

NAG Fortran Library Routine Document

F06HQP

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

F06HQP generates the parameters of a complex unitary matrix P , of order $n + 1$, chosen so as to set to zero the elements of a supplied n element complex vector x .

If PIVOT = 'F' and DIRECT = 'F', or if PIVOT = 'V' and DIRECT = 'B',

$$P \begin{pmatrix} \alpha \\ x \end{pmatrix} = \begin{pmatrix} \beta \\ 0 \end{pmatrix};$$

If PIVOT = 'F' and DIRECT = 'B', or if PIVOT = 'V' and DIRECT = 'F',

$$P \begin{pmatrix} x \\ \alpha \end{pmatrix} = \begin{pmatrix} 0 \\ \beta \end{pmatrix}.$$

Here α and β are complex scalars.

P is represented as a sequence of n plane rotations P_k , as specified by PIVOT and DIRECT; P_k is chosen to annihilate x_k , and its 2 by 2 plane rotation part has the form

$$\begin{pmatrix} c_k & \bar{s}_k \\ -s_k & c_k \end{pmatrix},$$

with c_k real. The tangent of the rotation, t_k , is overwritten on x_k .

2 Specification

```
SUBROUTINE F06HQP (PIVOT, DIRECT, N, ALPHA, X, INCX, C, S)
  INTEGER          N, INCX
  double precision C(*)
  complex*16      ALPHA, X(*), S(*)
  CHARACTER*1      PIVOT, DIRECT
```

3 Description

None.

4 References

None.

5 Parameters

1: PIVOT – CHARACTER*1

Input

On entry: specifies the plane rotated by P_k :

if PIVOT = 'V' (variable pivot), P_k rotates the $(k, k + 1)$ plane;
 if PIVOT = 'F' (fixed pivot), P_k rotates the $(1, k + 1)$ plane if DIRECT = 'F', or the $(k, n + 1)$ plane if DIRECT = 'B'.

Constraint: PIVOT = 'V' or 'F'.

- 2: DIRECT – CHARACTER*1 *Input*
 On entry: specifies the sequence direction:
 if DIRECT = 'F' (forward sequence), $P = P_n \cdots P_2 P_1$;
 if DIRECT = 'B' (backward sequence), $P = P_1 P_2 \cdots P_n$.
 Constraint: DIRECT = 'F' or 'B'.
- 3: N – INTEGER *Input*
 On entry: n , the number of elements in x .
- 4: ALPHA – **complex*16** *Input/Output*
 On entry: the scalar α .
 On exit: the scalar β .
- 5: X(*) – **complex*16** array *Input/Output*
 On entry: the vector x .
 On exit: the tangents of the rotations P_k , for $k = 1, 2, \dots, n$.
- 6: INCX – INTEGER *Input*
 On entry: the increment in the subscripts of X between successive elements of x .
 Constraint: INCX > 0.
- 7: C(*) – **double precision** array *Output*
 On exit: the values c_k , the cosines of the rotations.
- 8: S(*) – **complex*16** array *Output*
 On exit: the values s_k , the sines of the rotations.

6 Error Indicators and Warnings

None.
