

Processador de Propósito Geral

Mateus Arnaud Santos de Sousa, Rodrigo Sarmento Xavier

Dezembro 2018, Natal-RN

1 Introdução

A atividade tem por objetivo projetar um processador com as dez instruções mostradas na tabela 1 via linguagem de descrição VHDL. Os resultados simulados podem ser visualizados por Waveform, e os resultados práticos podem ser testados na placa Cyclone II e visualizados pelos displays de sete segmentos.

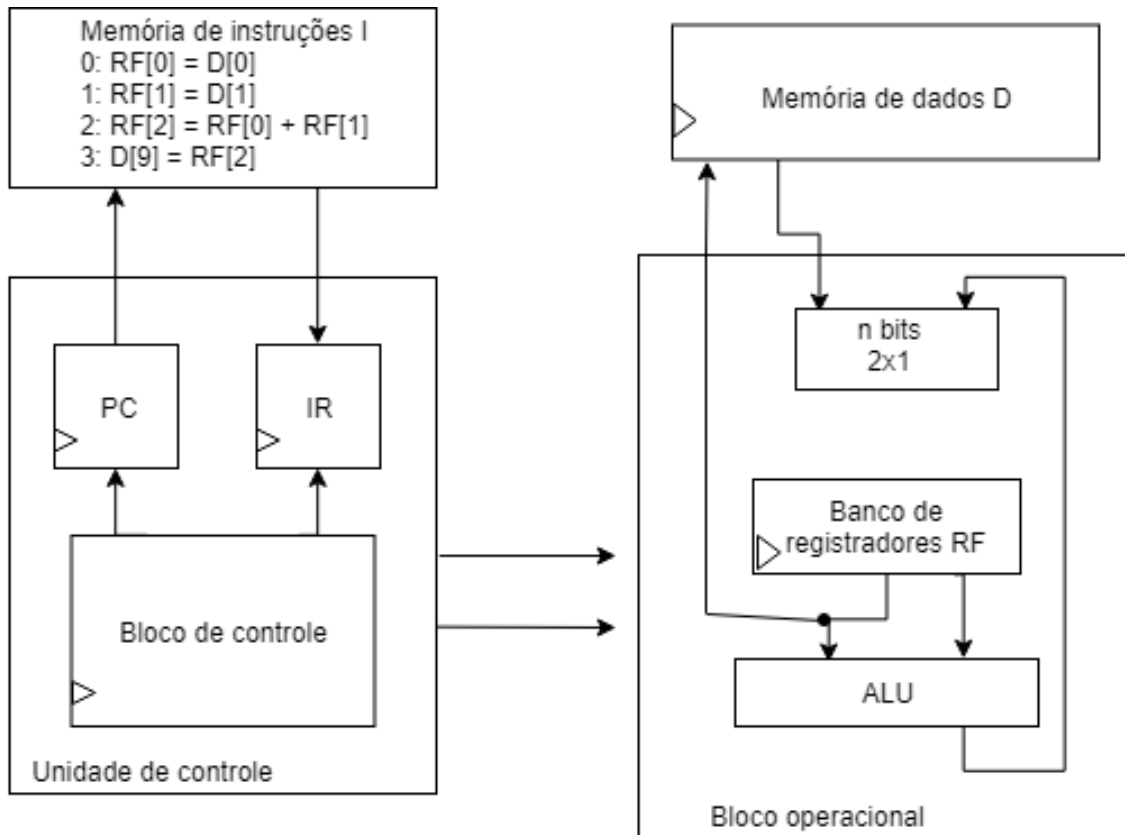
Figure 1: Tabela de instruções

Instrução	OPCode	Função
MOVA Rd	0000 dd00	Accumulator = Register [dd]
MOVR Rd	0001 dd00	Register [dd] = Accumulator
LOAD Imm	0010 iiii	Accumulator = Immediate
ADD Rd	0011 dd00	Accumulator = Accumulator + Register[dd]
SUB Rd	0100 dd00	Accumulator = Accumulator - Register[dd]
ANDR Rd	0101 dd00	Accumulator = Accumulator AND Register[dd]
ORR Rd	0110 dd00	Accumulator = Accumulator OR Register[dd]
JMP Address	0111 0000 aaaa aaaa	PC = Address[aaaa]
INV	1000 0000	Accumulator = NOT Accumulator
HALT	1001 1111	Stop execution

