



**Universidade Federal de Santa Catarina**  
**Centro Tecnológico**  
Departamento de Informática e Estatística  
Ciências da Computação & Engenharia Eletrônica



# **Sistemas Digitais**

**INE 5406**

## **Aula 12-P**

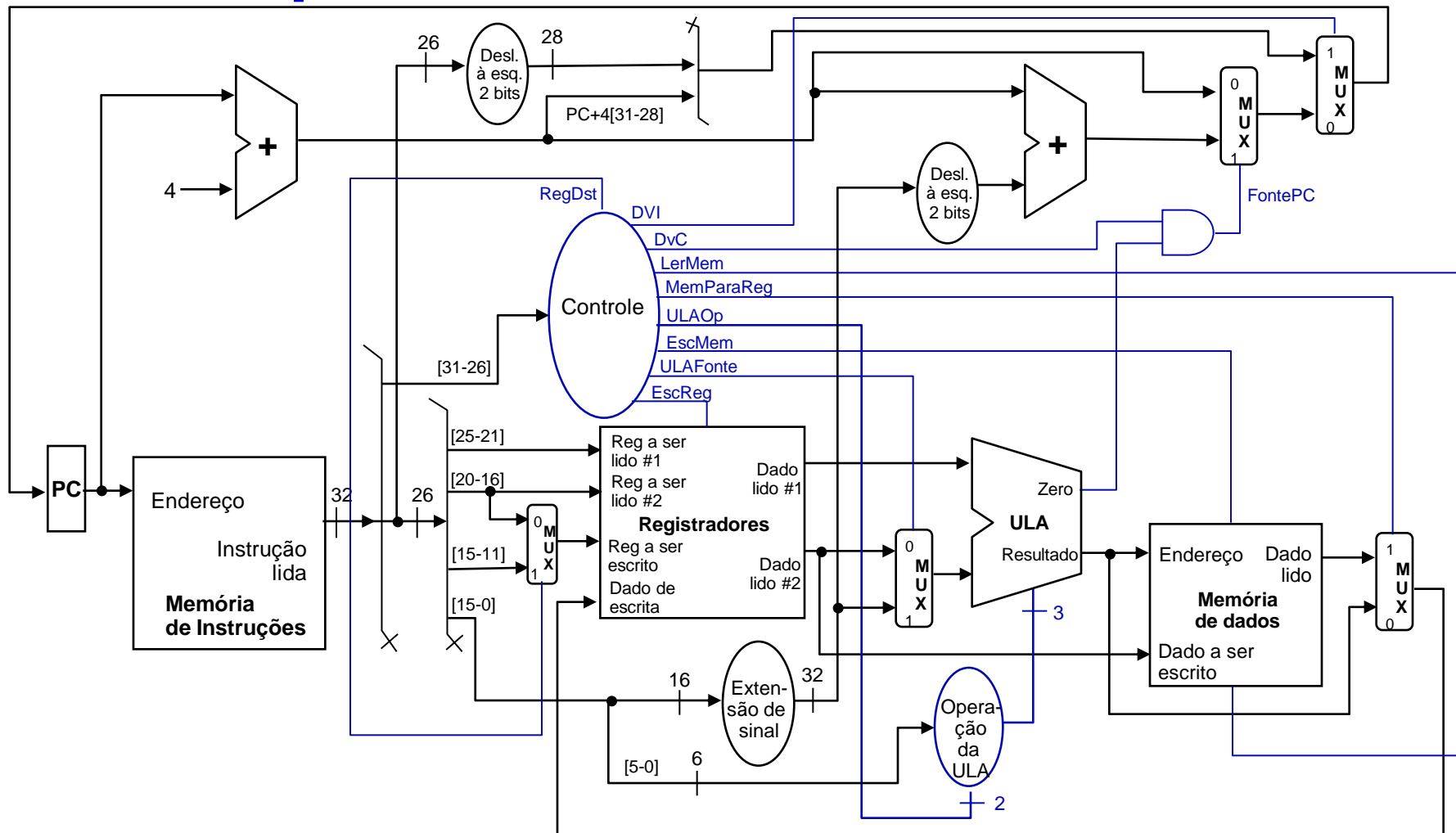
**Implementação do controle da ULA e integração com banco de registradores**

