

Programa para Excelência em Microeletrônica

Módulo: Nivelamento

Matéria: Sistemas Digitais

Desenvolvimento de um Microcomputador

Aluno: Wesley Wevertton De Azevedo Palmeira

Sumário

Introdução.....	3
Teoria.....	3
Desenvolvimento do Projeto.....	4
Conclusão.....	8

Introdução

No mundo de hoje, é comum nos encontrarmos com microprocessadores, eles estão presentes em nossos computadores, celulares...Um microcomputador é definido como uma maquina no qual o processamento de dados é dado por um microprocessador, neste trabalho foi procurado desenvolver um microcomputador (uP) de arquitetura SAP (Simple As Possible), o intuito dessa arquitetura é introduzir as idéias do funcionamento de um processador sem entrar em detalhes que deixariam a aprendizagem mais cansativa.

Teoria

Qualquer sistema de computação precisa apresentar, em sua construção, pelo menos 3 partes:

- 1 - Processador (Conhecido como CPU)
- 2 - Memória (Onde são armazenadas as instruções e os dados do computador)
- 3 – Conjuntos de entradas e saídas.

O microcomputador objeto de desenvolvimento do trabalho possui todas essas partes abordadas de uma forma diferente mais especificamente podem ver na Figura 1 seu diagrama de blocos. Mais adiante será mostrada uma breve introdução sobre o conceito de cada bloco.

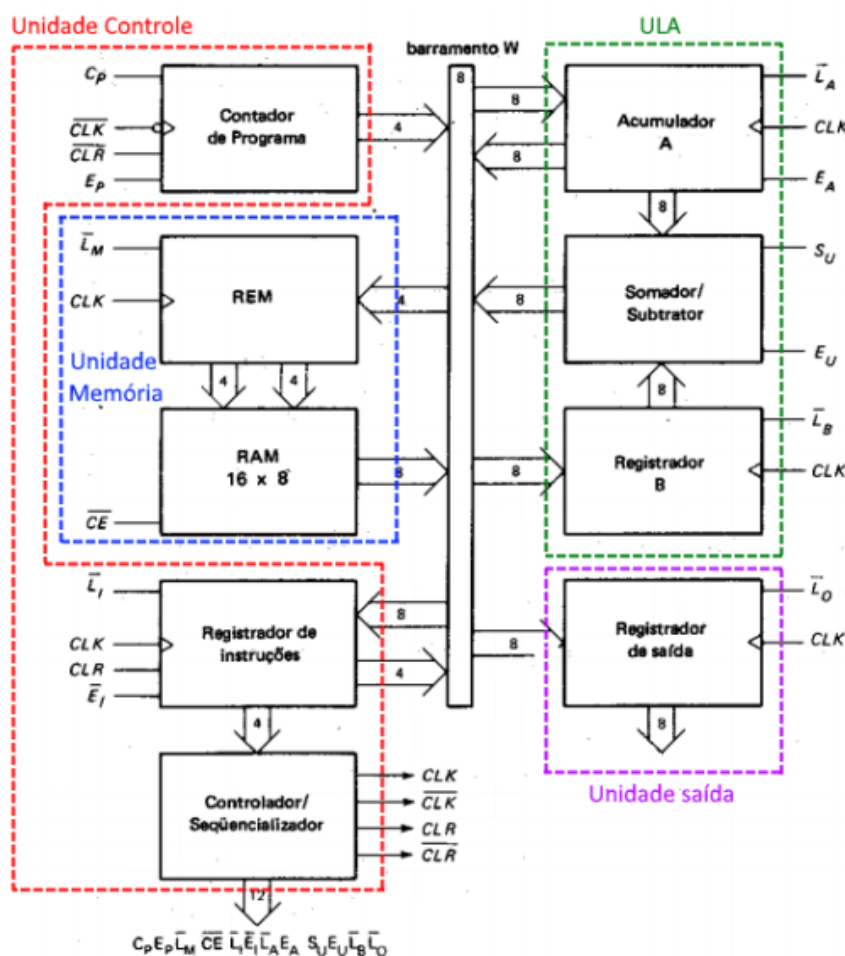


Figura 1 – Diagrama de blocos de um microcomputador SAP

Unidade de Controle

Esta unidade do nosso microcomputador é composta por três partes: Controlador, Registrador de instruções e o Contador de programa, cada parte tem um papel bem especial.

