Programa para Excelência em Microeletrônica

Módulo: Nivelamento

Matéria: Sistemas Digitais

Desenvolvimento de um Microcomputador

Aluno: Thiago Henrique Nascimento Guimarães

  

Sumário

[Introdução 3](#_Toc465017354)

[Teoria 3](#_Toc465017355)

[Desenvolvimento do Projeto 4](#_Toc465017356)

[Contador 4](#_Toc465017357)

[REM 5](#_Toc465017358)

[Acumulador 5](#_Toc465017358)

[Somador e subtrator 6](#_Toc465017358)

[Memória 7](#_Toc465017358)

[Controlador 7](#_Toc465017358)

[Conclusões 8](#_Toc465017359)

# Introdução

O projeto teve como meta construir em software simulador um microcomputador partindo dos conhecimento em circuitos básicos e integração dos mesmo adiquiridos nas etapas iniciais do Projeto de Excelencia em Microeletrônica.

# Teoria

A partir de conhecmentos básicos de portas lógicas e circuitos combinacionais é possível se construir microprocessadores básicos com funções semelhantes a qualque processador no mercado.

A escolha de modelos de processadores para o ensino de conceitos de arquitetura e organização de computadores é alvo de estudos freqüentes pelos educadores da área.

“...determinados conceitos utilizados nas disciplinas da área de Algoritmos e Programação são melhor compreendidos pelos alunos quando eles entendem os seus significados ao estudar a arquitetura do computador.” (MORANDI, 2006, p.74)

Fundamentalmente, todos os processadores fazem as mesmas coisas. Eles captam os sinais sob a forma de 0s (zeros) e 1s (uns) - portanto, sinais binários -, manipulam estes sinais conforme um conjunto de instruções e produzem o resultado novamente no formato binário. A voltagem na linha no momento em que o sinal é enviado determina se o sinal é 0 ou 1. Em um sistema de 3,3 volts, uma aplicação de 3,3 volts significa que o sinal é 1, enquanto uma aplicação de 0 volts quer dizer que o sinal é 0.

Os processadores trabalham por meio de uma reação específica à entrada de um valor 0 ou 1, que gera um resultado (saída) baseado na decisão que foi tomada. A decisão ocorre em um circuito denominado porta lógica, e cada uma dessas portas precisa pelo menos de um transistor, com as entradas e saídas organizadas de modo diferente para diferentes operações. O fato de que os processadores atuais contêm milhões de transistores dá uma idéia do quanto um sistema lógico é complexo. As portas lógicas do processador trabalham em conjunto para tomar as decisões usando lógica Booleana, que é baseada no sistema algébrico estabelecido pelo matemático George Boole.

Os principais operadores Booleanos são AND (E), OR (OU), NOT (NÃO), e NAND (combinação de NOT e AND); e são possíveis muitas combinações destes operadores. Uma porta AND resulta em um valor 1 somente se ambas as entradas forem iguais a 1. Uma porta OR resulta em 1 se pelo menos uma das entradas for igual a 1. E uma porta NOT considera uma única entrada e a reverte, resultando em 1 se a entrada for 0, e vice-versa. As portas NAND são muito populares; uma vez que elas usam somente dois transistores em vez de três em uma porta AND, elas ainda oferecem muito mais funcionalidade. Além disso, o processador usa portas em combinação para desenvolver funções aritméticas; ele também pode utilizá-las para acionar o armazenamento de dados na memória.

As portas lógicas operam via através de uma hardware conhecido como comutador - particularmente, um comutador digital. Nos bons e velhos tempos em que os computadores eram do tamanho de uma sala (parecendo muito mais impressionantes nos filmes do que as máquinas de hoje), os comutadores eram, na verdade, comutadores físicos, mas hoje em dia, a única coisa que se move é a corrente em si. O tipo de comutador mais comum nos computadores atuais é um transistor conhecido como MOSFET (de metal-oxide semiconductor field-effect transistor). Este tipo de transistor desenvolve uma função simples, mas fundamental: quando uma voltagem é aplicada, ele reage ativando ou desativando o circuito.

# Desenvolvimento do Projeto

O projeto foi sub dividido em módulos, segue abaixo a montagem e proposito de cada.

## **Contador**

O contador (PC) é um contador de bits, feito a partir de portas and, or, nand e flip flops tipo D, etc., que indica qual é a posição atual na sequência de execução de um processo. Dependendo dos detalhes da arquitetura, ele armazena o endereço da instrução sendo executada ou o endereço da próxima instrução Muito simples o contador desenvolvido também pode ser feito apenas com flip-flops jk.