**Est. Doc. André Bräscher & Luiz H. Cancellier**

**Prof. José Luís Güntzel**  
**j.guntzel@ufsc.br**

# O Processador MIPS Monociclo

## Bloco Operativo + Bloco de Controle

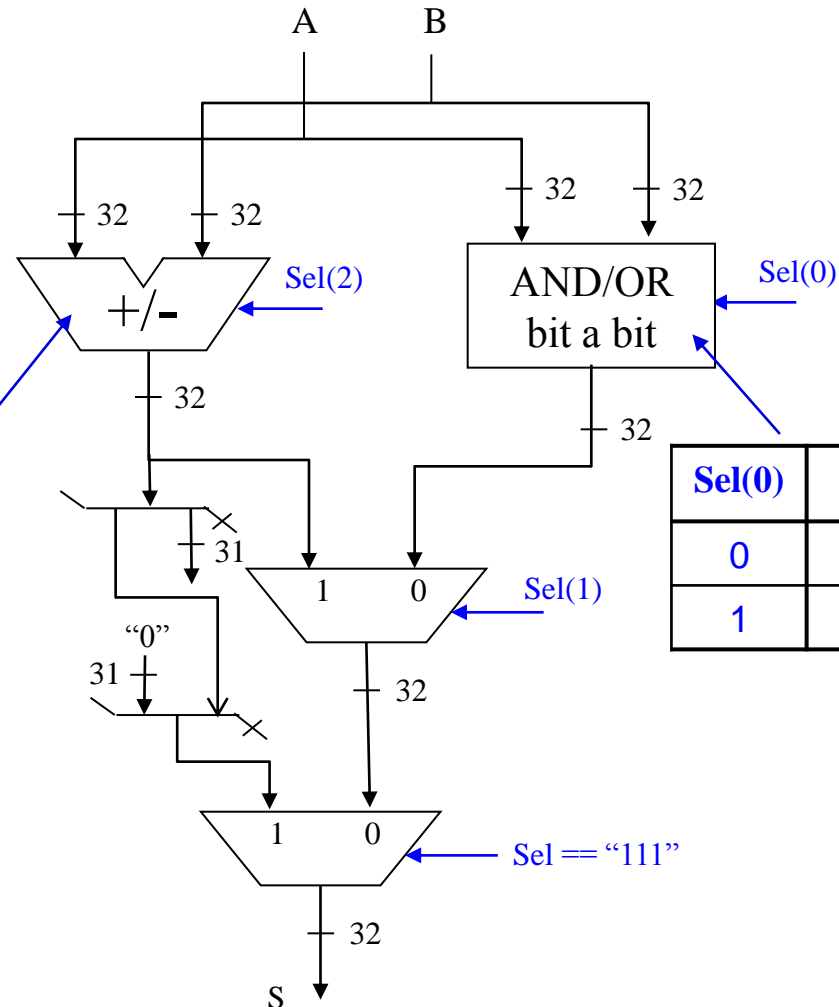


# O Processador MIPS Monociclo

## A ULA do MIPS

sel	operação
000	$S = A \text{ AND } B$
001	$S = A \text{ OR } B$
010	$S = A + B$
110	$S = A - B$
111	$S = 1 \text{ se } A < B \text{ senão, } S = 0$

Sel(2)	operação
0	$X = A + B$
1	$X = A - B$

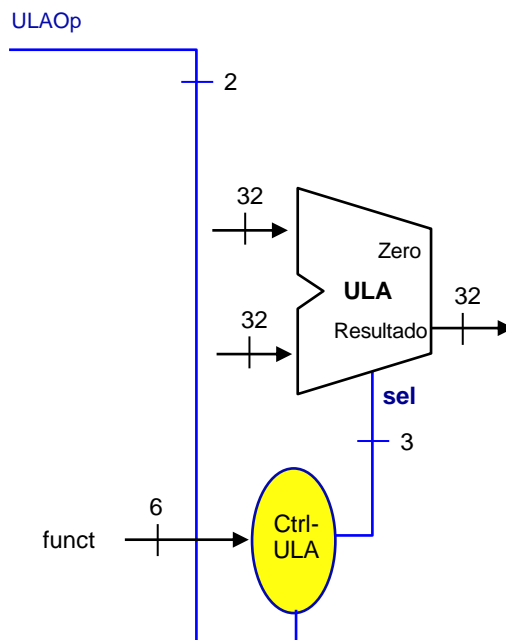


Sel(0)	operação
0	$Y = A \text{ AND } B$
1	$Y = A \text{ OR } B$

# O Processador MIPS Monociclo

## O Controle da ULA

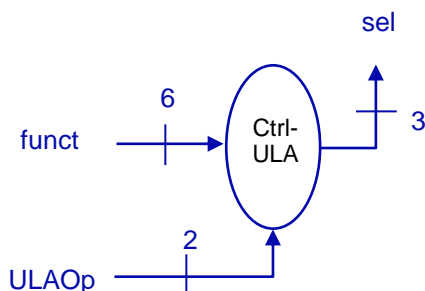
- A ULA terá um controle próprio, separado do bloco de controle principal (e portanto, menos complexo e mais rápido)



# O Processador MIPS Monociclo

## O Controle da ULA

Correspondência entre “funct” & “ULAOp” com “controle da ULA”



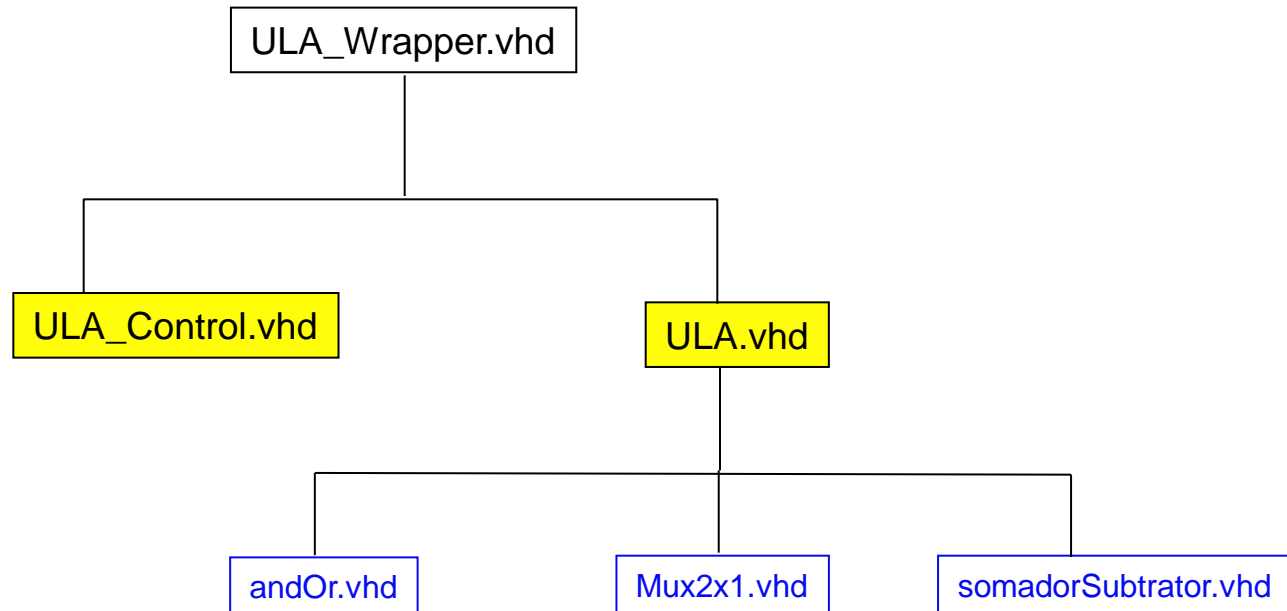
Instrução	ULAOp	Campo “funct”	Operação da ULA	sel
lw	00	XXXXXX	adição	010
sw	00	XXXXXX	adição	010
beq	01	XXXXXX	subtração	110
add	10	100000	adição	010
sub	10	100010	subtração	110
and	10	100100	and	000
or	10	100101	or	001
slt	10	101010	set on less than	111