O Contador de programa, é um contador de 4 bits, que tem o papel de apontar para o endereço de uma instrução para ser buscada e executada.

O Registrador de instruções armazena a instrução que está na memória e foi disponibilizada no barramento.

O Controlador/Sequencializador é responsável, como o próprio nome já diz, por gerar sinais para controlar todos os outros blocos.

Portanto, como vimos, a unidade de controle é a responsável por controlar as operações enviadas para o microcomputador.

Unidade de Memória

Esta unidade possui basicamente duas partes: Memória REM e Memória RAM, a memória REM tem finalidade de manter constante o valor de endereço o valor de endereço apontado pelo PC, já a memória RAM é onde está armazenado o programa e os dados a serem buscados e executados. Esse um bloco com a função básica de armazenamento.

ULA

A unidade lógica e aritmética (ULA) na verdade, uma "grande calculadora eletrônica" ele é composta por três blocos: Acumulador A, Somador/Subtrator e Registrador B.

O Acumulador A é um registrador de 8 bits que armazena os resultados intermediários calculados pelo uP.

O Somador/Subtrator é, como o próprio nome já diz, um somador/subtrator de 8 bits que trabalha em complemento de 2.

O Registrador B é outro registrador de 8 bits usado para fornecer o número em operações aritméticas.

Portando, podemos perceber que a ULA executa as principais operações lógicas e aritméticas do computador, no nosso caso, soma e subtração de até 8 bits trabalhando com complemento de 2.

Unidade de Saída

Esta unidade possui apenas um bloco, o Registrador de saída, que simplesmente que transfere o conteúdo presente no acumulador A para o mundo externo.

Desenvolvimento do Projeto

Para o desenvolvimento do projeto, foi escolhido o software Quartus *Prime* versão 16.1, o desenvolvimento consistiu em uma adaptação do modelo proposto no livro “Digital Computer Electronics”, Third Edition de Albert P. Malvino para que o uP trabalhe com um clock de 1 Mhz e que os Cis sejam construídos desde o nível mais baixo.

O modelo de desenvolvimento adotado foi o Bottom-Up (de baixo para cima) ou

seja, cada parte do microcomputador que está especificado na Figura 1 foi construída separadamente e só após todas estarem prontas, elas foram ligadas entre si.

Contador de Programa

Para a implementação do contador de programas, foram utilizados flip-flops JK com suas entradas JK interligadas entre si e ligadas na entrada Cp do bloco, essa entrada determina quando o clock passa pelo contador, a sua saída tem um controle chamado Ep, que controla por meio de portas tri buffer quando a saída vai para o barramento. Ver Figura 2.

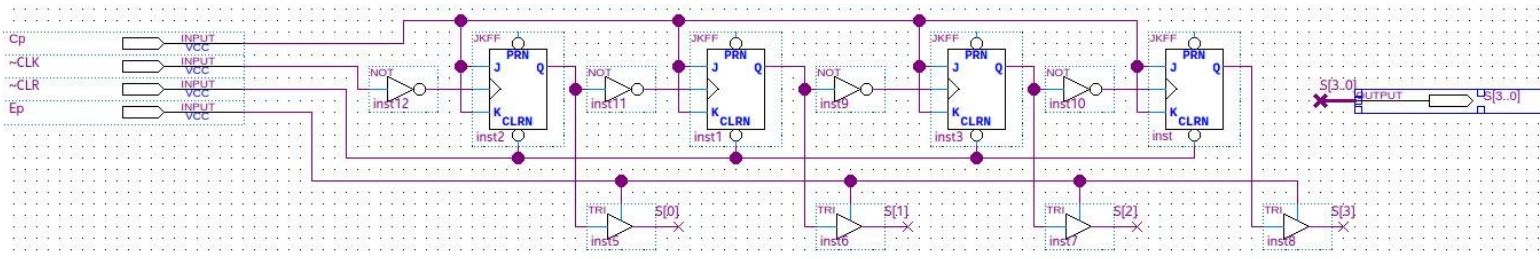


Figura 2 – Controlador de Programa

Registrador de Instruções

Na implementação do registrador de instruções, foram usados flip-flops do tipo D para guardar os dados que existem no barramento, que permite ou não guardar esses dados, as saídas desse bloco vão para dois lugares diferentes, uma para o controlador/sequencializador e outra para o barramento, a primeira são os 4 bits mais significativos e a segunda os bits menos significativos, vide na Figura 3.

