

Formato das instruções

ADDI = $[[11:8] [7:4] [3:0]]$, onde do bit 11 ao 8 é o registrador destino, do bit 7 ao 4 é lugar do imediato, e do 3 ao 0 o opcode.

Para o ADD e as demais operações do tipo R = $[[11:8] [7:4] [3:0]]$, onde o bit 11 ao 7 está o registrador destino que é o mesmo registrador que será feita a operação, do 7 ao 4 o segundo registrador da operação e do bit 3 ao bit 0, significa o opcode, ficando assim no formato $rs1 = rs1 + rs2$.

Formato das instruções:

R-type: operação entre registradores

Campo	Bits
opcode	4
rs1	4
rs2	4
rd = rs1	4

I-type: imediato ou memória

Campo	Bits
opcode	4
rd = rs1	4
imm	4

J-type: salto

Campo	Bits
opcode	4
offset	8

Instruções

NOT	R	not rs2	0000
AND	R	rs1 = rs1 and rs2	0001
OR	R	rs1 = rs1 or rs2	0010
ADD	R	rs1 = rs1 + rs2	0011
SUB	R	rs1 = rs1 - rs2	0100
MUL	R	rs1 = rs1 * rs2	0101
ADDI	I	rs1 = 0 + imm	0110
LW	I	rs1 = M [rs1 + imm][0:3]	0111
SW	I	M [rs1 + imm][0:3] = rs1	1000
BEQ	I	se rs1 == rs2 pc + imm	1001
BLT	I	se rs1 < rs2 pc + imm	1010
JAL	J	Salto com link	1011
J	J	pc + imm	1100

Código para testes

addi r1, r0, 3 → 0001 0011 0110 → 136
addi r2, r0, 5 → 0010 0101 0110 → 256
add r2, r2, r1 → 0010 0001 0011 → 213
sub r2, r2, r1 → 0010 0001 0100 → 214
mul r2, r2, r1 → 0010 0001 0101 → 215
addi r4, r0, 7 → 0100 0111 0110 → 476
and r4, r4, r1 → 0100 0001 0001 → 411
or r4, r4, r1 → 0100 0001 0010 → 412
addi r3, r0, 4 → 0011 0100 0110 → 346
sw r3, r3, 1 → 0011 0001 1000 → 318
lw r4, r4, 5 → 0100 0101 0111 → 457
sub r3, r3, r4 → 0011 0100 0100 → 344
beq r0, r3, 2 → 0011 0010 1001 → 329
addi r1, r0, 5 → 0001 0101 0110 → 156
sub r1, r1, r3 → 0001 0011 0100 → 134
j r0, 2 → 00000010 1100 → 2c
addi r1, r0, 8 → 0001 1000 0110 → 186
blt r0, r1, -4 → 0001 0100 1010 →