

Processador Dynasty

EDU – 2022007610

MARCUS 2022007001

FILIPPE - 2022010099

Descrição do Processador

- Processador RISC;

- Baseado no MIPS;

- Processador de 16bits;

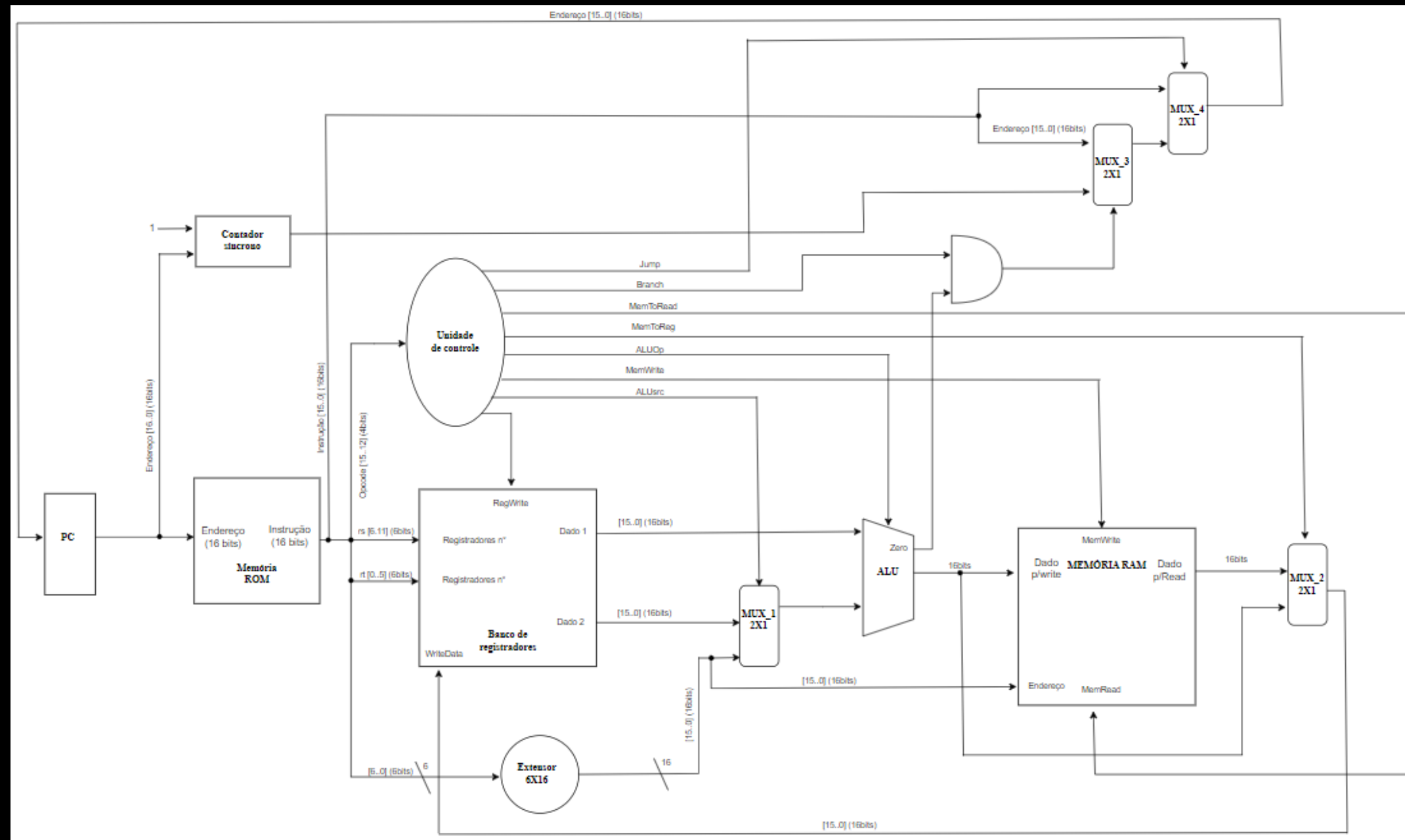
Formatos das Instruções

Instrução do tipo R		
Opcode	rs	rt
4bits	<u>6bits</u>	<u>6bits</u>
15-12	11-6	5-0
Instrução do tipo I		
Opcode	rs	Imediato
4bits	<u>6bits</u>	<u>6bits</u>
15-12	11-6	5-0
Instrução do tipo J		
Opcode	Endereço	
4bits	<u>12bits</u>	
15-12	11-0	

