77-FM-37

JSC-12960

Euler Angles, Quaternions, and Transformation Matrices

(NASA-TM-74839) SHUTTLE PROGRAM. EULER ANGLES, QUATERNIONS, AND TRANSFORMATION MATRICES WORKING RELATIONSHIPS (NASA) 42 p HC A03/MF A01 CSCL 22A

N77-31234

Unclas
G3/16 37332

Working Relationships

Mission Planning and Analysis Division

July 1977

National Aeronautics and Space Administration

Lyndon B. Johnson Space Center
Houston, Texas

77-FM-37

JSC-12960

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

EULER ANGLES, QUATERNIONS, AND TRANSFORMATION MATRICES - WORKING RELATIONSHIPS

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

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,

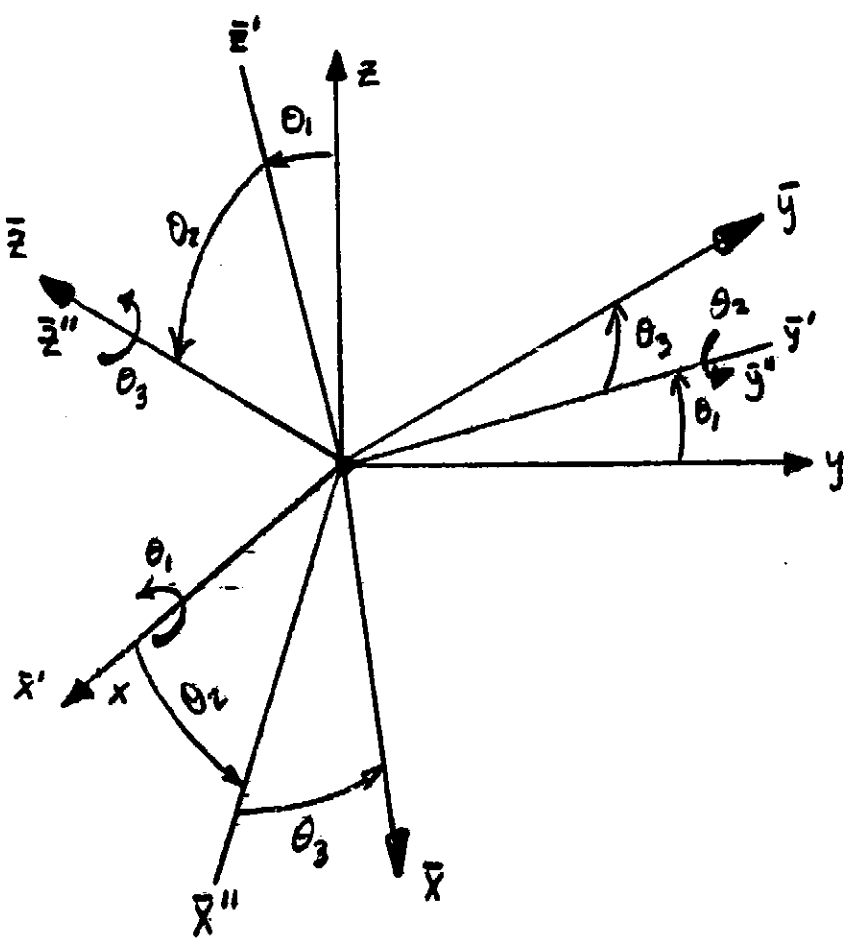

Figure 1.- Coordinate system and Euler angles.

The transformation matrix M, is defined to transform vectors in the $\bar{x}$- system ($\bar{x}$, $\bar{y}$, $\bar{z}$) into the original x-system (x, y, z) and is given by the equation,

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

where (1)

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

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 θ_1. The single rotation about the x-axis results in the following transformation,

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_1 & -\sin\theta_1 \\ 0 & \sin\theta_1 & \cos\theta_1 \end{pmatrix} \begin{pmatrix} \bar{x}' \\ \bar{y}' \\ \bar{z}' \end{pmatrix} \qquad (2)$$

or $x = X\bar{x}'$ in matrix form. Rotation about the $\bar{y}'$-axis by the amount θ_2 yields the intermediate transformation matrix:

$$\begin{pmatrix} \bar{x}' \\ \bar{y}' \\ \bar{z}' \end{pmatrix} = \begin{pmatrix} \cos\theta_2 & 0 & \sin\theta_2 \\ 0 & 1 & 0 \\ -\sin\theta_2 & 0 & \cos\theta_2 \end{pmatrix} \begin{pmatrix} \bar{x}'' \\ \bar{y}'' \\ \bar{z}'' \end{pmatrix} \qquad (3)$$

or $\bar{x}' = Y\bar{x}'$ in matrix form. Finally rotation about the $\bar{z}''$-axis by the amount θ_3 yields the intermediate transformation matrix,

$$\begin{pmatrix} \bar{x}'' \\ \bar{y}'' \\ \bar{z}'' \end{pmatrix} = \begin{pmatrix} \cos\theta_3 & -\sin\theta_3 & 0 \\ \sin\theta_3 & \cos\theta_3 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \bar{x} \\ \bar{y} \\ \bar{z} \end{pmatrix} \tag{4}$$

