

NAG Library Routine Document

F06SFF (ZTRMV)

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

F06SFF (ZTRMV) computes the matrix-vector product for a complex triangular matrix, its transpose or its conjugate transpose.

2 Specification

```
SUBROUTINE F06SFF (UPLO, TRANS, DIAG, N, A, LDA, X, INCX)
  INTEGER          N, LDA, INCX
  COMPLEX (KIND=nag_wp) A(LDA,*), X(*)
  CHARACTER(1)    UPLO, TRANS, DIAG
```

The routine may be called by its BLAS name *ztrmv*.

3 Description

F06SFF (ZTRMV) performs one of the matrix-vector operations

$$x \leftarrow Ax, \quad x \leftarrow A^T x \quad \text{or} \quad x \leftarrow A^H x,$$

where A is an n by n complex triangular matrix, and x is an n -element complex vector.

4 References

None.

5 Arguments

- 1: UPLO – CHARACTER(1) *Input*
On entry: specifies whether A is upper or lower triangular.
 UPLO = 'U'
 A is upper triangular.
 UPLO = 'L'
 A is lower triangular.
Constraint: UPLO = 'U' or 'L'.
- 2: TRANS – CHARACTER(1) *Input*
On entry: specifies the operation to be performed.
 TRANS = 'N'
 $x \leftarrow Ax$.
 TRANS = 'T'
 $x \leftarrow A^T x$.
 TRANS = 'C'
 $x \leftarrow A^H x$.
Constraint: TRANS = 'N', 'T' or 'C'.

- 3: DIAG – CHARACTER(1) *Input*
On entry: specifies whether A has nonunit or unit diagonal elements.
 DIAG = 'N'
 The diagonal elements are stored explicitly.
 DIAG = 'U'
 The diagonal elements are assumed to be 1, and are not referenced.
Constraint: DIAG = 'N' or 'U'.
- 4: N – INTEGER *Input*
On entry: n , the order of the matrix A .
Constraint: $N \geq 0$.
- 5: A(LDA,*) – COMPLEX (KIND=nag_wp) array *Input*
Note: the second dimension of the array A must be at least N .
On entry: the n by n triangular matrix A .
 If UPLO = 'U', A is upper triangular and the elements of the array below the diagonal are not referenced.
 If UPLO = 'L', A is lower triangular and the elements of the array above the diagonal are not referenced.
 If DIAG = 'U', the diagonal elements of A are assumed to be 1, and are not referenced.
- 6: LDA – INTEGER *Input*
On entry: the first dimension of the array A as declared in the (sub)program from which F06SFF (ZTRMV) is called.
Constraint: $LDA \geq \max(1, N)$.
- 7: X(*) – COMPLEX (KIND=nag_wp) array *Input/Output*
Note: the dimension of the array X must be at least $\max(1, 1 + (N - 1) \times |\text{INCX}|)$.
On entry: the vector x .
 If $\text{INCX} > 0$, x_i must be stored in $X(1 + (i-1) \times \text{INCX})$, for $i = 1, 2, \dots, N$.
 If $\text{INCX} < 0$, x_i must be stored in $X(1 - (N-i) \times \text{INCX})$, for $i = 1, 2, \dots, N$.
On exit: the updated vector x stored in the array elements used to supply the original vector x .
- 8: INCX – INTEGER *Input*
On entry: the increment in the subscripts of X between successive elements of x .
Constraint: $\text{INCX} \neq 0$.

6 Error Indicators and Warnings

None.

7 Accuracy

Not applicable.

8 Parallelism and Performance

F06SFF (ZTRMV) is not threaded in any implementation.

9 Further Comments

None.

10 Example

None.
