Implementação de um Processador de 8 bits - Logisim

Step-by-Step Design and Simulation of a Simple CPU Architecture

Allex Lima Paulo Moraes Renan Barroncas

Bacharelado em Engenharia da Computação
Escola de Exatas
Centro Universitário do Norte - UniNorte Laureate

Organização e Arquitetura de Computadores, 2016

Sumário

Uma breve introdução

Derek C. Schuurman, Ph.D., P.Eng.
Professor of Computer Science
Redeemer University College
Phone: (905) 648-2131 ext. 4273
Office: 221H
e-mail: dschuurman (at) cs.redeemer.ca
twitter: @DerekSchuurman

Computer science curriculum, 2008

"A professional in any field of computing should not regard the computer as just a black box that executes programs by magic ... Students need to understand computer architecture in order to make best use of the software tools and computer languages they use to create programs"

Sumário

Modelo Teórico

Diagrama de Blocos

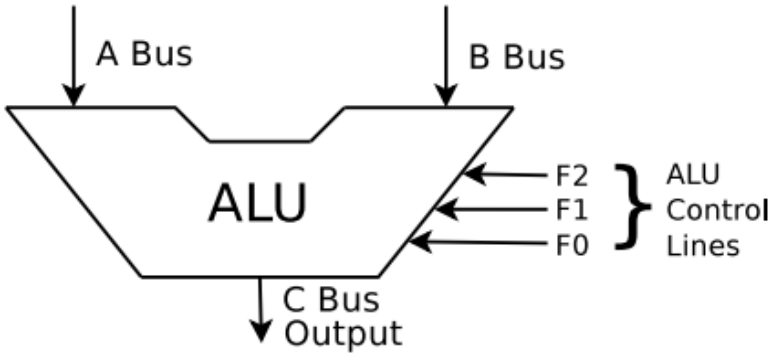


Tabela Verdade

F2	F1	F0	Output
0	0	0	A
0	0	1	B
0	1	0	$A + 1$
0	1	1	$B + 1$
1	0	0	$A + B$
1	0	1	$A - B$
1	1	0	A *AND* B
1	1	1	A *OR* B

Implementação no *Logisim*

Diagrama de Blocos

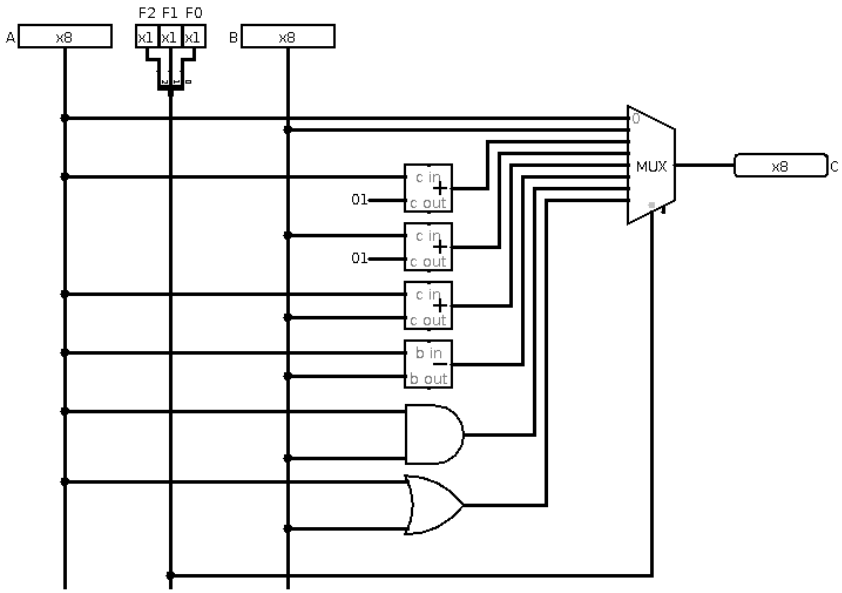
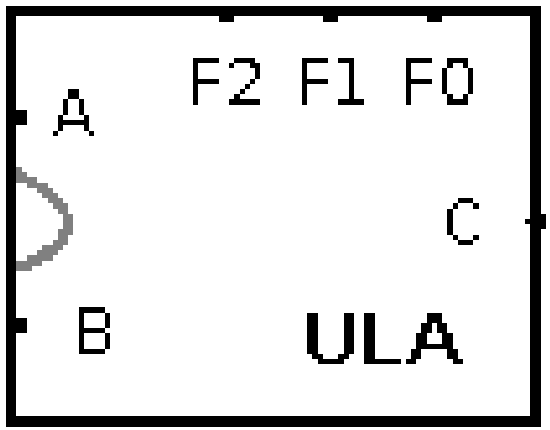
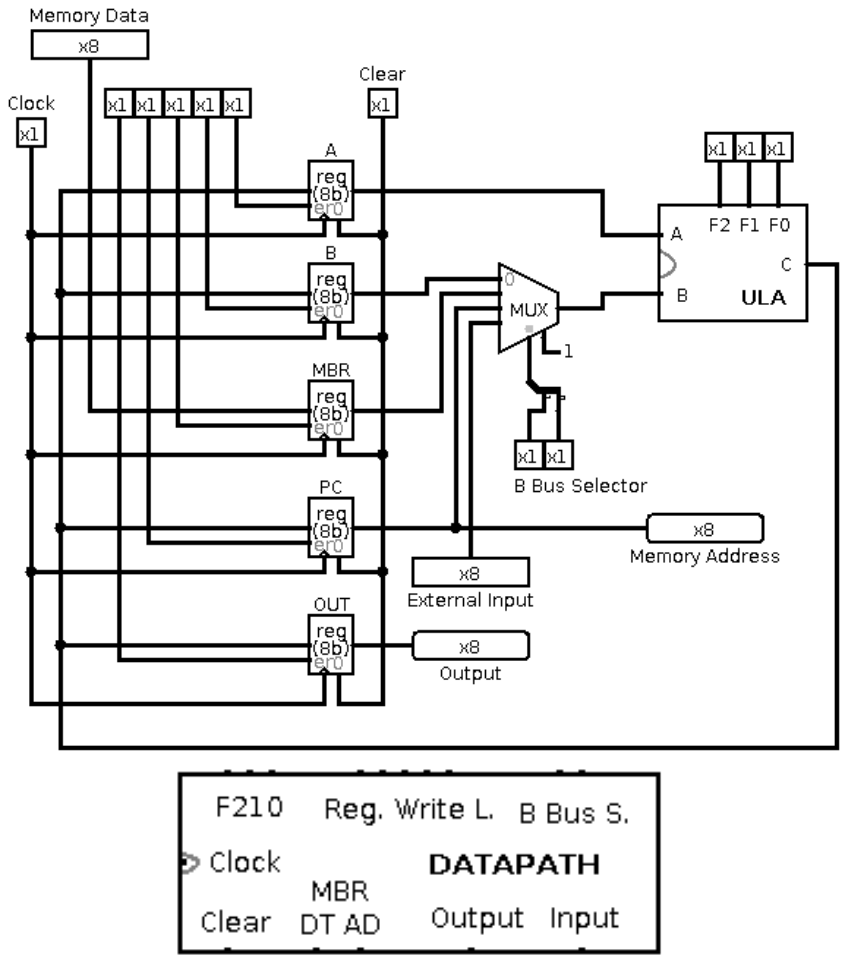