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

$$\begin{aligned} x &= X\bar{x}' \\ \bar{x}' &= Y\bar{x}'' \\ \bar{x}'' &= Z\bar{x} \end{aligned} \tag{5}$$

by substitution

$$x = (X\ Y\ Z)\ \bar{x}. \tag{6}$$

Then from equation 1,

$$M = (X\ Y\ Z) \tag{7}$$

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

$$M = \begin{pmatrix} (\cos\theta_2 \cos\theta_3) & (-\cos\theta_2 \sin\theta_3) & (\sin\theta_2) \\ (\cos\theta_1 \sin\theta_3 + \sin\theta_1 \sin\theta_2 \cos\theta_3) & (\cos\theta_1 \cos\theta_3 - \sin\theta_1 \sin\theta_2 \sin\theta_3) & (-\sin\theta_1 \cos\theta_2) \\ (\sin\theta_1 \sin\theta_3 - \cos\theta_1 \sin\theta_2 \cos\theta_3) & (\sin\theta_1 \cos\theta_3 + \cos\theta_1 \sin\theta_2 \sin\theta_3) & (\cos\theta_1 \cos\theta_2) \end{pmatrix} \tag{8}$$

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

(1) The three Euler angles θ_1 θ_2 and θ_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 permutations of the rotational order represents the twelve possible Euler angle sets using three rotations,

$$\begin{array}{ccc} X\,Y\,Z & Y\,X\,Z & Z\,X\,Y \\ X\,Z\,Y & Y\,Z\,X & Z\,Y\,X \\ X\,Y\,X & Y\,X\,Y & Z\,X\,Z \\ X\,Z\,X & Y\,Z\,Y & Z\,Y\,Z \end{array} \tag{9}$$

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

$$M = X\,Y\,Z = M(\theta_x, \theta_y, \theta_z) \tag{10}$$

and from (9)

$$M = X\ Z\ X = M(\theta_x, \theta_z, \theta_x') \text{ etc.} \tag{11}$$

This convention is assumed in both subroutines and also used throughout this design note when Euler angles are used. A brief explaination 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.

$$M^T = (X\ Y\ Z)^T = (\ Y\ Z)^T X^T = Z^T\ Y^T\ X^T. \tag{12}$$

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,

$$M^T(\theta_x, \theta_y, \theta_z) = M(-\theta_z, -\theta_y, -\theta_x). \tag{13}$$

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. $X = M\overline{x}$ 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;

$$
\begin{aligned}
q_1 &= \cos \omega/2 \\
q_2 &= \cos \alpha \sin \omega/2 \\
q_3 &= \cos \beta \sin \omega/2 \\
q_4 &= \cos \gamma \sin \omega/2 ,
\end{aligned}
\qquad (14)
$$

where ω is the rotation angle about the rotation axis with α, β, and γ 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, ω, is assumed positive according to the right-hand rule of axis rotation.

The matrix M becomes

$$
M = \begin{pmatrix} (q_1^2 + q_2^2 - q_3^2 - q_4^2) & 2(q_2q_3 - q_1q_4) & 2(q_2q_4 + q_1q_3) \\ 2(q_2q_3 + q_1q_4) & (q_1^2 - q_2^2 + q_3^2 - q_4^2) & 2(q_3q_4 - q_1q_2) \\ 2(q_2q_4 - q_1q_3) & 2(q_3q_4 + q_1q_2) & (q_1^2 - q_2^2 - q_3^2 + q_4^2) \end{pmatrix} . \qquad (15)
$$

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

$$
M = M(q_1, q_2, q_3, q_4). \qquad (16)
$$

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:

$$\begin{array}{ccc} q_1 & & -q_1 \\ q_2 & \text{and} & -q_2 \\ q_3 & & -q_3 \\ q_4 & & -q_4 \end{array} . \qquad (17)$$

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 (Reference 3) presented the quaternion as having a scalar and a vector part, i.e.,

$$q_1 = S \qquad \vec{V} = (q_2, q_3, q_4) \qquad (18)$$

and equation (16) could be expressed as,

$$M = M(q_1, q_2, q_3, q_4) = M(S, \vec{V}). \qquad (19)$$

For a given quaternion the following relationship is true (from (17) above),

$$M(S, \vec{V}) = M(-S, -\vec{V}). \tag{20}$$

The transpose of the transformation matrix is given by,

$$M^T(S, \vec{V}) = M(-S, \vec{V}) = M(S, -\vec{V}). \tag{21}$$

2.3 Euler Angle and Quaternion Relationships

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

$$M(X(\theta_1), Y(\theta_2), Z(\theta_3)) = M(\theta_1, \theta_2, \theta_3) = M(q_1, q_2, q_3, q_4) \tag{22}$$

