

Universidade Federal de Roraima – DCC
Arquitetura e Organização de Computadores
Professor: Herbert Oliveira Rocha



UFRR



Aluna: Nataly Almeida

Formato das Instruções

O processador LP possui 16 registradores, nomeados: \$zero, \$highReg, \$lowReg, \$compareReg, \$SO, \$S1, \$S2, \$S3, \$S4, \$S5, \$S6, \$S7, \$TO, \$T1, \$T2 e \$T3.

1

5 bits	4 bits	4 bits	3 bits
Opcode 15-11	Registrador 1 10-7	Registrador 2 6-3	Funct 2-0

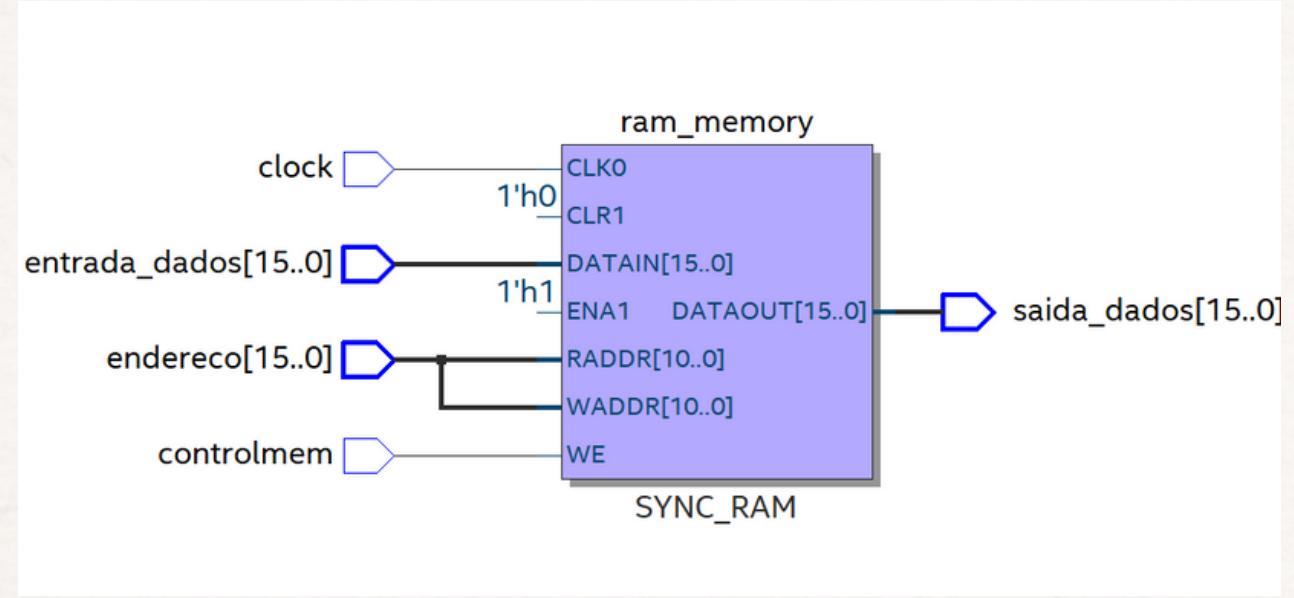
2

5 bits	11 bits
Opcode 15-11	Endereço de Destino 10-0

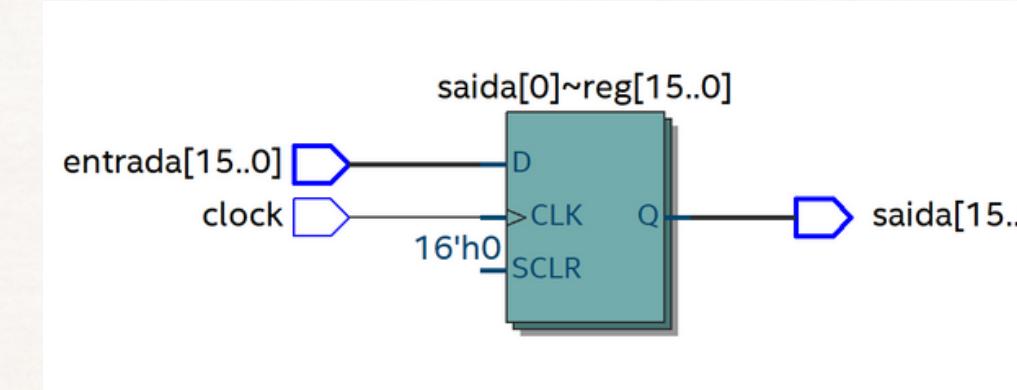
3

Este formato aborda instruções baseadas carregamentos imediatos na memória.

