

**a) add $t1, $t0, $t2**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Instruction | RegDst | ALUSrc | Memto-  Reg | Reg  Write | Mem  Read | Mem  Write | Branch |
| add | 1 | 0 | 0 | 1 | 0 | 0 | 0 |

**c) addi $t3, $t0, 5**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Instruction | RegDst | ALUSrc | Memto-  Reg | Reg  Write | Mem  Read | Mem  Write | Branch |
| addi | 0 | 0 | 0 | 1 | 0 | 0 | 0 |

**e) lw $t0, 24($s0)**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Instruction | RegDst | ALUSrc | Memto-  Reg | Reg  Write | Mem  Read | Mem  Write | Branch |
| lw | 0 | 1 | 1 | 1 | 1 | 0 | 0 |

**f) sw $t1, 4($s1)**

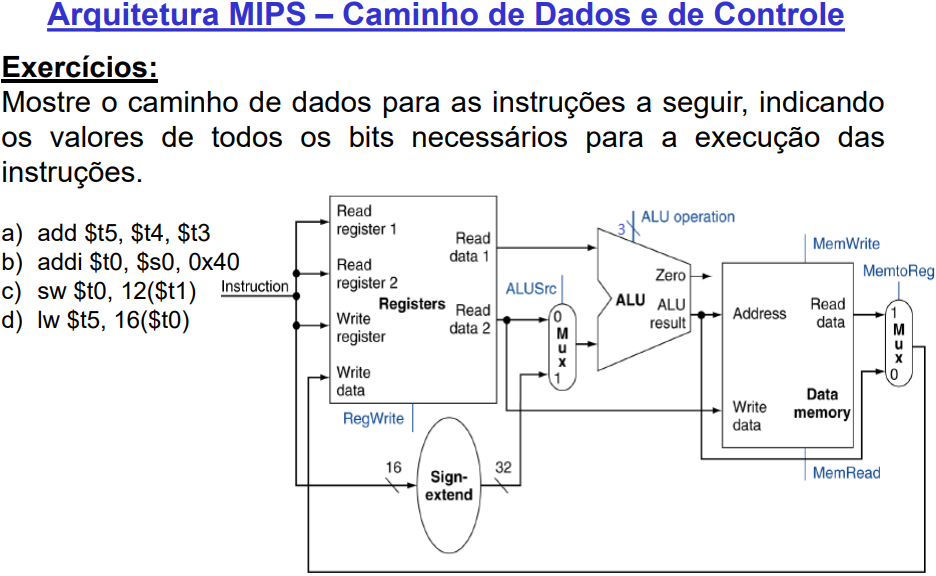
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Instruction | RegDst | ALUSrc | Memto-  Reg | Reg  Write | Mem  Read | Mem  Write | Branch |
| sw | X | 1 | X | 0 | 0 | 1 | 0 |

**g) beq $s0, $s1, 100**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Instruction | RegDst | ALUSrc | Memto-  Reg | Reg  Write | Mem  Read | Mem  Write | Branch |
| beq | X | 0 | X | 0 | 0 | 0 | 1 |

**h) j 350**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Instruction | RegDst | ALUSrc | Memto-  Reg | Reg  Write | Mem  Read | Mem  Write | Branch |
| j | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**a) add $t5, $t4, $t3**

1. (BUSCA) - Busca da instrução e incremento do PC

2. (DECODIFICAÇÃO) - $t4 e $t3 são lidos no banco de registradores e os sinais de controle são definidos

3. (ULA) - A ULA opera os dados de entrada

4. (WB) - O resultado da ALU é escrito no registrador designado ($t5).

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Instruction | RegDst | ALUSrc | Memto-  Reg | Reg  Write | Mem  Read | Mem  Write |
| add | 1 | 0 | 0 | 1 | 0 | 0 |

**b) addi $t0, $s0, 0x40**

1. (BUSCA) - Busca da instrução e incremento do PC

2. (DECODIFICAÇÃO) - Leitura do registrador ($s0)

3. (ULA) - Cálculo da soma do registrador ($s0) com a parte imediata da instrução

4. (WB) - O resultado da ALU é escrito no registrador designado ($t0).

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Instruction | RegDst | ALUSrc | Memto-  Reg | Reg  Write | Mem  Read | Mem  Write |
| addi | 0 | 0 | 0 | 1 | 0 | 0 |

**c) sw $t0, 12($t1)**

1. (BUSCA) - Busca da instrução e incremento do PC

2. (DECODIFICAÇÃO) - Leitura do registrador ($t1)

3. (ULA) - Cálculo da soma do registrador ($t1) com a parte imediata da instrução

4. (MEM) - A soma da ULA é usada como endereço para a memória de dados

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Instruction | RegDst | ALUSrc | Memto-  Reg | Reg  Write | Mem  Read | Mem  Write |
| sw | 1 | 1 | 0 | 0 | 0 | 1 |

**d) lw $t5, 16($t0)**

1. (BUSCA) - Busca da instrução e incremento do PC

2. (DECODIFICAÇÃO) - Leitura do registrador ($t0)

3. (ULA) - Cálculo da soma do registrador ($t0) com a parte imediata da instrução

4. (MEM) - A soma da ULA é usada como endereço para a memória de dados

5. (WB) - O dado da unidade de memória é escrito no banco de registradores no registrador destino ($t1)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Instruction | RegDst | ALUSrc | Memto-  Reg | Reg  Write | Mem  Read | Mem  Write |
| lw | 0 | 0 | 0 | 0 | 0 | 0 |

BUSCA / DECODIFICAÇÃO / CALCULO (ULA) / MEM (Memory Access) / WB (Write back)