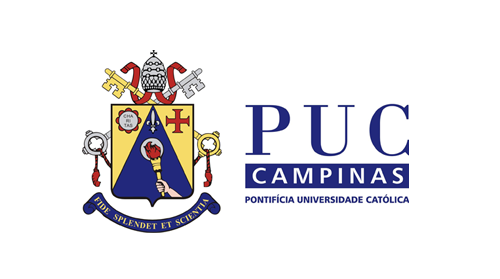
****

Engenharia da computação

Lab. de Arquitetura de Sistemas Computacionais

Projeto CPU

|  |  |
| --- | --- |
| Nome | RA |
| Júlio Ferro Neto | 15265135 |
| Robson Quero | 15124422 |
| Daniel Augusto Silveira Ramos | 15589500 |
| José Carlos Clausen Neto | 15055825 |

*25/10/2016*

**Sumário**

**1. Introdução**

* 1. Descrição textual do projeto
  2. Topologia da CPU

**2. Especificação**

2.1 Registradores (quantidade, endereço, tamanho)

2.2 Formato das instruções (OPCODE);

2.3 Unidade de controle: diagrama e tabela de estados, sinais e seus significados

**3. Resultados**

3.1 Descrição dos testes realizados

3.2 Resultados e descrição

**4. Bibliografia**

**5. Código em Anexo**

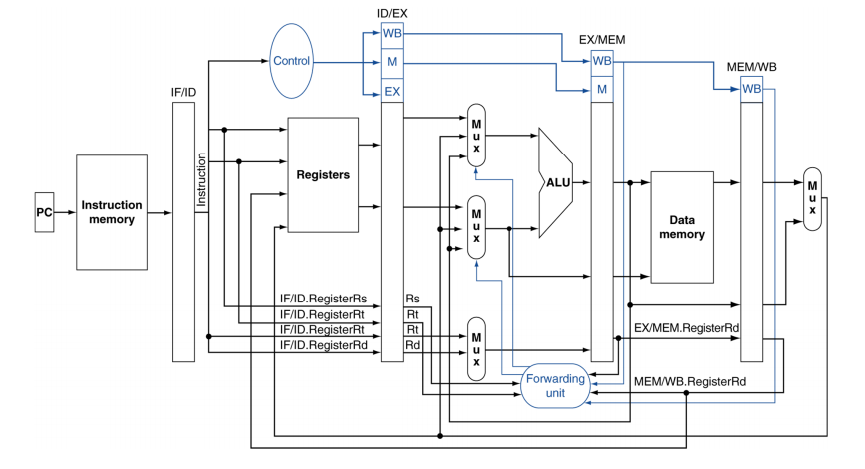
**Relatório Projeto CPU**

1. **Introdução**

1.1 Descrição textual do projeto

O projeto descreve a realização do desenvolvimento de uma CPU que realiza treze instruções (ADD, ADD Imediate, Subtract, Subtract Imediate, Load Word, Store Word, AND, AND Imediate, OR, OR Imediate, Branch on Equal, Jump e Jump Register). A CPU deve possuir pelo menos quatro registradores endereçáveis com capacidade de 8 bits de armazenamento cada, todos devem ser ligados por um barramento de dados.

* 1. Topologia da CPU



1. **Especificação**
   1. Registradores (quantidade, endereço, tamanho)

|  |  |  |
| --- | --- | --- |
| Registradores | Endereço | Tamanho (bits) |
| R0 | 00 | 8 |
| R1 | 01 | 8 |
| R2 | 10 | 8 |
| R3 | 11 | 8 |

* 1. Formato das instruções (OPCODE);

Tipo R:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 6 bits | 5 bits | 5 bits | 5 bits | 5 bits | 6 bits |
| OPCODE | RD | RS | RT | Shamt | Funct |

Tipo I:

|  |  |  |  |
| --- | --- | --- | --- |
| 6 bits | 5 bits | 5 bits | 16 bits |
| OPCODE | RD | RS | Imediato |