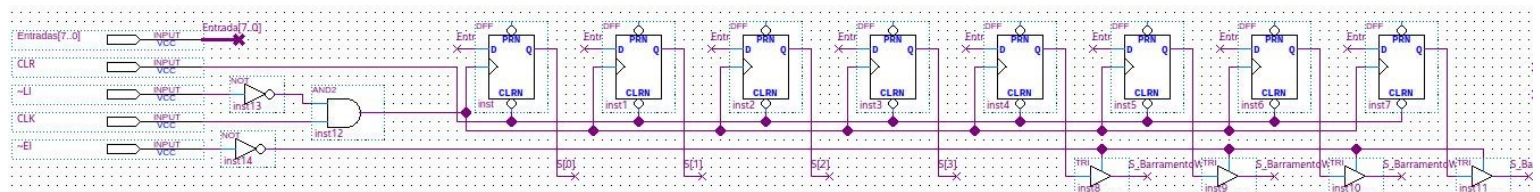


Figura 3 – Registrador de Instruções

Controlador/Sequencializador

Este é o maior de todos os blocos, responsável por controlar todos os outros blocos, como já especificado, ele possui dois blocos dentro dele, o Contador em Anel e o Fetch.

O Fetch consiste na caracterização dos 3 primeiros pulsos de clocks do computador, ele foi construído baseando-se nas observações das saídas e foi construído usando apenas as três primeiras saídas do contador em anel como entradas, portas Not e auxílio de Vcc e GND. Ver Figura 4.

O Contador em Anel é simplesmente um circuito de deslocamento de bits em que ao chegar no ultimo bit seu valor é retornado para o primeiro, por isso o nome “Anel”, ele foi confeccionado utilizando flip-flops do tipo D, ver Figura 5.

The diagram illustrates a 3-to-8 decoder circuit implemented using two 7414 hex inverters. The circuit has three inputs, T1, T2, and T3, each connected to a VCC supply. The outputs of the first inverter (inst4) are connected to the inputs of the second inverter (inst3). The outputs of inst3 are connected to the outputs of the decoder: CP, EP, ~LM, ~CE, ~LI, ~EI, ~LA, EA, SU, EU, ~LB, and ~LO. A VCC supply and a GND connection are also shown.

Para implementação da memória REM foram utilizados flip-flops do tipo D para armazenar os dados, com uma variável de controle as 4 saídas da memória REM vão para a memória RAM, muito semelhante ao registrador de instruções, ver Figura 6.

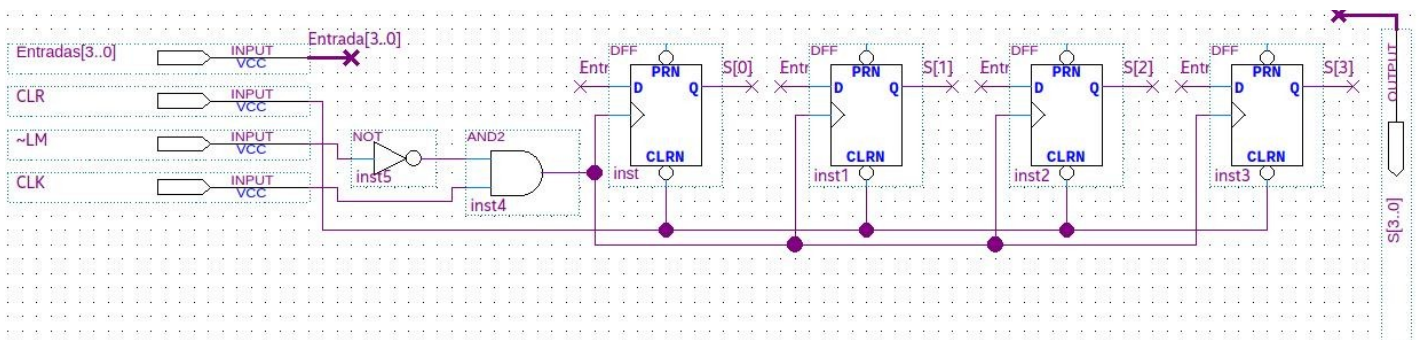


Figura 6 – REM

Memória RAM

A memória RAM foi de fácil implementação graças a utilização do bloco ROM que o próprio quartus disponibiliza, foi utilizado esse bloco/memória ROM com endereço de 4 bits para armazenar um arquivo .hex que contem os comandos a serem executados e as informações a serem processadas pelo microprocessador uP. A saída desses dados são conectadas ao barramento por meio de um tri buffer somente quando a variável de controle CE estiver ativa. Ver Figura 7.

