|  |  |  |
| --- | --- | --- |
| Address | Binary Value | Assembly Language |
| 0x000 | 0001000000100100 | addi $2, $0, 0100 |
| 0x004 | 0001000000110001 | Addi $3, $0, 0001 |
| 0x008 | 0001000001000101 | Addi $4, $0, 0101 |
| 0x00C | 0001000001100001 | addi $6, $0, 0001 |
| 0x010 | 1110001000100100 | slli $2, $2, 4 |
| 0x014 | 1110001100110100 | slli $3, $3, 4 |
| 0x018 | 1111010001000011 | muli $4,$4,0011 |
| 0x01C | 1110011001100100 | Slli $6, $6, 4 |
| 0x020 | 0001000001110101 | Addi $7, $0, 0101 |
| 0x024 | 1110001100110100 | Slli $3, $3, 4 |
| 0x028 | 1110010001010100 | slli $5,$4,4 |
| 0x02C | 0001000010000000 | addi $8, $0, 0000 |
| 0x030 | 1110011011000100 | Slli $12,$6,4 |
| 0x034 | 0001001100110001 | Addi $3, $3,0001 |
| 0x038 | 0010010001011110 | add $14, $4, $5 |
| 0x03C | 1110001100110100 | Slli $3, $3, 4 |
| 0x040 | 1110111011010100 | Slli $13, $14, 0004 |
| 0x044 | 1010000001110001 | While: slt $1, $0, $7 |
| 0x048 | 0001000010010001 | Do: addi $9, $0, 0001 |
| 0x04C | 0101011010000000 | lw $8, 0[$6] |
| 0x050 | 1110110111110100 | Slli $15,$13,0004 |
| 0x054 | 1011000000010001 | bne $1, $0, 0001 |
| 0x058 | 0111000000010000 | j return |
| 0x05C | 0011011110010111 | sub $7, $7, $9 |
| 0x060 | 1010110010000001 | slt $1, $12, $8 |
| 0x064 | 0001000010010100 | then: addi $9, $9, 0100 |
| 0x068 | 1011000000010001 | bne $1, $0, 0001 |
| 0x06C | 0111000000000101 | J else |
| 0x070 | 0110011011110000 | sw $15, 0[$6] |
| 0x074 | 1110100110010001 | Slli $9, $9, 0001 |
| 0x078 | 1000001010010010 | div $2, $2, $9 |
| 0x07C | 1001001100100011 | or $3, $3, $2 |
| 0x080 | 0111000000000011 | J end if |
| 0x084 | 1111010001000100 | Else: muli $4, $4, 0100 |
| 0x088 | 0100010101000101 | xor $5, $5, $4 |
| 0x08C | 0110011011100000 | sw $14, 0[$6] |
| 0x090 | 0001011001100010 | End if: addi $6, $6, 0010 |
| 0x094 | 0111111111101011 | j While |
| 0x098 | 0000000000000000 | return |