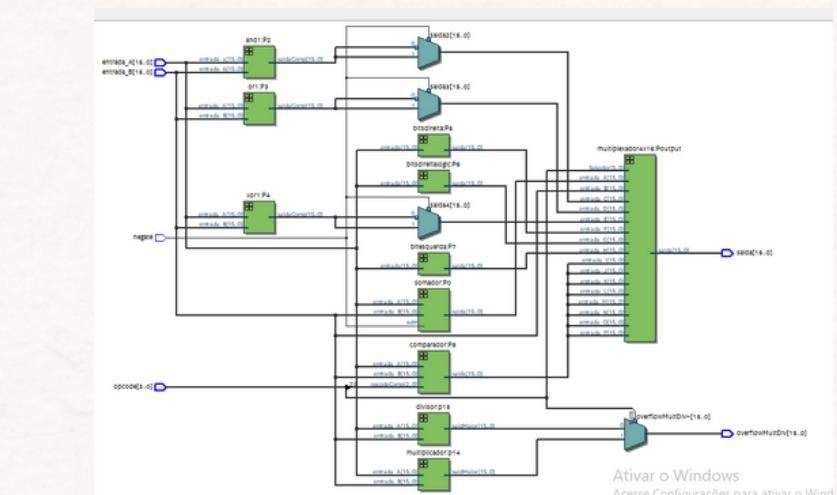
Memória RAM e ROM



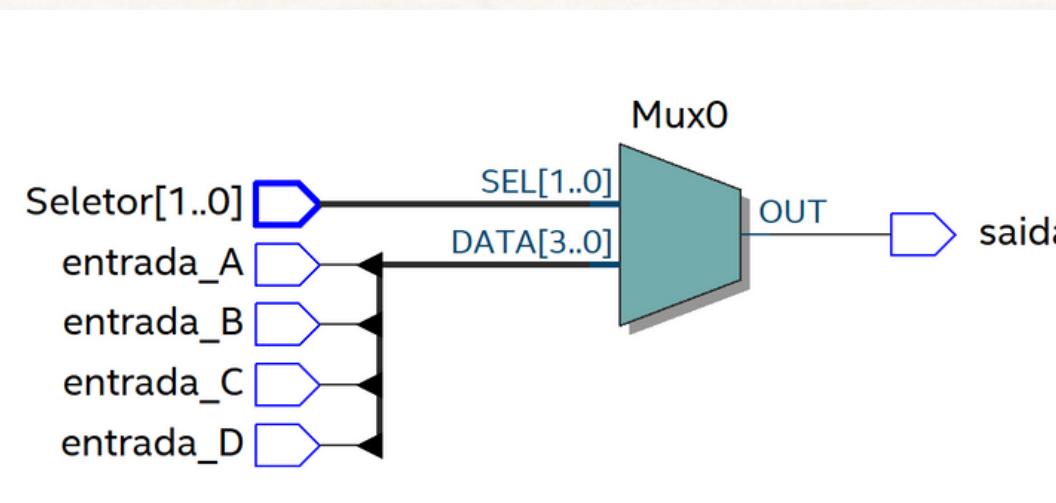
PC



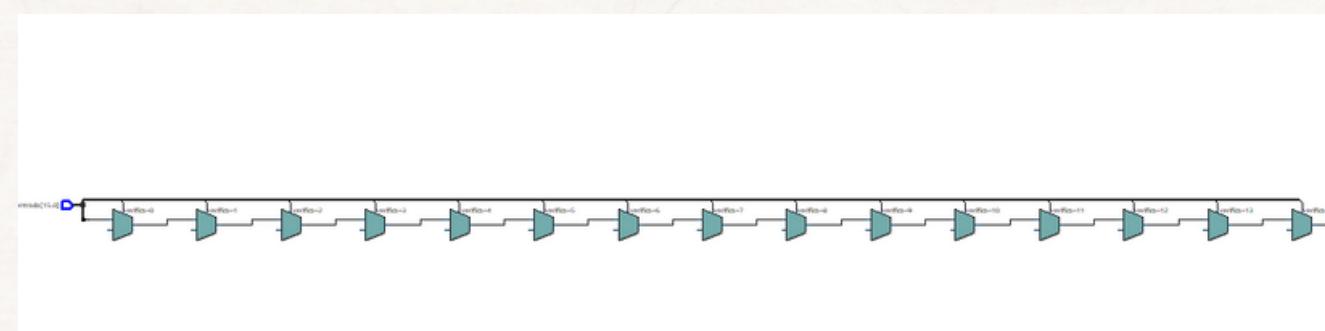
ULA



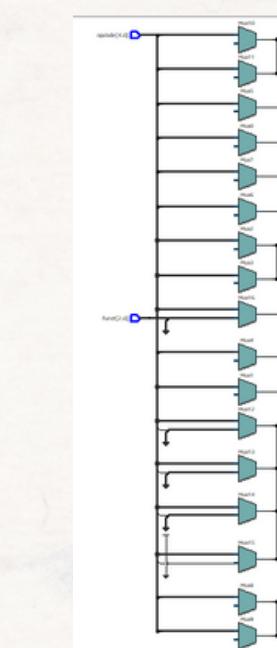