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

$$\begin{aligned}
\cos\theta_2 \cos\theta_3 &= q_1^2 + q_2^2 - q_3^2 - q_4^2 \\
-\cos\theta_2 \sin\theta_3 &= 2(q_2 q_3 - q_1 q_4) \\
\sin\theta_2 &= 2(q_2 q_4 + q_1 q_3) \\
\cos\theta_1 \sin\theta_3 + \sin\theta_1 \sin\theta_2 \cos\theta_3 &= 2(q_2 q_3 + q_1 q_4) \\
\cos\theta_1 \cos\theta_3 - \sin\theta_1 \sin\theta_2 \sin\theta_3 &= q_1^2 - q_2^2 + q_3^2 - q_4^2 \\
-\sin\theta_1 \cos\theta_2 &= 2(q_3 q_4 - q_1 q_2) \\
\sin\theta_1 \sin\theta_3 - \cos\theta_1 \sin\theta_2 \cos\theta_3 &= 2(q_3 q_4 - q_1 q_3) \\
\sin\theta_1 \cos\theta_3 + \cos\theta_1 \sin\theta_2 \sin\theta_3 &= 2(q_3 q_4 + q_1 q_2) \\
\cos\theta_1 \cos\theta_2 &= q_1^2 - q_2^2 - q_3^2 + q_4^2 .
\end{aligned} \tag{23}$$

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;

$$
\begin{aligned}
q_1 &= -\sin\tfrac{1}{2}\theta_1 \sin\tfrac{1}{2}\theta_2 \sin\tfrac{1}{2}\theta_3 + \cos\tfrac{1}{2}\theta_1 \cos\tfrac{1}{2}\theta_2 \cos\tfrac{1}{2}\theta_3 \\
q_2 &= +\sin\tfrac{1}{2}\theta_1 \cos\tfrac{1}{2}\theta_2 \cos\tfrac{1}{2}\theta_3 + \sin\tfrac{1}{2}\theta_2 \sin\tfrac{1}{2}\theta_3 \cos\tfrac{1}{2}\theta_1 \\
q_3 &= -\sin\tfrac{1}{2}\theta_1 \sin\tfrac{1}{2}\theta_3 \cos\tfrac{1}{2}\theta_2 + \sin\tfrac{1}{2}\theta_2 \cos\tfrac{1}{2}\theta_1 \cos\tfrac{1}{2}\theta_3 \\
q_4 &= +\sin\tfrac{1}{2}\theta_1 \sin\tfrac{1}{2}\theta_2 \cos\tfrac{1}{2}\theta_3 + \sin\tfrac{1}{2}\theta_3 \cos\tfrac{1}{2}\theta_1 \cos\tfrac{1}{2}\theta_2
\end{aligned}
\tag{24}
$$

Appendix A gives the quaternion as a function of the Euler angles for each of the twelve Euler rotaticnal 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

1. Working Paper: MDTSCO, TM No. 1.4-MPB-304, E914-8A/B-003, "Quaternions and Quaternion Transformations," David M. Henderson, 23 June 1976.
2. Transmittal Memo: MDTSCO, 1.4-MPB-229, "Improving Computer Accuracy in Extracting Quaternions," David M. Henderson, 9 March 1976.
3. Sir William Rowan Hamilton, LLD, LL.D. MRIA, D.C.L. CANTAB., "Elements of Quaternions" 2 Volumes, Chelsea Publishing Company, New York, N. Y., 3rd Edition 1969, Library of Congress 68-54711 #8284-0219-1.

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

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

$$q_1 = -\sin\tfrac{1}{2}\theta_1\sin\tfrac{1}{2}\theta_2\sin\tfrac{1}{2}\theta_3 + \cos\tfrac{1}{2}\theta_1\cos\tfrac{1}{2}\theta_2\cos\tfrac{1}{2}\theta_3$$

$$q_2 = \sin\tfrac{1}{2}\theta_1\cos\tfrac{1}{2}\theta_2\cos\tfrac{1}{2}\theta_3 + \sin\tfrac{1}{2}\theta_2\sin\tfrac{1}{2}\theta_3\cos\tfrac{1}{2}\theta_1$$

$$q_3 = -\sin\tfrac{1}{2}\theta_1\sin\tfrac{1}{2}\theta_3\cos\tfrac{1}{2}\theta_2 + \sin\tfrac{1}{2}\theta_2\cos\tfrac{1}{2}\theta_1\cos\tfrac{1}{2}\theta_3$$

$$q_4 = \sin\tfrac{1}{2}\theta_1\sin\tfrac{1}{2}\theta_2\cos\tfrac{1}{2}\theta_3 + \sin\tfrac{1}{2}\theta_3\cos\tfrac{1}{2}\theta_1\cos\tfrac{1}{2}\theta_2$$

$$\theta_1 = \tan^{-1}\left(\frac{-m_{23}}{m_{33}}\right)$$

$$\theta_2 = \tan^{-1}\left(\frac{m_{13}}{\sqrt{1-m_{13}^2}}\right)$$

$$\theta_3 = \tan^{-1}\left(\frac{-m_{12}}{m_{11}}\right)$$

(2) $M = M(X(\theta_1), Z(\theta_2), Y(\theta_3)) = XZY$

Axis Rotation Sequence: 1, 3, 2

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

$$q_1 = +\sin\tfrac{1}{2}\theta_1\sin\tfrac{1}{2}\theta_2\sin\tfrac{1}{2}\theta_3 + \cos\tfrac{1}{2}\theta_1\cos\tfrac{1}{2}\theta_2\cos\tfrac{1}{2}\theta_3$$

$$q_2 = +\sin\tfrac{1}{2}\theta_1\cos\tfrac{1}{2}\theta_2\cos\tfrac{1}{2}\theta_3 - \sin\tfrac{1}{2}\theta_2\sin\tfrac{1}{2}\theta_3\cos\tfrac{1}{2}\theta_1$$

$$q_3 = -\sin\tfrac{1}{2}\theta_1\sin\tfrac{1}{2}\theta_2\cos\tfrac{1}{2}\theta_3 + \sin\tfrac{1}{2}\theta_3\cos\tfrac{1}{2}\theta_1\cos\tfrac{1}{2}\theta_2$$

$$q_4 = +\sin\tfrac{1}{2}\theta_1\sin\tfrac{1}{2}\theta_3\cos\tfrac{1}{2}\theta_2 + \sin\tfrac{1}{2}\theta_2\cos\tfrac{1}{2}\theta_1\cos\tfrac{1}{2}\theta_3$$