Figura 1 - Contador

## **REM**

## Feito pra manter durante um período o valor de endereço apontado pelo PC e baseado em um demux de endereco feito com portas lógicas básicas e enquanto a REM retém o valor de endereço da RAM, o PC pode ser incrementado.



Figura 2 – Demix de endereço



Figura 3 - REM

## **Acumulador**

Feito com latchs de instrução que foram compostos por portas logicas basicas e flip flops tido D e que armazenam instrruções o acumulador é um dos registradores do microcomputador, no qual se armazena os resultados de uma operação aritimética



Figura 4 – Latch de instrucao



Figura 5 - Acumulador

**Somador e subtrator**

Em sistemas digitas necessitamos de circuitos que utlizem básicas somas e subtrações, no ramo dos microcomputadores ele ocupam posição impotante pois são responsáveis por parte das operações aritmeticas. Nesse projeto montamos um latch para servir ao nosso somador subtrator.

C:\Users\ThiHenrique\AppData\Local\Microsoft\Windows\INetCache\Content.Word\drawing 2.eps

Figura 6 – Latch circuito somador

## **Memória**

A memória RAM (Random Access Memory) é onde está armazenada o programa e os dados a serem buscados e executadosRegistrador de Saida

## Registrador de instruções

Responsável por armazenar a instrução endereçada na memória e que foi disponibilizada no barramento esse modulo foi construido a partr de flip flops tipo D.

C:\Users\ThiHenrique\AppData\Local\Microsoft\Windows\INetCache\Content.Word\reg de inst.eps

Figura 7 – Registrador de instruções

## **Controlador**

Responsável pela geração dos sinais que controlará todos os blocos e, assim, sincronizando suas operações no devido tempo



Figura 8 - Controlador

# Conclusões

Este é um modelo de documento no qual você deve se guiar para escrever o relatório do projeto e, assim, ter um documento organizado. Tente ser claro nas explicações de como chegou-se as suas soluções para o projeto.

O projeto possibilitou uma experiência mais próxima da realidade de desenvolvimento de hardware, fazendo os detalhes desse processo evidente e mostrando os problemas envolvidos nesse desenvolvimento.

Foi de grande valia o conhecimento sobre a ferramente Quartus BLA, pois o software oferece uma gama de possibilidades muito interessante para o projeto proposto, porém, o software tem nuances que podem tornar o trabalho complexo em aspectos nos quais não deveriam. Pela complexidade do programa em si se perde um tempo significativo na tentativa de superar essas dificuldades quando esse mesmo tempo poderia ser usado para o desenvolvimento do hardware em si. Talvez programas que possuissem as mesmas possibilidades de desenvolvimento e fossem mais simples em seu manuseio em si (Proteus BLA como exemplo aproximado) auxiliasse o desenvolvimento das tarefas.

Microcomputadores apesar de possuirem uma estrutura definida e organizada podem ter seus módulos implementados de maneiras diferentes em seus componentes mínimos desde o uso de um flip flop já conhecido á implementação de um circuito inteiro utilizando-se portas lógicas básicas o que torna o desenvolvimento de um microcomputador algo bem interessante. Pesquisar pelo funcionamento de cada módulo do projeto ajudou a entender mais a fundo o funcionamento de um núcleo e a implementa-los no programa. Cada bloco construido trouxe um novo conhecimento através da busca pelo funcionamento e da resoluções dos problemas.

Houveram problemas na integração dos módulos, alguns erros foram apontados e a falta de familiaridade com o programa Quartus não contribuiu para a solução dos mesmos. Mas a experiencia do desenvolvimento do projeto ajudou a elucidar muito o desenvolvimento de hardware através de circuitos básicos.

O repositório com o projeto supra citado se encontra em https://github.com/ThiHenrique/ThiagoHenrique\_uC\_PEM

**Referências Bibliográficas**

* RONALDI J. Tocci, Neal S. Widmer e Gregory L. Moss[.](http://www.pearsonhighered.com/tocci_br/)**Sistemas digitais: princípios e aplicações**. 10a ed, Prentice-Hall, 2008.
* MORANDI, et all. **Um Processador Básico para o Ensino de Conceitos de Arquitetura e Organização de Computadores** . CTTMar (Univali).2006.