NASA TECHNICAL MEMORANDUM

NASA TM-78252

TORQUE EQUILIBRIUM ATTITUDE CONTROL FOR SKYLAB REENTRY

By John R. Glaese and Hans F. Kennel
Systems Dynamics Laboratory

November 1979

(NASA-TM-78252) TORQUE EQUILIBRIUM ATTITUDE CONTROL FOR SKYLAB REENTRY (NASA) 173 p HC A08/MF A01 CSCL 22B

N80-14169

Unclas
G3/15 46460

NASA

George C. Marshall Space Flight Center
Marshall Space Flight Center, Alabama

TECHNICAL REPORT STANDARD TITLE PAGE

1. REPORT NO. NASA TM-78252	2. GOVERNMENT ACCESSION NO.	3. RECIPIENT'S CATALOG NO.
4. TITLE AND SUBTITLE Torque Equilibrium Attitude Control for Skylab Reentry		5. REPORT DATE November 1979
		6. PERFORMING ORGANIZATION CODE
7. AUTHOR(S) John R. Glaese and Hans F. Kennel		8. PERFORMING ORGANIZATION REPORT #
9. PERFORMING ORGANIZATION NAME AND ADDRESS George C. Marshall Space Flight Center, Marshall Space Flight Center, Alabama 35812		10. WORK UNIT NO.
		11. CONTRACT OR GRANT NO.
12. SPONSORING AGENCY NAME AND ADDRESS National Aeronautics and Space Administration, Washington, D.C. 20456		13. TYPE OF REPORT & PERIOD COVERED Technical Memorandum
		14. SPONSORING AGENCY CODE

15. SUPPLEMENTARY NOTES

Prepared by Systems Dynamics Laboratory, Science and Engineering

16. ABSTRACT

Deliberate relocation of the Skylab footprint (to lower the chances of injury or damage) required controlling to a high-drag and a low-drag attitude (with a transition capability from one to the other at any time) to an altitude below 150 km (80 n.mi.).

However, the Skylab attitude control system was designed for holding a solar inertial attitude (temporary Earth-pointing was possible at the expense of cold-gas thruster fuel) at an altitude (435 km or 235 n.mi.) where the gravity gradient torques were predominant and the aerodynamic torques were practically negligible.

To control Skylab to below 150 km (where the aerodynamic torques were 10 times higher than the gravity gradient torques), without the use of thruster fuel (once the attitude was established), required the design, development, and software implementation of a completely new method: Active seeking of a torque equilibrium attitude.

This report describes all the available torque equilibrium attitudes (most were useless from the standpoint of lack of electrical power), the equilibrium seeking method, as well as the actual successful application during the 3 weeks prior to Skylab reentry.

17. KEY WORDS Skylab, Attitude Control, Aerodynamic Torques, Angular Momentum Management, Quaternions, Spacecraft, Reentry		18. DISTRIBUTION STATEMENT Unclassified-Unlimited	
19. SECURITY CLASSIF. (of this report) Unclassified	20. SECURITY CLASSIF. (of this page) Unclassified	21. NO. OF PAGES 174	22. PRICE NTIS

MSFC - Form 3292 (Rev December 1972)

For sale by National Technical Information Service, Springfield, Virginia 22151

ACKNOWLEDGMENTS

We hereby express our appreciation for the excellent support we received in the simulation area from the MSFC simulation division, especially Mr. Jack Lucas and Dr. Frank Hay of MSFC and Mr. Ray Nix of Honeywell.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS (Continued)

LIST OF ILLUSTRATIONS (Concluded)

LIST OF TABLES

TECHNICAL MEMORANDUM 78252

TORQUE EQUILIBRIUM ATTITUDE CONTROL FOR SKYLAB REENTRY

INTRODUCTION

With the decision in December 1978 to discontinue efforts to keep the Skylab Space Station (Fig. 1) in orbit came the decision to terminate the highly successful end-on-velocity-vector (EOVV) mode of operation [1] and to reestablish the solar inertial (SI) mode as soon as practical. This would put the spacecraft in an ideal power attitude and greatly reduce ground management of systems. It would also increase the drag on the spacecraft and cause it to reenter sooner. Unfortunately the SI attitude was not going to be maintainable much below 280 km (150 n.mi.) since the growing density of the atmosphere was going to cause the aerodynamics to grow to the point where the storage capacity of the control moment gyroscopes (CMG's) for angular momentum was inadequate. Since control of attitude to 150 km or below would be required to be able to influence reentry and the amount of thruster gas was far too low to consider control with thrusters, another new attitude control scheme had to be developed. Aerodynamic torques are proportional to density and become nearly overwhelming at 150 km, and therefore any attitude control scheme which would work at 150 km would have to accurately take into account the aerodynamic disturbances. Thus it became clear that we must look at the torque equilibrium attitudes (TEA's) if any existed and plan our control schemes about these. Thus the important early questions were:

1) Are there equilibrium attitudes?