### Conclusões:

- Apenas algumas das 64 combinações possíveis a partir dos 6 bits do campo “funct” são de interesse
- O campo “funct” somente interessa quando ULAOp = 10

# O Processador MIPS Monociclo

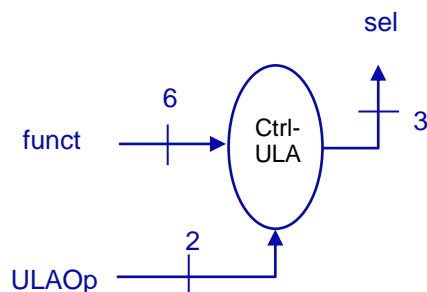
## ► Estrutura do Projeto



# O Processador MIPS Monociclo

## Atividade 1

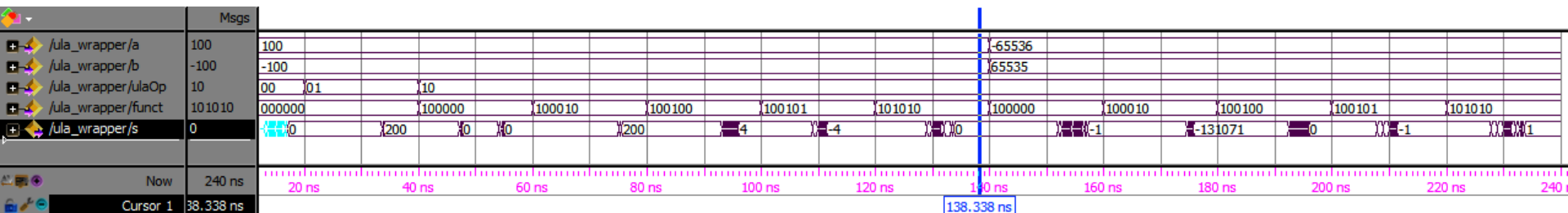
- Baixe os arquivos da Atividade 1 (disponíveis no moodle) e complete o arquivo “ULA\_Control.vhd”.
- Use o arquivo de estímulos disponibilizado para verificar o funcionamento do sistema



Instrução	ULAOp	Campo “funct”	Operação da ULA	sel
lw/sw	00	XXXXXX	adição	010
beq	01	XXXXXX	subtração	110
add	10	100000	adição	010
sub	10	100010	subtração	110
and	10	100100	and	000
or	10	100101	or	001
slt	10	101010	set on less than	111

# O Processador MIPS Monociclo

## Simulação



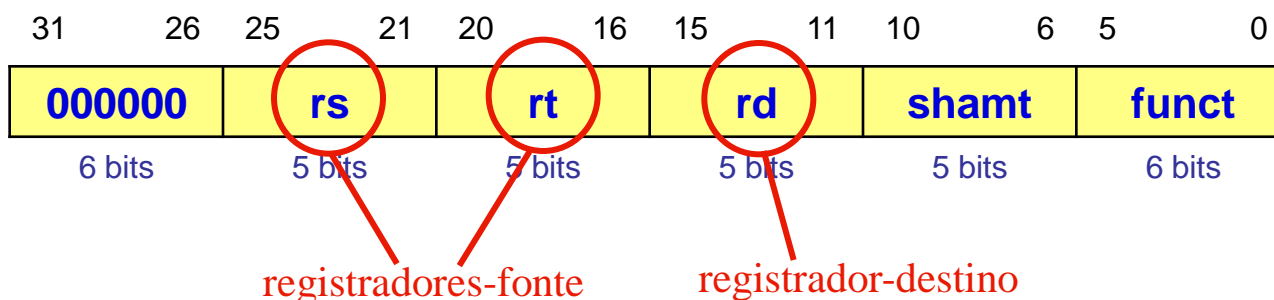
ULAOp	Campo "funct"	Operação da ULA	sel
00	XXXXXX	adição	010
01	XXXXXX	subtração	110
10	100000	adição	010
10	100010	subtração	110
10	100100	and	000
10	100101	or	001
10	101010	set on less than	111