Multiplexadores



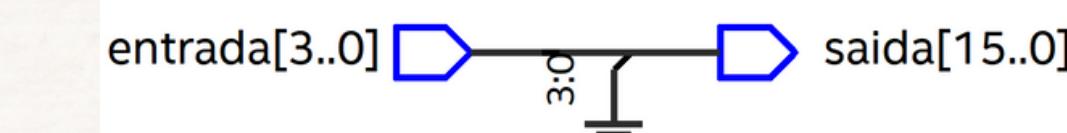
Encurtador



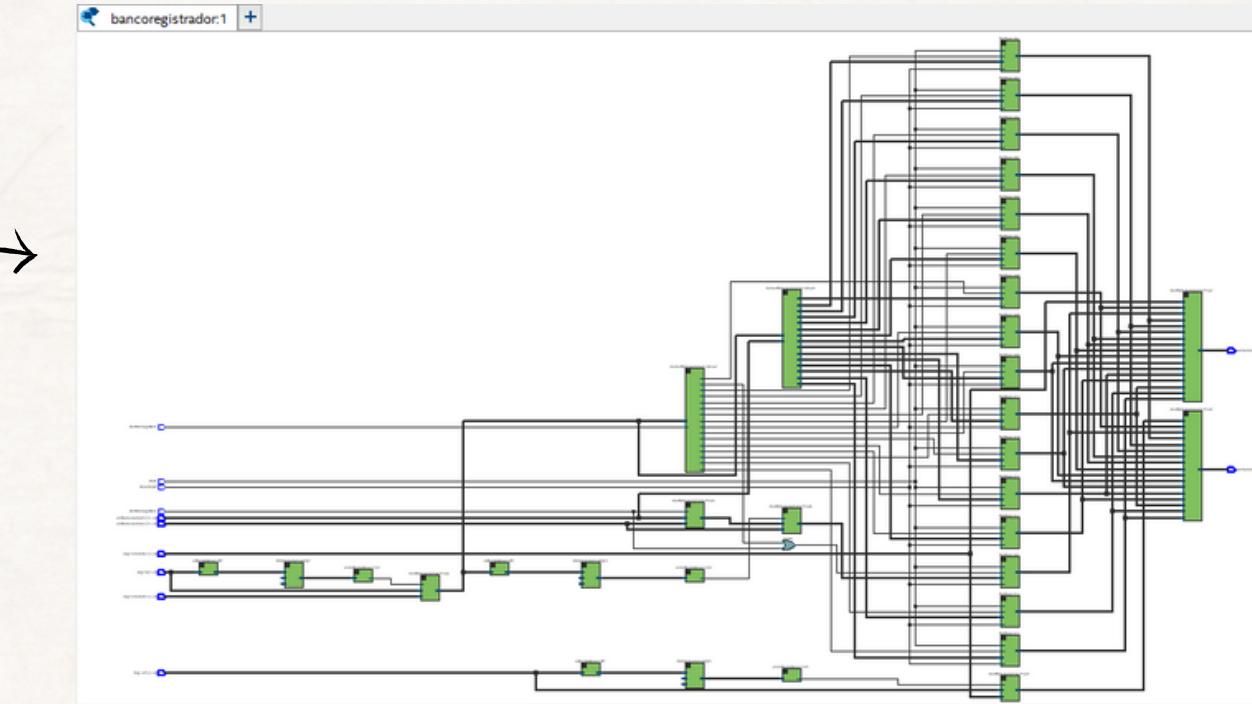
UC



Banco de Registradores



Extensores

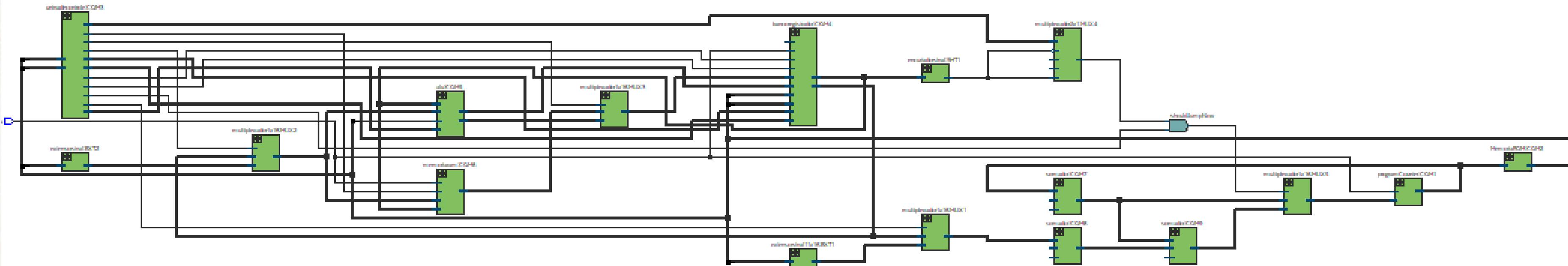