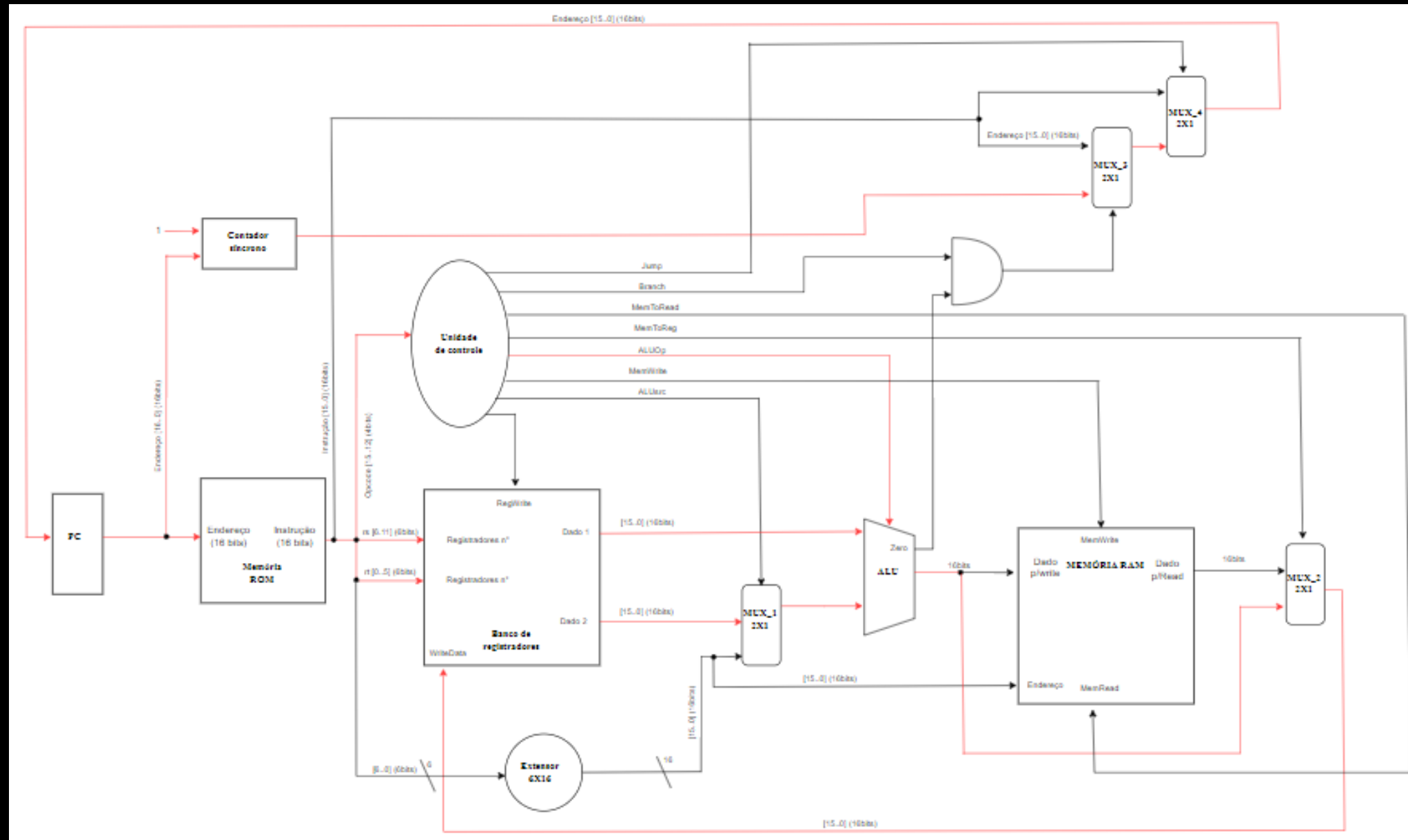
Lista de operações suportadas

Opcode	Sintaxe	Formato	Significado	Exemplo
0000	ADD	R	Soma	<i>add \$s0, \$s1</i>
0001	ADDI	I	Soma imediata	<i>addi \$s0, 1</i>
0010	SUB	R	Subtração	<i>sub \$s0, \$s1</i>
0011	SUBI	I	Subtração imediata	<i>subi \$s0, 3</i>
0100	LW	I	Load	<i>lw \$s0 ram (00)</i>
0101	SW	I	Store	<i>sw \$s0 ram (00)</i>
0110	LI	I	Load imediato	<i>li \$s0 2</i>
0111	BEQ	J	Branch Equal	<i>beq endereço</i>
1000	IF	J	If Equal	<i>if \$s0 \$s1</i>
1001	JUMP	J	Jump	<i>j endereço (0000)</i>

