

Lab#7 - Arquitetura de Computadores  
Pedro Francescon Cittolin e Viviane Lima Bonfim

Codificação das instruções

**r** = bit registrador  
destino ou único

**R** = bit registrador  
secundário (reg1)

**i** = bit de constante

**d** = bit de  
deslocamento

**opcode**

ADD reg2,reg1	=> 0001 rrr RRR 00000
ADD reg2,immediate	=> 0010 rrr iiiiiiiii
ADDI reg2,reg1,immediate	=> 1101 rrr RRR iiiii
SUB reg2,reg1	=> 0011 rrr RRR 00000
SUB reg2,immediate	=> 0100 rrr iiiiiiiii
MOV reg2,reg1	=> 0101 rrr RRR 00000
JMP reg2	=> 0110 rrr 00000000
BL immediate	=> 1010 iiiiiiiiiiiii
LD.W disp[reg1],reg2	=> 1001 rrr RRR iiiii
ST.W reg2,disp[reg1]	=> 0111 rrr RRR iiiii

Assembly

1. ADD reg1,0
2. ADD reg2,30
3. ST.W reg2,10[reg1] # salva 30 no endereço 10
4. ADD reg3,5
5. LD.W 5[reg3],reg4 # valor da RAM no endr 10 => reg4
6. MOV reg7,reg4
7. ST.W reg7,1[reg1]
8. ADD reg1,1
9. LD.W 0[reg1],reg6
10. MOV reg2,reg6
11. MOV reg1,reg2

Hexadecimal

1. 0010 001 00000000
2. 0010 010 00011110
3. 0111 010 001 01010
4. 0010 011 00000101
5. 1001 100 011 00101
6. 0101 111 100 00000
7. 0111 111 001 00001

8. 0010 001 00000001  
9. 1001 110 001 00000  
10. 0101 010 110 00000  
11. 0101 001 010 00000