Opcodes

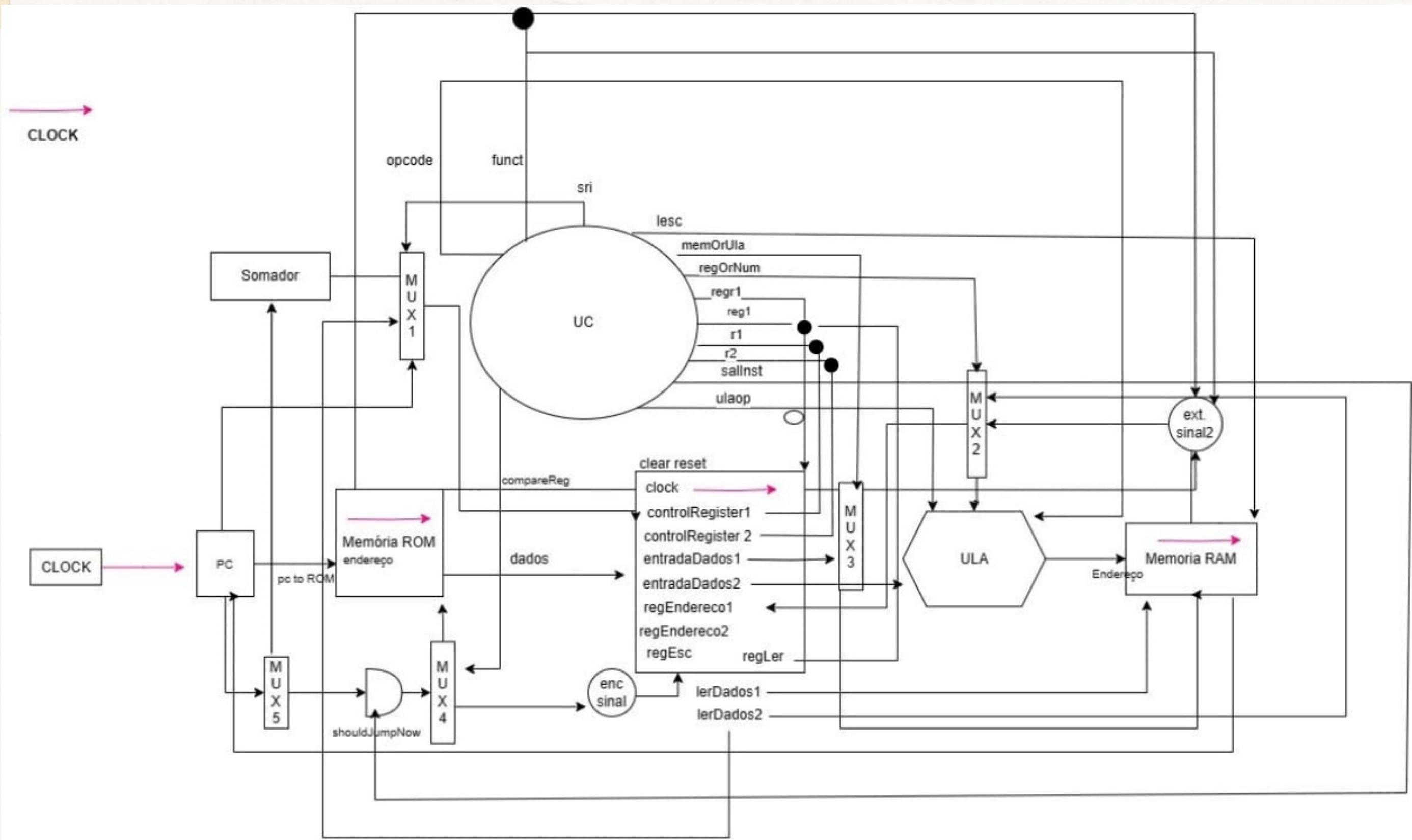
- São disponibilizados 5 bits para Opcode
- 27 Opcodes no total
- Incluindo os imediatos, comparadores, saltos e move