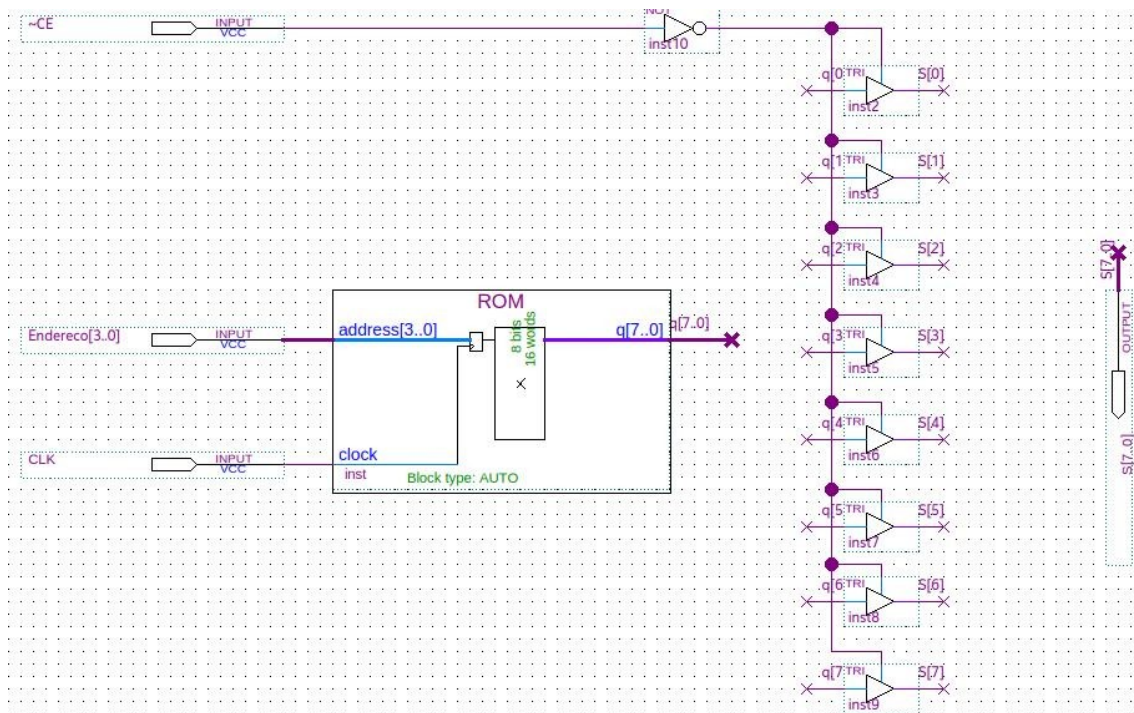


Figura 7 – REM

Acumulador A

Para a implementação do acumulador, foram utilizados flip-flops do tipo D que serviram para armazenar os dados que sempre que permitido, pela variável de controle LA, pega os dados do barramento e coloca-os na saídas dos respectivos flip-flops, ver Figura 8.

