

# Exercício 05

## Nome do aluno:

Fabio Volkmann Coelho

## Objetivo

Consolidar o aprendizado da linguagem Assembly RISC-V e compreender como as instruções de desvio condicional são executadas.

## Instruções

1. Abra o simulador de linguagem RISC-V.
2. No editor de texto do simulador, transcreva o código abaixo:

```
# -----  
# Exercício 05 - Baseado em Patterson pág. 67 (versão RISC-V)  
# Trecho em C: while (save[i] == k) i = i + 1;  
# -----  
  
.data  
Array_save: .word 3, 3, 1, 3, 3, 1, 4, 3, 1 # save[0] até save[8]  
  
.text  
main:  
    la    s6, Array_save    # endereço base de save[]  
    addi s5, zero, 3        # k = 3  
    addi s3, zero, 0        # i = 0  
  
Loop:  
    add  t1, s3, s3        # t1 = s3 * 2  
    add  t1, t1, t1        # t1 = s3 * 4  
    add  t1, t1, s6        # t1 = endereço de save[i]  
    lw   t0, 0(t1)          # t0 = save[i]  
    bne t0, s5, Exit       # se save[i] != k, vai para Exit  
    addi s3, s3, 1          # i = i + j  
    j    Loop               # repete o laço  
  
Exit:  
    nop                  # fim do laço
```

## Montagem e Execução

Clique no botão **Assemble** para montar o programa.

The screenshot shows the RARS 1.5 assembly debugger interface. The assembly code in the editor window is:

```

1  # -----
2  # Exercicio 05 - Baseado em Patterson pág. 67 (versão RISC-V)
3  # Trecho em C: while (save[i] == k) i = i + j;
4  #
5
6  .data
7  Array_save: .word 3, 3, 1, 3, 3, 1, 4, 3, 1  # save[0] até save[8]
8
9  .text
10 main:
11     la    s6, Array_save      # endereço base de save[]
12     addi s5, zero, 3        # k = 3
13     addi s4, zero, 1        # j = 1
14     addi s3, zero, 0        # i = 0
15
16 Loop:
17     add  t1, s3, s3        # t1 = s3 * 2
18     add  t1, t1, t1        # t1 = s3 * 4
19     add  t1, t1, s6        # t1 = endereço de save[i]
20     lw    t0, 0(t1)         # t0 = save[i]
21     bne  t0, s5, Exit      # se save[i] != k, vai para Exit
22     add  s3, s3, s4        # i = i + j
23     j    Loop              # repete o laço
24
25 Exit:
26     nop                  # fim do laço

```

The Registers window shows the initial state of the registers:

| Registers | Floating Point | Control and Status |
|-----------|----------------|--------------------|
| zero      |                | 0 0x00000000       |
| ra        |                | 1 0x00000000       |
| sp        |                | 2 0x7ffffeffc      |
| gp        |                | 3 0x10008000       |
| tp        |                | 4 0x00000000       |
| t0        |                | 5 0x00000000       |
| t1        |                | 6 0x00000000       |
| t2        |                | 7 0x00000000       |
| s0        |                | 8 0x00000000       |
| s1        |                | 9 0x00000000       |
| a0        |                | 10 0x00000000      |
| a1        |                | 11 0x00000000      |
| a2        |                | 12 0x00000000      |
| a3        |                | 13 0x00000000      |
| a4        |                | 14 0x00000000      |
| a5        |                | 15 0x00000000      |
| a6        |                | 16 0x00000000      |
| a7        |                | 17 0x00000000      |
| s2        |                | 18 0x00000000      |
| s3        |                | 19 0x00000000      |
| s4        |                | 20 0x00000000      |
| s5        |                | 21 0x00000000      |
| s6        |                | 22 0x00000000      |
| s7        |                | 23 0x00000000      |
| s8        |                | 24 0x00000000      |
| s9        |                | 25 0x00000000      |
| s10       |                | 26 0x00000000      |
| s11       |                | 27 0x00000000      |
| t3        |                | 28 0x00000000      |
| t4        |                | 29 0x00000000      |
| t5        |                | 30 0x00000000      |
| t6        |                | 31 0x00000000      |
| pc        |                | 0x00400000         |

The Messages window shows the output of the program execution:

```

Reset: reset completed.

-- program is finished running(dropped off bottom) --
-- program is finished running(dropped off bottom) --

```

Faça a execução passo-a-passo do programa e, a cada instrução, preencha a tabela abaixo cada vez que o valor de um registrador ou posição da memória de dados for modificado.

