Euler Angles, Quaternions, and Transformation Matrices

Working Relationships

Mission Planning and Analysis Division

July 1977
SHUTTLE PROGRAM

EULER ANGLES, QUATERNIONS, AND TRANSFORMATION MATRICES: WORKING RELATIONSHIPS

By D. M. Henderson
McDonnell Douglas Technical Services Co., Inc.

JSC Task Monitor: B. F. Cockrell

Approved: 
Emil R. Schiesser, Chief
Mathematical Physics Branch

Approved: 
Ronald L. Berry, Chief
Mission Planning and Analysis Division

National Aeronautics and Space Administration
Lyndon B. Johnson Space Center
Mission Planning and Analysis Division
Houston, Texas

July 1977
## CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>2.0</td>
<td>DISCUSSION</td>
<td>2</td>
</tr>
<tr>
<td>2.1</td>
<td>Euler Angle Transformation Matrices</td>
<td>2</td>
</tr>
<tr>
<td>2.2</td>
<td>Transformation Matrices Using the Hamilton Quaternion</td>
<td>7</td>
</tr>
<tr>
<td>2.3</td>
<td>Euler Angle and Quaternion Relationships</td>
<td>9</td>
</tr>
<tr>
<td>3.0</td>
<td>REFERENCES</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td>APPENDIX A - RELATIONS FOR THE THREE-AXIS EULER ANGLE ROTATION SEQUENCES</td>
<td>A-1</td>
</tr>
<tr>
<td></td>
<td>APPENDIX B - COMPUTER SUBROUTINES FOR THE RELATIONSHIPS</td>
<td>B-1</td>
</tr>
</tbody>
</table>
1.0 INTRODUCTION

Due to the extensive use of the quaternion in the onboard Space Shuttle Computer System, considerable analysis is being performed using relationships between the quaternion, the transformation matrix, and the Euler angles. This Internal Note offers a brief mathematical development of the relationships between the Euler angles and the transformation matrix, the quaternion and the transformation matrix, and the Euler angles and the quaternion. The analysis and equations presented here apply directly to current Space Shuttle problems. Appendix A presents the twelve three-axis Euler transformation matrices as functions of the Euler angles, the equations for the quaternion as a function of the Euler angles, and the Euler angles as a function of the transformation matrix elements.

The equations of Appendix A are a valuable reference in Shuttle analysis work and this Internal Note is the only known document where each of the twelve Euler angle to quaternion relationships are given. Appendix B presents a group of utility subroutines to accomplish the Euler-matrix, quaternion-matrix, and Euler-quaternion relationships of Appendix A.
2.0 DISCUSSION

2.1 Euler Angle Transformation Matrices

The following analysis and utility subroutines are offered to simplify computer programs when working with coordinate transformation matrices and their relationships with the Euler Angles and the Quaternions. The coordinate transformation matrices discussed here are defined using the following figure.

Figure 1.- Coordinate system and Euler angles.
The transformation matrix $M$, is defined to transform vectors in the $x$-system $(x, y, z)$ into the original $x$-system $(x, y, z)$ and is given by the equation,

$$x = M \bar{x}$$

where

$$x = (x, y, z) \text{ and } \bar{x} = (\bar{x}, \bar{y}, \bar{z}).$$

Using the right-hand rule for positive rotations, the $M$ matrix in (1) above is constructed by the following analysis. The first rotation in Figure 1 above is about the $x$-axis by the amount $\theta_1$. The single rotation about the $x$-axis results in the following transformation,

$$
\begin{pmatrix}
  x \\
  y \\
  z
\end{pmatrix} =
\begin{pmatrix}
  1 & 0 & 0 \\
  0 & \cos \theta_1 & -\sin \theta_1 \\
  0 & \sin \theta_1 & \cos \theta_1
\end{pmatrix}
\begin{pmatrix}
  \bar{x}' \\
  \bar{y}' \\
  \bar{z}'
\end{pmatrix}
$$

or $x = x' \in$ matrix form. Rotation about the $\bar{y}'$-axis by the amount $\theta_2$ yields the intermediate transformation matrix:

$$
\begin{pmatrix}
  \bar{x}' \\
  \bar{y}' \\
  \bar{z}'
\end{pmatrix} =
\begin{pmatrix}
  \cos \theta_2 & 0 & \sin \theta_2 \\
  0 & 1 & 0 \\
  -\sin \theta_2 & 0 & \cos \theta_2
\end{pmatrix}
\begin{pmatrix}
  \bar{x}'' \\
  \bar{y}'' \\
  \bar{z}''
\end{pmatrix}
$$

or $\bar{x}' = Yx''$ in matrix form. Finally rotation about the $\bar{z}''$-axis by the amount $\theta_3$ yields the intermediate transformation matrix,
\[
\begin{pmatrix}
\bar{x}'
\bar{y}'
\bar{z}'
\end{pmatrix}
= \begin{pmatrix}
\cos \theta_3 & -\sin \theta_3 & 0 \\
\sin \theta_3 & \cos \theta_3 & 0 \\
0 & 0 & 1
\end{pmatrix}
\begin{pmatrix}
\bar{x} \\
\bar{y} \\
\bar{z}
\end{pmatrix}
\]

(4)

and in matrix form \( \bar{x}' = \bar{x}_1 \). Now using the three equations,

\[
x = \bar{x}'
\]

\[
\bar{x}' = \bar{y} \bar{x}^n
\]

\[
\bar{x}^n = \bar{z} \bar{x}
\]

by substitution

\[
x = (X \ Y \ Z) \ \bar{x}.
\]

(5)

Then from equation 1,

\[
M = (X \ Y \ Z)
\]

(7)

Computation for the \( M \) matrix from the indicated matrix multiplication in equation (7) yields,

\[
M = \begin{pmatrix}
\cos \theta_2 \cos \theta_3 & -\cos \theta_2 \sin \theta_3 & \sin \theta_2 \\
\cos \theta_1 \sin \theta_3 + \sin \theta_1 \sin \theta_2 \cos \theta_3 & \cos \theta_1 \cos \theta_3 - \sin \theta_1 \sin \theta_2 \sin \theta_3 & -\sin \theta_1 \cos \theta_2 \\
\sin \theta_1 \sin \theta_3 - \cos \theta_1 \sin \theta_2 \cos \theta_3 & \cos \theta_1 \cos \theta_3 + \cos \theta_1 \sin \theta_2 \sin \theta_3 & \cos \theta_1 \cos \theta_2
\end{pmatrix}
\]

(8)

The matrix \( M \) in equation (8) is a function of;

(1) The three Euler angles \( \theta_1, \theta_2 \) and \( \theta_3 \) and

(2) The sequence of rotations used to generate the matrix.

