Author(s) : M. Metcalf **Library:** KERNLIB or Fortran library
Submitter :
Submitted: 10.12.1984
Language : Fortran with ISA extensions **Revised:** 18.10.1985

Handling Bits and Bytes, Bit Zero the Least Significant

BITPAK handles bits and bytes in a single word, with bit zero being the least significant bit.

Structure:

SUBROUTINE and FUNCTION subprograms
User Entry Names: IOR, IAND, NOT, IEOR, ISHFT, ISHFTC, IBITS, MVBITS, BTEST, IBSET, IBCLR

Usage:

A numeric storage unit is considered to consist of a string of bits numbered from *right to left*, starting at *zero*. The standard MIL-STD-l753 defines ll bit manipulation functions on such units, 8 of which are the ANSI/ISA functions found as intrinsic functions in many compilers. This package complements the functions available in compilers, ensuring that the full range is available on all machines. This description includes all the functions for the sake of completeness.

Logical operations:

IOR(M,N)	provides the inclusive OR of the two integer arguments.
IAND(M,N)	provides the logical AND of the two integer arguments.
NOT(M)	provides the logical complement of the integer argument.
IEOR(M,N)	provides the exclusive OR of the two integer arguments.

Shift operations:

$$\text{A shift count K specifies} \begin{cases} \text{a left shift} & \text{for } K > 0 \\ \text{no shift} & \text{for } K = 0 \\ \text{a right shift} & \text{for } K < 0. \end{cases}$$

ISHFT(M,K)	provides the value of the integer argument M with the bits shifted. Bits shifted out to the left or right are lost, and zeros are shifted in from the opposite end.
ISHFTC(M,K,IC)	provides the value of the integer argument M with the rightmost IC bits shifted, and the remaining bits untouched. The shift is circular; no bits are lost.

Bit subfields:

IBITS(M,I,LEN)	provides, right justified, the value of the LEN bits of the integer argument M, starting from position I.
CALL MVBITS(M,I,LEN,N,J)	moves LEN bits of integer argument M, starting at position I, to the integer argument N, starting at position J. All other bits of N are left untouched. The arguments M and N may refer to the same numeric storage unit.

Bit testing:

`BTEST(N,I)`	has the value `.TRUE.` if bit `I` of the integer argument `N` is set, and `.FALSE.` otherwise. Note that many compilers require `BTEST` to be declared type `LOGICAL`.
`IBSET(N,I)`	has the value of the integer argument `N` with bit `I` set to `1`.
`IBCLR(N,I)`	has the value of the integer argument `N` with bit `I` set to `0`.

Notes:

If bits are specified outside the range of one numeric storage unit, or if fields are specified which are longer than one numeric storage unit or zero, or if shifts are specified which are longer than the fields being shifted, then the results are undefined.

•