Diagrama de Blocos



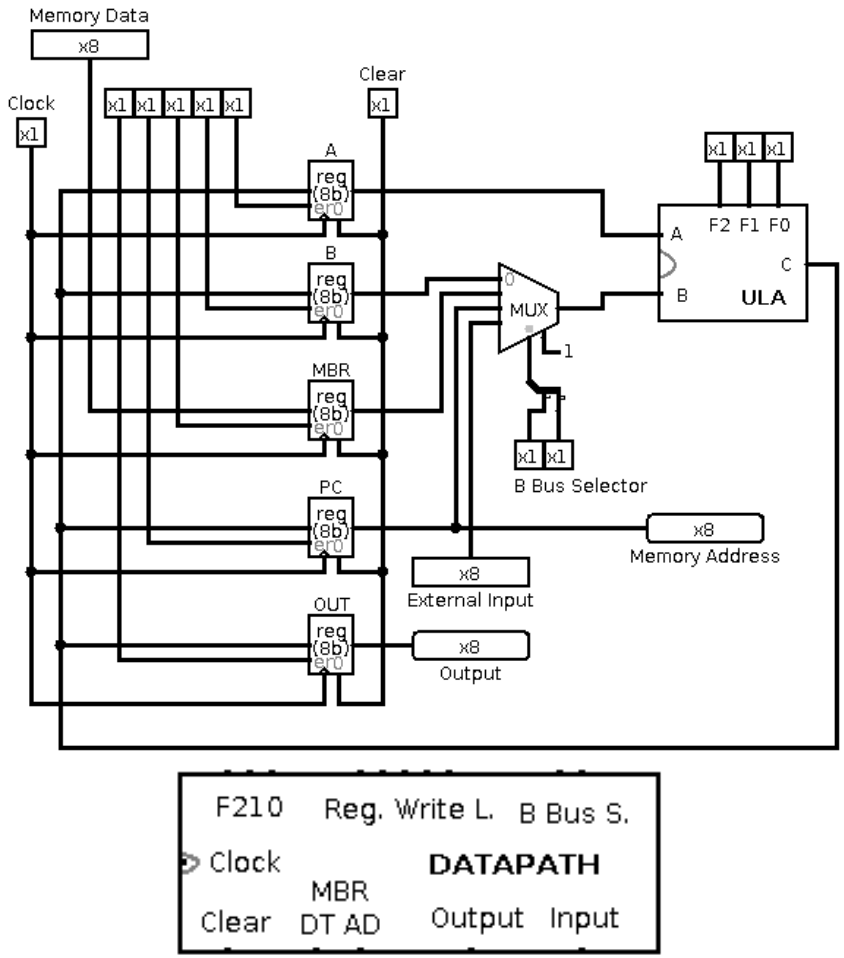
Sumário

O modelo utilizado



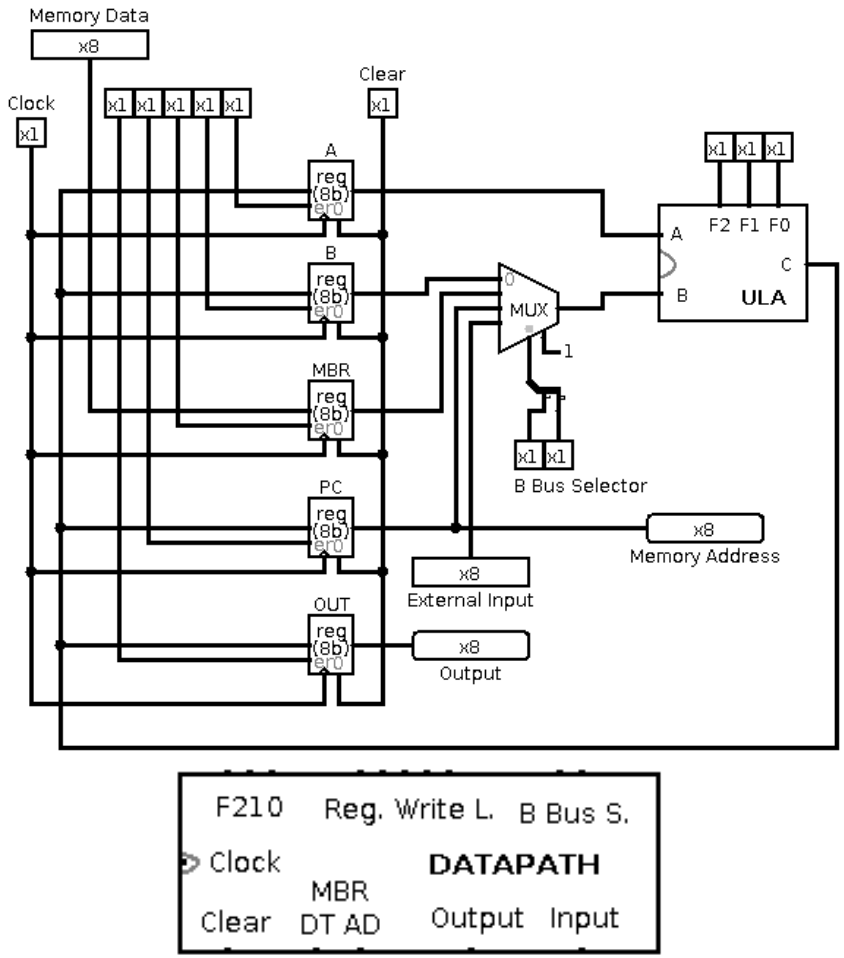
- A ULA sempre terá o registrador A associado a uma de suas entradas;
- A outra entrada da ULA possui como origem:
  - O registrador B;
  - O MBR;
  - O PC; ou
  - Uma fonte externa.
- A saída da ULA terá como destino o registrador que estiver habilitado a receber dados (write ativo).