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

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

$$\theta_3 = \tan^{-1}\left(\frac{m_{13}}{m_{11}}\right)$$

Axis Rotation Sequence: 1, 2, 1

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

$$q_1 = \cos\tfrac{1}{2}\theta_2\cos(\tfrac{1}{2}(\theta_1 + \theta_3))$$

$$q_2 = \cos\tfrac{1}{2}\theta_2\sin(\tfrac{1}{2}(\theta_1 + \theta_3))$$

$$q_3 = \sin\tfrac{1}{2}\theta_2\cos(\tfrac{1}{2}(\theta_1 - \theta_3))$$

$$q_4 = \sin\tfrac{1}{2}\theta_2\sin(\tfrac{1}{2}(\theta_1 - \theta_3))$$

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

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

$$\theta_3 = \tan^{-1}\left(\frac{m_{12}}{m_{13}}\right)$$

(4) $M = M(X(\theta_1), Z(\theta_2), X(\theta_3)) = XZX$

Axis Rotation Sequence: 1, 3, 1

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

$$q_1 = \cos\tfrac{1}{2}\theta_2\cos(\tfrac{1}{2}(\theta_1 + \theta_3))$$

$$q_2 = \cos\tfrac{1}{2}\theta_2\sin(\tfrac{1}{2}(\theta_1 + \theta_3))$$

$$q_3 = -\sin\tfrac{1}{2}\theta_2\sin(\tfrac{1}{2}(\theta_1 - \theta_3))$$

$$q_4 = \sin\tfrac{1}{2}\theta_2\cos(\tfrac{1}{2}(\theta_1 - \theta_3))$$

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

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

$$\theta_3 = \tan^{-1}\left(\frac{m_{13}}{-m_{12}}\right)$$

(5) $M = M(Y(\theta_1), X(\theta_2), Z(\theta_3)) = YXZ$

Axis Rotation Sequence: 2, 1, 3

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

$$q_1 = \sin\tfrac{1}{2}\theta_1\sin\tfrac{1}{2}\theta_2\sin\tfrac{1}{2}\theta_3 + \cos\tfrac{1}{2}\theta_1\cos\tfrac{1}{2}\theta_2\cos\tfrac{1}{2}\theta_3$$

$$q_2 = \sin\tfrac{1}{2}\theta_1\sin\tfrac{1}{2}\theta_3\cos\tfrac{1}{2}\theta_2 + \sin\tfrac{1}{2}\theta_2\cos\tfrac{1}{2}\theta_1\cos\tfrac{1}{2}\theta_3$$

$$q_3 = \sin\tfrac{1}{2}\theta_1\cos\tfrac{1}{2}\theta_2\cos\tfrac{1}{2}\theta_3 - \sin\tfrac{1}{2}\theta_2\sin\tfrac{1}{2}\theta_3\cos\tfrac{1}{2}\theta_1$$

$$q_4 = -\sin\tfrac{1}{2}\theta_1\sin\tfrac{1}{2}\theta_2\cos\tfrac{1}{2}\theta_3 + \sin\tfrac{1}{2}\theta_3\cos\tfrac{1}{2}\theta_1\cos\tfrac{1}{2}\theta_2$$

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

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

$$\theta_3 = \tan^{-1}\left(\frac{m_{21}}{m_{22}}\right)$$

(6) $M = M(Y(\theta_1), Z(\theta_2), X(\theta_3)) = YZX$

Axis Rotation Sequence: 2, 3, 1

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

$$q_1 = -\sin\tfrac{1}{2}\theta_1\sin\tfrac{1}{2}\theta_2\sin\tfrac{1}{2}\theta_3 + \cos\tfrac{1}{2}\theta_1\cos\tfrac{1}{2}\theta_2\cos\tfrac{1}{2}\theta_3$$

$$q_2 = +\sin\tfrac{1}{2}\theta_1\sin\tfrac{1}{2}\theta_2\cos\tfrac{1}{2}\theta_3 + \sin\tfrac{1}{2}\theta_3\cos\tfrac{1}{2}\theta_1\cos\tfrac{1}{2}\theta_2$$

$$q_3 = +\sin\tfrac{1}{2}\theta_1\cos\tfrac{1}{2}\theta_2\cos\tfrac{1}{2}\theta_3 + \sin\tfrac{1}{2}\theta_2\sin\tfrac{1}{2}\theta_3\cos\tfrac{1}{2}\theta_1$$

$$q_4 = -\sin\tfrac{1}{2}\theta_1\sin\tfrac{1}{2}\theta_3\cos\tfrac{1}{2}\theta_2 + \sin\tfrac{1}{2}\theta_2\cos\tfrac{1}{2}\theta_1\cos\tfrac{1}{2}\theta_3$$