Tipo J:

|  |  |
| --- | --- |
| 6 bits | 26 bits |
| OPCODE | Imediato |

**Resultados**

* 1. **Testes realizados:**
     1. ADD (Tipo R):

|  |  |  |  |
| --- | --- | --- | --- |
| 4 bits | 2 bits | 2 bits | 8 bits |
| 0000 | 01 | 00 | 00000010(b) |
| Movi | R1 | xx | 2(10) |

* + 1. ADD Imediato (Tipo I):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 4 bits | 2 bits | 2 bits | 2 bits | 6 bits |
| 0000 | 01 | 00 | 00 | 000000 |
| Mov | R1 | R0 | xx | xxxxxx |

* + 1. Add (Tipo R):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 4 bits | 2 bits | 2 bits | 2 bits | 6 bits |
| 0011 | 10 | 01 | 00 | 000000 |
| Add | R2 | R1 | R0 | xxxxxx |

* + 1. Add Imediato (Tipo I):

|  |  |  |  |
| --- | --- | --- | --- |
| 4 bits | 2 bits | 2 bits | 8 bits |
| 0100 | 01 | 00 | 00000101(b) |
| Addi | R1 | R0 | 5(d) |

* + 1. SUB (Tipo R):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 4 bits | 2 bits | 2 bits | 2 bits | 6 bits |
| 0010 | 00 | 01 | 00 | 000000 |
| Xchg | R0 | R1 | xx | xxxxxx |

* + 1. SUB Imediato (Tipo I):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 4 bits | 2 bits | 2 bits | 2 bits | 6 bits |
| 0011 | 10 | 01 | 00 | 000000 |
| Add | R2 | R1 | R0 | xxxxxx |

* + 1. Load Word (Tipo I):

|  |  |  |  |
| --- | --- | --- | --- |
| 4 bits | 2 bits | 2 bits | 8 bits |
| 0110 | 01 | 00 | 00000001(b) |
| Subi | R1 | R0 | 1(d) |

* 1. **Resultados e discussão:**
     1. Store Word (Tipo I):

|  |  |  |  |
| --- | --- | --- | --- |
| 4 bits | 2 bits | 2 bits | 8 bits |
| 0000 | 01 | 00 | 00000010(b) |
| Movi | R1 | xx | 2(10) |
| Final | 00000010(b) | xx | xxxxxxxx |

* + 1. Branch on Equal (Tipo I):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 4 bits | 2 bits | 2 bits | 2 bits | 6 bits |
| 0000 | 01 | 00 | 00 | 000000 |
| Mov | R1 | R0 | xx | xxxxxx |
| Final | [R0] | R0 | xx | xxxxxx |

* + 1. Jump (Tipo J):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 4 bits | 2 bits | 2 bits | 2 bits | 6 bits |
| 0011 | 10 | 01 | 00 | 000000 |
| Add | R2 | R1 | R0 | xxxxxx |
| Final | [R1+R0] | R1 | R0 | xxxxxx |

* + 1. Jump Register (Tipo I):

|  |  |  |  |
| --- | --- | --- | --- |
| 4 bits | 2 bits | 2 bits | 8 bits |
| 0100 | 01 | 00 | 00000101(b) |
| Addi | R1 | R0 | 5(d) |
| Final | [R0+5(d)] | R0 | xxxxxxxx |

* + 1. AND (Tipo R):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 4 bits | 2 bits | 2 bits | 2 bits | 6 bits |
| 0010 | 00 | 01 | 00 | 000000 |
| Xchg | R0 | R1 | xx | xxxxxx |
| Final | [R1] | [R0] | xx | xxxxxx |

AND Imediato (Tipo I):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 4 bits | 2 bits | 2 bits | 2 bits | 6 bits |
| 0011 | 10 | 01 | 00 | 000000 |
| Add | R2 | R1 | R0 | xxxxxx |
| Final | [R1-R0] | R1 | R0 | xxxxxx |