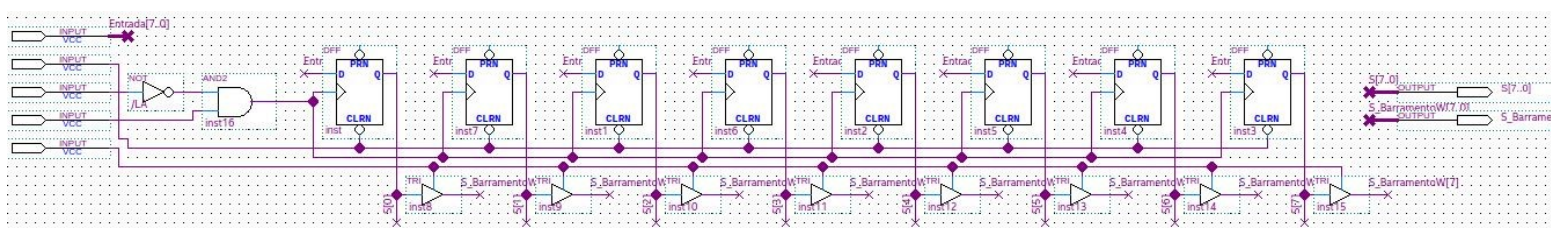


Figura 8 – Acumulador

Somador/Subtrator

Para confecção desse bloco, foi utilizado a fundo o conceito de bottom-up, foram construídos um meio somador e um meio subtrator, somente com portas lógicas,

a partir deles, construídos um somador completo e um subtrator completo e a partir desses um somador completo de 8 bits e um subtrator completo de 8 bits e por fim a partir desses dois, o somador/subtrator. Não será mostrado por meio de figuras os circuitos que compõem o somador/subtrator pois são circuitos triviais e geralmente já conhecidos. Já conhecido os circuitos Somador de 8 bits e Subtrator de 8 bits, para juntar os dois foi construído uma lógica por meio de portas AND e OR que simulam um MUX para selecionar se iremos utilizar a saída do Subtrator ou do Somador para o Barramento.

Registrador B

É um bloco muito semelhante ao bloco Acumulador, também confeccionado utilizando flip-flops do tipo D, mas agora possui apenas uma saída que vai para o Y do nosso bloco Somador/Subtrator. Ver Figura 9.

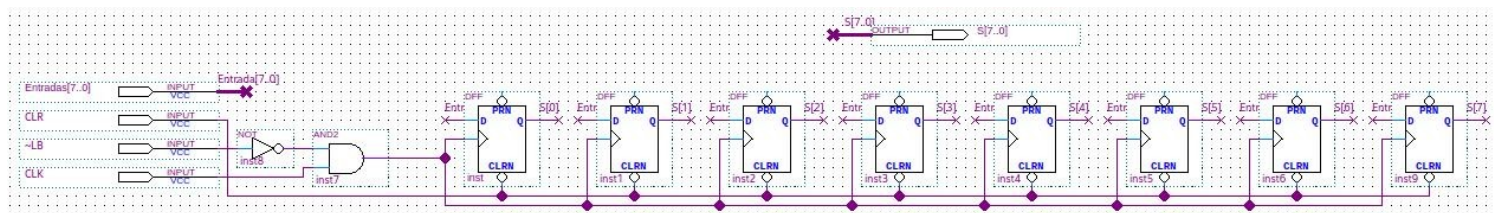


Figura 9 – Registrador B

Registrador De Saída

É um registrador bastante simples, pega o que está no barramento quando a variável de controle LO estiver ativa, e guarda em flip-flops tipo D para ser lida no mundo externo. Ver Figura 10.

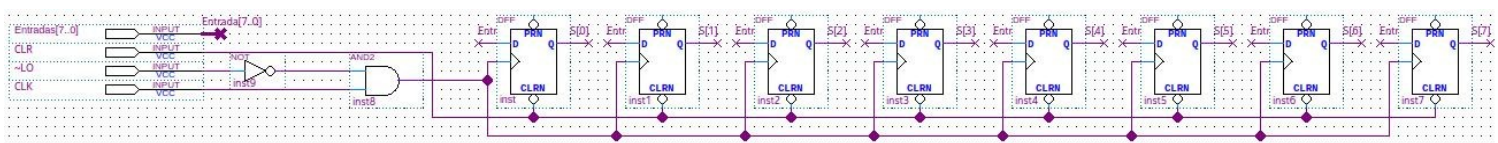


Figura 10 – Registrador de Saída

Conclusão

Portanto, feito a implementação de cada bloco separado como mostrado na descrição do trabalho ficou fácil uni-los com o diagrama de blocos da Figura 1. A confecção do trabalho, apesar de muito trabalhosa, por ser um trabalho inteiramente esquemático, foi muito gratificante, pois deu para entender realmente como se dá o processamento de um computador, tendo em vista que entendendo o microcomputador da arquitetura proposta, fica fácil saber como funciona os atuais, ligar fio por fio foi trabalhoso mas foi recompensador, pois dessa forma cansativa, entendemos realmente como funciona todo o processo de processamento.