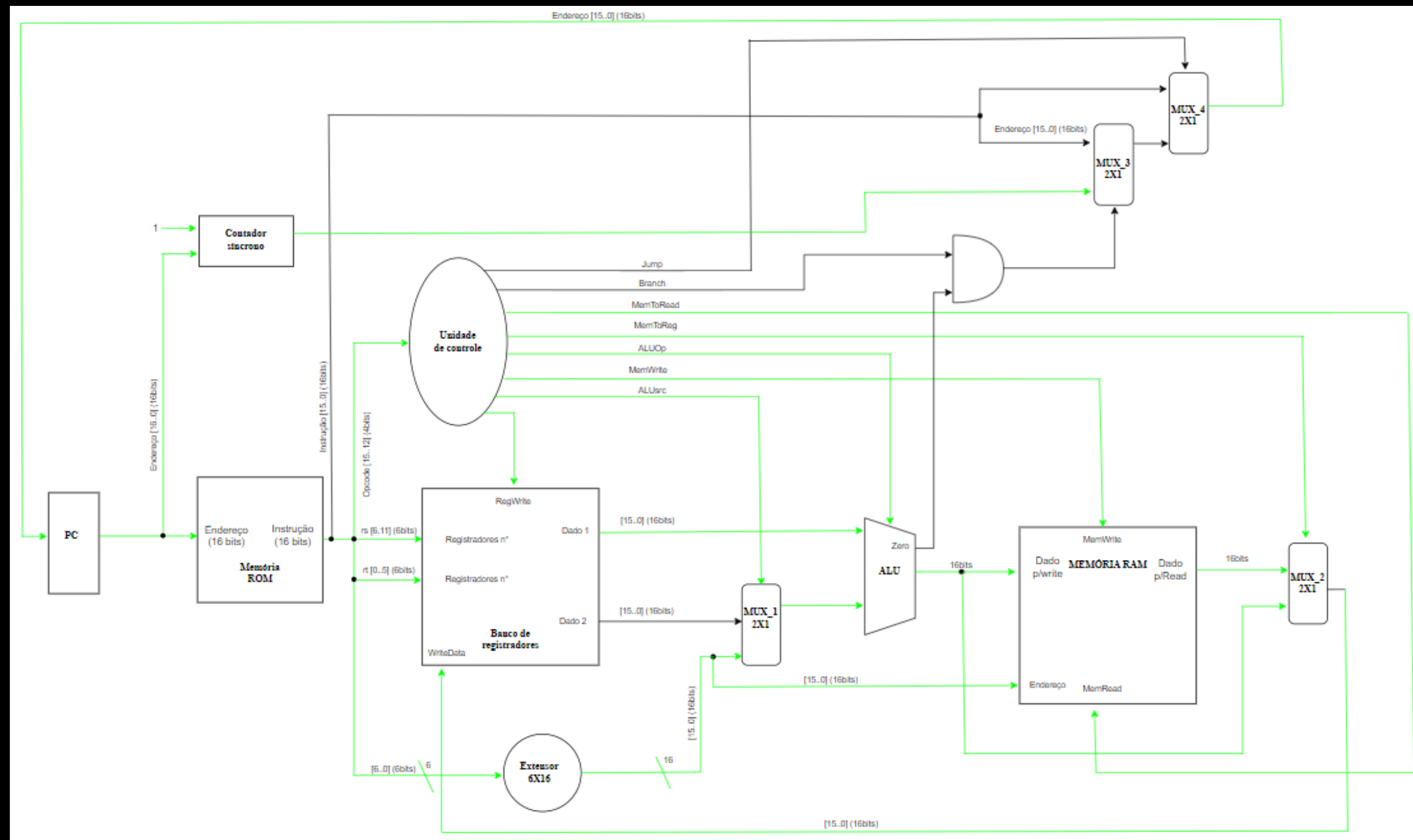
Datapath do Processador



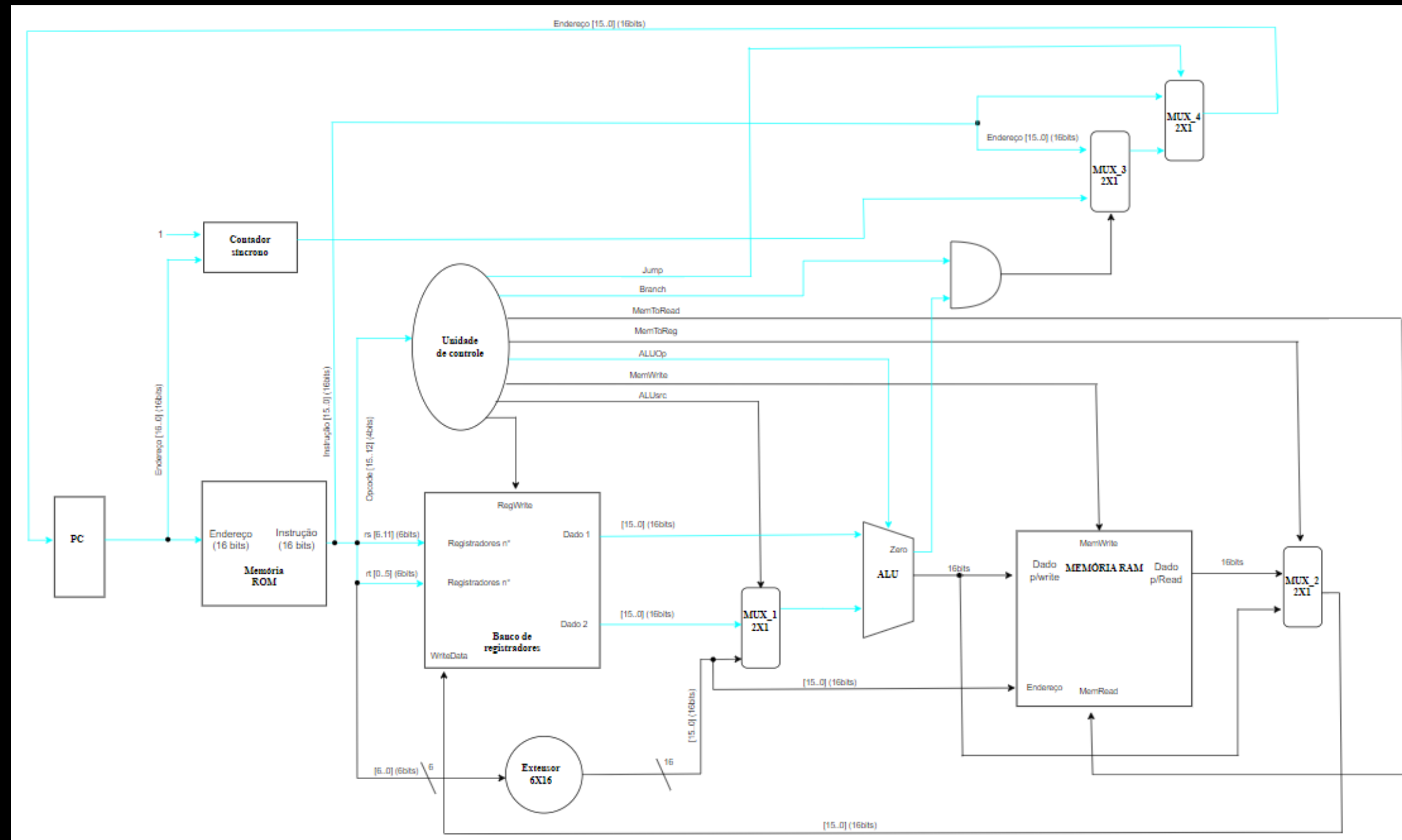
Datapath do tipo R



Datapath do tipo I