* + 1. OR (Tipo R):

|  |  |  |  |
| --- | --- | --- | --- |
| 4 bits | 2 bits | 2 bits | 8 bits |
| 0110 | 01 | 00 | 00000001(b) |
| Subi | R1 | R0 | 1(d) |
| Final | [R0-1(d)] | R0 | xxxxxxxx |

* + 1. OR Imediato (Tipo I):

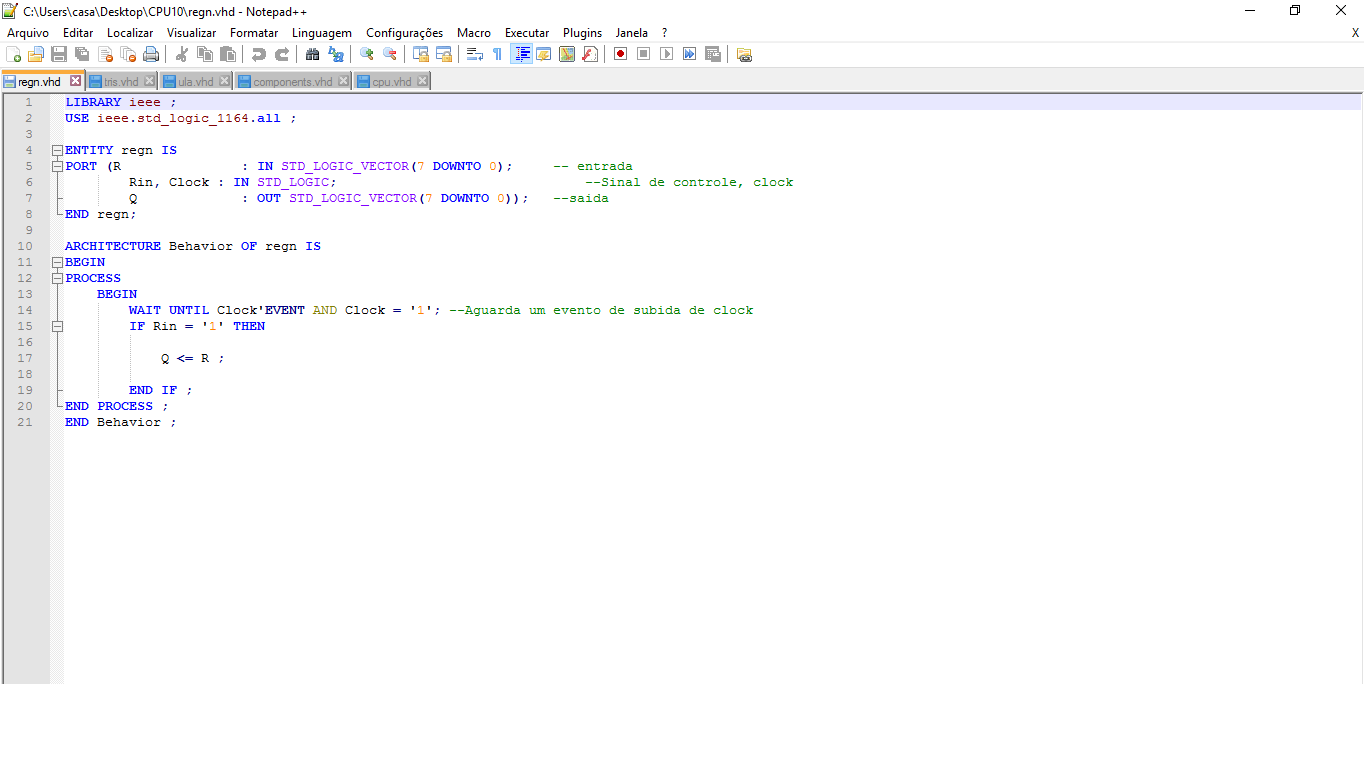
|  |  |  |  |
| --- | --- | --- | --- |
| 4 bits | 2 bits | 2 bits | 8 bits |
| 0110 | 01 | 00 | 00000001(b) |
| Subi | R1 | R0 | 1(d) |
| Final | [R0-1(d)] | R0 | xxxxxxxx |