1.1 Funcionamento do processador

O processador tem nove linhas de instruções(Não está implementado o JMP). As linhas de instruções seguem o padrão pedido na tabela1. Todo o sistema do funcionamento de um processador programável está mostrado na figura 2 abaixo, junto com um exemplo de linhas de instrução no box "Memória de instruções"

Figure 2: Unidade de controle



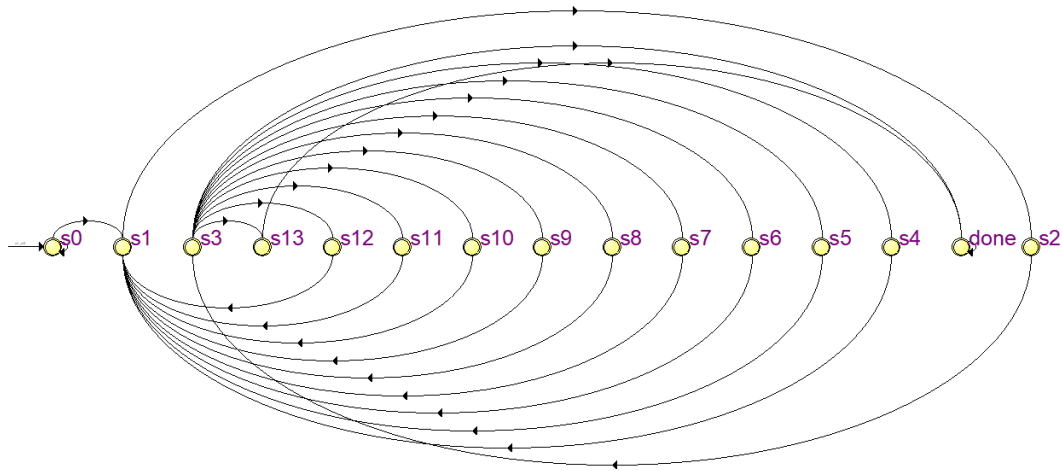
1.1.1 Controlador

O controlador possui quatorze máquinas de estado³ O funcionamento das máquinas de estado estão descritas abaixo:

- S0, Estado inicial, inicialização do processador
- S1, Busca as instruções
- S2, Incrementa a linha de instrução
- S3, Procura o comando equivalente aquela linha de instrução
- S4, Instrução "MOVA"
- S5, Instrução "MOVR"
- S6, Instrução "LOAD"
- S7, Instrução "ADD"
- S8, Instrução "SUB"
- S9, Instrução "ANDR"
- S10, Instrução "ORR"

- S11, Instrução "JMP"
- S12, Instrução "INV"
- S13, Instrução "HALT"
- S4, DONE

Figure 3: FSM



2 Resultados

Nessa sessão estão os resultados simulados de algumas das instruções em formato de Waveform. Nos exemplos da 4 temos o mesmo padrão de como uma instrução é chamada e um resultado é mostrado¹:

Primeiro é mostrado o valor inicial do registrador (sempre começando com 0), o segundo valor mostrado é o novo valor que o registrador recebeu, após o acumulador ser carregado e o registrador receber o acumulador (MOVR), o terceiro valor sempre é 0, o quarto valor é o atual do registrador, o quinto valor é uma constante carregada no acumulador (LOAD Imm), e por último, o sexto valor é o resultado da instrução.

¹ Para o resultado de Figura 4(c) Não tem o uso de registrador, então são apenas 4 valores, sendo o segundo o acumulador e o quarto o resultado