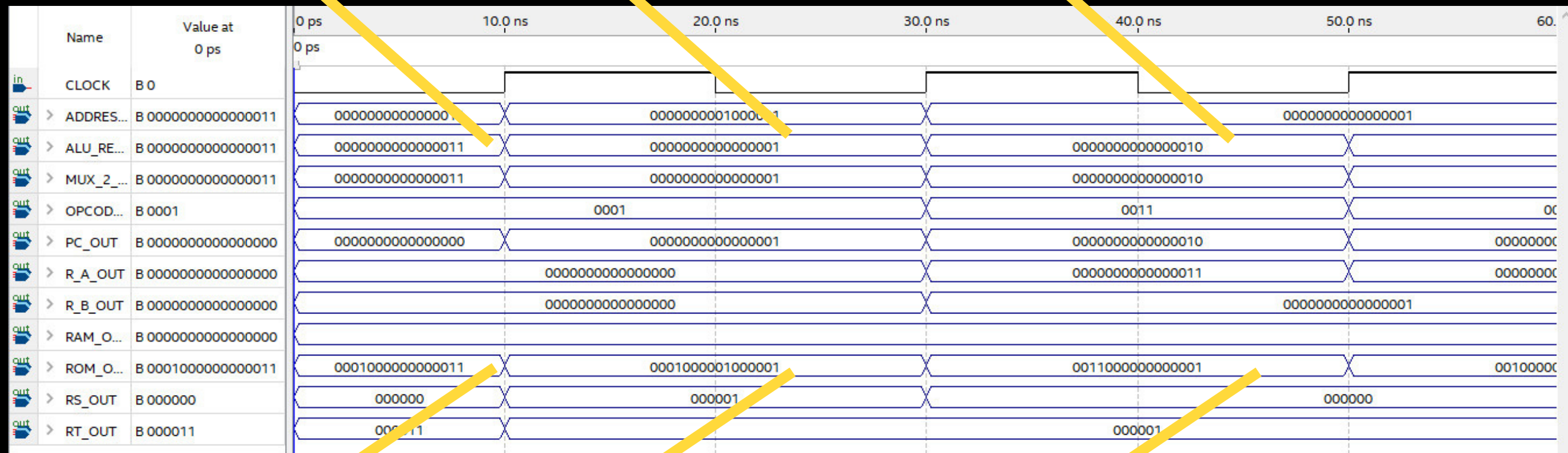
Datapath do tipo J



Teste do ADDI, SUB e SUBI

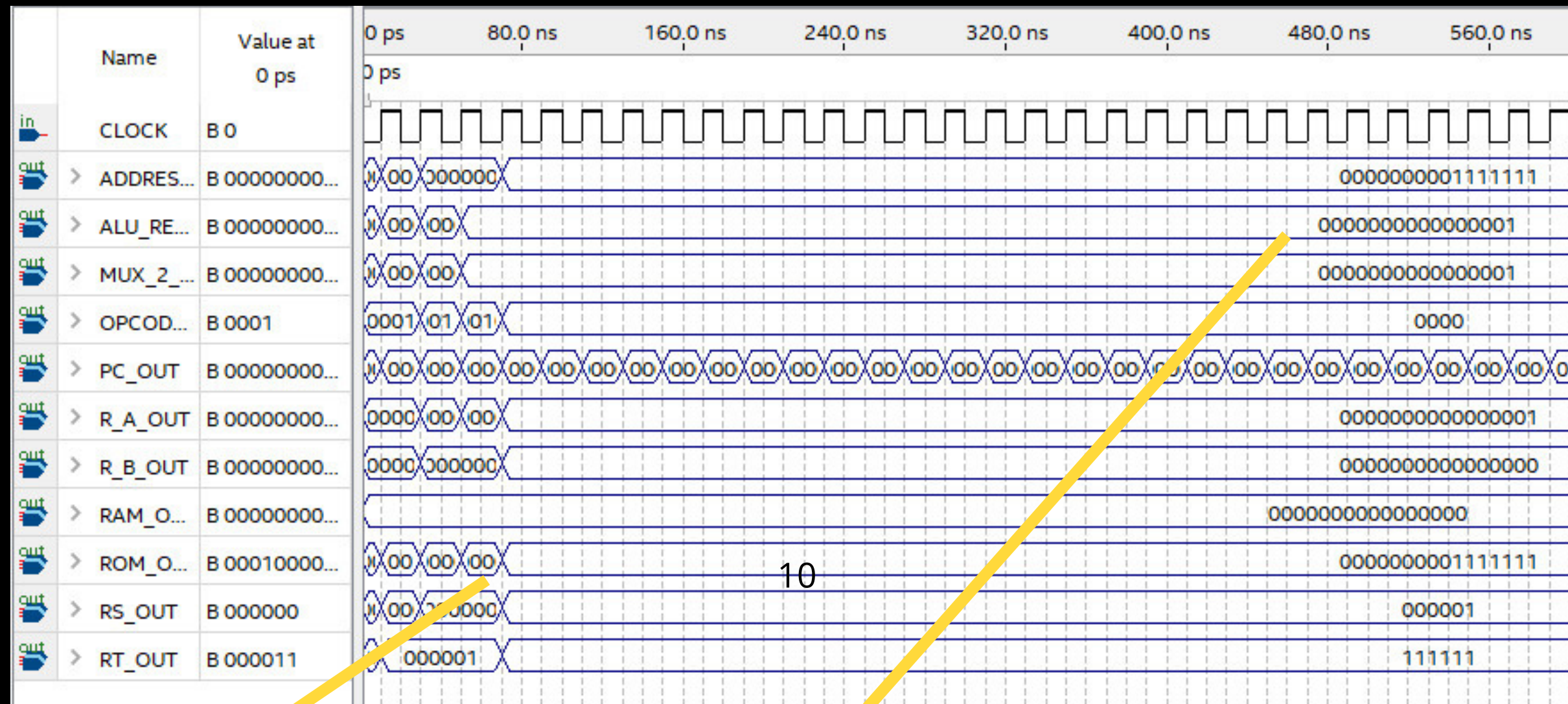
```
----- TESTE DE ADDI, SUB E SUBI
0 => "0001000000000001", -- ADDI S0 3
1 => "0001000001000001", -- ADDI S1 1
2 => "0011000000000001", -- SUBI S0 1
3 => "0010000000000001", -- SUB S0 S1
```