By examination of equation (7) it is possible to show that there are twelve possible Euler rotational sequences. If the \((X \ Y \ Z)\) notation in equation (7) represents a rotation about the \( X \) axis, then the \( Y \) axis and finally the \( Z \) axis, then the following per-
mutations of the rotational order represents the twelve possible Euler angle sets using three rotations,

\[ \begin{align*}
X & \ Y & \ Z & \ \\
Y & \ Z & \ X & \ \\
Z & \ X & \ Y & \ \\
X & \ Y & \ X & \ \\
X & \ Z & \ X & \ \\
Y & \ Z & \ Y & \ \\
\end{align*} \]  \quad (9)

Any repeated axis rotation such as XXY does not represent a three axis rotation but reduces to the two axis rotation XY. Hence the rotations described in (9) above represent all twelve possible sets of Euler angle defining sequences. Conversely then, for a given transformation matrix there are twelve Euler angle sets which can be extracted from the matrix. The Euler matrices corresponding to all possible rotational sequences of (9) above are presented in Appendix A.

The utility subroutines "EULMAT" generates the transformation matrix from a given Euler sequence and the Euler angles. The utility subroutine "MATEUL" extracts the Euler angles from a given Euler rotational sequence. The convention is established that the Euler angles occur in the same sequences as the axis rotations. Using this concept and functional notation, equation (7) could be expressed as

\[ M = X \ Y \ Z = M(\theta_x, \theta_y, \theta_z) \]  \quad (10)

and from (9)
\[ M = X Z X = M(\theta_x, \theta_z, \theta_x') \text{ etc.} \]  

(11)

This convention is assumed in both subroutines and also used throughout this design note when Euler angles are used. A brief explanation of the use of these two utility subroutines is given in Appendix B.

It is interesting to note that a negative rotation in the single axis rotation matrices of equations (2), (3) and (4) will result in the formation of the transpose of the matrix. However the transpose of \( M \) in equation (7) is formed from reversing the order of multiplication and transposing the individual axis rotation equations, i.e.

\[ M^T = (X \ Y \ Z)^T = (Y \ Z)X^T = Z^T Y^T X^T. \]  

(12)

Hence the transposes of the matrices of (9) are easily formed by reversing the order of multiplication and transposing each single axis rotation equation. Using the notation in equations (10) and (11) above equation (12) could be written,

\[ M^T(\theta_x, \theta_y, \theta_z) = M(-\theta_z, -\theta_y, -\theta_x). \]  

(13)

It is recommended to avoid confusion that the forward transformation be computed and simply transposed to yield the reverse transformation matrix. All matrices of Appendix A are in the forward form, i.e. \( X = Mx \) and formed from (9).
2.2 Transformation Matrices Using the Hamilton Quaternion

The transformation matrix of equation (1) can be written as a function of the Hamilton Quaternion:

\[ q_1 = \cos \omega/2 \]
\[ q_2 = \cos \alpha \sin \omega/2 \]
\[ q_3 = \cos \beta \sin \omega/2 \]
\[ q_4 = \cos \gamma \sin \omega/2 , \]

where \( \omega \) is the rotation angle about the rotation axis with \( \alpha, \beta, \) and \( \gamma \) direction angles with the x, y and z axes respectively. Notice also that \( q_1^2 + q_2^2 + q_3^2 + q_4^2 = 1 \), since \( \cos^2 \alpha + \cos^2 \beta + \cos^2 \gamma = 1 \). The rotation angle, \( \omega \), is assumed positive according to the right-hand rule of axis rotation.

The matrix \( M \) becomes

\[
M = \begin{pmatrix}
(q_1^2 + q_2^2 - q_3^2 - q_4^2) & 2(q_2 q_3 - q_1 q_4) & 2(q_2 q_4 + q_1 q_3) \\
2(q_2 q_3 + q_1 q_4) & (q_1^2 - q_2^2 + q_3^2 - q_4^2) & 2(q_3 q_4 - q_1 q_2) \\
2(q_2 q_4 - q_1 q_3) & 2(q_3 q_4 + q_1 q_2) & (q_1^2 - q_2^2 - q_3^2 + q_4^2)
\end{pmatrix}
\]

(15)

For a more detailed discussion of the derivation of equation (15), see Reference 1. Using functional notation, equation (15) can be written,

\[ M = M(q_1, q_2, q_3, q_4). \]

Unlike the Euler angle rotational sequences to describe the transformation matrix of equation (1), only two quaternions can be found from equation (15). The two quaternions are:
These two quaternions represent a positive rotation about the rotation axis pointing in one direction and a positive rotation about the same line of rotation pointing in the opposite direction. Both quaternions of (17) satisfy equation (15).

The utility subroutine "QMAT" generates the transformation matrix from a given quaternion. The "QMAT" algorithm generates the matrix as given in equation (15) without duplicating any arithmetic operations. The subroutine "MATQ" extracts the positive quaternion, i.e., $q_1 > 0$, from the transformation matrix and normalizes the results to guarantee an orthogonal matrix. In order to avoid any discontinuity in extracting the quaternion from the transformation matrix, the procedure as described in Reference 2 is used.

Early works by Hamilton (Référence 3) presented the quaternion as having a scalar and a vector part, i.e.,

$$q_1 = S \quad \mathbf{v} = (q_2, q_3, q_4)$$  \hspace{1cm} (18)

and equation (16) could be expressed as,

$$M = M(q_1, q_2, q_3, q_4) = M(S, \mathbf{v}).$$  \hspace{1cm} (19)
For a given quaternion the following relationship is true (from (17) above),

\[ M(S, V) = M(-S, -V). \]  

(20)

The transpose of the transformation matrix is given by,

\[ M^T(S, V) = M(-S, V) = M(S, -V). \]  

(21)

2.3 Euler Angle and Quaternion Relationships

By examination of equations (10) and (16) the equality,

\[ M(X(\theta_1), Y(\theta_2), Z(\theta_3)) = M(\theta_1, \theta_2, \theta_3) = M(q_1, q_2, q_3, q_4) \]  