O modelo utilizado



- A ULA sempre terá o registrador A associado a uma de suas entradas;
- A outra entrada da ULA possui como origem:
  - O registrador B;
  - O MBR;
  - O PC; ou
  - Uma fonte externa.
- A saída da ULA terá como destino o registrador que estiver habilitado a receber dados (write ativo).

O modelo utilizado



- A ULA sempre terá o registrador A associado a uma de suas entradas;
- A outra entrada da ULA possui como origem:
  - O registrador B;
  - O MBR;
  - O PC; ou
  - Uma fonte externa.
- A saída da ULA terá como destino o registrador que estiver habilitado a receber dados (write ativo).

Sumário

A Unidade de Controle

- Uma ROM (*Read Only Memory*) de 32x24-bits é utilizada para armazenar o *microprogama*;
- O registrador MPC contém os 5 bits do *endereço* da instrução;
- O seletor JMPC irá encaminhar para o MPC um endereço do MBR ou da próxima instrução;
- O MIR armazena os 16 bits da *instrução em si*;
- O MPC e o MIR são atualizados a cada *clock*;

A Unidade de Controle

- Uma ROM (*Read Only Memory*) de 32x24-bits é utilizada para armazenar o *microprogama*;
- O registrador MPC contém os 5 bits do *endereço* da instrução;
- O seletor JMPC irá encaminhar para o MPC um endereço do MBR ou da próxima instrução;
- O MIR armazena os 16 bits da *instrução em si*;
- O MPC e o MIR são atualizados a cada *clock*;

A Unidade de Controle

- Uma ROM (*Read Only Memory*) de 32x24-bits é utilizada para armazenar o *microprogama*;
- O registrador MPC contém os 5 bits do *endereço* da instrução;
- O seletor JMPC irá encaminhar para o MPC um endereço do MBR ou da próxima instrução;
- O MIR armazena os 16 bits da *instrução em si*;
- O MPC e o MIR são atualizados a cada *clock*;

