F01ZDF – NAG Fortran Library Routine Document

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

F01ZDF copies a complex band matrix stored in a packed array into an unpacked array, or vice versa.

2 Specification

```
SUBROUTINE F01ZDF(JOB, M, N, KL, KU, A, LDA, B, LDB, IFAIL)
INTEGER          M, N, KL, KU, LDA, LDB, IFAIL
complex          A(LDA,N), B(LDB,*)
CHARACTER*1      JOB
```

3 Description

F01ZDF unpacks a band matrix that is stored in a packed array, or packs a band matrix that is stored in an unpacked array. The band matrix has m rows, n columns, k_l non-zero sub-diagonals, and k_u non-zero super-diagonals. This routine is intended for possible use in conjunction with routines from the F06 Chapter Introduction, the F07 Chapter Introduction or the F08 Chapter Introduction, where routines that use band matrices store them in the packed form described below.

4 References

None.

5 Parameters

1: JOB — CHARACTER*1 *Input*

On entry: specifies whether the band matrix is to be packed or unpacked, as follows:

JOB = 'P' (Pack) the band matrix is to be packed into array B.

JOB = 'U' (Unpack) the band matrix is to be unpacked into array A.

Constraint: JOB must be one of 'P' or 'U'.

2: M — INTEGER *Input*

3: N — INTEGER *Input*

On entry: m and n, the number of rows and columns of the band matrix, respectively.

Constraint: M, N > 0.

4: KL — INTEGER *Input*

On entry: k_l, the number of sub-diagonals of the band matrix.

Constraint: KL ≥ 0.

5: KU — INTEGER *Input*

On entry: k_u, the number of super-diagonals of the band matrix.

Constraint: KU ≥ 0.

6: A(LDA,N) — ***complex*** array *Input/Output*

On entry: if JOB = 'P', then the leading m by n part of A must contain the band matrix stored in unpacked form. Elements of the array that lie outside the banded part of the matrix are not referenced and need not be assigned.

On exit: if JOB = 'U', then the leading m by n part of A contains the band matrix stored in unpacked form. Elements of the leading m by n part of A that are not within the banded part of the matrix are assigned the value zero.

7: LDA — INTEGER *Input*

On entry: the first dimension of the array A as declared in the (sub)program from which F01ZDF is called.

Constraint: LDA $\geq$ M.

8: B(LDB,$*$) — ***complex*** array *Input/Output*

Note: the second dimension of the array B must be at least min(M+KU,N)..

On entry: if JOB = 'U', then B must contain the band matrix in packed form, in the leading $(k_l + k_u + 1)$ by $\min(m + k_u, n)$ part of the array. The matrix is packed column by column, with the leading diagonal of the matrix in row $(k_u + 1)$ of B, the first super-diagonal starting at position 2 in row k_u, the first sub-diagonal starting at position 1 in row $(k_u + 2)$, and so on. Elements of B that are not needed to store the band matrix, for instance the leading k_u by k_u triangle, are not referenced and need not be assigned.

On exit: if JOB = 'P', then B contains the band matrix stored in packed form. Elements of B that are not needed to store the band matrix are not referenced.

9: LDB — INTEGER *Input*

On entry: the first dimension of the array B as declared in the (sub)program from which F01ZDF is called.

Constraint: LDB $\geq$ (KL+KU+1).

10: IFAIL — INTEGER *Input/Output*

On entry: IFAIL must be set to 0, -1 or 1. For users not familiar with this parameter (described in Chapter P01) the recommended value is 0.

On exit: IFAIL = 0 unless the routine detects an error (see Section 6).

6 Error Indicators and Warnings

If on entry IFAIL = 0 or -1, explanatory error messages are output on the current error message unit (as defined by X04AAF).

Errors detected by the routine:

IFAIL =1

On entry, JOB $\neq$ 'P', 'p', 'U' or 'u'.

IFAIL =2

On entry, KL $<$ 0.

IFAIL =3

On entry, KU $<$ 0.

IFAIL =4

On entry, LDA $<$ M.

IFAIL =5

On entry, LDB < KL + KU + 1.

IFAIL =6

On entry, M < 1,
or N < 1.

