NAG Toolbox for MATLAB

f08bx

1 Purpose

f08bx multiplies a general complex m by n matrix C by the complex unitary matrix Z from an RZ factorization computed by f08bv.

2 Syntax

```
[c, info] = f08bx(side, trans, l, a, tau, c, 'm', m, 'n', n, 'k', k)
```

3 Description

f08bx is intended to be used following a call to f08bv, which performs an RZ factorization of a real upper trapezoidal matrix A and represents the unitary matrix Z as a product of elementary reflectors.

This function may be used to form one of the matrix products

$$ZC, \quad Z^{\mathrm{H}}C, \quad CZ, \quad CZ^{\mathrm{H}},$$

overwriting the result on C, which may be any complex rectangular m by n matrix.

4 References

Anderson E, Bai Z, Bischof C, Blackford S, Demmel J, Dongarra J J, Du Croz J J, Greenbaum A, Hammarling S, McKenney A and Sorensen D 1999 *LAPACK Users' Guide* (3rd Edition) SIAM, Philadelphia URL: http://www.netlib.org/lapack/lug

5 Parameters

5.1 Compulsory Input Parameters

1: **side – string**

Indicates how Z or Z^{H} is to be applied to C.

side = 'L'

Z or Z^{H} is applied to C from the left.

side = 'R'

Z or Z^{H} is applied to C from the right.

Constraint: **side** = 'L' or 'R'.

2: **trans – string**

If **trans** = 'N', no transpose, apply Z.

If **trans** = 'C', transpose, apply Z^{H}.

Constraint: **trans** = 'N' or 'C'.

3: **l – int32 scalar**

The number of columns of the matrix A containing the meaningful part of the Householder reflectors.

Constraints:

if **side** = 'L', $\mathbf{m} \geq \mathbf{l} \geq 0$;
if **side** = 'R', $\mathbf{n} \geq \mathbf{l} \geq 0$.

4: **a(lda,∗) – complex array**

The first dimension of the array **a** must be at least $\max(1, \mathbf{k})$

The second dimension of the array must be at least $\max(1, \mathbf{m})$ if **side** = 'L' and at least $\max(1, \mathbf{n})$ if **side** = 'R'

The ith row of **a** must contain the vector which defines the elementary reflector H_i, for $i = 1, 2, \ldots, k$, as returned by f08bv.

5: **tau(∗) – complex array**

Note: the dimension of the array **tau** must be at least $\max(1, \mathbf{k})$.

tau(i) must contain the scalar factor of the elementary reflector H_i, as returned by f08bh.

6: **c(ldc,∗) – complex array**

The first dimension of the array **c** must be at least $\max(1, \mathbf{m})$

The second dimension of the array must be at least $\max(1, \mathbf{n})$

The m by n matrix C.

5.2 Optional Input Parameters

1: **m – int32 scalar**

Default: The first dimension of the array **c**.

m, the number of rows of the matrix C.

Constraint: $\mathbf{m} \geq 0$.

2: **n – int32 scalar**

Default: The second dimension of the array **c**.

n, the number of columns of the matrix C.

Constraint: $\mathbf{n} \geq 0$.

3: **k – int32 scalar**

Default: The dimension of the array **tau**.

k, the number of elementary reflectors whose product defines the matrix Z.

Constraints:

if **side** = 'L', $\mathbf{m} \geq \mathbf{k} \geq 0$;
if **side** = 'R', $\mathbf{n} \geq \mathbf{k} \geq 0$.

5.3 Input Parameters Omitted from the MATLAB Interface

lda, ldc, work, lwork

5.4 Output Parameters

1: **c(ldc,∗) – complex array**

The first dimension of the array **c** must be at least $\max(1, \mathbf{m})$

The second dimension of the array must be at least $\max(1, \mathbf{n})$

c contains ZC or $Z^{\mathrm{H}}C$ or CZ or $Z^{\mathrm{H}}C$ as specified by **side** and **trans**.

2: **info – int32 scalar**

info $= 0$ unless the function detects an error (see Section 6).

6 Error Indicators and Warnings

Errors or warnings detected by the function:

info $= -i$

If **info** $= -i$, parameter i had an illegal value on entry. The parameters are numbered as follows:

1: **side**, 2: **trans**, 3: **m**, 4: **n**, 5: **k**, 6: **l**, 7: **a**, 8: **lda**, 9: **tau**, 10: **c**, 11: **ldc**, 12: **work**, 13: **lwork**, 14: **info**.

It is possible that **info** refers to a parameter that is omitted from the MATLAB interface. This usually indicates that an error in one of the other input parameters has caused an incorrect value to be inferred.

7 Accuracy

The computed result differs from the exact result by a matrix E such that

$$\|E\|_2 = O\epsilon\|C\|_2$$

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

8 Further Comments

The total number of floating point operations is approximately $16nlk$ if **side** $=$ 'L' and $16mlk$ if **side** $=$ 'R'.

The real analogue of this function is f08bk.

9 Example

```
side = 'Left';
trans = 'Conjugate transpose';
l = int32(1);
a   =   [complex(2.503143210347317,   -0),   complex(0.4809038260097097,   -
0.680193998646756), ...
                    complex(-0.3515766495221888,     -1.049569215740192),
complex(0.02428150997532811, -0.2110246999702111);
          complex(-0.1419878310874442,  +0.008154310665875074),  complex(-
1.969705879310746, ...
       -0),  complex(0.6273709827062415,  +0.2047611531910448),  complex(-
0.06167299671596834, +0.1246040219830579);
          complex(-0.09061318086343964,  -0.05909460446106325),  complex(-
0.4734396725777993, ...
       +0.1684395728926726),  complex(-1.220364144932806,  -0),  complex(-
0.2462396805294381, -0.2636616658757083)];
tau = [complex(1.913653988303099, -0);
     complex(1.962073663852821, +0);
     complex(1.769674314319312, +0)];
c       =       [complex(-0.7078523389712563,       -0.01175074625576926),
complex(0.2684286127201276, -0.4324811203946267);
             complex(-4.263793865758445,   -2.462208743121598),   complex(-
0.3145423694693609, -1.491394594372636);
                      complex(-2.079678997260836,    -6.227516091951813),
complex(1.745764522877727, +2.300097951222233);
     complex(0, +0), complex(0, +0)];
[cOut, info] = f08bx(side, trans, l, a, tau, c)
```

```
cOut =
   0.6467 + 0.0107i  -0.2453 + 0.3951i
   4.1764 + 2.3435i   0.2908 + 1.4900i
   1.3486 + 5.5027i  -1.4418 - 1.6465i
   1.1669 - 3.3224i  -0.5023 + 1.8323i
info =
           0
```