| Antes da execução da instrução |                  |                   | Depois da execução da instrução |            |            |            |            |            |
|--------------------------------|------------------|-------------------|---------------------------------|------------|------------|------------|------------|------------|
| PC                             | Instrução        | Pseudoinstrução   | R5                              | R6         | R19        | R20        | R21        | R22        |
|                                |                  |                   | (t0)                            | (t1)       | (s3)       | (s4)       | (s5)       | (s6)       |
|                                |                  |                   | 0x00000000                      | 0x00000000 | 0x00000000 | 0x00000000 | 0x00000000 | 0x00000000 |
| 0x00400000                     | auipc x22, 64528 | la s6, Array_save |                                 |            |            |            |            | 0x10010000 |
| 0x00400004                     | addi x22 x22 0   |                   |                                 |            |            |            |            |            |

| Antes da execução da instrução |                     |                   | Depois da execução da instrução |            |            |            |            |            |
|--------------------------------|---------------------|-------------------|---------------------------------|------------|------------|------------|------------|------------|
| PC                             | Instrução           | Pseudoinstrução   | R5                              | R6         | i          | j          | k          | save       |
|                                |                     |                   | (t0)                            | (t1)       | (s3)       | (s4)       | (s5)       | (s6)       |
|                                |                     |                   | 0x00000000                      | 0x00000000 | 0x00000000 | 0x00000000 | 0x00000000 | 0x00000000 |
| 0x00400008                     | addi s5, zero, 3    | k = 3             |                                 |            |            |            | 0x00000003 | 0x10010000 |
| 0x0040000C                     | addi s3, zero, zero | i = 0             |                                 |            | 0x00000000 |            | 0x00000003 | 0x10010000 |
| 0x00400010                     | add t1, s3, s3      | t1 = i + i        |                                 | 0x00000000 | 0x00000000 |            | 0x00000003 | 0x10010000 |
| 0x00400014                     | add t1, t1, t1      | t1 = t1 + t1      |                                 | 0x00000000 | 0x00000000 |            | 0x00000003 | 0x10010000 |
| 0x00400018                     | add t1, t1, s6      | calcula offset    |                                 | 0x10010000 | 0x00000000 |            | 0x00000003 | 0x10010000 |
| 0x0040001c                     | lw t0, 0(t1)        |                   | 0x00000003                      | 0x10010000 | 0x00000000 |            | 0x00000003 | 0x10010000 |
| 0x00400020                     | bne t0, s5, Exit    |                   | 0x00000003                      | 0x10010000 | 0x00000000 |            | 0x00000003 | 0x10010000 |
| 0x00400024                     | addi s3, s3, 1      | i++               | 0x00000003                      | 0x10010000 | 0x00000001 |            | 0x00000003 | 0x10010000 |
| 0x00400028                     | j Loop              | volta para o loop | 0x00000003                      | 0x10010000 | 0x00000001 |            | 0x00000003 | 0x10010000 |
| 0x00400030                     |                     |                   |                                 | 0x00000002 |            |            |            |            |
| 0x00400034                     |                     |                   |                                 | 0x00000004 |            |            |            |            |
| 0x00400038                     |                     |                   |                                 | 0x10010004 |            |            |            |            |
| 0x0040003c                     |                     |                   | 0x00000003                      |            |            |            |            | 0x10010004 |
| 0x00400040                     |                     |                   |                                 |            |            |            |            |            |
| 0x00400044                     |                     |                   |                                 |            | 0x00000002 |            |            |            |
| 0x00400048                     |                     |                   |                                 |            |            |            |            |            |
| 0x00400050                     |                     |                   |                                 | 0x00000004 |            |            |            |            |
| 0x00400054                     |                     |                   |                                 | 0x00000008 |            |            |            |            |
| 0x00400058                     |                     |                   |                                 | 0x10010008 |            |            |            |            |
| 0x0040005c                     |                     |                   | 0x00000001                      |            |            |            |            |            |
| 0x00400060                     | bne t0, s5, Exit    |                   | 0x00000001                      |            |            |            |            |            |
| 0x00400064                     | nop                 |                   |                                 |            |            |            |            |            |
|                                |                     |                   |                                 |            |            |            |            |            |
|                                |                     |                   |                                 |            |            |            |            |            |

OBS: Salve o PDF em formato A2 e Paisagem para garantir que todas as informações da página fiquem visíveis

Adicionar linha

Salvar como PDF

Se desejar reiniciar o programa, clique no botão **Reset**.

C:\Users\eduar\OneDrive\Área de Trabalho\riscv1.asm - RARS 1.5

File Edit Run Settings Tools Help

Run speed at max (no interaction)

Reset memory and registers

Text Segment

| Bkpt | Address    | Code       | Basic           | Source   |
|------|------------|------------|-----------------|--|
|      | 0x00400000 | 0x00400313 | addi x6,x0,4    | 10: addi t1, zero, 4 # g = 4                       |
|      | 0x00400004 | 0x00300393 | addi x7,x0,3    | 11: addi t2, zero, 3 # h = 3                       |
|      | 0x00400008 | 0x00200e13 | addi x28,x0,2   | 12: addi t3, zero, 2 # i = 2                       |
|      | 0x0040000c | 0x00100e93 | addi x29,x0,1   | 13: addi t4, zero, 1 # j = 1                       |
|      | 0x00400010 | 0x00730f33 | add x30,x6,x7   | 15: add t5, t1, t2 # t5 = g + h                    |
|      | 0x00400014 | 0x01de0fb3 | add x31,x28,x29 | 16: add t6, t3, t4 # t6 = i + j                    |
|      | 0x00400018 | 0x41ff02b3 | sub x5,x30,x31  | 17: sub t0, t5, t6 # f = t5 - t6 (resultado em t0) |

