

# NAG Fortran Library Routine Document

## X04AAF

**Note:** before using this routine, please read the Users' Note for your implementation to check the interpretation of *bold italicised* terms and other implementation-dependent details.

### 1 Purpose

X04AAF returns the value of the current error message unit number, or sets the current error message unit number to a new value.

### 2 Specification

```
SUBROUTINE X04AAF ( IFLAG, NERR )
INTEGER           IFLAG, NERR
```

### 3 Description

This routine enables those library routines which output error messages, to determine the number of the output unit to which the error messages are to be sent; in this case X04AAF is called with IFLAG = 0. X04AAF may also be called with IFLAG = 1 to set the unit number to a specified value. Otherwise a default value (stated in the Users' Note for your implementation) is returned.

Records written to this output unit by other library routines are at most 80 characters long (including a line-printer carriage control character).

Note that if the unit number is set < 0, no messages will be output.

### 4 References

None.

### 5 Parameters

1: IFLAG – INTEGER *Input*

*On entry:* the action to be taken (see NERR).

*Constraint:* IFLAG = 0 or 1.

2: NERR – INTEGER *Input/Output*

*On entry:*

if IFLAG = 0, NERR need not be set;

if IFLAG = 1, NERR must specify the new error message unit number.

*On exit:*

if IFLAG = 0, NERR is set to the current error message unit number,

if IFLAG = 1, NERR is unchanged.

Note that Fortran unit numbers must be positive or zero. If NERR is set < 0, output of error messages is totally suppressed.

### 6 Error Indicators and Warnings

None.

## 7 Accuracy

Not applicable.

## 8 Further Comments

The time taken by the routine is negligible.

## 9 Example

In this example X04AAF is called by the user's main program to make the error message from the routine DUMMY appear on the same unit as the rest of the output (unit 6). Normally a NAG Fortran Library routine with an IFAIL parameter (see Chapter P01) would take the place of DUMMY.

### 9.1 Program Text

**Note:** the listing of the example program presented below uses *bold italicised* terms to denote precision-dependent details. Please read the Users' Note for your implementation to check the interpretation of these terms. As explained in the Essential Introduction to this manual, the results produced may not be identical for all implementations.

```
*      X04AAF Example Program Text
*      Mark 14 Revised.  NAG Copyright 1989.
*      .. Parameters ..
      INTEGER          NOUT
      PARAMETER       (NOUT=6)
*      .. External Subroutines ..
      EXTERNAL        DUMMY, X04AAF
*      .. Executable Statements ..
      WRITE (NOUT,*) 'X04AAF Example Program Results'
*
      CALL X04AAF(1,NOUT)
      CALL DUMMY
*
      STOP
      END
*
      SUBROUTINE DUMMY
*      .. Local Scalars ..
      INTEGER          NERR
*      .. External Subroutines ..
      EXTERNAL        X04AAF
*      .. Executable Statements ..
      CALL X04AAF(0,NERR)
      WRITE (NERR,*)
      WRITE (NERR,*) 'This is a dummy error message'
      RETURN
      END
```

### 9.2 Program Data

None.

### 9.3 Program Results

```
X04AAF Example Program Results
```

```
This is a dummy error message
```

---