**Bits decodificados da memória de controle**

| **PC** | **W** | **ALUSRC** | **WSRC** | **DISPLAY** | **ALUOP** | | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| **Operação** | **Op code** | **Memória de controle** |
| --- | --- | --- |
| nop | 0000 | 00000000 |
| li | 0001 | 01010000 |
| add | 0010 | 01000000 |
| sub | 0011 | 01000001 |
| mult | 0100 | 01000010 |
| and | 0101 | 01000011 |
| or | 0110 | 01000100 |
| xor | 0111 | 01000101 |
| sll | 1000 | 01000110 |
| addi | 1001 | 01100000 |
| subi | 1010 | 01100001 |
| jump | 1011 | 10000000 |
| branch on equal | 1100 | 10000001 |
| show | 1101 | 00001000 |
| halt | 1110 | 10000000 |

**Assembly:**

*li r1, 0*

*li, r2, 7*

*li r3, 18*

*li r4, 10*

*li r5, 0*

*loop: beq r4, r5, fim*

*mult r6, r5, r3*

*add r1, r1, r2*

*addi r5, r5, 1*

*show r1*

*j loop*

*fim: show r1*

*halt*

| Binário |
| --- |
| 00010001000000000000000000000000 |
| 00010010000000000000000000000111 |
| 00010011000000000000000000010010 |
| 00010100000000000000000000001010 |
| 00010101000000000000000000000000 |
| 11000100010100000000000000000000 |
| 01000110010100110000000000000000 |
| 00100001000100100000000000000000 |
| 10010101010100000000000000000001 |
| 11010000000100000000000000000000 |
| 10110000000000000000000000000000 |
| 11010000000100000000000000000000 |
| 11100000000000000000000000000000 |
|  |