(22)

can be written. Based on an equality for each element of the matrix the following nine equations must be true;

\[
\begin{align*}
\cos \theta_2 \cos \theta_3 &= q_1^2 + q_2^2 - q_3^2 - q_4^2 \\
-\cos \theta_2 \sin \theta_3 &= 2(q_2 q_3 - q_1 q_4) \\
\sin \theta_2 &= 2(q_2 q_4 + q_1 q_3) \\
\cos \theta_1 \sin \theta_3 + \sin \theta_1 \sin \theta_2 \cos \theta_3 &= 2(q_3 q_4 + q_1 q_2) \\
\cos \theta_1 \cos \theta_3 - \sin \theta_1 \sin \theta_2 \sin \theta_3 &= q_1^2 - q_2^2 + q_3^2 - q_4^2 \\
-\sin \theta_1 \cos \theta_2 &= 2(q_3 q_4 - q_1 q_2) \\
\sin \theta_1 \sin \theta_3 - \cos \theta_1 \sin \theta_2 \cos \theta_3 &= 2(q_3 q_4 - q_1 q_3) \\
\sin \theta_1 \cos \theta_3 + \cos \theta_1 \sin \theta_2 \sin \theta_3 &= 2(q_3 q_4 + q_1 q_2) \\
\cos \theta_1 \cos \theta_2 &= q_1^2 - q_2^2 - q_3^2 + q_4^2 .
\end{align*}
\]

(23)

It is possible to solve for the values of the quaternion using the trigonometric half angle identities. For this Euler sequence, i.e. \( X(\theta_1) Y(\theta_2) Z(\theta_3) \), the following quaternion results;
\[ q_1 = -\sin \frac{\theta_1}{2} \sin \frac{\theta_2}{2} \sin \frac{\theta_3}{2} + \cos \frac{\theta_1}{2} \cos \frac{\theta_2}{2} \cos \frac{\theta_3}{2} \]
\[ q_2 = +\sin \frac{\theta_1}{2} \cos \frac{\theta_2}{2} \cos \frac{\theta_3}{2} + \sin \frac{\theta_2}{2} \sin \frac{\theta_3}{2} \cos \frac{\theta_1}{2} \]
\[ q_3 = -\sin \frac{\theta_1}{2} \sin \frac{\theta_3}{2} \cos \frac{\theta_2}{2} + \sin \frac{\theta_2}{2} \cos \frac{\theta_1}{2} \cos \frac{\theta_3}{2} \]
\[ q_4 = +\sin \frac{\theta_1}{2} \sin \frac{\theta_2}{2} \cos \frac{\theta_3}{2} + \sin \frac{\theta_3}{2} \cos \frac{\theta_1}{2} \cos \frac{\theta_2}{2} \]  

(24)

Appendix A gives the quaternion as a function of the Euler angles for each of the twelve Euler rotational sequence presented in Section 2.1. The special equations for the quaternion as functions of the Euler angles, like equations (24) above, are sometimes cumbersome to use, especially when multiple Euler sequences are utilized. Less coding, but perhaps more computer operations, are required by using the more general method of first generating the matrix \( M \) from the given Euler angle set and then simply extracting the quaternion from the matrix. This method is effected by first a call to "EULMAT" and then a call to "MATQ".
3.0 REFERENCES


APPENDIX A

RELATIONSHIPS FOR THE THREE-AXIS EULER ANGLE ROTATION SEQUENCES

The twelve Euler matrices for each rotation sequence are given based on the single axis rotation equations (2), (3) and (4). The Euler matrices transform vectors from the system that has been rotated into vectors in the stationary system. Also presented here are the equations for the quaternion as a function of the Euler angles and the Euler angles as a function of the matrix elements for each rotation sequence.
(1) \[ M = M(X(\theta_1), Y(\theta_2), Z(\theta_3)) = XYZ \]

Axis Rotation Sequence: 1, 2, 3

\[
M = \begin{bmatrix}
\cos\theta_2 \cos\theta_3 & -\cos\theta_2 \sin\theta_3 & \sin\theta_2 \\
\sin\theta_1 \sin\theta_2 \cos\theta_3 & -\sin\theta_1 \sin\theta_2 \sin\theta_3 + \cos\theta_1 \cos\theta_3 & -\sin\theta_1 \cos\theta_2 \\
-\cos\theta_1 \sin\theta_2 \cos\theta_3 & \cos\theta_1 \sin\theta_2 \sin\theta_3 & \sin\theta_1 \cos\theta_2
\end{bmatrix}
\]

\[
q_1 = -\sin\theta_1 \sin\theta_2 \sin\theta_3 + \cos\theta_1 \cos\theta_2 \cos\theta_3 \\
q_2 = \sin\theta_1 \cos\theta_2 \cos\theta_3 + \sin\theta_2 \sin\theta_3 \cos\theta_1 \\
q_3 = -\sin\theta_1 \sin\theta_2 \cos\theta_3 + \sin\theta_2 \cos\theta_1 \cos\theta_3 \\
q_4 = \sin\theta_1 \sin\theta_2 \cos\theta_3 + \sin\theta_3 \cos\theta_2 \cos\theta_1 \\
\]

\[
\theta_1 = \tan^{-1}\left(\frac{-m_{23}}{m_{33}}\right) \\
\theta_2 = \tan^{-1}\left(\frac{m_{13}}{\sqrt{1-m_{13}^2}}\right) \\
\theta_3 = \tan^{-1}\left(\frac{-m_{12}}{m_{11}}\right)
\]
(2) \( M = M(X(\theta_1), Z(\theta_2), Y(\theta_3)) = XZY \)

Axis Rotation Sequence: 1, 3, 2

\[
M = \begin{bmatrix}
\cos \theta_2 \cos \theta_3 & -\sin \theta_2 & \cos \theta_2 \sin \theta_3 \\
\cos \theta_1 \sin \theta_2 \cos \theta_3 & \cos \theta_1 \cos \theta_2 & \cos \theta_1 \sin \theta_2 \sin \theta_3 \\
+\sin \theta_1 \sin \theta_3 & \sin \theta_1 \cos \theta_3 & -\sin \theta_1 \sin \theta_2 \sin \theta_3 \\
-\cos \theta_1 \sin \theta_3 & -\cos \theta_1 \cos \theta_3 & +\cos \theta_1 \cos \theta_2 \sin \theta_3 \\
\end{bmatrix}
\]

\[
q_1 = +\sin \theta_1 \sin \theta_2 \sin \theta_3 + \cos \theta_1 \cos \theta_2 \cos \theta_3
\]

\[
q_2 = +\sin \theta_1 \cos \theta_2 \cos \theta_3 - \sin \theta_2 \sin \theta_3 \cos \theta_1
\]

\[
q_3 = -\sin \theta_1 \sin \theta_2 \cos \theta_3 + \sin \theta_2 \cos \theta_1 \cos \theta_2
\]

\[
q_4 = +\sin \theta_1 \sin \theta_2 \cos \theta_3 + \sin \theta_2 \cos \theta_1 \cos \theta_3
\]

\[
\theta_1 = \tan^{-1} \left( \frac{m_{32}}{m_{22}} \right)
\]