Data Segment

| Address    | Value (+0) | Value (+4) | Value (+8) | Value (+c) | Value (+10) | Value (+14) | Value (+18) | Value (+1c) |
|------------|------------|------------|------------|------------|-------------|-------------|-------------|-------------|
| 0x10010000 | 0x00000000 | 0x00000000 | 0x00000000 | 0x00000000 | 0x00000000  | 0x00000000  | 0x00000000  | 0x00000000  |
| 0x10010020 | 0x00000000 | 0x00000000 | 0x00000000 | 0x00000000 | 0x00000000  | 0x00000000  | 0x00000000  | 0x00000000  |
| 0x10010040 | 0x00000000 | 0x00000000 | 0x00000000 | 0x00000000 | 0x00000000  | 0x00000000  | 0x00000000  | 0x00000000  |
| 0x10010060 | 0x00000000 | 0x00000000 | 0x00000000 | 0x00000000 | 0x00000000  | 0x00000000  | 0x00000000  | 0x00000000  |
| 0x10010080 | 0x00000000 | 0x00000000 | 0x00000000 | 0x00000000 | 0x00000000  | 0x00000000  | 0x00000000  | 0x00000000  |
| 0x100100a0 | 0x00000000 | 0x00000000 | 0x00000000 | 0x00000000 | 0x00000000  | 0x00000000  | 0x00000000  | 0x00000000  |
| 0x100100c0 | 0x00000000 | 0x00000000 | 0x00000000 | 0x00000000 | 0x00000000  | 0x00000000  | 0x00000000  | 0x00000000  |
| 0x100100e0 | 0x00000000 | 0x00000000 | 0x00000000 | 0x00000000 | 0x00000000  | 0x00000000  | 0x00000000  | 0x00000000  |
| 0x10010100 | 0x00000000 | 0x00000000 | 0x00000000 | 0x00000000 | 0x00000000  | 0x00000000  | 0x00000000  | 0x00000000  |
| 0x10010120 | 0x00000000 | 0x00000000 | 0x00000000 | 0x00000000 | 0x00000000  | 0x00000000  | 0x00000000  | 0x00000000  |
| 0x10010140 | 0x00000000 | 0x00000000 | 0x00000000 | 0x00000000 | 0x00000000  | 0x00000000  | 0x00000000  | 0x00000000  |
| 0x10010160 | 0x00000000 | 0x00000000 | 0x00000000 | 0x00000000 | 0x00000000  | 0x00000000  | 0x00000000  | 0x00000000  |
| 0x10010180 | 0x00000000 | 0x00000000 | 0x00000000 | 0x00000000 | 0x00000000  | 0x00000000  | 0x00000000  | 0x00000000  |
| 0x100101a0 | 0x00000000 | 0x00000000 | 0x00000000 | 0x00000000 | 0x00000000  | 0x00000000  | 0x00000000  | 0x00000000  |

Messages

Run I/O

Reset: reset completed.

Clear

0x10010000 (.data)  Hexadecimal Addresses  Hexadecimal Values  ASCII

Registers Floating Point Control and Status

| Name | Number | Value      |
|------|--------|------------|
| zero | 0      | 0x00000000 |
| ra   | 1      | 0x00000000 |
| sp   | 2      | 0x7fffffe0 |
| gp   | 3      | 0x10008000 |
| tp   | 4      | 0x00000000 |
| t0   | 5      | 0x00000000 |
| t1   | 6      | 0x00000004 |
| t2   | 7      | 0x00000000 |
| s0   | 8      | 0x00000000 |
| s1   | 9      | 0x00000000 |
| a0   | 10     | 0x00000000 |
| a1   | 11     | 0x00000000 |
| a2   | 12     | 0x00000000 |
| a3   | 13     | 0x00000000 |
| a4   | 14     | 0x00000000 |
| a5   | 15     | 0x00000000 |
| a6   | 16     | 0x00000000 |
| a7   | 17     | 0x00000000 |
| s2   | 18     | 0x00000000 |
| s3   | 19     | 0x00000000 |
| s4   | 20     | 0x00000000 |
| s5   | 21     | 0x00000000 |
| s6   | 22     | 0x00000000 |
| s7   | 23     | 0x00000000 |
| s8   | 24     | 0x00000000 |
| s9   | 25     | 0x00000000 |
| s10  | 26     | 0x00000000 |
| s11  | 27     | 0x00000000 |
| t3   | 28     | 0x00000000 |
| t4   | 29     | 0x00000000 |
| t5   | 30     | 0x00000000 |
| t6   | 31     | 0x00000000 |
| pc   |        | 0x00400004 |

[← Voltar ao tutorial](#)