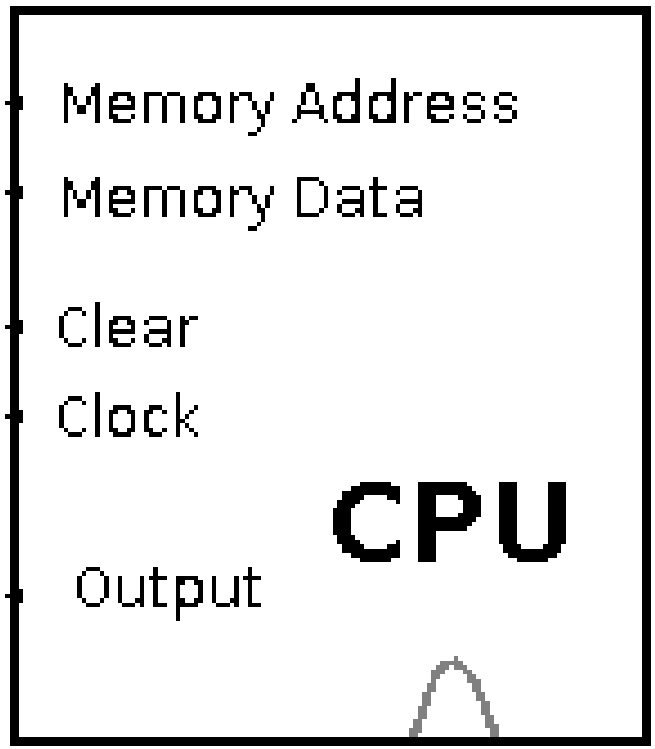
A Unidade de Controle

- Uma ROM (*Read Only Memory*) de 32x24-bits é utilizada para armazenar o *microprogama*;
- O registrador MPC contém os 5 bits do *endereço* da instrução;
- O seletor JMPC irá encaminhar para o MPC um endereço do MBR ou da próxima instrução;
- O MIR armazena os 16 bits da *instrução em si*;
- O MPC e o MIR são atualizados a cada *clock*;

