Decrement and Branch Instruction

It is used for implementing Loop control

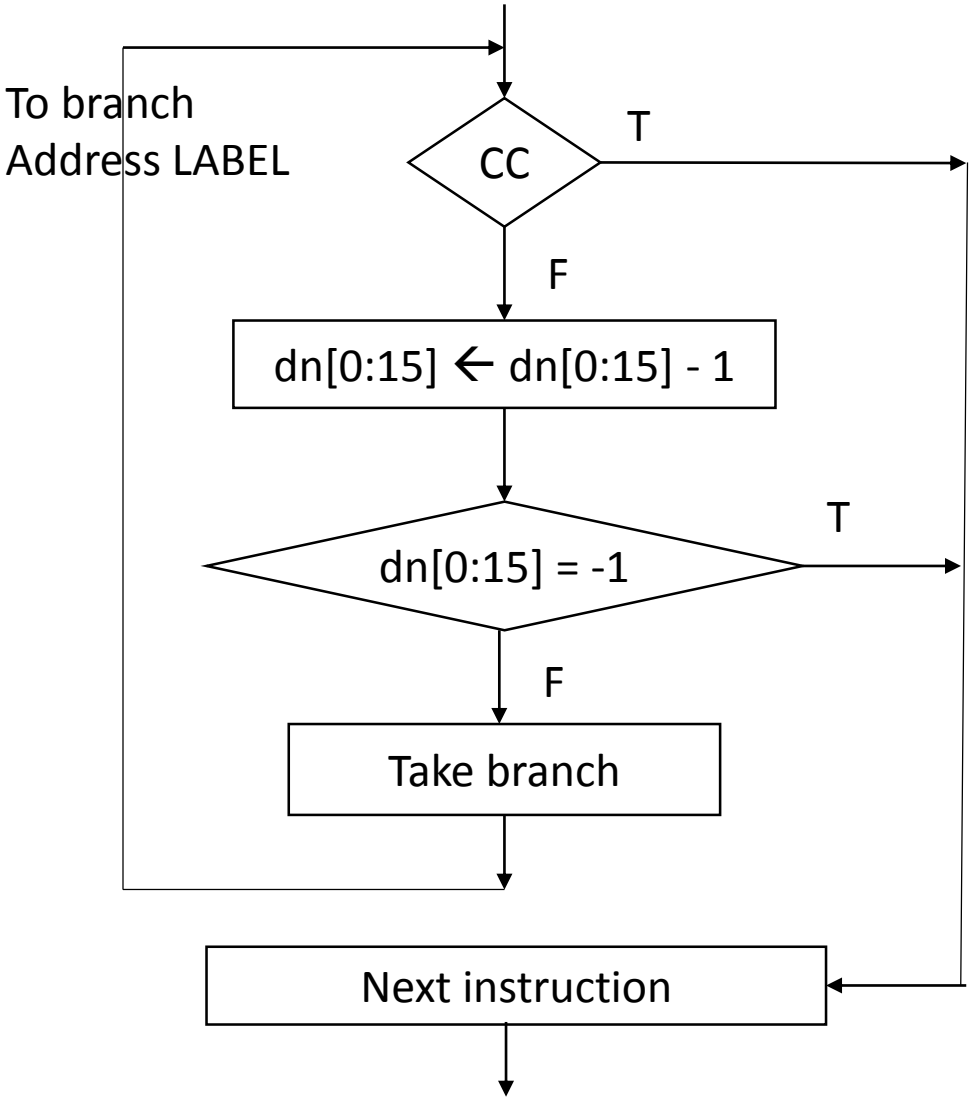
dbcc dn, LABEL

Example: dbgt (decrement and branch unless greater than)

The branch instruction used in these instructions is opposite to the way it is used in other branch instructions

- If condition specified by cc is satisfied, then the next instruction is executed
- If condition not satisfied, then
  - $dn_{16-bits} \leftarrow dn_{16-bits} - 1$
  - If $dn = -1$, next instruction is executed
  - If $dn \neq -1$, branch is made to LABEL

Decrement and Branch Instruction cont.




```
there   add.w d1,d2
        .....
        .....
        .....
        dbeq d3, there
here    sub d1,d2
```

If Z bit is 1, CC is True

- here is executed

If Z bit is 0, CC is False

d3 = d3 -1

If d3 = -1, here is executed

else there is executed

Decrement and Branch Instruction cont.

```
        dbcc d3, LOOP
        next instruction
```

Is equivalent to the sequence

```
        bcc NEXT
        subq #1, d3
        bge LOOP
NEXT    next instruction
```

dbra (decrement and branch always)

- Loop is executed a predetermined number of times

```
        moveq #31,d2
Loop    add ….
        sub ….
        dbra d2, Loop
```

When a loop is to be executed *n* times, the number (*n-1*) is loaded into dn register, and

```
        dbra dn, loop
```

Is executed