

command

= option

control

Module 5

- 2.20] srl \$t2, \$t0, 10 # move bits 16-11 down to the very right, Bits we want at the start)
sll \$t2, \$t2, 26 # shift \$t2 26 bits to the left, bits 0-25 will be zeros.
sll \$t1, \$t1, 6 # cut off \$t1 bits 26-31
srl \$t1, \$t1, 6 # shift \$t1 6 bits to the right, so last 6 bits are zeros
add \$t1, \$t1, \$t2 # replaces the zeroes in \$t1 in bits 26-31 with the 6 bits in \$t2

2.25.1] I format to hold an address, and two registers for comparison.

2.25.2] LOOP : slt \$t1, \$zero, \$t2 # if ~~\$t2 < 0~~ \$t2 < \$t2, \$t1 = 1, else \$t1 = 0
beg \$t3, \$zero, EXIT
addi \$t1, \$t1, -1 # decrement, no subi

j Loop # if R[rs] > 0, go to Loop

EXIT :

| Loop | \$t1 | \$t2 |
|------|------|------------|
| 1 | 9 | 2 |
| 2 | 8 | 4 |
| 3 | 7 | 6 |
| 4 | 6 | 8 |
| 5 | 5 | 10 |
| 6 | 4 | 12 |
| 7 | 3 | 14 |
| 8 | 2 | 16 |
| 9 | 1 | 18 |
| 10 | 0 | 20 |
| 11 | fail | go to DONE |

$$[\$t2 = 20]$$

2.26.1] while (i > 0) {
 i = i - 1
 B = B + 2

}

2.26.2] if N = 0 = \$t1 then only 2 instructions are executed.

LOOP sll \$t2, \$0, \$t1 ← first
beg \$t2, \$0, DONE ← second.

If N = 1, 5? instruction

N = 2, 12.

so Number of instructions = $(\$t1 \cdot 5) + 2$ or $\downarrow N$