\[
\theta_2 = \tan^{-1} \left( \frac{-m_{12}}{\sqrt{1-m_{12}^2}} \right)
\]

\[
\theta_3 = \tan^{-1} \left( \frac{m_{13}}{m_{11}} \right)
\]
Axis Rotation Sequence: 1, 2, 1

\[
M = \begin{bmatrix}
\cos \theta_2 & \sin \theta_2 \sin \theta_3 & \sin \theta_2 \cos \theta_3 \\
\sin \theta_1 \sin \theta_2 & \cos \theta_1 \cos \theta_3 & -\cos \theta_1 \sin \theta_3 \\
-\cos \theta_1 \sin \theta_2 & +\sin \theta_1 \cos \theta_3 & -\sin \theta_1 \sin \theta_3 \\
\end{bmatrix}
\]

\[
q_1 = \cos \theta_2 \cos \left( \frac{1}{2}(\theta_1 + \theta_3) \right)
\]

\[
q_2 = \cos \theta_2 \sin \left( \frac{1}{2}(\theta_1 + \theta_3) \right)
\]

\[
q_3 = \sin \theta_2 \cos \left( \frac{1}{2}(\theta_1 - \theta_3) \right)
\]

\[
q_4 = \sin \theta_2 \sin \left( \frac{1}{2}(\theta_1 - \theta_3) \right)
\]

\[
\theta_1 = \tan^{-1} \left( \frac{m_{21}}{-m_{31}} \right)
\]

\[
\theta_2 = \tan^{-1} \left( \sqrt{ \frac{1 - m_{11}^2}{m_{11}} } \right)
\]

\[
\theta_3 = \tan^{-1} \left( \frac{m_{12}}{m_{13}} \right)
\]
(4) \[ M = M(x(\theta_1), z(\theta_2), x(\theta_3)) = XZX \]

\[ M = \begin{bmatrix}
\cos \theta_2 & -\sin \theta_2 \cos \theta_3 & \sin \theta_2 \sin \theta_3 \\
\cos \theta_1 \sin \theta_2 & \cos \theta_1 \cos \theta_2 \cos \theta_3 & -\cos \theta_1 \cos \theta_2 \sin \theta_3 \\
\sin \theta_1 \sin \theta_2 & \sin \theta_1 \cos \theta_2 \cos \theta_3 & -\sin \theta_1 \cos \theta_2 \sin \theta_3 \\
\end{bmatrix} \]

\[ q_1 = \cos \theta_2 \cos \left(\frac{1}{2}(\theta_1 + \theta_3)\right) \]
\[ q_2 = \cos \theta_2 \sin \left(\frac{1}{2}(\theta_1 + \theta_3)\right) \]
\[ q_3 = -\sin \theta_2 \sin \left(\frac{1}{2}(\theta_1 - \theta_3)\right) \]
\[ q_4 = \sin \theta_2 \cos \left(\frac{1}{2}(\theta_1 - \theta_3)\right) \]

\[ \theta_1 = \tan^{-1} \left( \frac{m_{31}}{m_{21}} \right) \]
\[ \theta_2 = \tan^{-1} \left( \sqrt{1 - \frac{m_{11}^2}{m_{11}}} \right) \]
\[ \theta_3 = \tan^{-1} \left( \frac{m_{13}}{-m_{12}} \right) \]
(5) \( M = M(Y(\theta_1), X(\theta_2), Z(\theta_3)) = YXZ \)

Axis Rotation Sequence: 2, 1, 3

\[
M = \begin{bmatrix}
\sin \theta_1 \sin \theta_2 \sin \theta_3 & \sin \theta_1 \sin \theta_2 \cos \theta_3 & \sin \theta_1 \cos \theta_2 \\
+ \cos \theta_1 \cos \theta_3 & - \cos \theta_1 \sin \theta_3 & \\
\cos \theta_2 \sin \theta_3 & \cos \theta_2 \cos \theta_3 & - \sin \theta_2 \\
\cos \theta_1 \sin \theta_2 \sin \theta_3 & \cos \theta_1 \sin \theta_2 \cos \theta_3 & \cos \theta_1 \cos \theta_2 \\
- \sin \theta_1 \cos \theta_3 & + \sin \theta_1 \sin \theta_3 & 
\end{bmatrix}
\]

\[
q_1 = \sin \frac{\theta_1}{2} \sin \frac{\theta_2}{2} \sin \frac{\theta_3}{2} + \cos \frac{\theta_1}{2} \cos \frac{\theta_2}{2} \cos \frac{\theta_3}{2}
\]

\[
q_2 = \sin \frac{\theta_1}{2} \sin \frac{\theta_2}{2} \cos \frac{\theta_3}{2} + \sin \frac{\theta_2}{2} \cos \frac{\theta_1}{2} \cos \frac{\theta_3}{2}
\]

\[
q_3 = \sin \frac{\theta_1}{2} \cos \frac{\theta_2}{2} \cos \frac{\theta_3}{2} - \sin \frac{\theta_2}{2} \sin \frac{\theta_3}{2} \cos \frac{\theta_1}{2}
\]

\[
q_4 = - \sin \frac{\theta_1}{2} \sin \frac{\theta_2}{2} \cos \frac{\theta_3}{2} + \sin \frac{\theta_2}{2} \cos \frac{\theta_1}{2} \cos \frac{\theta_3}{2}
\]

\[
\theta_1 = \tan^{-1} \left( \frac{m_{31}}{m_{33}} \right)
\]

\[
\theta_2 = \tan^{-1} \left( \frac{-m_{23}}{\sqrt{1-m_{23}^2}} \right)
\]

\[
\theta_3 = \tan^{-1} \left( \frac{m_{21}}{m_{22}} \right)
\]
(6) \(M = M(Y(\theta_1), Z(\theta_2), X(\theta_3)) = YZX\)

Axis Rotation Sequence: 2, 3, 1

\[
M = \begin{bmatrix}
\cos \theta_1 \cos \theta_2 & -\cos \theta_1 \sin \theta_2 \cos \theta_3 & \cos \theta_1 \sin \theta_2 \sin \theta_3 \\
\sin \theta_2 & \cos \theta_2 \cos \theta_3 & -\cos \theta_2 \sin \theta_3 \\
-sin \theta_1 \cos \theta_2 & \sin \theta_1 \sin \theta_2 \cos \theta_3 & -\sin \theta_1 \sin \theta_2 \sin \theta_3 \\
\end{bmatrix}
\]

\[q_1 = -\sin \theta_1 \sin \theta_2 \sin \theta_3 + \cos \theta_1 \cos \theta_2 \cos \theta_3 \]
\[q_2 = +\sin \theta_1 \sin \theta_2 \cos \theta_3 + \sin \theta_2 \cos \theta_1 \cos \theta_2 \]
\[q_3 = +\sin \theta_1 \cos \theta_2 \cos \theta_3 + \sin \theta_2 \sin \theta_3 \cos \theta_1 \]
\[q_4 = -\sin \theta_1 \sin \theta_2 \cos \theta_2 + \sin \theta_2 \cos \theta_1 \sin \theta_3 \]

\[\theta_1 = \tan^{-1} \left( \frac{-m_{31}}{m_{11}} \right) \]
\[\theta_2 = \tan^{-1} \left( \frac{m_{21}}{\sqrt{1-m_{21}^2}} \right) \]
\[\theta_3 = \tan^{-1} \left( \frac{-m_{23}}{m_{22}} \right) \]
(7) \( M = M(Y(\theta_1), X(\theta_2), Y(\theta_3)) = YXY \)