$$\theta_1 = \tan^{-1}\left(\frac{-m_{31}}{m_{11}}\right)$$

$$\theta_2 = \tan^{-1}\left(\frac{m_{21}}{\sqrt{1-m_{21}^2}}\right)$$

$$\theta_3 = \tan^{-1}\left(\frac{-m_{23}}{m_{22}}\right)$$

(7) $M = M(Y(\theta_1), X(\theta_2), Y(\theta_3)) = YXY$

Axis Rotation Sequence: 2, 1, 2

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

$$q_1 = +\cos\tfrac{1}{2}\theta_2\cos(\tfrac{1}{2}(\theta_1 + \theta_3))$$
$$q_2 = +\sin\tfrac{1}{2}\theta_2\cos(\tfrac{1}{2}(\theta_1 - \theta_3))$$
$$q_3 = +\cos\tfrac{1}{2}\theta_2\sin(\tfrac{1}{2}(\theta_1 + \theta_3))$$
$$q_4 = -\sin\tfrac{1}{2}\theta_2\sin(\tfrac{1}{2}(\theta_1 - \theta_3))$$

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

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

$$\theta_3 = \tan^{-1}\left(\frac{m_{21}}{-m_{23}}\right)$$

(8) $M = M(Y(\theta_1), Z(\theta_2), Y(\theta_3)) = YZY$

Axis Rotation Sequence: 2, 3, 2

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

$$q_1 = +\cos\tfrac{1}{2}\theta_2\cos(\tfrac{1}{2}(\theta_1 + \theta_3))$$
$$q_2 = +\sin\tfrac{1}{2}\theta_2\sin(\tfrac{1}{2}(\theta_1 - \theta_3))$$
$$q_3 = +\cos\tfrac{1}{2}\theta_2\sin(\tfrac{1}{2}(\theta_1 + \theta_3))$$
$$q_4 = +\sin\tfrac{1}{2}\theta_2\cos(\tfrac{1}{2}(\theta_1 - \theta_3))$$

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

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

$$\theta_3 = \tan^{-1}\left(\frac{m_{23}}{m_{21}}\right)$$

(9) $M = M(Z(\theta_1), X(\theta_2), Y(\theta_3)) = ZXY$

Axis Rotation Sequence: 3, 1, 2

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

$$q_1 = -\sin\tfrac{1}{2}\theta_1\sin\tfrac{1}{2}\theta_2\sin\tfrac{1}{2}\theta_3 + \cos\tfrac{1}{2}\theta_1\cos\tfrac{1}{2}\theta_2\cos\tfrac{1}{2}\theta_3$$

$$q_2 = -\sin\tfrac{1}{2}\theta_1\sin\tfrac{1}{2}\theta_3\cos\tfrac{1}{2}\theta_2 + \sin\tfrac{1}{2}\theta_2\cos\tfrac{1}{2}\theta_1\cos\tfrac{1}{2}\theta_3$$

$$q_3 = +\sin\tfrac{1}{2}\theta_1\sin\tfrac{1}{2}\theta_2\cos\tfrac{1}{2}\theta_3 + \sin\tfrac{1}{2}\theta_3\cos\tfrac{1}{2}\theta_1\cos\tfrac{1}{2}\theta_2$$

$$q_4 = +\sin\tfrac{1}{2}\theta_1\cos\tfrac{1}{2}\theta_2\cos\tfrac{1}{2}\theta_3 + \sin\tfrac{1}{2}\theta_2\sin\tfrac{1}{2}\theta_3\cos\tfrac{1}{2}\theta_1$$

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

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

$$\theta_3 = \tan^{-1}\left(\frac{-m_{31}}{m_{33}}\right)$$

(10) $M = M(Z(\theta_1), Y(\theta_2), X(\theta_3)) = ZYX$

Axis Rotation Sequence: 3, 2, 1

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

$$q_1 = +\sin\tfrac{1}{2}\theta_1\sin\tfrac{1}{2}\theta_2\sin\tfrac{1}{2}\theta_3 + \cos\tfrac{1}{2}\theta_1\cos\tfrac{1}{2}\theta_2\cos\tfrac{1}{2}\theta_3$$

$$q_2 = -\sin\tfrac{1}{2}\theta_1\sin\tfrac{1}{2}\theta_2\cos\tfrac{1}{2}\theta_3 + \sin\tfrac{1}{2}\theta_3\cos\tfrac{1}{2}\theta_1\cos\tfrac{1}{2}\theta_2$$

$$q_3 = +\sin\tfrac{1}{2}\theta_1\sin\tfrac{1}{2}\theta_3\cos\tfrac{1}{2}\theta_2 + \sin\tfrac{1}{2}\theta_2\cos\tfrac{1}{2}\theta_1\cos\tfrac{1}{2}\theta_3$$

$$q_4 = +\sin\tfrac{1}{2}\theta_1\cos\tfrac{1}{2}\theta_2\cos\tfrac{1}{2}\theta_3 - \sin\tfrac{1}{2}\theta_2\sin\tfrac{1}{2}\theta_3\cos\tfrac{1}{2}\theta_1$$