1. **Bibliografia**

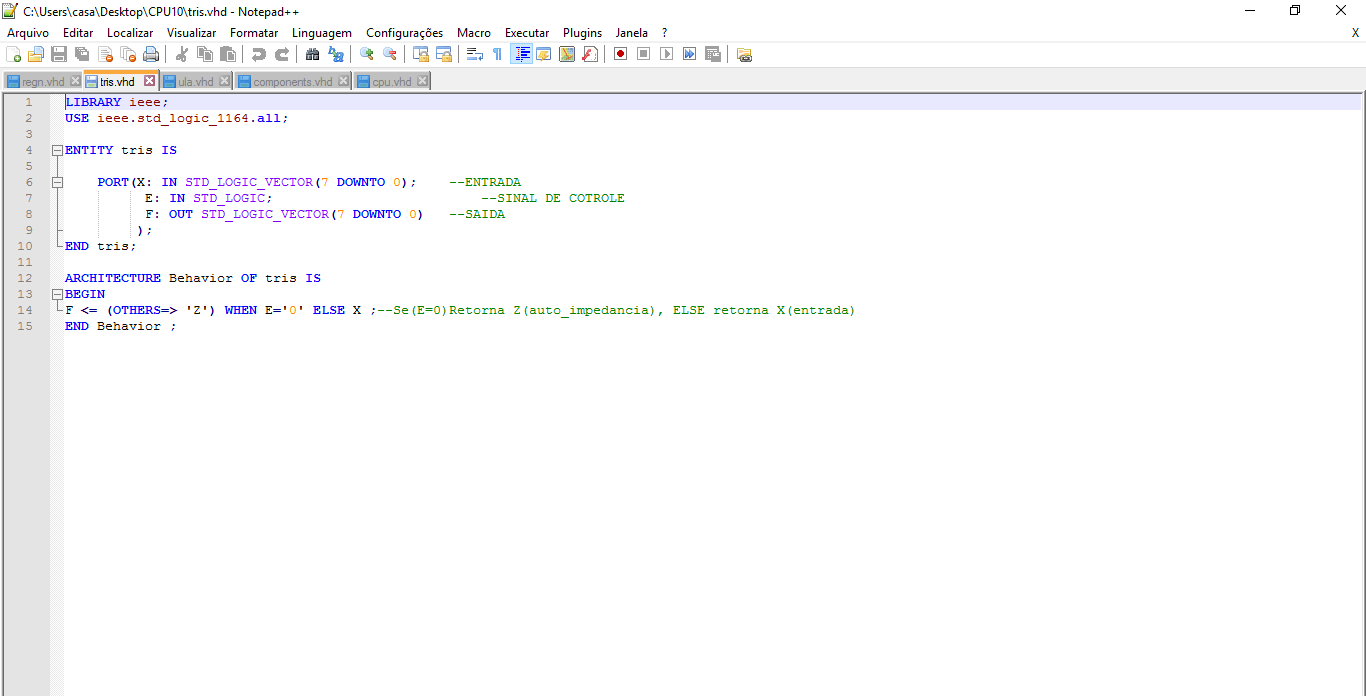
PATTERSON, David A. e HENNESSY, John L. Computer Organization and Design – The Hardware and Software Interface. Estados Unidos, Ed. Morgan Kauffman – 4° edição. Cap 3 e 4.

Stephen Brown; Svonko Vranesic; “Fundamentals of Digital Logic with VHDL Design”.