Recebe o valor de 3 Recebe o valor de 1 Subtrai 1 do valor de S0 ficando 2



Primeira instrução Segunda instrução Terceira instrução

Teste do ADDI, SUB e SUBI



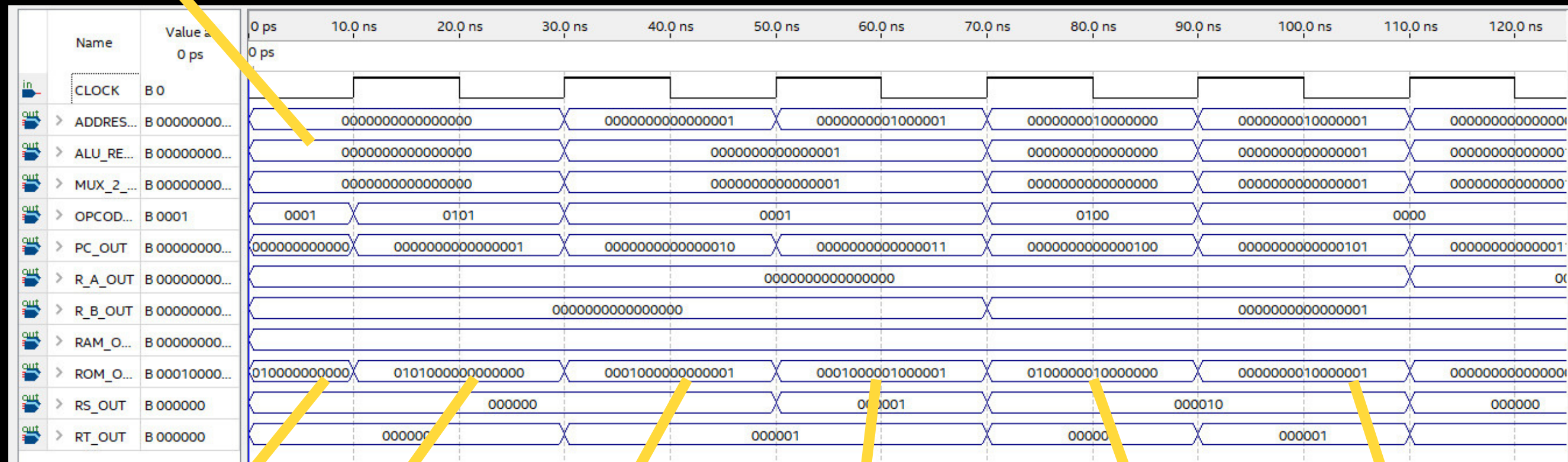
Quarta instrução
Sub S0 S1

O resultado é 1 pois
 $S0 = 2$ e $S1 = 1$
 $2 - 1 = 1$

Teste do Fibonacci

```
-- TESTE FIBONACCI
0 => "0001000000000000", -- ADDI S0 0
1 => "0101000000000000", -- SW S0
2 => "0001000000000001", -- ADDI S0 1
3 => "0001000001000001", -- ADDI S1 1
4 => "0100000010000000", -- LW S2 0
5 => "0000000010000001", -- ADD S2 S1
6 => "0000000000000000", -- ADD S1 S0
7 => "0100000001000000", -- LW S0 00
8 => "0000000000000010", -- ADD S0 S2
9 => "1001000001010100", -- J 0100
```

