Consider the following fragment of C Code:

for(i=1; i < 9; i++) {

sum \*= 2;

if (sum < 400)

A[i] = A[i+1]

else

A[i] = A[i-1]

}

Assume that A is an array of words and that the base address of A is in $s0, sum is in $s1, and i is in $s2.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Instr # | Address | Label | Instruction | Comment |
| 0 | 8000 |  | addi $s2, $zero, 1 | #i = 1 |
| 1 | 8004 | loop: | slti $t0, $s2, 9 | #t0 = 1 if i<9 |
| 2 | 8008 |  | beq $t0, $zero, exit | #goto exit if t0=0 |
| 3 | 8012 |  | sll $s1, $s1, 1 | #sum = sum \* 2 |
| 4 | 8016 |  | sll $t0, $s2, 2 | #t0 = i \* 4 |
| 5 | 8020 |  | add $t0, $t0, $s0 | #t0 = t0 + base A |
| 6 | 8024 |  | slti $t1, $s1, 400 | #t1=1 if sum<400 |
| 7 | 8028 |  | beq $t1, $zero, else | #goto else if t1=0 |
| 8 | 8032 |  | lw $t2, 4($t0) | #t2 = A[i+1] |
| 9 | 8036 |  | sw $t2, 0($t0) | #A[i] = t2 |
| 10 | 8040 |  | j finif | #goto fin if |
| 11 | 8044 | else: | lw $t2, -4($t0) | #t2 = A[i-1] |
| 12 | 8048 |  | sw $t2, 0($t0) | #A[i] = t2 |
| 13 | 8052 | finif: | addi $s2, $s2, 1 | #i++ |
| 14 | 8056 |  | j loop | #goto loop |
| 15 | 8060 | exit: | nop | #end of for loop |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Instr # | Address | Label | Instruction | Comment |
| 0 | 8000 |  | addi $s2, $zero, 1 | #i = 1 |
| 1 | 8004 | loop: | slti $t0, $s2, 9 | #t0 = 1 if i<9 |
| 2 | 8008 |  | beq $t0, $zero, 12 | #goto exit if t0=0 |
| 3 | 8012 |  | sll $s1, $s1, 1 | #sum = sum \* 2 |
| 4 | 8016 |  | sll $t0, $s2, 2 | #t0 = i \* 4 |
| 5 | 8020 |  | add $t0, $t0, $s0 | #t0 = t0 + base A |
| 6 | 8024 |  | slti $t1, $s1, 400 | #t1=1 if sum<400 |
| 7 | 8028 |  | beq $t1, $zero, 3 | #goto else if t1=0 |
| 8 | 8032 |  | lw $t2, 4($t0) | #t2 = A[i+1] |
| 9 | 8036 |  | sw $t2, 0($t0) | #A[i] = t2 |
| 10 | 8040 |  | j 2013 | #goto fin if |
| 11 | 8044 | else: | lw $t2, -4($t0) | #t2 = A[i-1] |
| 12 | 8048 |  | sw $t2, 0($t0) | #A[i] = t2 |
| 13 | 8052 | finif: | addi $s2, $s2, 1 | #i++ |
| 14 | 8056 |  | j 2001 | #goto loop |
| 15 | 8060 | exit: | nop | #end of for loop |

PC-Relative Addressing: PC = PC + 4 + 4\*offset

Pseudodirect Addressing: PC = PC[31-28] : address \* 4

PC = 0000 : 2013 \* 4 = 0000 : 8052 = 8052