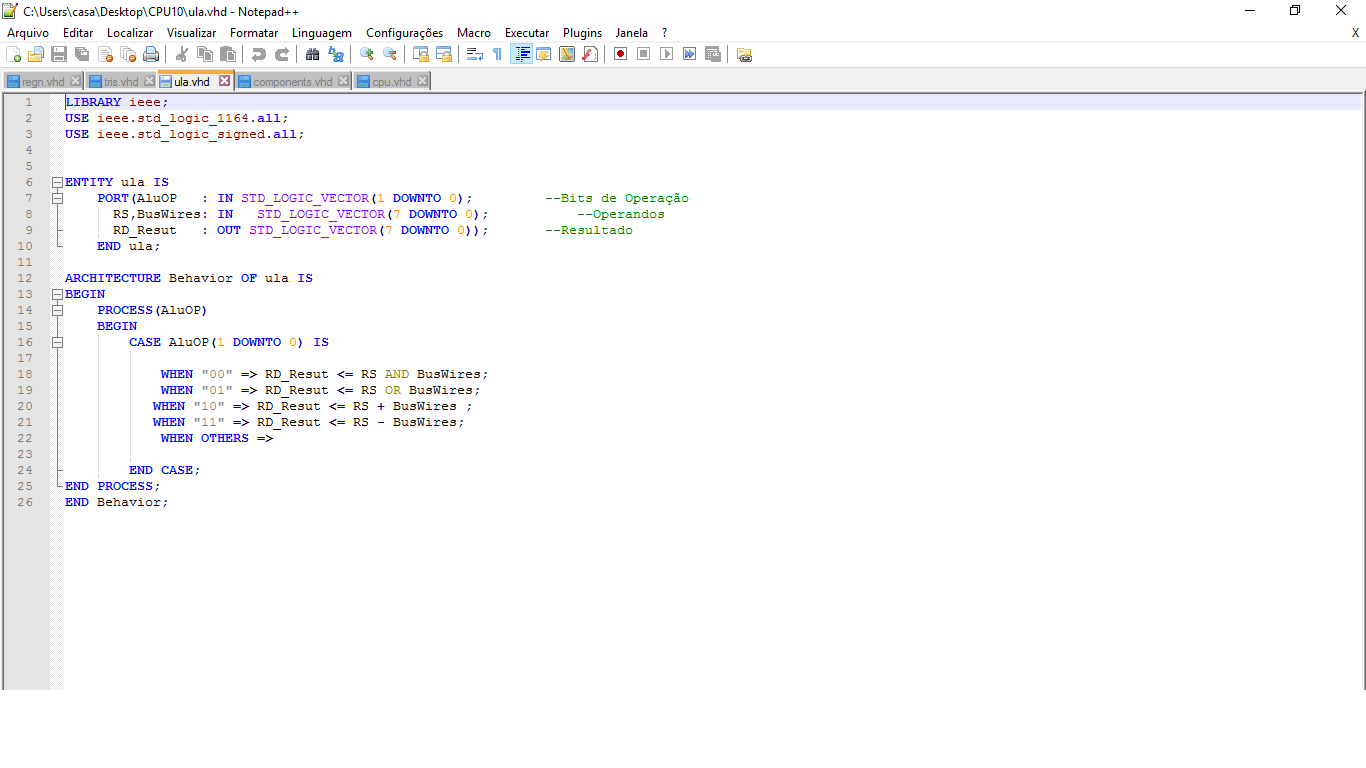
1. **Código em Anexo**
   1. – Regn.