# O Processador MIPS Monociclo

## Instruções formato R: add, sub, or, and

- opcode = 0
- “funct” define a operação a ser feita pela ALU

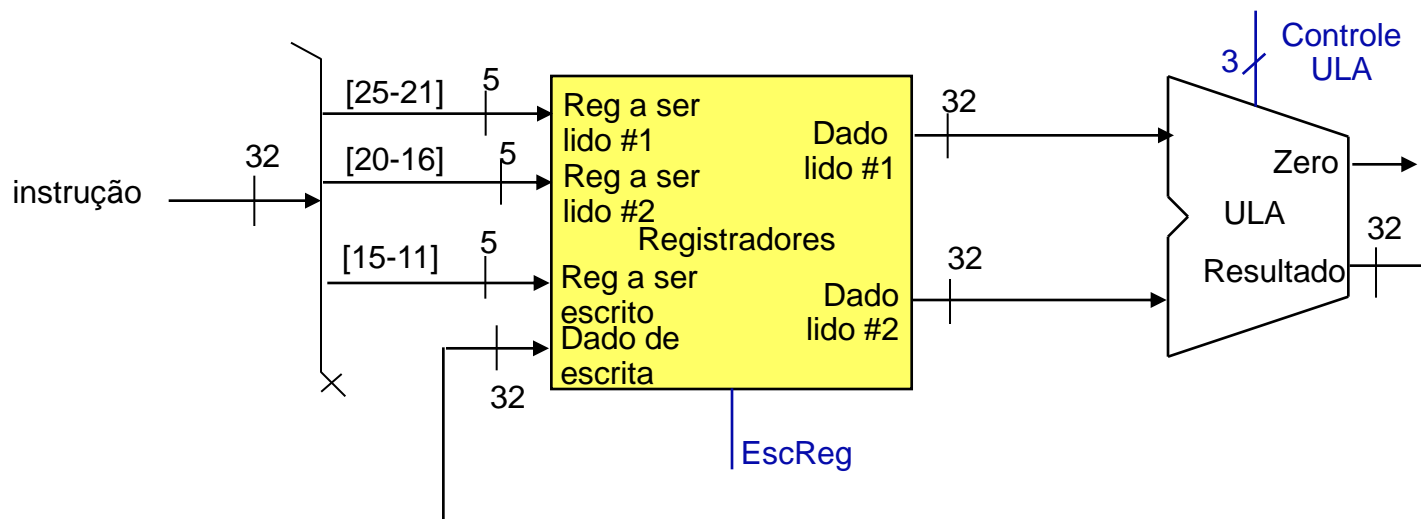


Simbólico (exemplo): `add $s1,$s2, $s3` ( $\$s1 \leftarrow \$s2 + \$s3$ )

# O Processador MIPS Monociclo

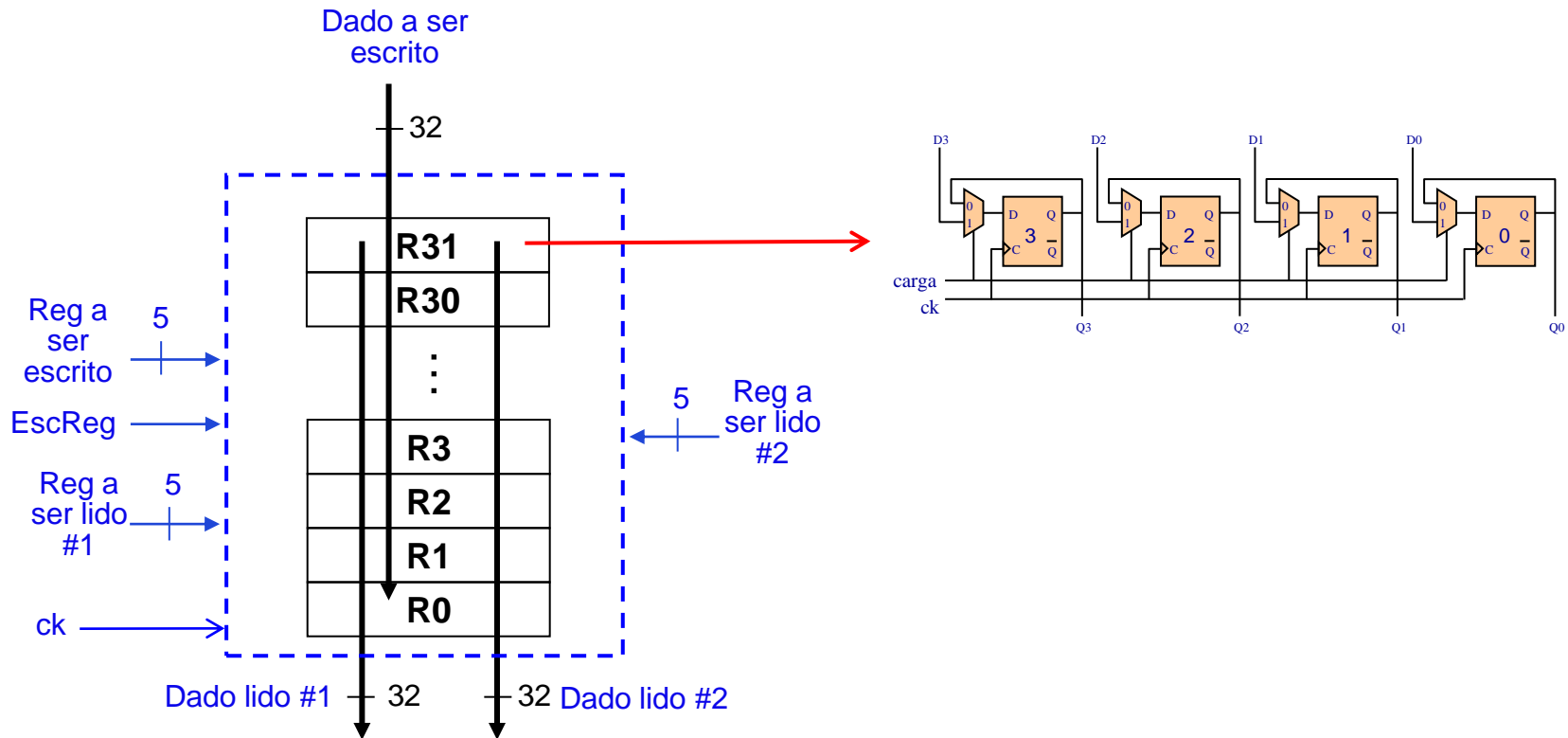
## Componentes do Bloco Operativo

- Um banco de registradores para armazenar os operandos e o resultado das operações
- Uma Unidade Lógica/Aritmética (ULA) que será utilizada para realizar as operações