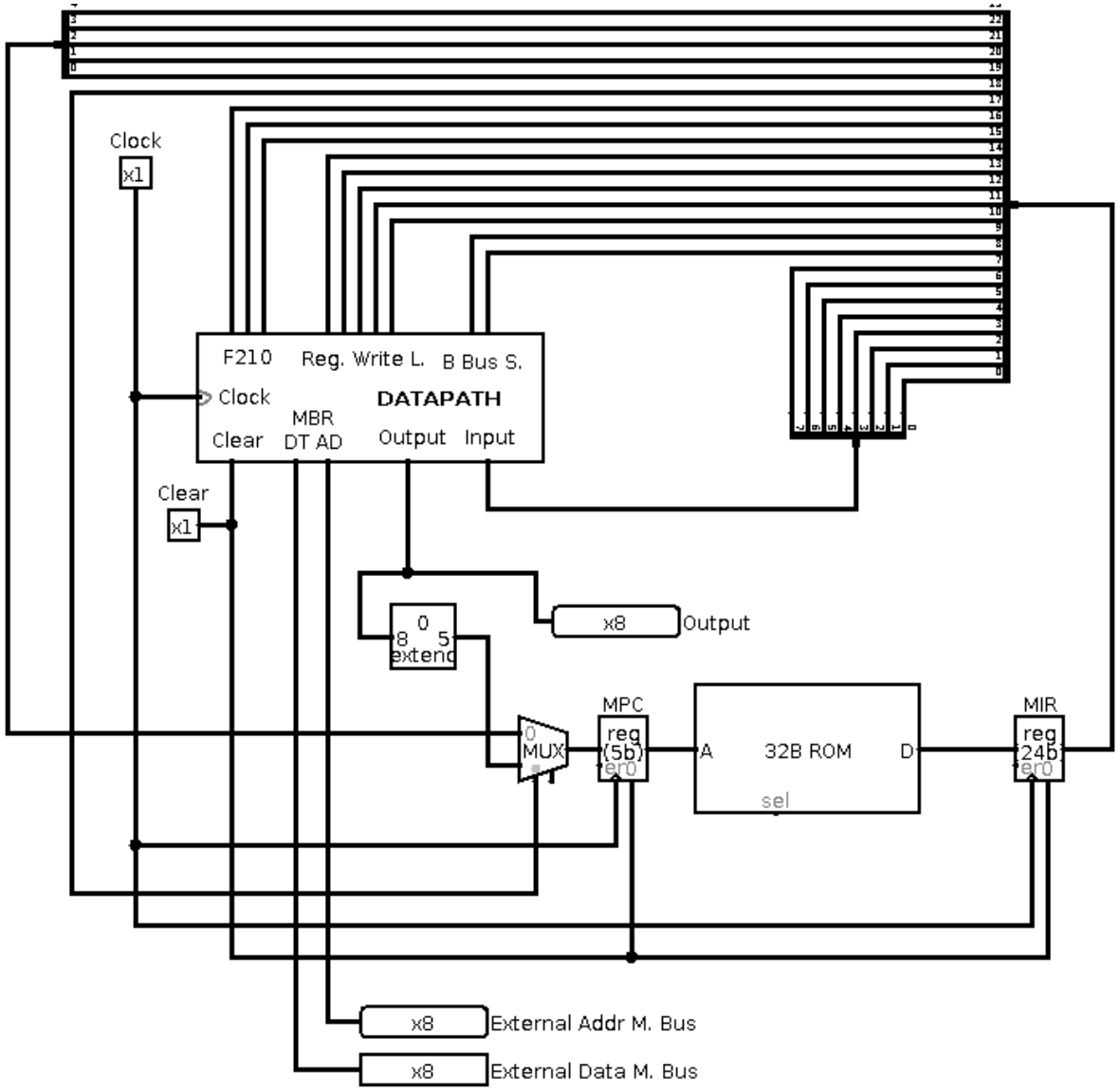
A Unidade de Controle

- Uma ROM (*Read Only Memory*) de 32x24-bits é utilizada para armazenar o *microprogama*;
- O registrador MPC contém os 5 bits do *endereço* da instrução;
- O seletor JMPC irá encaminhar para o MPC um endereço do MBR ou da próxima instrução;
- O MIR armazena os 16 bits da *instrução em si*;
- O MPC e o MIR são atualizados a cada *clock*;

A Unidade de Controle

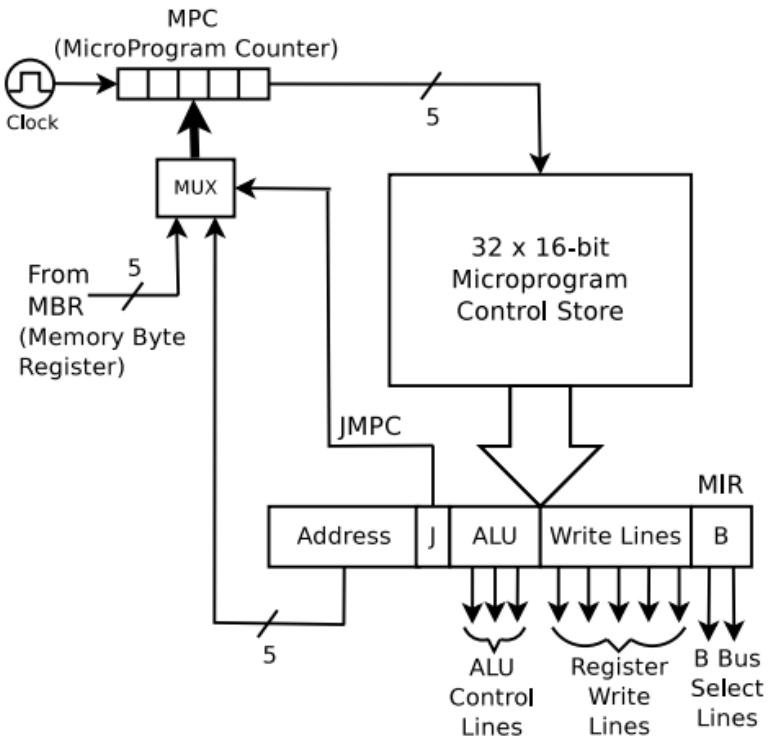


A Unidade de Controle



A Unidade de Controle

- Os bits que encontram-se no MIR poussem a organização:
  - Endereço: 5 bits;
  - Seletor JMPC (J): 1 bit;
  - Controladores da ULA: 3 bits;
  - Controle de escrita: 5 bits; e
  - Seletor da entrada B da ULA: 2 bits.



A Unidade de Controle

Instructions Supported		
Assembly	**Binary**	**Hexadecimal**
input .r, <value>	##### 0 001 000RR 11 VVVVVVVV	
print .r	##### 0 00L 10000 00 00000000	
add .r0, .r1	##### 0 100 000RR 00 00000000	
sub .r0, .r1	##### 0 101 000RR 00 00000000	
mov .r0, .r1	##### 0 00L 000RR 00 00000000	
inc .r	##### 0 0IC 000RR 00 00000000	
and	##### 0 110 10000 00 00000000	
or	##### 0 111 10000 00 00000000	
jump #<add>	##### 1 001 10000 11 AAAAAAAA	
load .r, #<add>	##### 0 001 01000 11 AAAAAAAA	
	##### 0 001 000RR 01 00000000	
end	##### 0 000 00000 00 00000000	