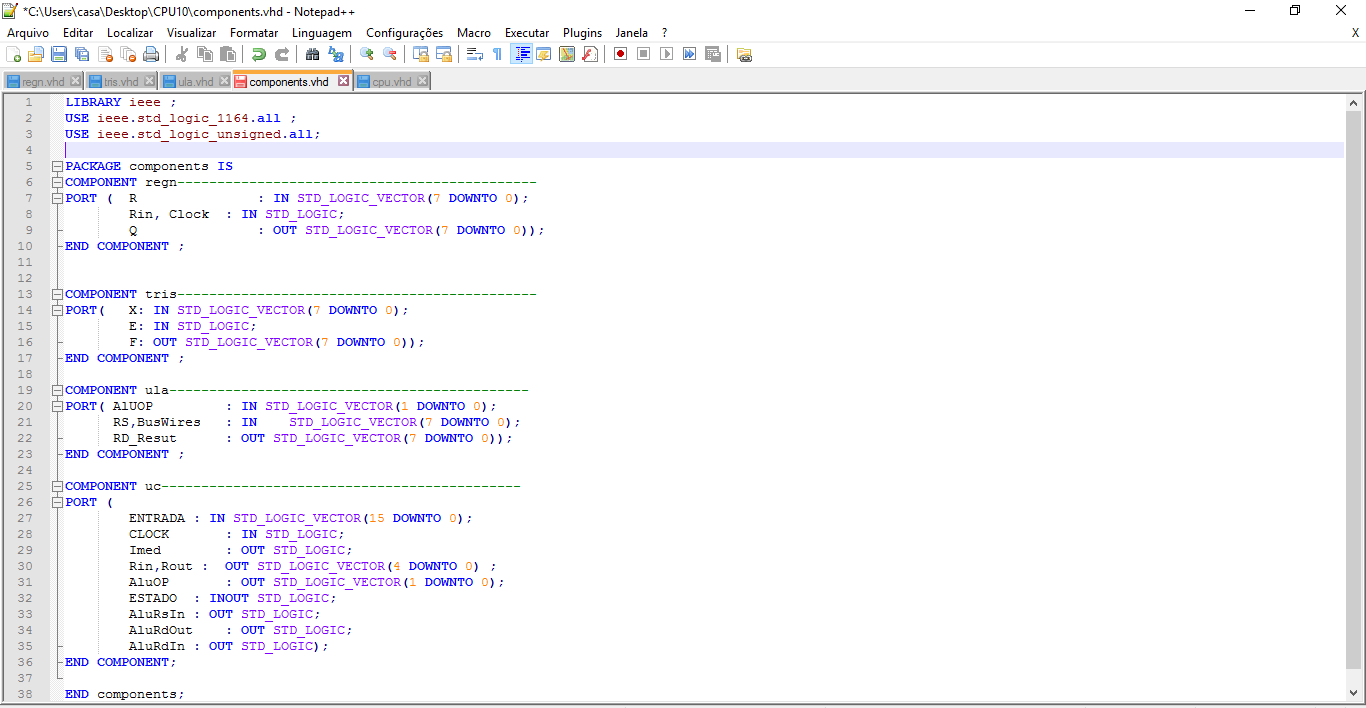
* 1. – Tris

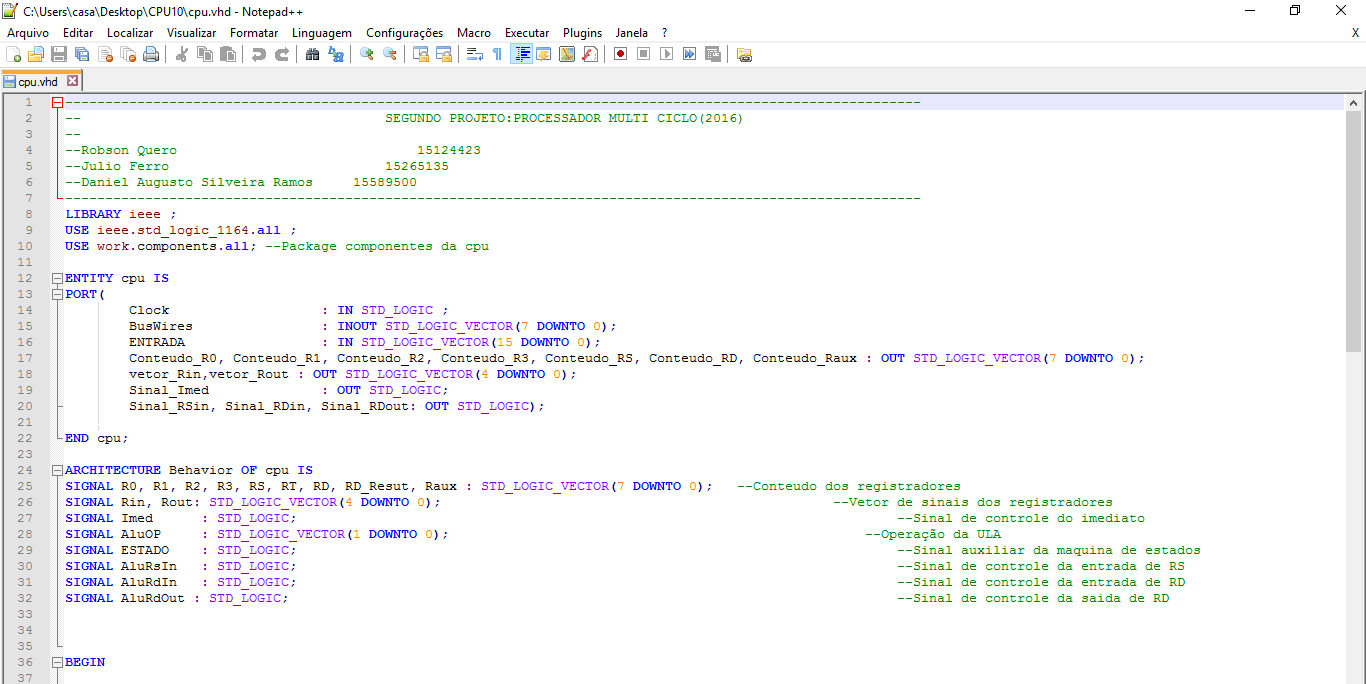