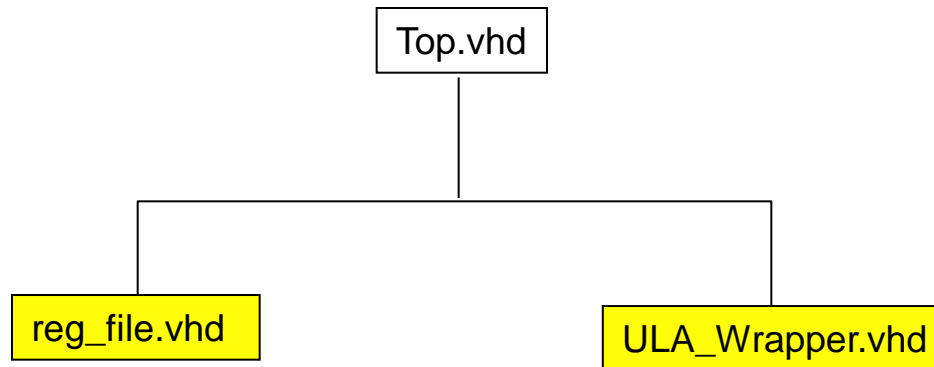
# Estrutura e Comportamento de Registradores

## O Banco de Registradores



# O Processador MIPS Monociclo

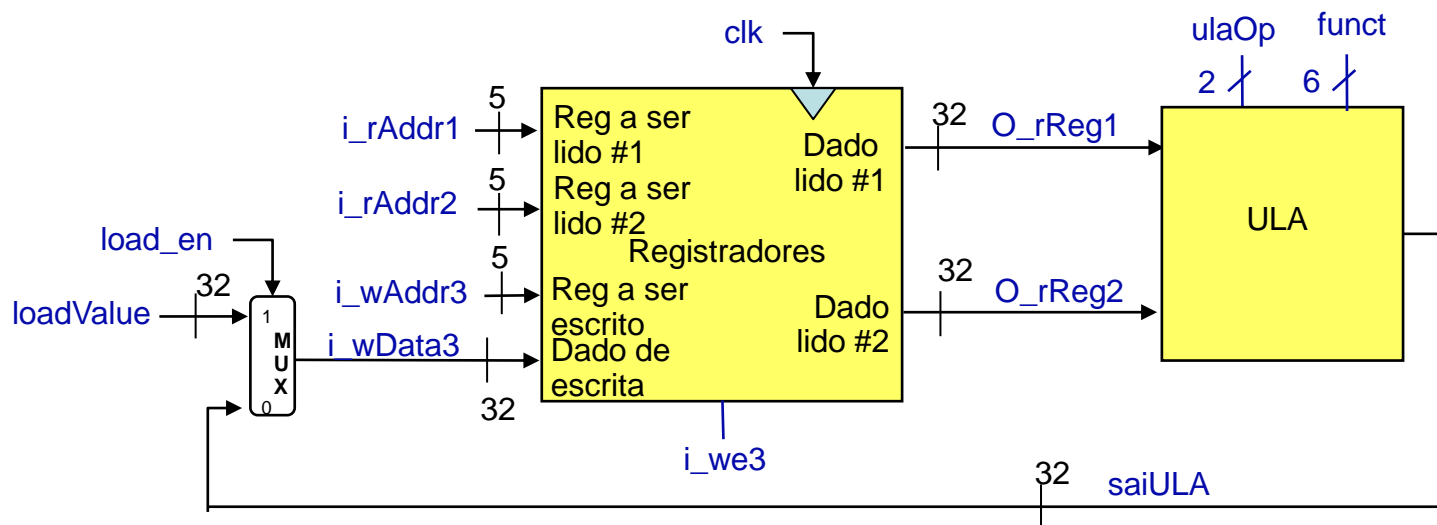
## ► Estrutura do Projeto



# O Processador MIPS Monociclo

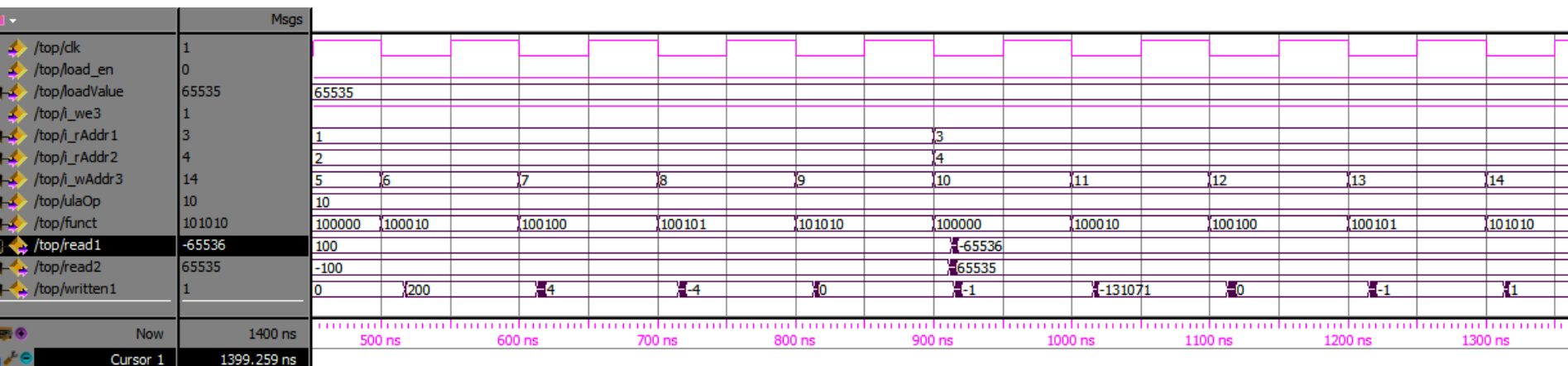
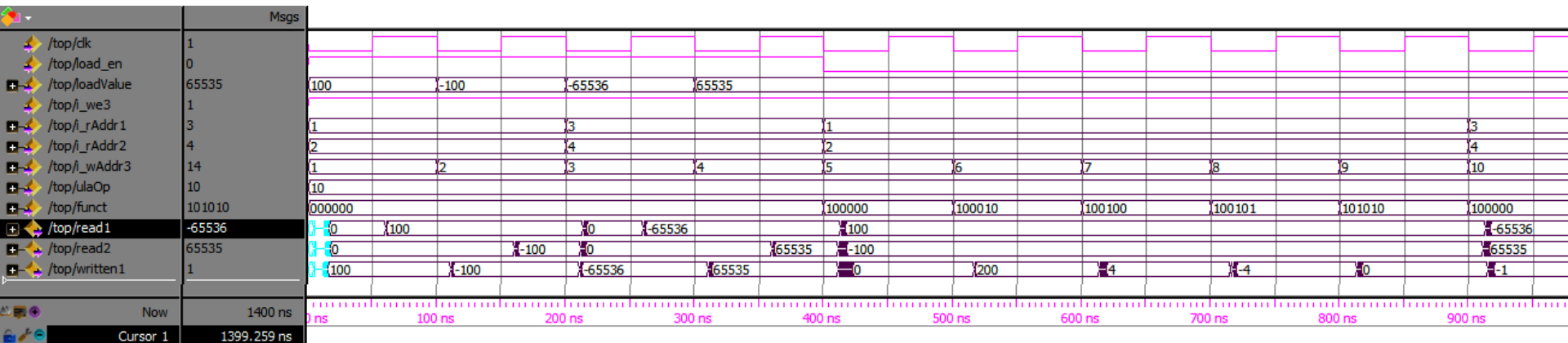
## Atividade 2