s0 == 0



Primeira instrução
ADDI s0 0

Segunda instrução
sW s0

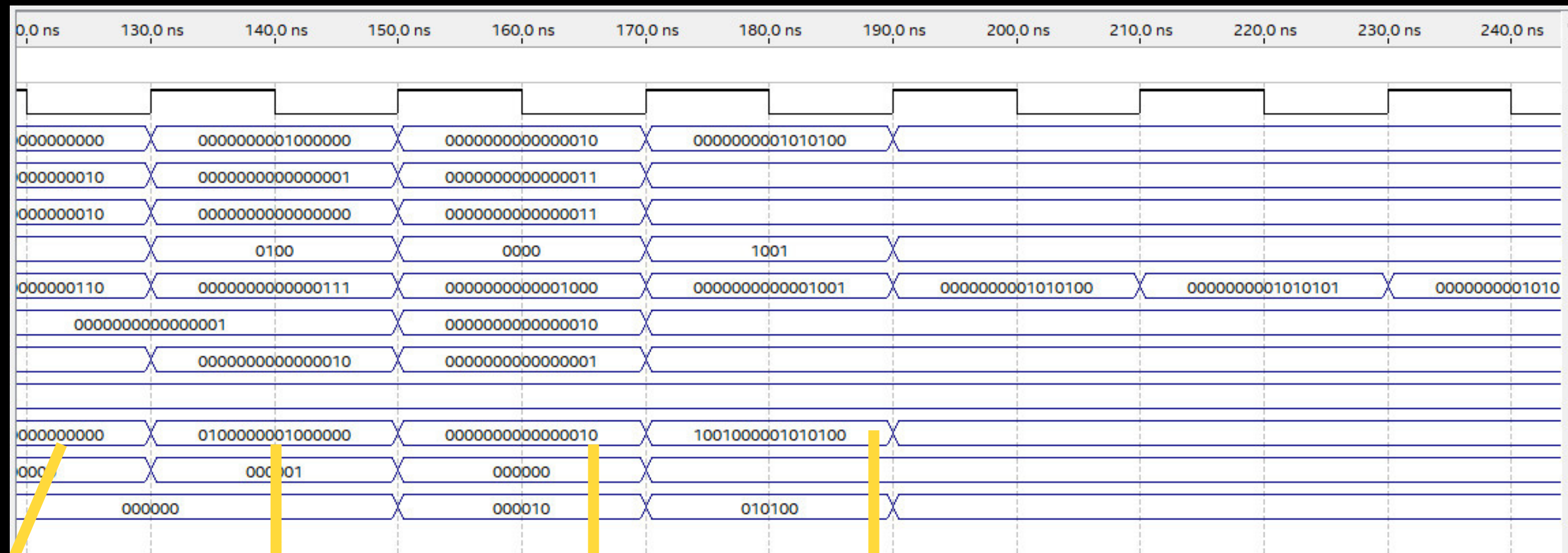
Terceira instrução
ADDI s0 1

Quarta instrução
ADDI s1 1

Quinta instrução
Lw s2 0

Sexta instrução
ADD s2 s1

Teste do Fibonacci



Limitações e dificuldades

- Falta de documentação online e conteúdo disponível;
.....
- IDE Quartus;
.....
- Simulações e testes usando o waveforms;
.....

Conclusão

Este trabalho apresentou o projeto e implementação do processador de 16 bits denominado de Dynasty, que foi uma rica oportunidade para pôr em prática o que nos foi ensinado na disciplina de AOC, e esclarecer diversos pontos que antes eram difíceis de se entender. “Uma das maiores dificuldades encontradas foi justamente a divisão bits”.

Referencias

- Stallings, William. Arquitetura e Organização de computadores. 8º Edição . São Paulo : Pearson Prattice Hall, 2010.

-
- https://github.com/DilliKel/AOC_Eduardo_Kelvin_UFRR_2022

-
- <https://embarcados.com.br/serie/vhdl-basico/>
-

OBRIQADO PELA ATENÇÃO!!!