F08ATF (CUNGQR/ZUNGQR) – 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

F08ATF (CUNGQR/ZUNGQR) generates all or part of the complex unitary matrix Q from a QR factorization computed by F08ASF (CGEQRF/ZGEQRF) or F08BSF (CGEQPF/ZGEQPF).

2 Specification

```
SUBROUTINE F08ATF(M, N, K, A, LDA, TAU, WORK, LWORK, INFO)
ENTRY      cungqr(M, N, K, A, LDA, TAU, WORK, LWORK, INFO)
INTEGER           M, N, K, LDA, LWORK, INFO
complex           A(LDA,*), TAU(*), WORK(LWORK)
```

The ENTRY statement enables the routine to be called by its LAPACK name.

3 Description

This routine is intended to be used after a call to F08ASF (CGEQRF/ZGEQRF) or F08BSF (CGEQPF/ZGEQPF), which perform a QR factorization of a complex matrix A. F08ASF and F08BSF represent the unitary matrix Q as a product of elementary reflectors.

This routine may be used to generate Q explicitly as a square matrix, or to form only its leading columns.

Usually Q is determined from the QR factorization of an m by p matrix A with $m \geq p$. The whole of Q may be computed by:

```
CALL CUNGQR (M,M,P,A,LDA,TAU,WORK,LWORK,INFO)
```

(note that the array A must have at least m columns) or its leading p columns by:

```
CALL CUNGQR (M,P,P,A,LDA,TAU,WORK,LWORK,INFO)
```

The columns of Q returned by the last call form an orthonormal basis for the space spanned by the columns of A; thus F08ASF followed by F08ATF can be used to orthogonalise the columns of A.

The information returned by the QR factorization routines also yields the QR factorization of the leading k columns of A, where $k < p$. The unitary matrix arising from this factorization can be computed by:

```
CALL CUNGQR (M,M,K,A,LDA,TAU,WORK,LWORK,INFO)
```

or its leading k columns by:

```
CALL CUNGQR (M,K,K,A,LDA,TAU,WORK,LWORK,INFO)
```

4 References

[1] Golub G H and van Loan C F (1996) *Matrix Computations* Johns Hopkins University Press (3rd Edition), Baltimore

5 Parameters

1: M — INTEGER *Input*

On entry: m, the order of the unitary matrix Q.

Constraint: M $\geq$ 0.

2: N — INTEGER *Input*

On entry: n, the number of columns of matrix Q that are required.

Constraint: M $\geq$ N $\geq$ 0.

3: K — INTEGER *Input*

On entry: k, the number of elementary reflectors whose product defines the matrix Q.

Constraint: N $\geq$ K $\geq$ 0.

4: A(LDA,∗) — ***complex*** array *Input/Output*

Note: the second dimension of the array A must be at least max(1,N).

On entry: details of the vectors which define the elementary reflectors, as returned by F08ASF (CGEQRF/ZGEQRF) or F08BSF (CGEQPF/ZGEQPF).

On exit: the m by n matrix Q.

5: LDA — INTEGER *Input*

On entry: the first dimension of the array A as declared in the (sub)program from which F08ATF (CUNGQR/ZUNGQR) is called.

Constraint: LDA $\geq$ max(1,M).

6: TAU(∗) — ***complex*** array *Input*

Note: the dimension of the array TAU must be at least max(1,K).

On entry: further details of the elementary reflectors, as returned by F08ASF (CGEQRF/ZGEQRF) or F08BSF (CGEQPF/ZGEQPF).

7: WORK(LWORK) — ***complex*** array *Workspace*

On exit: if INFO = 0, WORK(1) contains the minimum value of LWORK required for optimum performance.

8: LWORK — INTEGER *Input*

On entry: the dimension of the array WORK as declared in the (sub)program from which F08ATF (CUNGQR/ZUNGQR) is called.

Suggested value: for optimum performance LWORK should be at least N $\times$ nb, where nb is the ***blocksize***.

Constraint: LWORK $\geq$ max(1,N).

9: INFO — INTEGER *Output*

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

6 Error Indicators and Warnings

INFO < 0

If INFO $= -i$, the ith parameter had an illegal value. An explanatory message is output, and execution of the program is terminated.

7 Accuracy

The computed matrix Q differs from an exactly unitary matrix by a matrix E such that

$$\| E \|_2 = O(\epsilon),$$

where ϵ is the ***machine precision***.

8 Further Comments

The total number of real floating-point operations is approximately $16mnk - 8(m+n)k^2 + \frac{16k^3}{3}$; when $n = k$, the number is approximately $\frac{8}{3}n^2(3m - n)$.

