriormente testado em FPGA, utilizando interfaces de saída de dados destinados aos displays da placa, aliado a um código divisor de clock utilizado para reduzir o clock de operação da placa, 27 MHz, para um clock de operação de 10Hz.

Os teste físicos retornaram os resultados obtidos em simulador, constituindo um projeto implementável e sem sensibilidade a atrasos físicos dos circuitos do FPGA.

**5. Conclusão**

A utilização de circuitos microcontroladores e microprocessadores se tornou indispensável à vida humana nos últimos anos, e sua participação na composição de diversos dispositivos vem se tornando mais presente ao decorrer do tempo.

A máquina de Von Neumann apesar de seu caráter de construção e funcionamentos simples revolucionou o panorama da engenharia eletrônica e deu o pontapé inicial à construção dos processadores atuais, possibilitando os avanços à ciência que vieram acompanhados ao surgimento e aprimoramento dos computadores.

Neste documento foi possível a implementação bem sucedida de tal protótipo, possibilitando um maior conhecimento desta arquitetura e tornando mais fácil o entendimento do funcionamento da linguagem assembly como um todo.

**6. Referências Bibliográficas**

D’AMORE, R.; VHDL: descrição e síntese de circuitos digitais. 2.ed. Rio de Janeiro: LTC, 2015. 292p.

STALLINGS, William. Arquitetura e organização de computadores. 8. ed. São Paulo: Pearson Prentice Hall, 2010.

VAHID, F.; Sistemas Digitais: projeto, otimização e HSLs; tradução Anatólio Laschuk. – Porto Alegre: Artmed, 2008. 560p.