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

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

$$\theta_3 = \tan^{-1}\left(\frac{m_{32}}{m_{33}}\right)$$

(11) $M = M(Z(\theta_1), X(\theta_2), Z(\theta_3)) = ZXZ$

Axis Rotation Sequence: 3, 1, 3

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

$$q_1 = +\cos\tfrac{1}{2}\theta_2\cos(\tfrac{1}{2}(\theta_1 + \theta_3))$$
$$q_2 = +\sin\tfrac{1}{2}\theta_2\cos(\tfrac{1}{2}(\theta_1 - \theta_3))$$
$$q_3 = +\sin\tfrac{1}{2}\theta_2\sin(\tfrac{1}{2}(\theta_1 - \theta_3))$$
$$q_4 = +\cos\tfrac{1}{2}\theta_2\sin(\tfrac{1}{2}(\theta_1 + \theta_3))$$

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

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

$$\theta_3 = \tan^{-1}\left(\frac{m_{31}}{m_{32}}\right)$$

12) $M = M(Z(\theta_1), Y(\theta_2), Z(\theta_3)) = ZYZ$

Axis Rotation Sequence: 3, 2, 3

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

$$q_1 = +\cos\tfrac{1}{2}\theta_2\cos(\tfrac{1}{2}(\theta_1 + \theta_3))$$

$$q_2 = -\sin\tfrac{1}{2}\theta_2\sin(\tfrac{1}{2}(\theta_1 - \theta_3))$$

$$q_3 = +\sin\tfrac{1}{2}\theta_2\cos(\tfrac{1}{2}(\theta_1 - \theta_3))$$

$$q_4 = +\cos\tfrac{1}{2}\theta_2\sin(\tfrac{1}{2}(\theta_1 + \theta_3))$$

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

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

$$\theta_3 = \tan^{-1}\left(\frac{m_{32}}{-m_{31}}\right)$$

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

```
@FOR,IS     EULMAT,EULMAT
FOR S3E3-02/19/77-06:24:23 (,0)

    SUBROUTINE EULMAT     ENTRY POINT 000237

    STORAGE USED: CODE(1) 000250; DATA(0) 000134; BLANK COMMON(2)

    EXTERNAL REFERENCES (BLOCK, NAME)

     0003   SIN
     0004   COS
     0005   NERR3$

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

     0001    000105 100L        0001    000113 116G        0001    000010
     0001    000142 162G        0001    000144 165G        0001    000146
     0001    000177 30L         0000 R  000033 P           0000 R  000052
     0000    000057 INJP$       0000 I  000046 J           0000 I  000044
     0000 R  000047 STNA        0000 R  000053 TEMP        0000 R  000000

00101              SUBROUTINE EULMAT(ISEQ,EUL,A)
00103              DIMENSION ISEQ(3),EUL(3),A(3,3)
00104              DIMENSION X(3,3,3),R(3,3)
00105              DO 100 K=1,3
00110              DO 10 I=1,3
00113              DO 5 J=1,3
00116              X(I,J,K)=0.0
00117              IF(I.EQ.J) X(I,J,K)=1.0
00121           5  CONTINUE
00122          10  CONTINUE
00125              IF(ISEQ(K).LE.0) GO TO 100
00127              SINA=SIN(EUL(K))
00130              COSA=COS(EUL(K))
00131              IF(ISEQ(K).EQ.2) GO TO 20
00133              IF(ISEQ(K).EQ.3) GO TO 30
00135              X(2,2,K)=COSA
00136              X(2,3,K)=-SINA
00137              X(3,2,K)= SINA
00140              X(3,3,K)=COSA
00141              GO TO 100
00142          20  X(1,1,K)=COSA
00143              X(1,3,K)= SINA
00144              X(3,1,K)=-SINA
00145              X(3,3,K)=COSA
00146              GO TO 100
00147          30  X(1,1,K)=COSA
00150              X(1,2,K)=-SINA
00151              X(2,1,K)= SINA
00152              X(2,2,K)=COSA
```

EULER ANGLES TO THE TRANSFORMATION MATRIX

(CONTINUED)

```
00153        100 CONTINUE
00155            DO 400 L=1,2
00160            M=3-L
00161            DO 300 I=1,3
00164            DO 300 J=1,3
00167            TEMP=0.0
00170            DO 250 K=1,3
00173            IF(L.EQ.1) HOLD=X(K,J,3)
00175            IF(L.EQ.2) HOLD=B(K,J)
00177            IF(ABS(HOLD).LT.1.0E-10) GO TO 250
00201            IF(ABS(X(I,K,M)).LT.1.0E-10) GO TO 250
00203            TEMP=TEMP+X(I,K,M)*HOLD
00204        250 CONTINUE
00206            IF(L.EQ.1) B(I,J)=TEMP
00210            IF(L.EQ.2) A(I,J)=TEMP
00212        300 CONTINUE
00215        400 CONTINUE
00217            RETURN
00220            END
```