Axis Rotation Sequence: 2, 1, 2

\[
M = \begin{bmatrix}
-sin\theta_1cos\theta_2sin\theta_3 & sin\theta_1sin\theta_2 & sin\theta_1cos\theta_2cos\theta_3 \\
+cos\theta_1cos\theta_3 & cos\theta_2 & +cos\theta_1sin\theta_3 \\
-sin\theta_2sin\theta_3 & -sin\theta_2cos\theta_3 & cos\theta_1sin\theta_2 \\
-cos\theta_1cos\theta_2sin\theta_3 & cos\theta_1sin\theta_2 & cos\theta_1cos\theta_2cos\theta_3 \\
-sin\theta_1cos\theta_3 & -sin\theta_1sin\theta_3 & -sin\theta_1cos\theta_3
\end{bmatrix}
\]

\[
q_1 = +cos\theta_2cos(\frac{1}{2}(\theta_1 + \theta_3)) \\
q_2 = +sin\theta_2cos(\frac{1}{2}(\theta_1 - \theta_3)) \\
q_3 = +cos\theta_2sin(\frac{1}{2}(\theta_1 + \theta_3)) \\
q_4 = -sin\theta_2sin(\frac{1}{2}(\theta_1 - \theta_3))
\]

\[
\theta_1 = \tan^{-1}\left(\frac{m_{12}}{m_{32}}\right)
\]

\[
\theta_2 = \tan^{-1}\left(\frac{\sqrt{1-m_{22}^2}}{m_{22}}\right)
\]

\[
\theta_3 = \tan^{-1}\left(\frac{m_{21}}{-m_{23}}\right)
\]
(8) \( M = M(Y(\theta_1), Z(\theta_2), Y(\theta_3)) = YZY \)

**Axis Rotation Sequence: 2, 3, 2**

\[
M = \begin{bmatrix}
\cos \theta_1 \cos \theta_2 \cos \theta_3 & -\cos \theta_1 \sin \theta_2 & \cos \theta_1 \cos \theta_2 \sin \theta_3 \\
-\sin \theta_1 \sin \theta_3 & \cos \theta_2 & \sin \theta_2 \sin \theta_3 \\
\sin \theta_2 \cos \theta_3 & \sin \theta_1 \sin \theta_2 & -\sin \theta_1 \cos \theta_2 \sin \theta_3 \\
-\cos \theta_1 \sin \theta_3 & -\cos \theta_2 \sin \theta_3 & \cos \theta_1 \cos \theta_3
\end{bmatrix}
\]

\[ q_1 = \cos \theta_2 \cos \left( \frac{1}{2}(\theta_1 + \theta_3) \right) \]
\[ q_2 = \sin \theta_2 \sin \left( \frac{1}{2}(\theta_1 - \theta_3) \right) \]
\[ q_3 = \cos \theta_2 \sin \left( \frac{1}{2}(\theta_1 + \theta_3) \right) \]
\[ q_4 = \sin \theta_2 \cos \left( \frac{1}{2}(\theta_1 - \theta_3) \right) \]

\[ \theta_1 = \tan^{-1} \left( \frac{m_{32}}{-m_{12}} \right) \]
\[ \theta_2 = \tan^{-1} \left( \frac{\sqrt{1-m_{22}^2}}{m_{22}} \right) \]
\[ \theta_3 = \tan^{-1} \left( \frac{m_{23}}{m_{21}} \right) \]
(9) \( M = M(Z(\theta_1), X(\theta_2), Y(\theta_3)) = ZXY \)

Axis Rotation Sequence: 3, 1, 2

\[
M = \begin{bmatrix}
    -\sin \theta_1 \sin \theta_2 \sin \theta_3 & -\sin \theta_1 \cos \theta_2 & \sin \theta_1 \sin \theta_2 \cos \theta_3 \\
    \cos \theta_1 \cos \theta_3 & \cos \theta_1 \cos \theta_2 & -\cos \theta_1 \sin \theta_2 \cos \theta_3 \\
    \cos \theta_1 \sin \theta_2 \sin \theta_3 & +\sin \theta_1 \cos \theta_2 & +\sin \theta_1 \sin \theta_2 \cos \theta_3 \\
    -\cos \theta_2 \sin \theta_3 & \sin \theta_2 & \cos \theta_2 \cos \theta_3
\end{bmatrix}
\]

\[
q_1 = -\sin \theta_1 \sin \theta_2 \sin \theta_3 + \cos \theta_1 \cos \theta_2 \cos \theta_3 \\
q_2 = -\sin \theta_1 \sin \theta_2 \cos \theta_3 + \sin \theta_2 \cos \theta_1 \cos \theta_3 \\
q_3 = +\sin \theta_1 \sin \theta_2 \cos \theta_3 + \sin \theta_3 \cos \theta_1 \cos \theta_2 \\
q_4 = +\sin \theta_1 \cos \theta_2 \cos \theta_3 + \sin \theta_2 \sin \theta_3 \cos \theta_1
\]

\[
\theta_1 = \tan^{-1} \left( \frac{-m_{12}}{m_{22}} \right)
\]

\[
\theta_2 = \tan^{-1} \left( \frac{m_{32}}{\sqrt{1-m_{32}^2}} \right)
\]

\[
\theta_3 = \tan^{-1} \left( \frac{-m_{31}}{m_{33}} \right)
\]
\[ M = M(Z(\theta_1), Y(\theta_2), X(\theta_3)) = ZYX \]