RTL Viewer

Processador
LogPose



Datapath



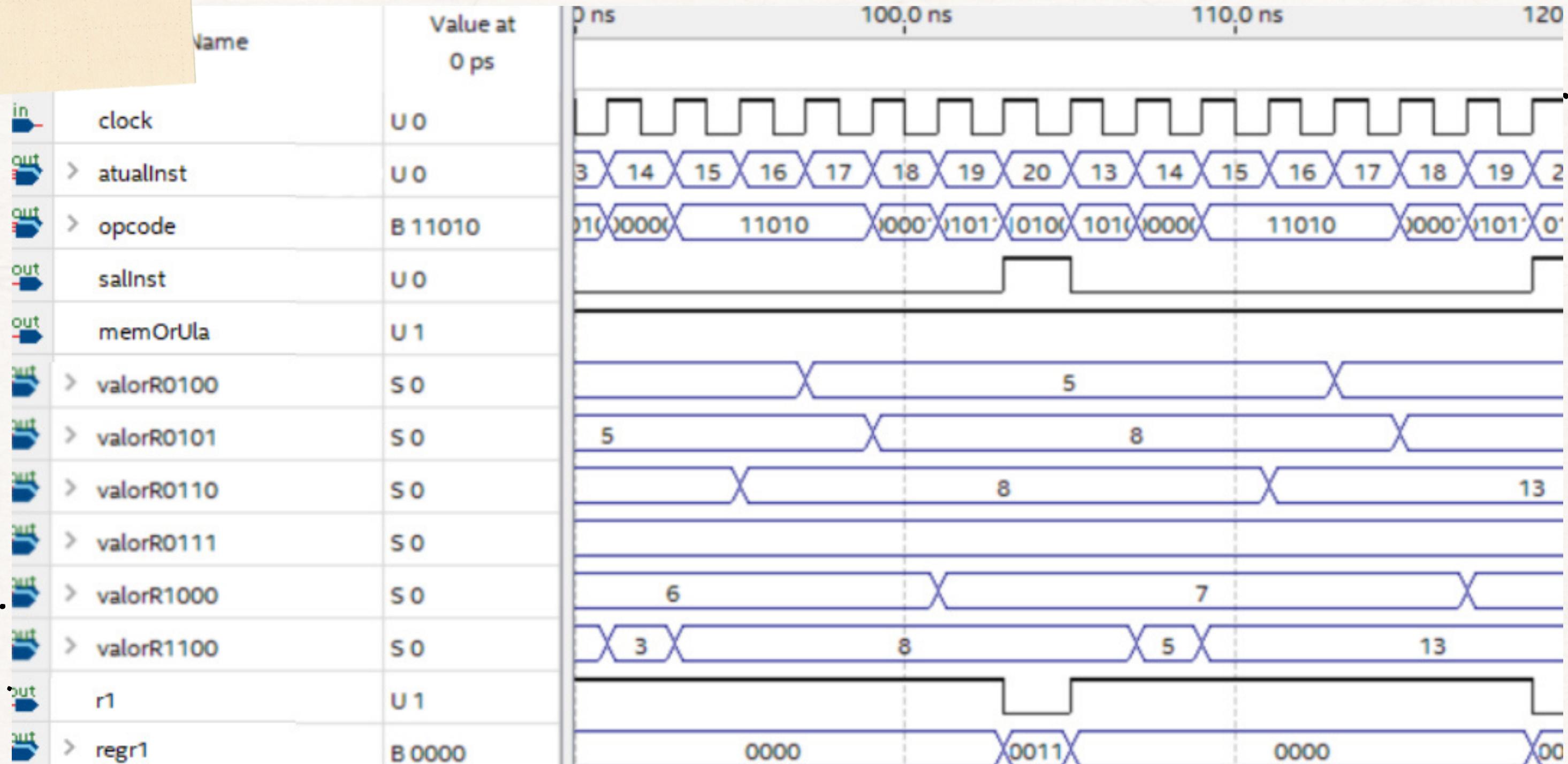
Fibonacci

Código detalhado para realização do Fibonacci

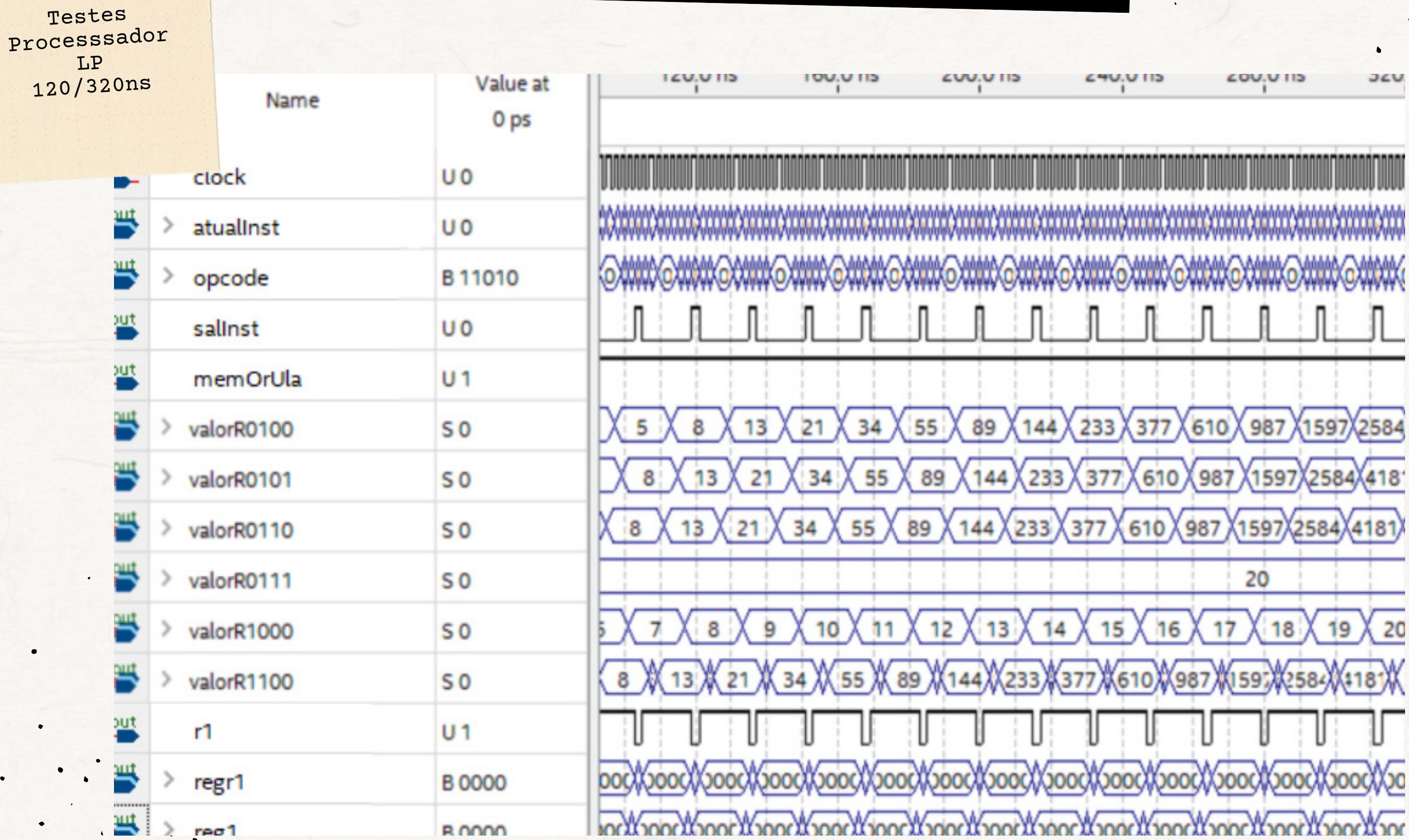
ENDEREÇO	SINTAXE	OPCODE	R1	R2	Instrução
01	main : movi \$s0 , 0;	11010	0100	0000	1101001000000001
02	movi \$s1 , 1;	11010	0101	0001	1101001010001001
03	movi \$s3 , 15;	11010	0111	1111	110100111111001
04	addi \$s3 , 5;	00001	0111	0101	0000101110101000
05	movi \$s4 , 2;	11010	1000	0010	110101000010001
06	smeq \$s3 , \$zero ;	01011	0111	0000	0101101110000000
07	jimbt Endg ;	10100	00000001111	1010000000001111	
08	move \$s2 , \$s0 ;	11010	0110	0100	1101001100100000

Waveform

Testes
Processador
90/120ns



Waveform



Algoritmo do Bubble Sort

ENDEREÇO SINTAXE

```
02  prt1 : movi $t1 , 0;  
03  lpn1 : equ $t1 , $s1 ;  
04  jimb t bck1 ;  
05  move $t3 , $s0 ;  
06  add $t3 , $t1 ;  
07  ldr $s7 , $t3 ;  
08  addi $t1 , 1;  
09  jim lpn1 ;  
10  prt2 : movi $t1 , 0;  
11  lpn2 : equ $t1 , $s1 ;  
12  jimb t bck2 ;  
13  move $t3 , $s0 ;  
14  add $t3 , $t1 ;  
15  ldr $s7 , $t3 ;  
16  addi $t4 , 1;  
17  jim lpn2 ;
```