7 Accuracy

Not applicable.

8 Further Comments

None.

9 Example

This example program reads a matrix A in unpacked form, and copies it to the packed matrix B.

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.

```
*     F01ZDF Example Program Text
*     Mark 14 Release.  NAG Copyright 1989.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          NMAX, LDA, LDB
      PARAMETER        (NMAX=10,LDA=NMAX,LDB=LDA)
*     .. Local Scalars ..
      INTEGER          I, IFAIL, J, KL, KU, M, N
*     .. Local Arrays ..
      complex          A(LDA,NMAX), B(LDB,NMAX)
*     .. External Subroutines ..
      EXTERNAL         F01ZDF, X04DAF
*     .. Executable Statements ..
      WRITE (NOUT,*) 'F01ZDF Example Program Results'
*     Skip heading in data file
      READ (NIN,*)
      WRITE (NOUT,*)
      READ (NIN,*) M, N, KL, KU
*     Read a banded matrix of size M by N. KL is the number of
*     sub-diagonals, KU the number of super-diagonals.
      DO 20 I = 1, N
         READ (NIN,*) (A(I,J),J=1,N)
   20 CONTINUE
*     Clear the packed matrix array B, so that no elements are
*     unassigned when we print B later.
      DO 60 J = 1, N
         DO 40 I = 1, KL + KU + 1
            B(I,J) = (0.0e+0,0.0e+0)
   40    CONTINUE
   60 CONTINUE
```

```
      IFAIL = 0
*     Print the unpacked matrix
      CALL X04DAF('G','X',N,N,A,LDA,'Unpacked Matrix A:',IFAIL)
      WRITE (NOUT,*)
*
*     Convert to packed matrix form
      CALL F01ZDF('Pack',M,N,KL,KU,A,LDA,B,LDB,IFAIL)
*
*     Print the packed matrix
      CALL X04DAF('G','X',KL+KU+1,N,B,LDB,'Packed Matrix B:',IFAIL)
      STOP
      END
```

9.2 Program Data

```
F01ZDF Example Program Data.
 5  5  1  1                                                       M  N  KL  KU
 (1.1,-1.1)  (1.2,-1.2)  (0.0, 0.0)  (0.0, 0.0)  (0.0, 0.0)  Unpacked Matrix A
 (2.1,-2.1)  (2.2,-2.2)  (2.3,-2.3)  (0.0, 0.0)  (0.0, 0.0)
 (0.0, 0.0)  (3.2,-3.2)  (3.3,-3.3)  (3.4,-3.4)  (0.0, 0.0)
 (0.0, 0.0)  (0.0, 0.0)  (4.3,-4.3)  (4.4,-4.4)  (4.5,-4.5)
 (0.0, 0.0)  (0.0, 0.0)  (0.0, 0.0)  (5.4,-5.4)  (5.5,-5.5)
```

9.3 Program Results

```
 F01ZDF Example Program Results

 Unpacked Matrix A:
             1          2          3          4          5
 1      1.1000     1.2000     0.0000     0.0000     0.0000
       -1.1000    -1.2000     0.0000     0.0000     0.0000

 2      2.1000     2.2000     2.3000     0.0000     0.0000
       -2.1000    -2.2000    -2.3000     0.0000     0.0000

 3      0.0000     3.2000     3.3000     3.4000     0.0000
        0.0000    -3.2000    -3.3000    -3.4000     0.0000

 4      0.0000     0.0000     4.3000     4.4000     4.5000
        0.0000     0.0000    -4.3000    -4.4000    -4.5000

 5      0.0000     0.0000     0.0000     5.4000     5.5000
        0.0000     0.0000     0.0000    -5.4000    -5.5000

 Packed Matrix B:
             1          2          3          4          5
 1      0.0000     1.2000     2.3000     3.4000     4.5000
        0.0000    -1.2000    -2.3000    -3.4000    -4.5000

 2      1.1000     2.2000     3.3000     4.4000     5.5000
       -1.1000    -2.2000    -3.3000    -4.4000    -5.5000

 3      2.1000     3.2000     4.3000     5.4000     0.0000
       -2.1000    -3.2000    -4.3000    -5.4000     0.0000
```