Sumário

Combinando os processos

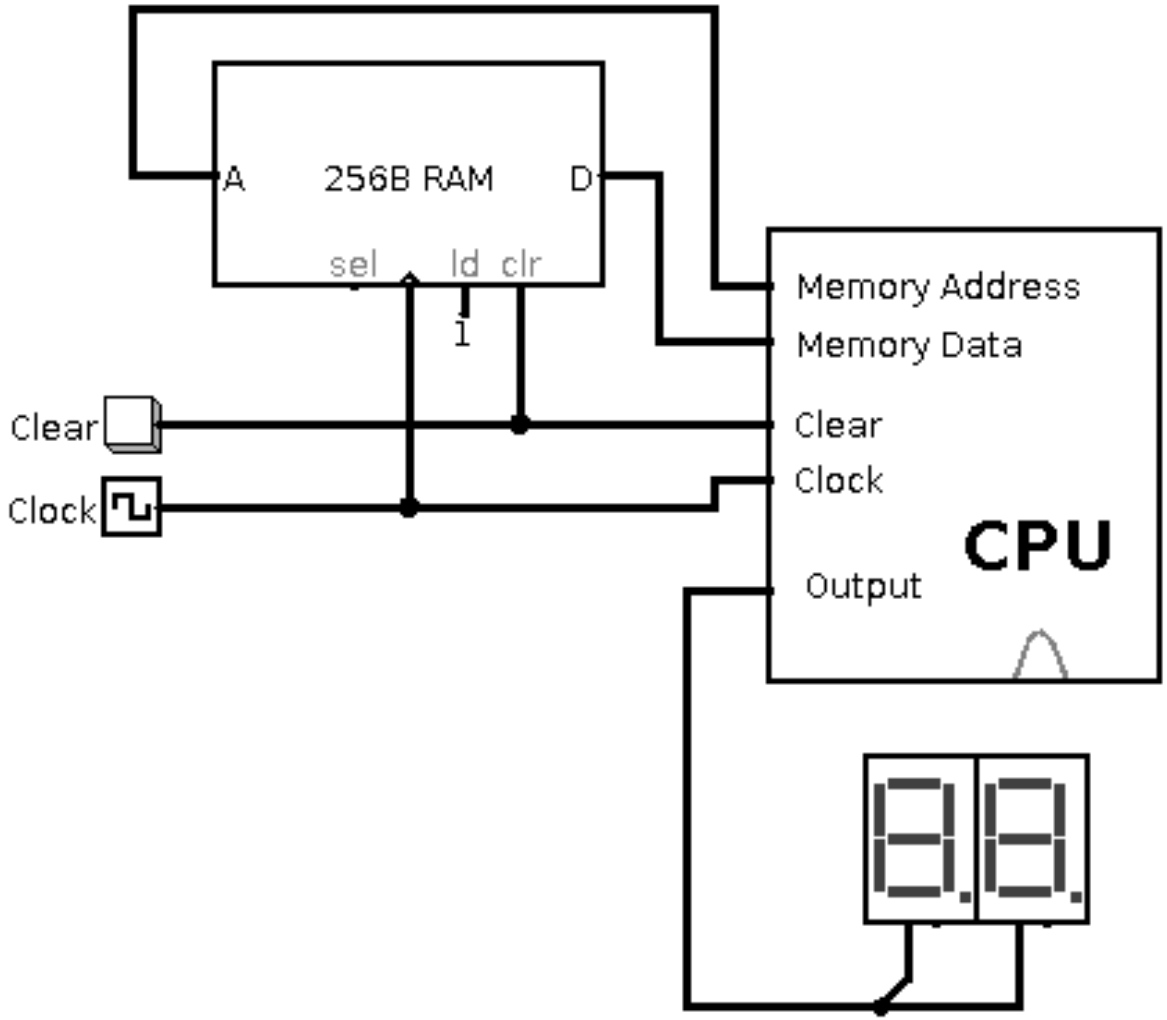


Figure: Resultado

Instruções - Testes

TESTES

Referências

Derek C. Schuurman.
Step-by-step design and simulation of a simple CPU architecture
Proceeding of the 44th ACM technical symposium on Computer science education (SIGCSE '13), ACM, New York, NY, USA, 335-340. 2013.