**Axis Rotation Sequence:** 3, 2, 1

\[
M = \begin{bmatrix}
\cos \theta_1 \cos \theta_2 & \cos \theta_1 \sin \theta_2 \sin \theta_3 & \cos \theta_1 \sin \theta_2 \cos \theta_3 \\
\sin \theta_1 \cos \theta_2 & \sin \theta_1 \sin \theta_2 \sin \theta_3 & \sin \theta_1 \sin \theta_2 \cos \theta_3 \\
-sin \theta_2 & cos \theta_2 \sin \theta_3 & cos \theta_2 \cos \theta_3 \\
\end{bmatrix}
\]

\[ q_1 = +\sin \frac{\theta_1}{2} \sin \frac{\theta_2}{2} \sin \frac{\theta_3}{2} + \cos \frac{\theta_1}{2} \cos \frac{\theta_2}{2} \cos \frac{\theta_3}{2} \]

\[ q_2 = -\sin \frac{\theta_1}{2} \sin \frac{\theta_2}{2} \cos \frac{\theta_3}{2} + \sin \frac{\theta_2}{2} \cos \frac{\theta_1}{2} \cos \frac{\theta_3}{2} \]

\[ q_3 = +\sin \frac{\theta_1}{2} \sin \frac{\theta_2}{2} \cos \frac{\theta_3}{2} + \sin \frac{\theta_2}{2} \cos \frac{\theta_1}{2} \cos \frac{\theta_3}{2} \]

\[ q_4 = +\sin \frac{\theta_1}{2} \cos \frac{\theta_2}{2} \cos \frac{\theta_3}{2} - \sin \frac{\theta_2}{2} \sin \frac{\theta_3}{2} \cos \frac{\theta_1}{2} \]

\[ \theta_1 = \tan^{-1} \left( \frac{m_{21}}{m_{11}} \right) \]

\[ \theta_2 = \tan^{-1} \left( \frac{-m_{31}}{\sqrt{1-m_{31}^2}} \right) \]

\[ \theta_3 = \tan^{-1} \left( \frac{m_{32}}{m_{33}} \right) \]
(11) \( M = M(Z(\theta_1), X(\theta_2), Z(\theta_3)) = ZXZ \)

Axis Rotation Sequence: 3, 1, 3

\[
M = \begin{bmatrix}
-sin\theta_1 cos\theta_2 sin\theta_3 & -sin\theta_1 cos\theta_2 cos\theta_3 & sin\theta_1 sin\theta_2 \\
+cos\theta_1 cos\theta_3 & -cos\theta_1 cos\theta_3 & -cos\theta_1 sin\theta_2 \\
-cos\theta_1 cos\theta_2 sin\theta_3 & cos\theta_1 cos\theta_2 cos\theta_3 & -sin\theta_1 sin\theta_3 \\
+sin\theta_1 cos\theta_3 & -sin\theta_1 sin\theta_3 & sin\theta_2 cos\theta_3 \\
sin\theta_2 sin\theta_3 & sin\theta_2 cos\theta_3 & cos\theta_2
\end{bmatrix}
\]

\[
q_1 = +cos^2\theta_2 cos(\frac{1}{2}(\theta_1 + \theta_3))
\]

\[
q_2 = +sin^2\theta_2 cos(\frac{1}{2}(\theta_1 - \theta_3))
\]

\[
q_3 = +sin^2\theta_2 sin(\frac{1}{2}(\theta_1 - \theta_3))
\]

\[
q_4 = +cos^2\theta_2 sin(\frac{1}{2}(\theta_1 + \theta_3))
\]

\[
\theta_1 = \tan^{-1} \left( \frac{m_{13}}{m_{23}} \right)
\]

\[
\theta_2 = \tan^{-1} \left( \frac{\sqrt{1-m_{33}^2}}{m_{33}} \right)
\]

\[
\theta_3 = \tan^{-1} \left( \frac{m_{31}}{m_{32}} \right)
\]
12) \( M = M(Z(\theta_1), Y(\theta_2), Z(\theta_3)) = ZYZ \)

Axis Rotation Sequence: 3, 2, 3

\[
M = \begin{bmatrix}
\cos \theta_1 \cos \theta_2 \cos \theta_3 & -\cos \theta_1 \cos \theta_2 \sin \theta_3 & \cos \theta_1 \sin \theta_2 \\
-\sin \theta_1 \sin \theta_3 & -\sin \theta_1 \cos \theta_3 & \sin \theta_1 \sin \theta_2 \\
\sin \theta_1 \cos \theta_2 \cos \theta_3 & -\sin \theta_1 \cos \theta_2 \sin \theta_3 & \sin \theta_1 \cos \theta_3 \\
+\cos \theta_1 \sin \theta_3 & +\cos \theta_1 \cos \theta_3 & \sin \theta_2 \sin \theta_3 \\
-\sin \theta_2 \cos \theta_3 & \sin \theta_2 \sin \theta_3 & \cos \theta_2 \\
\end{bmatrix}
\]

\[
q_1 = +\cos \frac{\theta_2}{2} \cos \left(\frac{1}{2}(\theta_1 + \theta_3)\right)
\]

\[
q_2 = -\sin \frac{\theta_2}{2} \sin \left(\frac{1}{2}(\theta_1 - \theta_3)\right)
\]

\[
q_3 = +\sin \frac{\theta_2}{2} \cos \left(\frac{1}{2}(\theta_1 - \theta_3)\right)
\]

\[
q_4 = +\cos \frac{\theta_2}{2} \sin \left(\frac{1}{2}(\theta_1 + \theta_3)\right)
\]

\[
\theta_1 = \tan^{-1}\left(\frac{m_{23}}{m_{13}}\right)
\]

\[
\theta_2 = \tan^{-1}\left(\frac{\sqrt{1-m_{33}^2}}{m_{33}}\right)
\]

\[
\theta_3 = \tan^{-1}\left(\frac{m_{32}}{-m_{31}}\right)
\]
APPENDIX B
COMPUTER SUBROUTINES FOR THE RELATIONSHIPS

The following subroutines with a brief description of their use are presented in this appendix.

(1) "EULMAT" - Generates the transformation matrix from a given set of Euler angles and an axis rotation sequence.

(2) "MATEUL" - Extracts the Euler angles from the given transformation matrix and an axis rotation sequence.

(3) "QMAT" - Generates the transformation matrix from a given quaternion.

(4) "MATQ" - Extracts the quaternion from a given transformation matrix.

(5) "YPRQ" - Generates the quaternion directly from the yaw-pitch-roll Euler angles.

(6) "POSNOR" - Computes the positive-normalized quaternion from the given quaternion.
NAME: EULMAT

PURPOSE: Generates a 3 x 3 transformation matrix from a given sequence and Euler angle set.