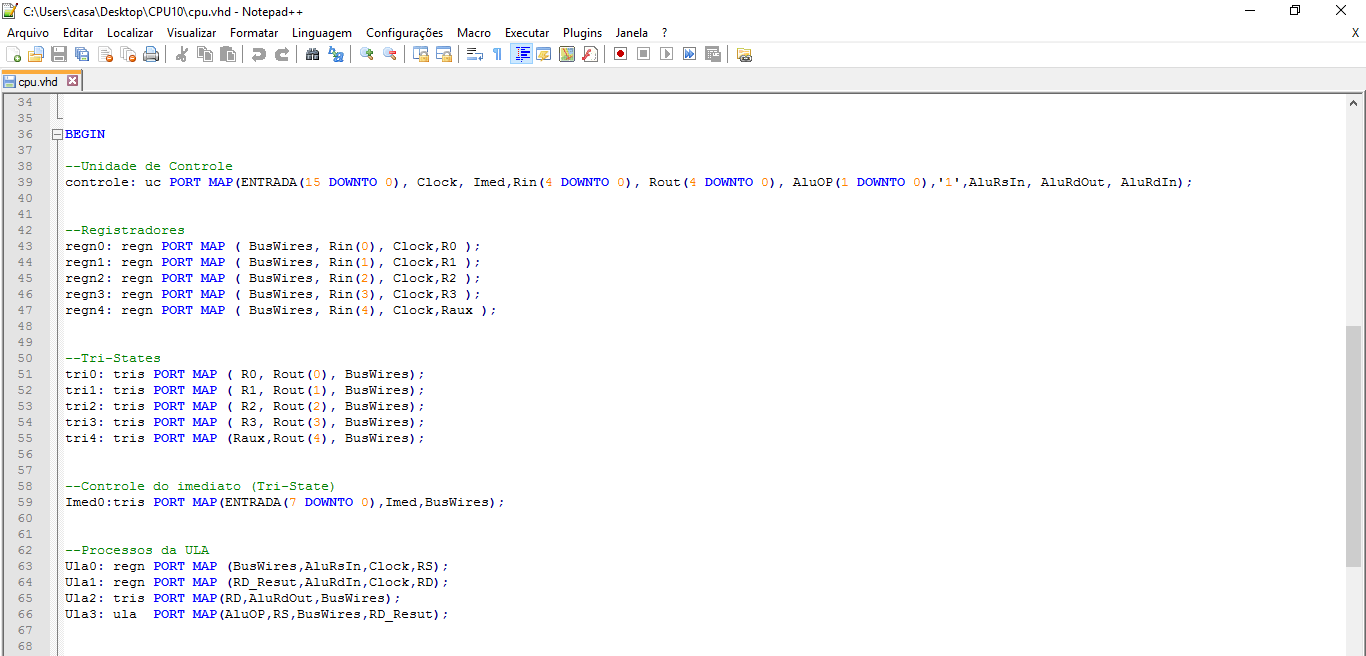
* 1. – ULA