- Baixe os arquivos da Atividade 2 (disponíveis no moodle) e complete o arquivo “top.vhd”.
- Use o arquivo de estímulos disponibilizado para verificar o funcionamento do sistema



# O Processador MIPS Monociclo

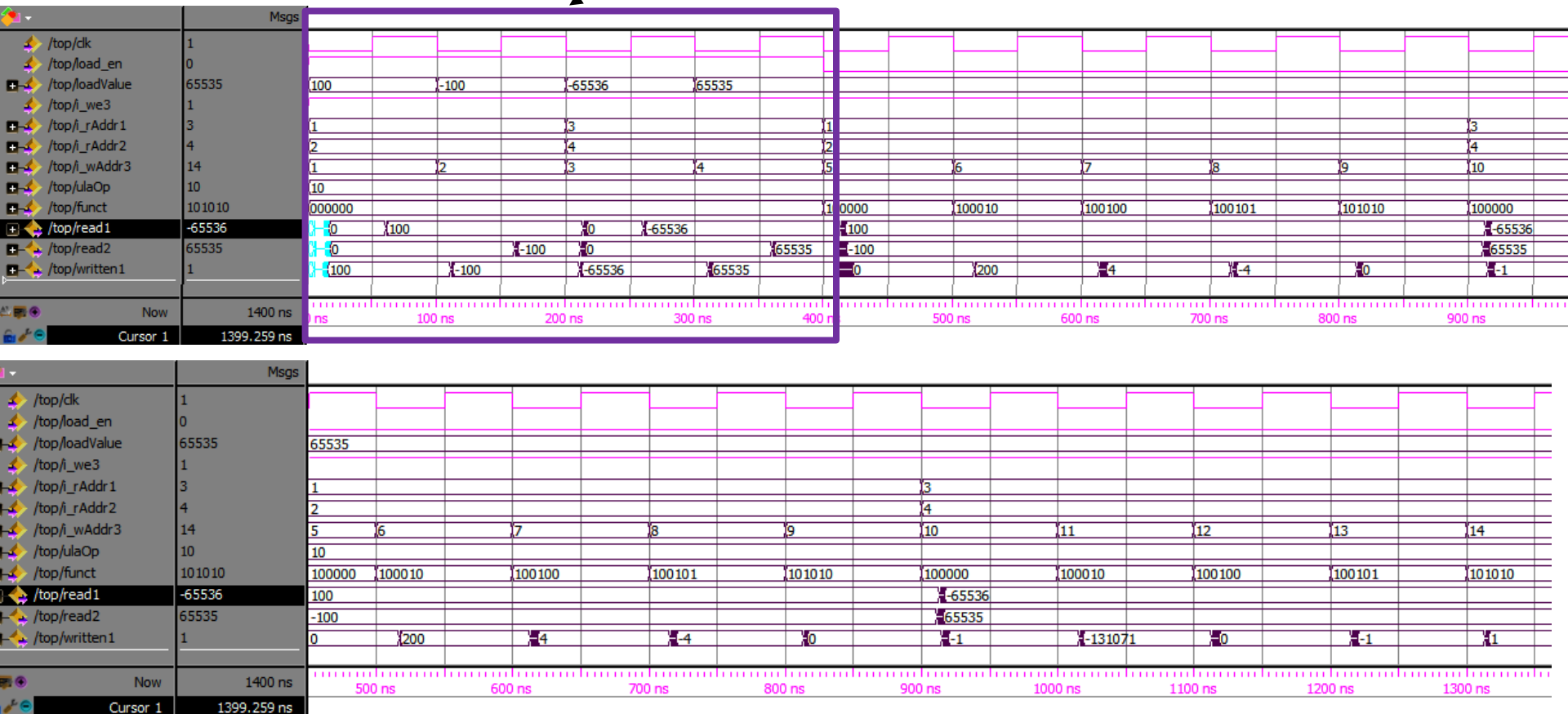
