Author(s) : H. Grote **Library:** KERNLIB
Submitter : **Submitted:** 01.12.1980
Language : CDC: Fortran and Compass, IBM: Assembler **Revised:**

Move Bit String

`BTMOVE` moves a contiguous string of `N` bits from any position in memory to any other position. Bits are numbered from left to right (most significant to least significant within words) and may be across word boundaries.

Structure:

`SUBROUTINE` subprogram
User Entry Names: `BTMOVE`
External References: `UCOPY` (V301) (CDC only)

Usage:

```
CALL BTMOVE(SOURCE,ISRC,TARGET,ITGT,N)
```

moves the string of `N` contiguous bits starting at position `ISRC` in word or array `SOURCE` to position `ITGT` in word or array `TARGET`. The other bits in `TARGET` are not changed, nor is `SOURCE`.

Notes:

Source and target strings must not overlap in storage, else the results may be unpredictable.

Examples:

IBM:
Move the highest bit (sign-bit) in word `A` to the lowest position of `I` so that it can be treated as an integer:

```
REAL A
INTEGER*4 I
I=0
CALL BTMOVE(A,1,I,32,1)
```

CDC:
Pack the five integers of array I5(5) into one word `IPACK`, using 12 bits per packed integer:

```
   DIMENSION I5(5)
   IPOS=1
   DO 1 I = 1,5
   CALL BTMOVE(I5(I),49,IPACK,IPOS,12)
 1 IPOS=IPOS+12
```

Move a string of 20 characters from positions 41-60 in array `A` to positions 7-26 in array `B`:

```
DIMENSION A(6),B(3)
CALL BTMOVE(A,241,B,37,120)
```

•