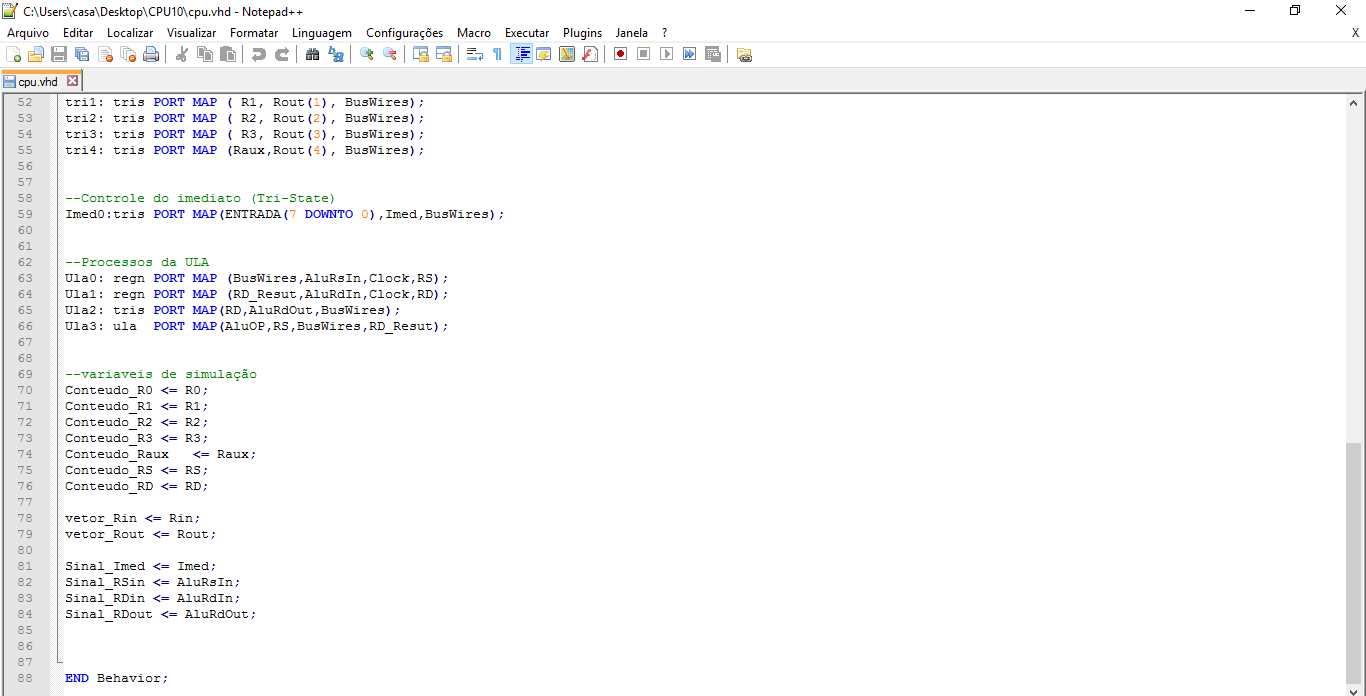


* 1. – CPU









* 1. – UC