## Simulação



# O Processador MIPS Monociclo

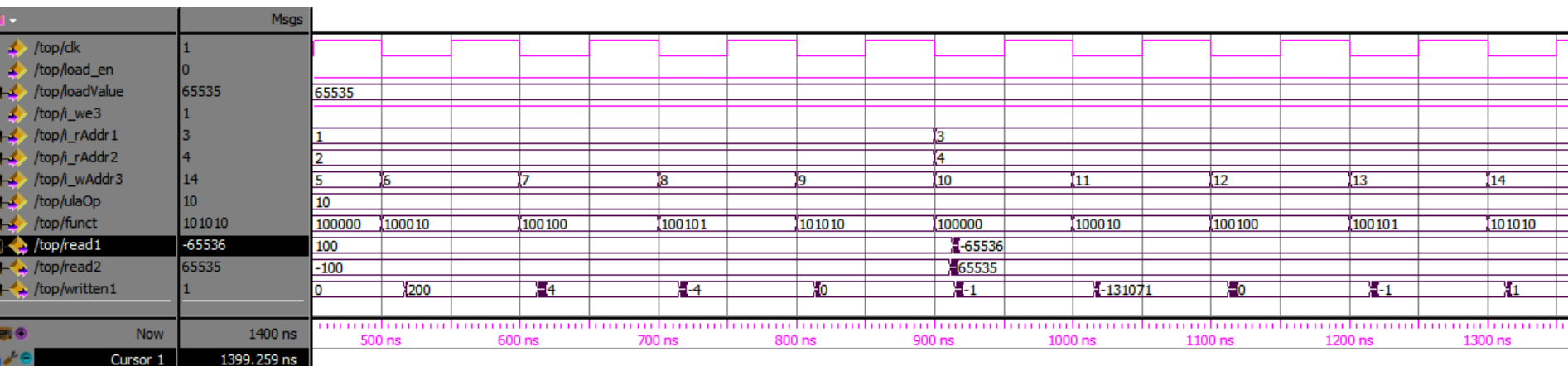
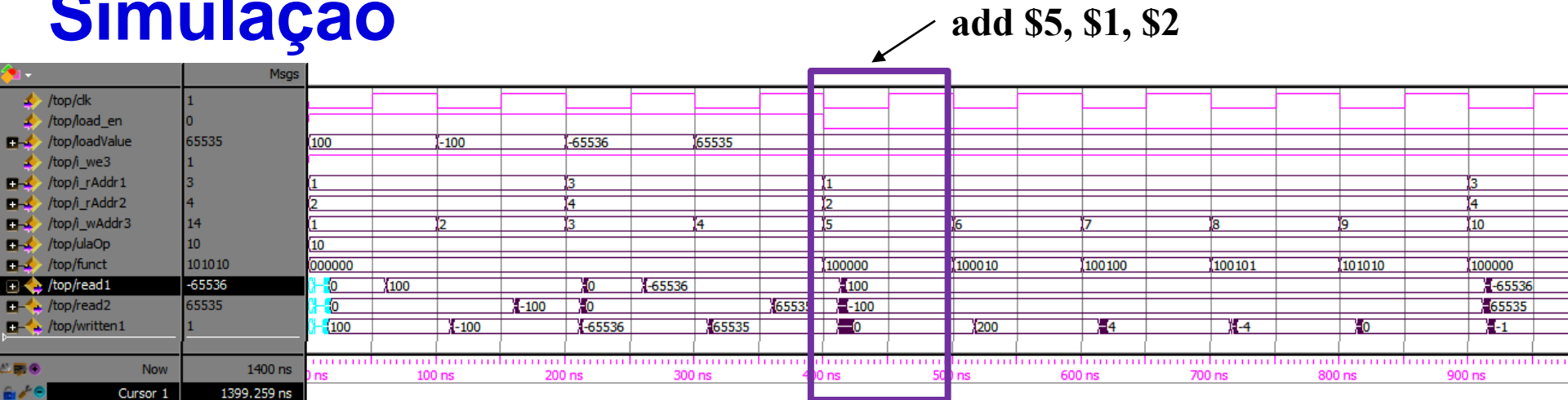
## Simulação

Carrega valores nos registradores



# O Processador MIPS Monociclo

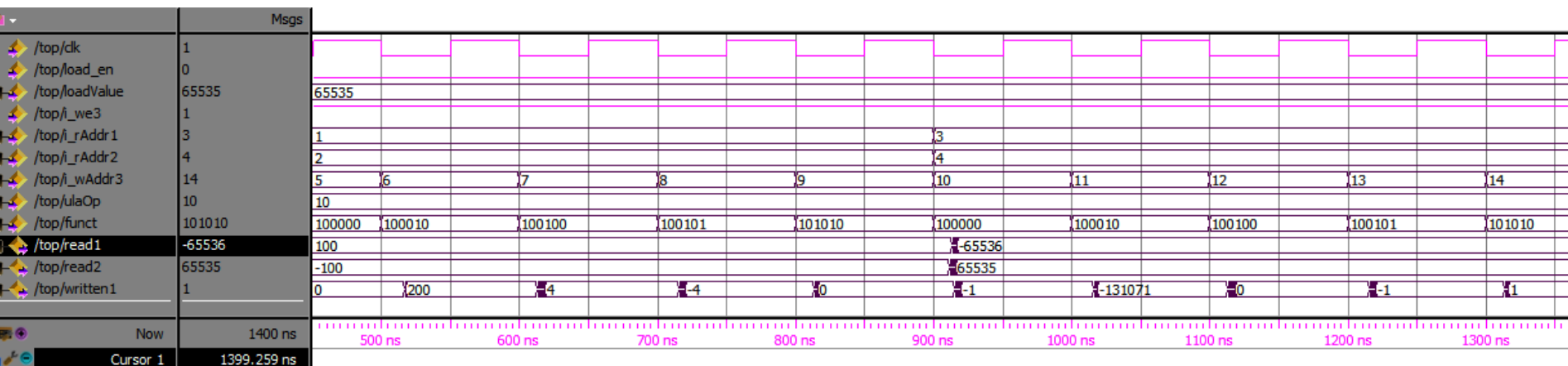
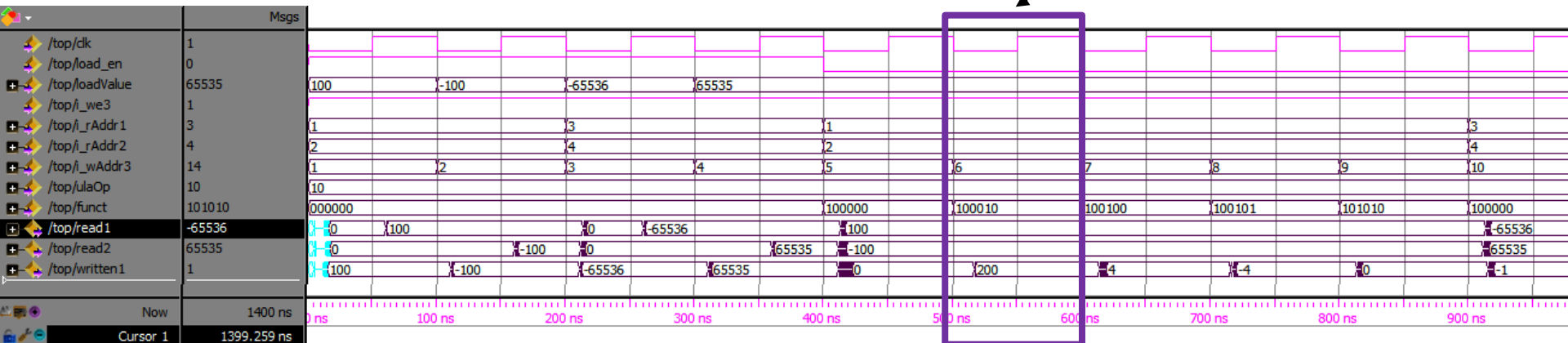
## Simulação