INPUT: ISEQ - Rotation Sequence (Integer Array (3); i.e., 1, 2, 3) 
        EUL - Euler Angles in radians, in "ISEQ" Order; ARRAY (3)

OUTPUT: A - The 3 x 3 transformation matrix

ALGORITHM REFERENCE: Appendix A; Euler Sequences (1) thru (12).
EULER ANGLES TO THE TRANSFORMATION MATRIX

SUBROUTINE LULMAT ENTRY POINT OCL237

STORAGE USEFUL CODE(1) OCL260; DATA(1) OCL104; BLANK COMMON(2)

EXTERNAL REFERENCES (BLOCK, NAME)

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

SUBROUTINE LULMAT(IS, LULMAT)
DIMENSION ISOM(3), LULMAT(3, 4)
DO 10 IS = 1, 3
DO 10 J = 1, 4
10 CONTINUE
IF(IS <= 3) LULMAT(IS, J) = 0 TO 10
SINA = SINA(IS, LULMAT(J))
COSA = COSA(IS, LULMAT(J))
DO 20 IS = 1, 3
20 CONTINUE
EULER ANGLES TO THE TRANSFORMATION MATRIX

(Continued)

END OF Compilation: NO DiagnostiCS.

ORIGINAL PAGE IS OF POOR QUALITY
NAME: MATEUL

PURPOSE: Extracts the Euler angles from the given transformation matrix and the required Euler rotational sequence.

INPUT: ISEQ - Rotation sequence, (Integer Array (3), i.e., 1, 2, 3.)

A - The 3 x 3 transformation

OUTPUT: EUL - The Euler angles, in "ISEQ" order; ARRAY(3).

ALGORITHM REFERENCE: Appendix A; Euler angles as a function of the matrix elements, sequences (1) thru (12).
TRANSFORMATION MATRIX TO THE EULER ANGLES

FOR IS  MATEUL, MATEUL
FOR S3L3 02/19/77-06/29/79 (0:0)

SUBROUTINE MATEUL  ENTRY POINT U*3355

STORAGE USED: CODE(11) 00553; DATA(5) 005525; BLANK COMMON(2)

EXTERNAL REFERENCES (BLOCK, NAME)
103 SORT
1064 A1AN0
2005 MPRZ3

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

101 I300550 ICL 4 123456 ICL5 3051 3-7-7
102 011 211 ICL 4 123456 ICL5 3051 3-7-7
103 011 111 ICL 4 123456 ICL5 3061 3-7-7
104 011 111 E5ign 4 123456 E5ign5 3050 3-7-7
105 011 111 FNU5 4 123456 FNU5 3050 3-7-7
106 011 111 J 4 123456 J 3050 3-7-7
107 011 111 T 4 123456 T 3050 3-7-7

SUBROUTINE MATEUL ISE (J, FUL)
DIMENSION A(3,21) (FUL)
DIMENSION ISE (0,3)

I.FLC=4 I.EOM=4.95
I.LSIG=1 I.EGM=1
I.T(2,1)=150 GO TO 10
I.T(2,2)=150 GO TO 10
I.T(3,1)=150 GO TO 10
I.T(3,2)=150 GO TO 10
I.C(SIC)=1 GO TO 6
I.C(SIC)=1 GO TO 6
10 IF(E.R.E.2) GO TO 15
I.SIGN=1 I.EGM=2 I.L=3
GO TO 3
11 IF(T(SIC) .LT. E.GM) L=1
GO TO 3
12 IF(T(SIC) .LT. E.GM) L=1
GO TO 3
13 IF(T(SIC) .LT. E.GM) L=1
GO TO 3
14 IF(T(SIC) .LT. E.GM) L=1
GO TO 3
15 IF(T(SIC) .LT. E.GM) L=1
GO TO 3
16 IF(T(SIC) .LT. E.GM) L=1
GO TO 3
17 IF(T(SIC) .LT. E.GM) L=1
GO TO 3
18 IF(T(SIC) .LT. E.GM) L=1
GO TO 3
19 IF(T(SIC) .LT. E.GM) L=1
GO TO 3
20 IF(W-R.E.2) GO TO 25
I.SIGN=1
TRANSFORMATION MATRIX TO THE EULER ANGLES

(Continued)

```plaintext
29* IF (IEQK .NE. 0) L = 1
30* GO TO 30
31* 25. CSIGN = -1.0
32* IF (IEQK .NE. 0) L = 3
33* 30. JJ = JJ + 1
34* FNSGN = 1.0
35* FDOSGN = 1.0
36* IF (N.EQ.2) GO TO 70
37* IF (N.EQ.4) GO TO 50
38* IF (IEQK .NE. 0) GO TO 40
39* FNSGN = 0
40* JJ = 1
41* GO TO 45
42* J = J + 1
43* IF (IEQK .GT. 0) FDOSGN = 1.0
44* FNUM = FNSGN * A(I,J)
45* FDCN = FDOSGN * A(I,J)
46* GO TO 90
47* 50. I = I + 1
48* IF (IEQK .GT. 0) GO TO 55
49* FNSGN = 0
50* JJ = K
51* GO TO 60
52* 55. FDOSGN = 0
53* JJ = J
54* J = J + 1
55* IF (IEQK .GT. 0) FNUM = FNSGN * A(J,K)
56* FDCN = FDOSGN * A(I,J)
57* GO TO 90
58* 60. I = I + 1
59* IF (IEQK .GT. 0) GO TO 65
60* FNUM = CSIGN * A(I,K)
61* FDCN = SQRT(1.0 - A(I,K)**2)
62* GO TO 90
63* 65. I = I + 1
64* IF (IEQK .GT. 0) GO TO 70
65* FNUM = SQRT(1.0 - A(I,I)**2)
66* FDCN = A(I)
67* GO TO 90
68* 70. I = I + 1
69* IF (IEQK .GT. 0) GO TO 75
70* FNUM = SQRT(1.0 - A(I,I)**2)
71* FDCN = A(I)
72* GO TO 90
73* 75. I = I + 1
74* IF (IEQK .GT. 0) GO TO 80
75* FNUM = SQRT(1.0 - A(I,J)**2)
76* FDCN = A(J)
77* GO TO 90
78* 80. I = I + 1
79* IF (IEQK .GT. 0) GO TO 85
80* FNUM = SQRT(1.0 - A(J,J)**2)
81* FDCN = A(J)
82* GO TO 90
83* 85. I = I + 1
84* IF (IEQK .GT. 0) GO TO 90
85* FNUM = SQRT(1.0 - A(J,J)**2)
86* FDCN = A(J)
87* GO TO 90
88* 90. RETURN
89* END
```

END OF COMPILATION, NO DIAGNOSTICS.

