O processador IA possui 04 registradores: $s0, $s1, $s2 e $s3. Assim como 2 formatos de instruçôes de 8bits cada:

\* instrução do tipo R (Operação aritmetica):

Formato na linguagem Quantum:

|  |  |  |
| --- | --- | --- |
| OPCODE | Reg1 | Reg2 |

Formato em codigo Binário

|  |  |  |
| --- | --- | --- |
| OPCODE | Reg1 | Reg2 |
| 4 Bits | 2 Bits | 2 Bits |
| 7-4 | 3-2 | 1-0 |

\* I(load e store):

Formato na linguagem Quantum:

|  |  |  |  |
| --- | --- | --- | --- |
| OPCODE | Reg1 | Reg2 | Endereço |

Formato em codigo Binário

|  |  |  |
| --- | --- | --- |
| OPCODE | Reg1 | Endereço |
| 4 Bits | 2 Bits | 2 Bits |
| 7-4 | 3-2 | 1-0 |

\* J (Jump):

Formato na linguagem Quantum:

|  |  |
| --- | --- |
| OPCODE | Endereço |

Formato em codigo Binário

|  |  |
| --- | --- |
| OPCODE | Endereço |
| 4 Bits | 4 Bits |
| 7-4 | 3-0 |

Tabela que mostra a lista de OPCODES UTILIZADAS PELO PROCESSADOR IA.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| OPCODE | INSTRUÇÃO | NOME | TIPO | EXEMPLO | SIGNIFICADO |
| 0000 | ADD | add | r | add $s0 , $s2,$s1 | $s0 = $s2 + $s1 |
| 0001 | SUB | sub | r | sub $s0 , $s2,$s1 | $s0 = $s2 - $s1 |
| 0010 | AND | and | r | and $s0 , $s2,$s1 | $s0 = $s2 & $s1 |
| 0011 | OR | or | r | or $s0 , $s2,$s1 | $s0 = $s2|$s1 |
| 0100 | MULTIPLICAÇÃO | mult | r | mult $s0 , $s2 | ( $s0 \* $s2 ) |
| 0101 | BEQ | beq | i | beq $s0 , $s2.25 | If ($s0 == $s2) go to 25 |
| 0110 | SLT | slt | i | slt $s0,$s2,$s1 | if($s2 < $s1) $s0=1 else $s0=0 |
| 0111 | LOAD IM | li | i | li $s0, 3 | $s0 = 3 |
| 1000 | LOAD | lw | i | lw $s0 , 4($s2) | $s0 =Memoria[$s2+4] |
| 1001 | STORE | sw | i | sw $s0 , 4 ($s2) | Memoria[$s2+4] = $s0 |
| 1010 | JUMP | jump | j | jump 1111 | Go to 1111 |
| 1111 | \*EXIT |  |  | Ele serve de encerramento ao um programa. e geralmente é de tipo J. |  |

implementação do processador ProIA foi utilizado a IDE: Quartus Prime version 17.0.