# O Processador MIPS Monociclo

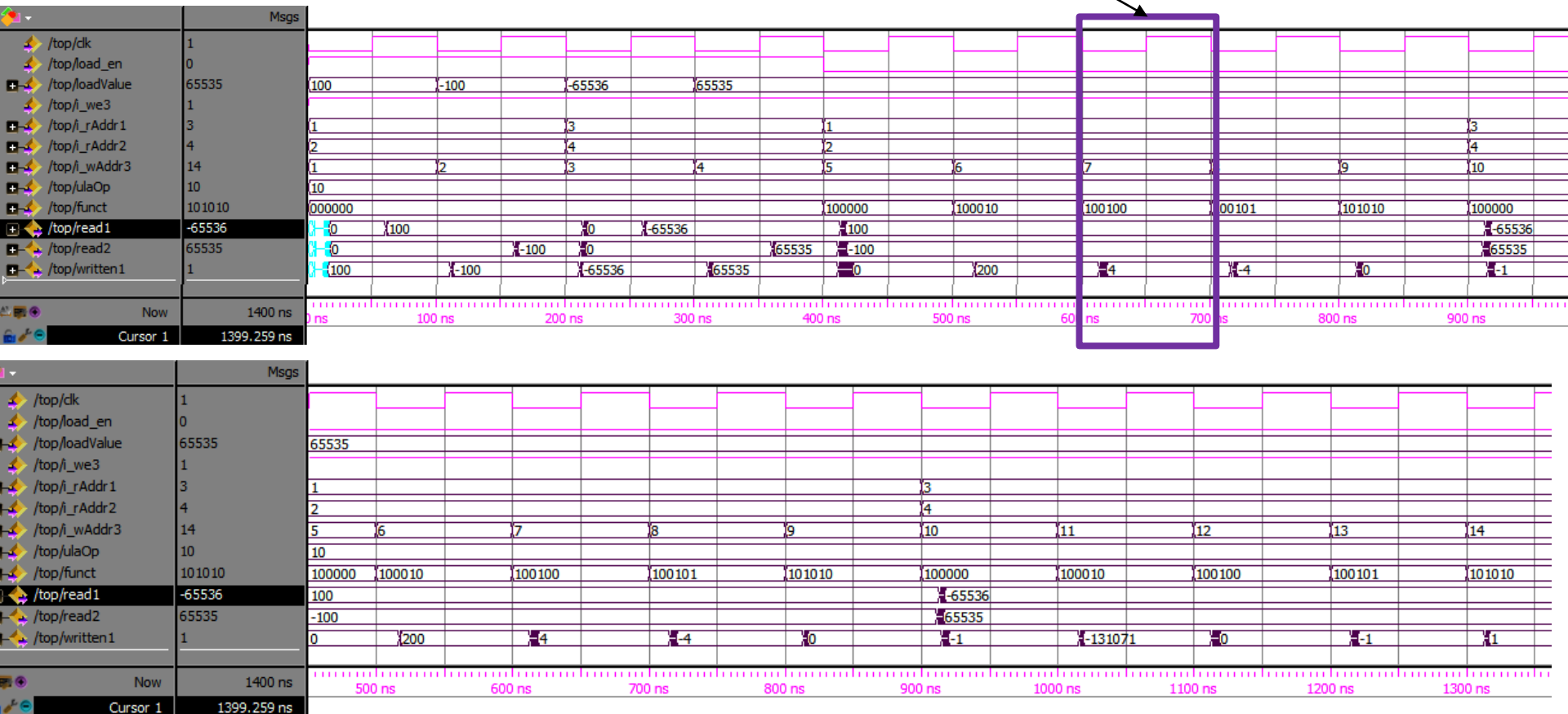
## Simulação



# O Processador MIPS Monociclo

## Simulação

and \$7, \$1, \$2



# O Processador MIPS Monociclo

## Simulação

and \$8, \$1, \$2



# O Processador MIPS Monociclo

## Simulação

or \$9, \$1, \$2



# O Processador MIPS Monociclo

## Simulação

slt \$10, \$1, \$2