The real analogue of this routine is F08AFF (SORGQR/DORGQR).

9 Example

To form the leading 4 columns of the unitary matrix Q from the QR factorization of the matrix A, where

$$A = \begin{pmatrix} 0.96-0.81i & -0.03+0.96i & -0.91+2.06i & -0.05+0.41i \\ -0.98+1.98i & -1.20+0.19i & -0.66+0.42i & -0.81+0.56i \\ 0.62-0.46i & 1.01+0.02i & 0.63-0.17i & -1.11+0.60i \\ -0.37+0.38i & 0.19-0.54i & -0.98-0.36i & 0.22-0.20i \\ 0.83+0.51i & 0.20+0.01i & -0.17-0.46i & 1.47+1.59i \\ 1.08-0.28i & 0.20-0.12i & -0.07+1.23i & 0.26+0.26i \end{pmatrix}$$

The columns of Q form an orthonormal basis for the space spanned by the columns of A.

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.

```
*     F08ATF Example Program Text
*     Mark 16 Release. NAG Copyright 1992.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          MMAX, NMAX, LDA, LWORK
      PARAMETER        (MMAX=8,NMAX=8,LDA=MMAX,LWORK=64*NMAX)
*     .. Local Scalars ..
      INTEGER          I, IFAIL, INFO, J, M, N
      CHARACTER*30     TITLE
*     .. Local Arrays ..
      complex          A(LDA,NMAX), TAU(NMAX), WORK(LWORK)
      CHARACTER        CLABS(1), RLABS(1)
*     .. External Subroutines ..
      EXTERNAL         X04DBF, cgeqrf, cungqr
*     .. Executable Statements ..
      WRITE (NOUT,*) 'F08ATF Example Program Results'
*     Skip heading in data file
      READ (NIN,*)
```

```
      READ (NIN,*) M, N
      IF (M.LE.MMAX .AND. N.LE.NMAX .AND. M.GE.N) THEN
*
*        Read A from data file
*
         READ (NIN,*) ((A(I,J),J=1,N),I=1,M)
*
*        Compute the QR factorization of A
*
         CALL cgeqrf(M,N,A,LDA,TAU,WORK,LWORK,INFO)
*
*        Form the leading N columns of Q explicitly
*
         CALL cungqr(M,N,N,A,LDA,TAU,WORK,LWORK,INFO)
*
*        Print the leading N columns of Q only
*
         WRITE (NOUT,*)
         WRITE (TITLE,99999) N
         IFAIL = 0
*
         CALL X04DBF('General',' ',M,N,A,LDA,'Bracketed','F7.4',TITLE,
     +               'Integer',RLABS,'Integer',CLABS,80,0,IFAIL)
*
      END IF
      STOP
*
99999 FORMAT ('The leading ',I2,' columns of Q')
      END
```

9.2 Program Data

```
F08ATF Example Program Data
  6  4                                                  :Values of M and N
 ( 0.96,-0.81) (-0.03, 0.96) (-0.91, 2.06) (-0.05, 0.41)
 (-0.98, 1.98) (-1.20, 0.19) (-0.66, 0.42) (-0.81, 0.56)
 ( 0.62,-0.46) ( 1.01, 0.02) ( 0.63,-0.17) (-1.11, 0.60)
 (-0.37, 0.38) ( 0.19,-0.54) (-0.98,-0.36) ( 0.22,-0.20)
 ( 0.83, 0.51) ( 0.20, 0.01) (-0.17,-0.46) ( 1.47, 1.59)
 ( 1.08,-0.28) ( 0.20,-0.12) (-0.07, 1.23) ( 0.26, 0.26)   :End of matrix A
```

9.3 Program Results

```
 F08ATF Example Program Results

 The leading  4 columns of Q
                    1                  2                  3                  4
 1  (-0.3110, 0.2624) (-0.3175, 0.4835) ( 0.4966,-0.2997) (-0.0072,-0.3718)
 2  ( 0.3175,-0.6414) (-0.2062, 0.1577) (-0.0793,-0.3094) (-0.0282,-0.1491)
 3  (-0.2008, 0.1490) ( 0.4892,-0.0900) ( 0.0357,-0.0219) ( 0.5625,-0.0710)
 4  ( 0.1199,-0.1231) ( 0.2566,-0.3055) ( 0.4489,-0.2141) (-0.1651, 0.1800)
 5  (-0.2689,-0.1652) ( 0.1697,-0.2491) (-0.0496, 0.1158) (-0.4885,-0.4540)
 6  (-0.3499, 0.0907) (-0.0491,-0.3133) (-0.1256,-0.5300) ( 0.1039, 0.0450)
```