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

ORIGINAL PAGE IS
OF POOR QUALITY

```
@FOR,IS    MATEUL,MATEUL
FOR S3E3-02/19/77-06:24:25 (,0)

    SUBROUTINE MATEUL      ENTRY POINT 000335

    STORAGE USED: CODE(1) 000353; DATA(0) 000052; BLANK COMMON(2)

    EXTERNAL REFERENCES (BLOCK, NAME)

     0003    SQRT
     0004    ATAN2
     0005    NERR3$

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

     0001      000054 10L         0001      000066 13L         0001      000217
     0001      000115 30L         0001      000022 4L          0001      000332
     0001      000051 50L         0001      000257 6L          0001      000327
     0000 R    000004 BSIGN       0000 R    000005 CSIGN       0000 R    000001
     0000 R    000023 ENUM        0000 I    000000 I           0000 I    000000
     0000 I    000001 J           0000 I    000012 JJ          0000 I    000003


 00101           SUBROUTINE MATEUL(ISEQ,A,EUL)
 00103           DIMENSION A(3,3),EUL(3)
 00104           DIMENSION ISEQ(3)
 00105           I=ISEQ(1)
 00106           J=ISEQ(2)
 00107           K=ISEQ(3)
 00110           IECK=0
 00111           IF(I.EQ.K) IECK=4.95
 00113           BSIGN=1.0
 00114           CSIGN=1.0
 00115           IF(I.EQ.1) GO TO 10
 00117           IF(I.EQ.2) GO TO 20
 00121           IF(J.NE.1) GO TO 5
 00123           BSIGN=-1.0
 00124           IF(IECK.NE.0) L=2
 00126           GO TO 30
 00127         5 CSIGN=-1.0
 00130           IF(IECK.NE.0) L=1
 00132           GO TO 30
 00133        10 IF(J.NE.2) GO TO 15
 00135           BSIGN=-1.
 00136           IF(IECK.NE.0) L=3
 00140           GO TO 30
 00141        15 CSIGN=-1.
 00143           IF(IECK.NE.0) L=2
 00144           GO TO 30
 00145        20 IF(J.NE.3) GO TO 25
 00147           BSIGN=-1.0
```

TRANSFORMATION MATRIX TO THE EULER ANGLES
(CONTINUED)

```
00150          IF(IEOK.NE.0) L=1
00152          GO TO 30
00153       25 CSIGN=-1.0
00154          IF(IEOK.NE.0) L=3
00156       30 DO 100 N=1,3
00161          FNSGN=1.0
00162          FDSGN=1.0
00163          IF(N.EQ.2) GO TO 70
00165          IF(N.EQ.1) GO TO 50
00167          IF(IEOK.NE.0) GO TO 40
00171          FNSGN=BSIGN
00172          JJ=I
00173          GO TO 45
00174       40 JJ=L
00175          IF(BSIGN.GT.0.0) FDSGN=-1.0
00177       45 FNUM=FNSGN*A(I,J)
00200          FDEN=FDSGN*A(I,JJ)
00201          GO TO 90
00202       50 IF(IEOK.NE.0) GO TO 55
00204          FNSGN=BSIGN
00205          II=K
00206          JJ=K
00207          GO TO 60
00210       55 FDSGN=BSIGN
00211          II=L
00212          JJ=I
00213       60 FNUM=FNSGN*A(J,K)
00214          FDEN=FDSGN*A(II,JJ)
00215          GO TO 90
00216       70 IF(IEOK.NE.0) GO TO 80
00220          FNUM=CSIGN*A(I,K)
00221          FDEN=SQRT(1.0-A(I,K)**2)
00222          GO TO 90
00223       80 FNUM=SQRT(1.0-A(I,I)**2)
00224          FDEN=A(I,I)
00225       90 EUL(N)=ATAN2(FNUM,FDEN)
00226      100 CONTINUE
00230          RETURN
00231          END

END OF COMPILATION:          NO  DIAGNOSTICS.
```

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

ORIGINAL PAGE IS OF POOR QUALITY

```
@FOR,IS        QMAT,QMAT
FOR S?E3-02/19/77-06:24:19 (,0)

   SUBROUTINE QMAT        ENTRY POINT 000077

   STORAGE USED: CODE(1) 000103; DATA(0) 000013; BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003    NERR3$

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

    0000     000007 INJP$      0000 R 000002 P2          0000 R 000001
    0000 R   000004 P5         0000 R 000005 TEMP


 00101          SUBROUTINE QMAT(Q,A)
 00103          DIMENSION Q(4),A(3,3)
 00104          P2=Q(2)+Q(2)
 00105          P3=Q(3)+Q(3)
 00106          P4=Q(4)+Q(4)
 00107          P5=P2*Q(2)
 00110          P6=P4*Q(4)
 00111          TEMP=1.0-P3*Q(3)
 00112          A(1,1)=TEMP-P6
 00113          A(2,2)=1.0-P5-P6
 00114          A(3,3)=TEMP-P5
 00115          P5=P2*Q(3)
 00116          P6=P4*Q(1)
 00117          A(1,2)=P5-P6
 00120          A(2,1)=P5+P6
 00121          P5=P2*Q(4)
 00122          P6=P3*Q(1)
 00123          A(1,3)=P5+P6
 00124          A(3,1)=P5-P6
 00125          P5=P3*Q(4)
 00126          P6=P2*Q(1)
 00127          A(2,3)=P5-P6
 00130          A(3,2)=P5+P6
 00131          RETURN
 00132          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.

TRANSFORMATION MATRIX TO THE QUATERNION

ORIGINAL PAGE IS OF POOR QUALITY

```
@FOR,SI MATQ,MATQ
FOR S1E3-02/19/77-06:24:21 (,0)