OPCODE / R1 /R2

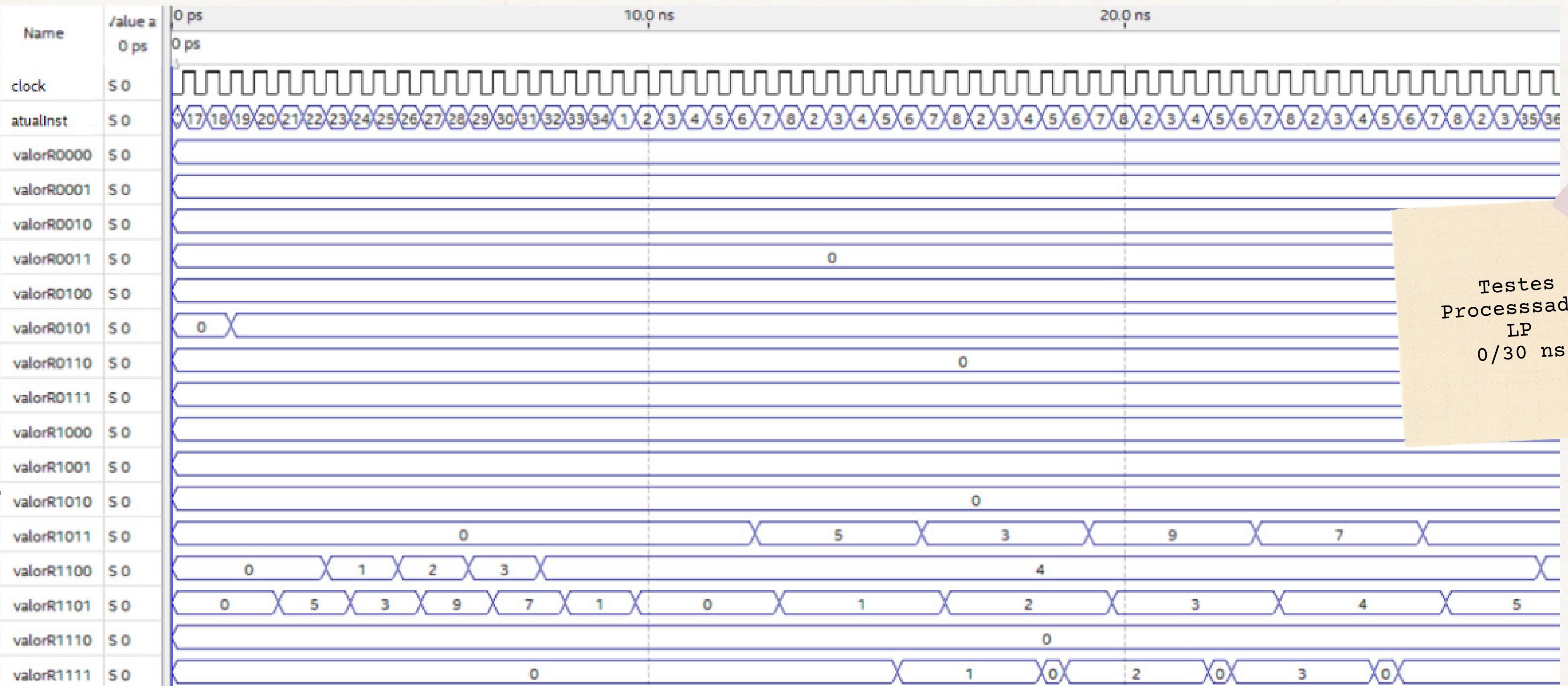
```
11010 / 1101 / 0000  
01000 / 1101 / 0101  
10100 / 00000100000  
11010 / 1111 / 0100  
00000 / 1111 / 1101  
10110 / 1011 / 1111  
00001 / 1101 / 0001  
10001 / 1111111010  
11010 / 1101 / 0000  
01000 / 1101 / 0101  
10100 / 0000010110  
11010 / 1111 / 0100  
00000 / 1111 / 1101  
10110 / 1011 / 1111  
00001 / 1101 / 0001  
10001 / 1111111010
```

Instrução

```
1101011010000001  
0100011010101000  
10100000000100000  
1101011101000000  
000001111101000  
1011010111110000  
0000111010001000  
100011111111010  
1101011010000001  
0100011010101000  
1010000000010110  
1101011101000000  
000001111101000  
1011010111111000  
0000111010001000  
100011111111010 10
```