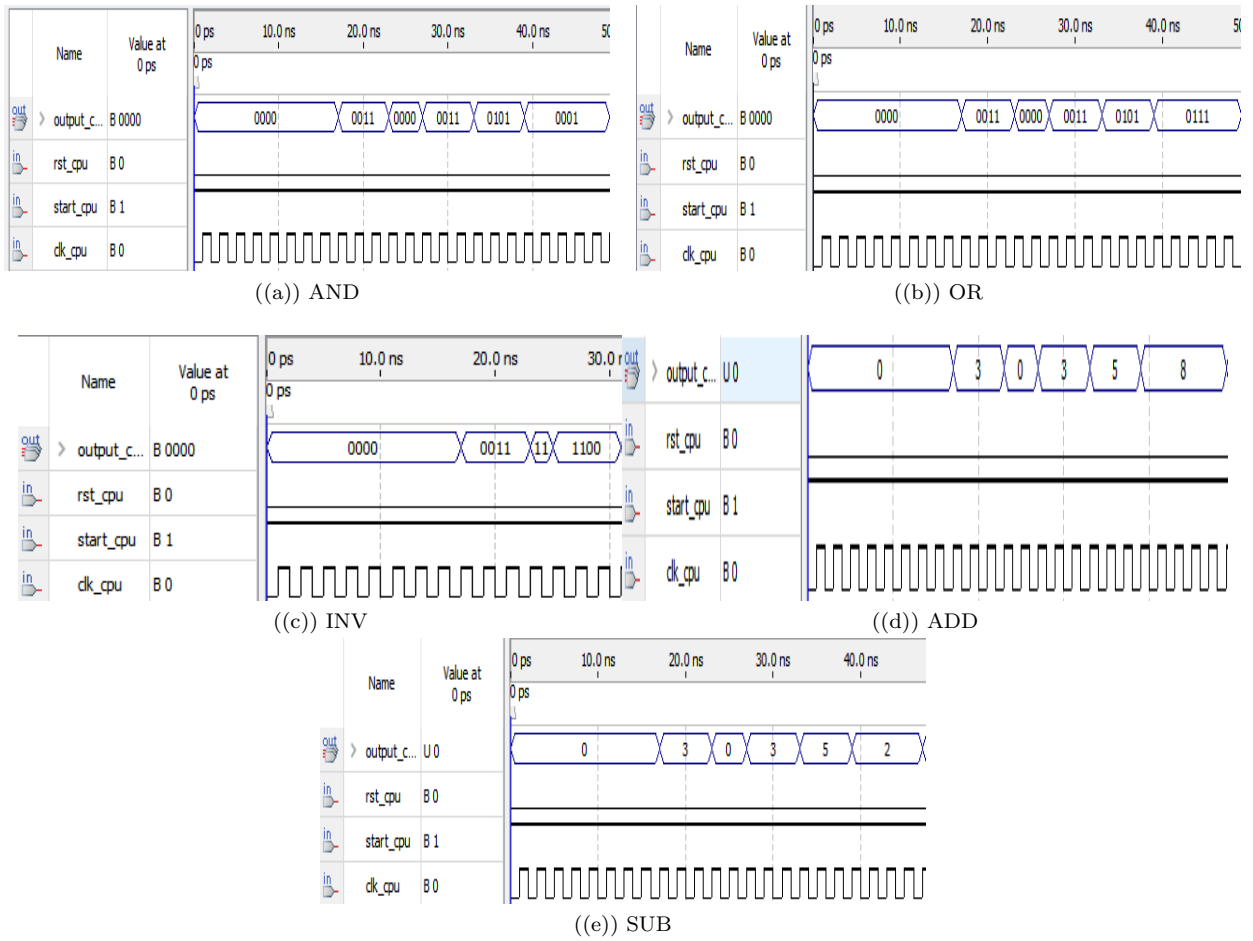


Figure 4: Exemplos de instruções funcionando

- Na imagem 4(a) podemos ver o funcionamento da instrução AND, a operação nessa situação é dos valores '0011' para o registrador e '0101' para o acumulador, resultando no valor esperado de '0001' ;
- Na imagem 4(b) podemos ver o funcionamento da instrução OR, a operação nessa situação é dos valores '0011' para o registrador e '0101' para o acumulador, resultando no valor esperado de '0111' ;
- Na imagem 4(c) podemos ver o funcionamento da instrução INV, a operação nessa situação é do valor '0011' para o acumulador, resultando no valor esperado de '1100' ;
- Na imagem 4(d) podemos ver o funcionamento da instrução ADD, a operação nessa situação é dos valores '3' para o registrador e '5' para o acumulador, resultando no valor esperado de '8' ;
- Na imagem 4(e) podemos ver o funcionamento da instrução SUB, a operação nessa situação é dos valores '3' para o registrador e '5' para o acumulador, resultando no valor esperado de '2' ;

3 Conclusão

Nesse trabalho foi feito um processador de nove instruções. As instruções foram testadas em simulador e na placa Cyclone II. Testes foram feitos usando várias instruções ao mesmo tempo, e os resultados obtidos estão dentro do previsto.