ORIGINAL PAGE IS OF POOR QUALITY.
**NAME:** QMAT

**PURPOSE:** Generates the transformation matrix from the given quaternion.

**INPUT:** Q - The quaternion; ARRAY(4).

**OUTPUT:** A - The 3 x 3 transformation matrix

**ALGORITHM REFERENCE:** Equation (15) from Section 2.2.
QUATERNION TO THE TRANSFORMATION MATRIX

FORIS OMAT,GMAT FOR S\textsuperscript{e}3-02/19/77-06:024:19 (u,)

SUBROUTINE GMAT ENTRY POINT 000077

STORAGE USED: CODE11 CODE13 DATA10 12 LUMA10 BLANK COMMON(2)

EXTERNAL REFERENCES (BLOCK, NAME)

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

SUBROUTINE GMAT(6,A)
DIMENSION Q(4),A(3,3)
P:R:2(4)+Q(2)
R:Q(3)+L(3)
Q:Q(4)+L(4)
K:R(4)
P:Q(3)+L(3)
T:2:0-P*Q(3)
K:M=KMP-P
A(2,2)+P-P
B(2,2)=KMP-P
D=P*Q(3)
P:PL:Q(3)
A(1,1)+P-P
A(1,1)=P-P
R:Q(3)+L(4)
P:PL:Q(3)
A(2,2)+P-P
A(2,2)=P-P
A(3,3)+P-P
A(3,3)=P-P
P5=P3*Q(4)
P5=P3*Q(4)
P5=P3*Q(4)
P5=P3*Q(4)
A(2,2)+P-P
A(2,2)=P-P
A(3,3)+P-P
A(3,3)=P-P
RETURN
END

END OF COMPILATION... NO DIAGNOSTICS.
NAME: MATQ

PURPOSE: Extracts the positive quaternion from the given transformation matrix.

INPUT: A - The 3 x 3 transformation matrix

OUTPUT: Q - The positive quaternion; ARRAY(4).

ALGORITHM REFERENCE: See Reference 2.
SUBROUTINE HATO
ENTRY POINT IPRINT

STORAGE USED: CODE(1) - 200,290; DATA(1) - 200; BLANK COMMUNI(2)

EXTERNAL REFERENCES (BLOCK, NAME)

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

SUBROUTINE HATO(A, Q)
DIMENSION A(3,3),Q(4,4)

I = 1

IF(J.EQ.0) GO TO 20
IF(J.EQ.1) GO TO 21
IF(J.EQ.2) GO TO 22
TEMP = A(1,1) + A(1,2) + A(1,3) + T

20 TEMP = A(1,2) + A(2,2) + A(2,3) + T

21 TEMP = A(1,3) + A(3,2) + A(3,3) + T

22 TEMP = A(2,3) + A(3,2) + A(3,3) + T

GO TO 30

30 IF(T(1,1).LT.T(1,1)) GO TO 40

40 IF(T(1,1).LT.T(1,1)) GO TO 50

50 CONTINUE

END

END OF COMPILATION: NO DIAGNOSTICS.
NAME:          YPRQ

PURPOSE:       Generates the quaternion directly from the yaw-
pitch-roll Euler angles, i.e., a 3, 2, 1 Euler sequence.

INPUT:         YPR - The yaw-pitch-roll Euler angles; ARRAY (3).

OUTPUT:        QO - The positive quaternion, ARRAY (4).

ALGORITHM REFERENCE: Appendix A, the quaternion equations for Euler sequence (10), a 3, 2, 1 sequence.

NOTE: This subroutine calls "POSNOR" to take the normal of the positive quaternion.
FOR S YPRQ YPRQ
FOR S0E3-02/19/77-00:34:17 (C)

SUBROUTINE YPRQ  ENTRY POINT YPRQ

STORAGE USED: CODE(1) MOD21: DATA(D) 00020: PLANK COMMON(2) CU

EXTERNAL REFERENCES (BLOCK, NAME)
J023 POSNOR
J024 COS
J025 SIN
J026 NEPR31

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)
J000 R 3071: CP
J000 R 3032:4 HY
J000 R 3032: SY
J010 R 3071: CR
J000 R 303007 C
J000 R 303006 P

J0101 1* SUBROUTINE YPRQ YPRQ, C01
J0103 1* DIMENSION YPR(3), C(4), L0(4)
J0104 1* HR=...5*YPR(1)
J0105 3* HY=...5*YPR(2)
J0106 5* CY=COS(HY)
J0107 6* CP=COS(HP)
J0110 9* CY=COS(HR)
J0112 6* CP=COS(HR)
J0113 6* SY=SIN(HY)
J0114 12* SP=SIN(HY)
J0115 12* SR=SIN(HR)
J0115 12* CP=CP*CR+SP*SR
J0116 12* CP=CP*SR-SP*CR
J0117 14* Q(14)=SP*CP=CR+SP*SR
J0118 14* Q(14)=CP*CP=CR+SP*SR
J0120 15* CALL POSNOR(Q, Q0)
J0122 17* RTUPN
J0123 15* END

END OF COMPILATION: NO DIAGNOSTICS.
NAME:            POSNOR

PURPOSE:        To output the positive and normalized quaternion
                from the given quaternion.

INPUT:          Q - The quaternion; ARRAY (4).

OUTPUT:         QO - The positive-normalized quaternion;
                ARRAY (4).

ALGORITHM REFERENCE:

1. If the sign of Q(I) is negative:
   Set QO(I) = -Q(I) for I = 1, 2, 3, 4.

2. Set QO(I) = Q0(I)/TEMP
   where TEMP = \sqrt{Q0_1^2 + Q0_2^2 + Q0_3^2 + Q0_4^2}
SELETS THE POSITIVE QUATERNION AND NORMALIZES

FOR IS=0 TO 13

SUBROUTINE POSNOR ENTRY POINT 070055

STORAGE USED: CODE(11) 00647; DATA(CF) 00613; BLANK COMMON(2) C

EXTERNAL REFERENCES (BLOCK, NAME)

2003 SUBR 2004 VRORIS

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

3001 CPG1 1110 6301_30036 1216 GOOD 1 COMMON

30101 1a SUBROUTINE POSNOR(D,401)
30103 2a DIMENSION QA(9),QO(4)
30105 3a TEMP=1.0
30107 4a IF (U111. LT .5) TEMP=-1.0
30110 5a SUM=0.0
30113 6a DO 1=1,4
30115 7a 30111=TEMP*Q(1)
30117 8a SUM=SUM+30111*QO(1)
30119 9a 50 CONTINUE
30121 10a TEMP=1.0/SQRT(SUM)
30123 11a DO 1=1,4
30125 12a 30121=TEMP*Q(1)
30127 13a 17 CONTINUE
30129 14a RETURN
30101 15a END

END OF Compilation: NO DIAGNOSTICS.

ORIGINAL PAGE IS OF POOR QUALITY