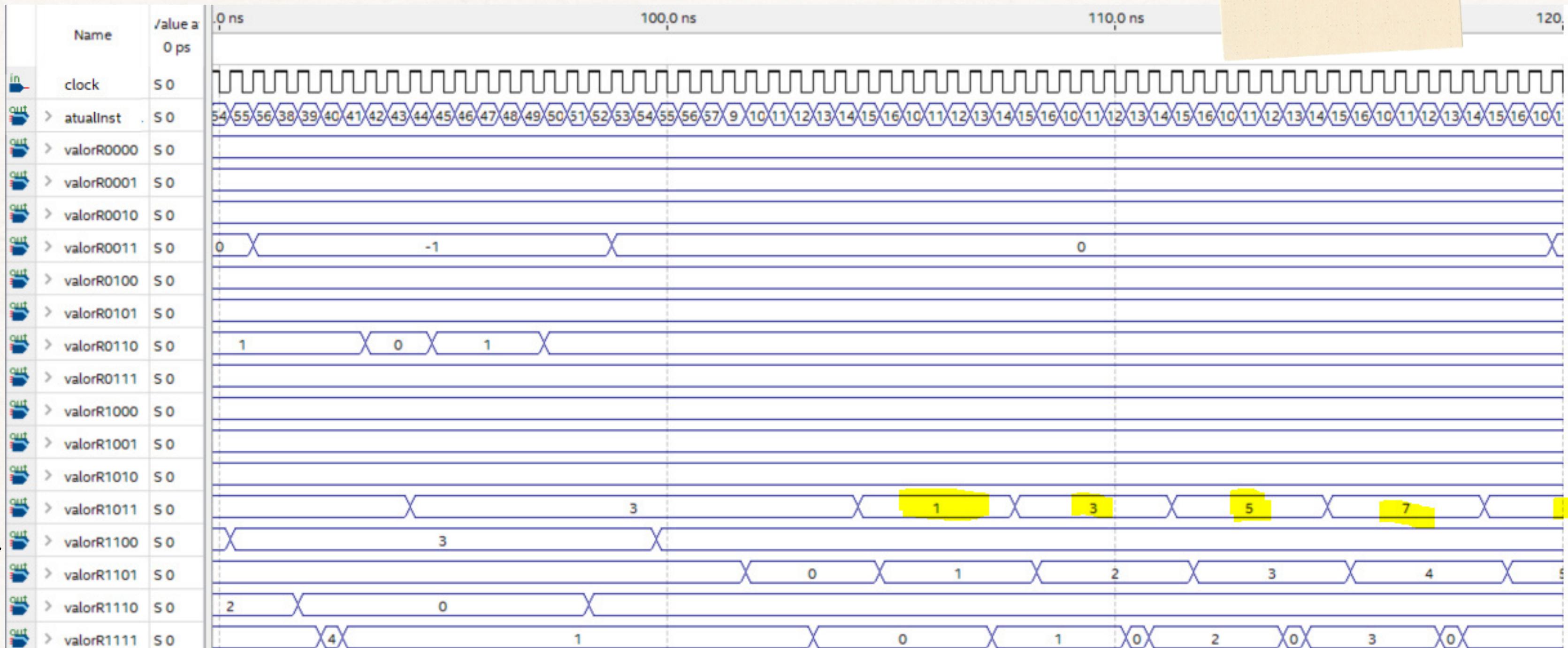
ENDEREÇO	SINTAXE	OPCODE / R1 /R2	Instrução
18	main : movi \$s0 , 0;	11010 / 0100 / 0000	1101001000000001
19	movi \$s1 , 5;	11010 / 0101 / 0101	1101001010101001
20	move \$t0 , \$s0 ;	11010 / 1100 / 0100	1101011000100000
21	movi \$t1 , 5;	11010 / 1101 / 0101	1101011010101001
22	str \$t1 , \$t0 ;	11000 / 1101 / 1100	1100011011100000
23	addi \$t0 , 1;	00001 / 1100 / 0001	0000111000001000
24	movi \$t1 , 3;	11010 / 1101 / 0011	1101011010011001
25	str \$t1 , \$t0 ;	11000 / 1101 / 1100	1100011011100000
26	addi \$t0 , 1;	00001 / 1100 / 0001	0000111000001000
27	movi \$t1 , 9;	11010 / 1101 / 1001	1101011011001001
28	str \$t1 , \$t0;	11000 / 1101 / 1100	1100011011100000
29	addi \$t0 , 1;	00001 / 1100 / 0001	0000111000001000
30	movi \$t1 , 7;	11010 / 1101 / 0111	1101011010111001
31	str \$t1 , \$t0 ;	11000 / 1101 / 1100	1100011011100000
32	addi \$t0 , 1;	00001 / 1100 / 0001	0000111000001000
33	movi \$t1 , 1;	11010 / 1101 / 0001	1101011010001001
34	str \$t1 , \$t0 ;	11000 / 1101 / 1100	1100011011100000

Waveform



Waveform

Testes
Processsador
LP
90/120ns



Referências

- PATTERSON, D.; HENNESSY, J. L. **Organização e projeto de computadores: a interface hardware/software.** 3^a Edição. São Paulo: Elsevier, 2005, 484 p.