1 SUMMARY

This function supplies real-valued machine constants relating to the floating-point storage and arithmetic of the machine in use.

A nonzero floating-point number is stored in the form $\pm m \beta^e$, where $\beta$ is known as the base (or radix) of the arithmetic, $m$ is the mantissa (or significand or fraction) and $e$ is the exponent (or characteristic). The mantissa is usually normalized so that any floating-point number has a unique representation. Individual machines differ in the way that the normalization is performed. The exponent is stored as a sequence of binary digits (bits); the sign of the exponent either occupies one of these digits, or, more commonly, the actual value of the exponent is obtained by adding the stored binary representation to a fixed negative bias. The mantissa is represented as $m = \sum_{i}^{n} m_i \beta^{i+j}$, where $0 \leq m_i < \beta$, $j$ is usually 0 or 1 and $m$ is usually normalized so that $m_i > 0$. (Under special circumstances, $m_i$ may be zero; such circumstances are typically associated with implementations of gradual underflow on a particular machine.)


2 HOW TO USE THE PACKAGE

Two versions of the function are available corresponding to the use of single and double precision arithmetic.

2.1 Argument list

The single precision version

\[ \text{RNUM} = \text{FD15A(MC)} \]

The double precision version

\[ \text{DNUM} = \text{FD15AD(MC)} \]

MC is a CHARACTER variable which must be set by the user to select which one of the real machine constants is required. Possible values of MC and the corresponding constants are:

1. FD15A/FD15AD (‘E’) is the smallest REAL/DOUBLE PRECISION number of the form $\beta^e$ such that $1.0 + \beta$ and 1.0 are different stored numbers. On most machines $i= 1 - n$. This value is that returned by the Numeric Inquiry Function EPSILON in the ISO Fortran 90 standard and is commonly called the machine precision.

2. FD15A/FD15AD (‘T’) is a close approximation to the smallest positive REAL/DOUBLE PRECISION number which may be stored on the machine to full precision, i.e., for which $m_i > 0$. This number is normally $\beta^{-e+1}$, where $e_{\text{min}}$ is the smallest allowable value of the exponent. This value is that returned by the Numeric Inquiry Function TINY in the ISO Fortran 90 standard.

3. FD15A/FD15AD (‘H’) is a close approximation to the largest finite positive REAL/DOUBLE PRECISION number which may be stored on the machine. This number is normally $\beta^{-e+1}(1.0 - \beta^{-n})$, where $e_{\text{max}}$ is the largest allowable value of the exponent, and is the value returned by the Numeric Inquiry Function HUGE in the ISO Fortran 90 standard.

4. FD15A/FD15AD (‘R’) gives $\beta$, the base used for the floating-point arithmetic. This is the same value returned as an integer by the ISO Fortran 90 standard Numeric Inquiry Function RADIX but here it is returned as a REAL/DOUBLE PRECISION number.
MC is not altered by the function. **Restriction:** it must be one of the set \[\{'E','T','H','R'\}\]. Note: FD15 does not indicate an error when MC is out of range but returns the value zero.

FD15A/FD15AD is a REAL (DOUBLE PRECISION in the D version) function whose value will be set to the required machine constant.

### 3 GENERAL INFORMATION

**Use of common:** None.

**Other routines called directly:** None.

**Input/output:** None.

**Restrictions:** MC must be one of the set \[\{'E','T','H','R'\}\].

### 4 METHOD

The constants have been set by the original implementor of HSL on your machine. Further detail of the machine representation of floating-point numbers may be found in, for instance, the Dictionary of Computing (Oxford University Press, 1983).

### 5 EXAMPLE OF USE

This is a very simple example which lists the four machine constants for the current machine.

```fortran
PROGRAM MAIN
  DOUBLE PRECISION DNUM, FD15AD
  INTEGER INUM
  DNUM = FD15AD( 'E' )
  WRITE( 6, 2000 ) DNUM
  DNUM = FD15AD( 'T' )
  WRITE( 6, 2010 ) DNUM
  DNUM = FD15AD( 'H' )
  WRITE( 6, 2020 ) DNUM
  INUM = INT( FD15AD( 'R' ) )
  WRITE( 6, 2030 ) INUM
2000 FORMAT( ' Machine precision (double precision) = ', 1P, E12.4 )
2010 FORMAT( ' Smallest floating-point number (double precision) = ',
              ' 1P, E12.4 ' )
2020 FORMAT( ' Largest floating-point number (double precision) = ',
              ' 1P, E12.4 ' )
2030 FORMAT( ' Base used for floating-point (double precision) = ',
              ' 110 ' )
END
```

This produces the following output

```
Machine precision (double precision) = 2.2204E-16
Smallest floating-point number (double precision) = 2.2251-308
Largest floating-point number (double precision) = 1.7977+308
Base used for floating-point (double precision) = 2
```