SUBROUTINE MATQ       ENTRY POINT 000003

STORAGE USED: CODE(1) 000220; DATA(0) 000050; BLANK COMMON(2)

EXTERNAL REFERENCES (BLOCK, NAME)

 0003    SQRT
 0004    NERR2$

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

 0001   000073 10L        0001   000052 1076       0001   000157
 0001   000111 35L        0001   000121 40L        0001   000163
 0001   000213 INJP$      0000 I 000000 J          0000 R 000000

                     SUBROUTINE MATQ(A,Q)
                     DIMENSION A(3,3),Q(4),T(4)
                     I=0
                     BIG=0.0
                     DO 40 J=1,4
                     Q(J)=0.0
                     IF(J.EQ.2) GO TO 10
                     IF(J.EQ.3) GO TO 20
                     IF(J.EQ.4) GO TO 30
                     Q(J)=1.0
                     TEMP=A(1,1)+A(2,2)+A(3,3)+1.0
                     T(J)=1.0
                     GO TO 35
                 10  TEMP=A(1,1)-A(2,2)-A(3,3)+1.0
                     T(J)=A(3,2)-A(2,3)
                     GO TO 35
                 20  TEMP=-A(1,1)+A(2,2)-A(3,3)+1.0
                     T(J)=A(1,3)-A(3,1)
                     GO TO 35
                 30  TEMP=-A(1,1)-A(2,2)+A(3,3)+1.0
                     T(J)=A(2,1)-A(1,2)
                 35  IF(TEMP.LT.BIG) GO TO 40
                     BIG=TEMP
                     I=J
                 40  CONTINUE
                     IF(I.EQ.0) GO TO 60
                     Q(I)=0.5*SQRT(BIG)
                     IF(I.NE.1) Q(1)=ABS(0.25*T(I)/Q(I))
                     TEMP=0.25/Q(1)
                     DO 50 J=2,4
                     Q(J)=TEMP*T(J)
                 50  CONTINUE
                 60  RETURN
                     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.

YAW-PITCH-ROLL EULER ANGLES TO THE QUATERNION

ORIGINAL PAGE IS
OF POOR QUALITY

```
@FOR,S.     YPRQ,YPRQ
FOR S0E3-02/19/77-06:24:33 (,0)

   SUBROUTINE YPRQ        ENTRY POINT 000114

   STORAGE USED: CODE(1) 000121; DATA(0) 000025; BLANK COMMON(2) 00

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003    POSNOR
    0004    COS
    0005    SIN
    0006    NERR3$

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

    0000 R 000001 CP         0000 R 000011 CR         0000 R 000007 C
    0000 R 000004 HY         0000   000016 INJP$      0000 R 000000 C
    0000 R 000012 SY

 00101        1*           SUBROUTINE YPRQ(YPR,Q0)
 00103        2*           DIMENSION YPR(3),Q(4),Q0(4)
 00104        3*           HY=0.50*YPR(1)
 00105        4*           HP=0.50*YPR(2)
 00106        5*           HR=0.50*YPR(3)
 00107        6*           CY=COS(HY)
 00110        7*           CP=COS(HP)
 00111        8*           CR=COS(HR)
 00112        9*           SY=SIN(HY)
 00113       10*           SP=SIN(HP)
 00114       11*           SR=SIN(HR)
 00115       12*           Q(1)=CY*CP*CR+SY*SP*SR
 00116       13*           Q(2)=CY*CP*SR-SY*SP*CR
 00117       14*           Q(3)=CY*SP*CR+SY*CP*SR
 00120       15*           Q(4)=-CY*SP*SR+SY*CP*CR
 00121       16*           CALL POSNOR(Q,Q0)
 00122       17*           RETURN
 00123       18*           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(1) is negative:

   Set QO(I) = -Q(I) for I = 1, 2, 3, 4.

2. Set QO(I) = QO(I)/TEMP

   where TEMP = $\sqrt{QO_1^2 + QO_2^2 + QO_3^2 + QO_4^2}$

SELECTS THE POSITIVE QUATERNION AND NORMALIZES

```
@FOR,IS     POSNOR,POSNOR
FOR S3E3-02/19/77-06:24:14 (,0)

   SUBROUTINE POSNOR    ENTRY POINT 000055

   STORAGE USED: CODE(1) 000067; DATA(0) 000017; BLANK COMMON(2) 0

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003    SQRT
    0004    NERR3$

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

    0001    000016 111G       0001    000036 121G       0000 I 000002
    0000 R  000000  TEMP


00101      1*                 SUBROUTINE POSNOR(Q,QO)
00103      2*                 DIMENSION Q(4),QO(4)
00104      3*                 TEMP=1.0
00105      4*                 IF(Q(1).LT.0.0) TEMP=-1.0
00107      5*                 SUM=0.0
00110      6*                 DO 50 I=1,4
00113      7*                 QO(I)=TEMP*Q(I)
00114      8*                 SUM=SUM+QO(I)*QO(I)
00115      9*            50   CONTINUE
00117     10*                 TEMP=1.0/SQRT(SUM)
00120     11*                 DO 100 I=1,4
00123     12*                 QO(I)=TEMP*QO(I)
00124     13*           100   CONTINUE
00126     14*                 RETURN
00127     15*                 END

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

ORIGINAL PAGE IS
OF POOR QUALITY

NASA-JSC