2) If so, is adequate solar power available?

An investigation with the mathematical aerodynamic model of the Skylab vehicle indicated there were no aerodynamic trim or equilibrium attitudes but there may be attitudes where aerodynamic, gravity gradient, and gyroscopic torques balance. Indeed, 12 such attitudes were found. Most were not useable as control attitudes since there would be insufficient solar energy available to power the spacecraft and battery power would be completely inadequate for the several weeks required. Only 3 of the 12 TEA's appeared viable if a control scheme could be developed for them. So work began on a TEA control scheme. A candidate scheme which was promising was developed, but there were large uncertainties

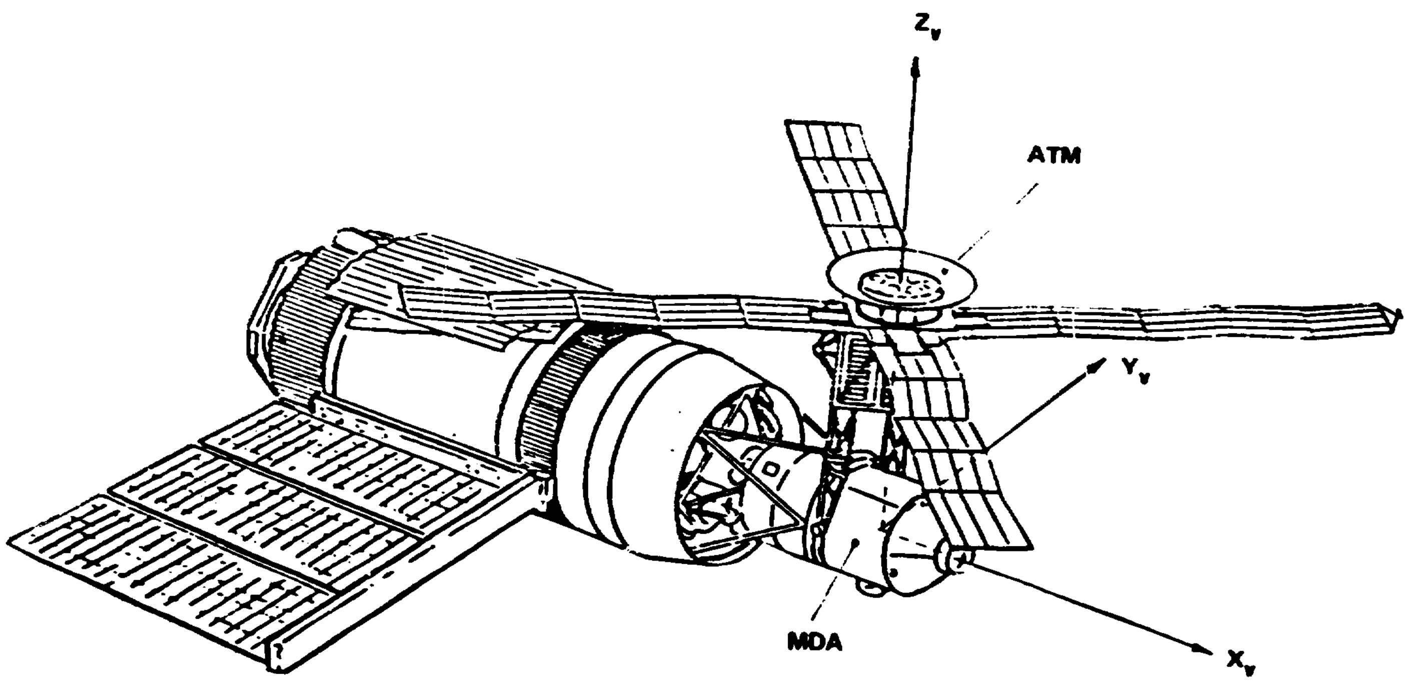


Figure 1. Skylab.

because of lack of confidence in the aerodynamic coefficients. This lack of confidence was due to the lack of test data confirming the aero moment model. As a result we did not have high confidence that the new TEA scheme would work. Later flight performance would show that our models were much better than we expected. The various mathematical models and tools needed for TEA control are described here. The operational aspects (time lines, command history, etc.) of TEA control as well as more complete background information and descriptions of the other essential Skylab systems (power, telemetry, etc.) are contained in Reference 2.

COORDINATE SYSTEMS

The coordinate systems which are pertinent to Skylab TEA control are defined in this section. Each system has some special geometrical or physical feature which simplifies the solution of a particular problem.

The following coordinate systems are described in this section: Orbital, Vehicle, Attitude Reference, Solar Inertial, and Z-Local Vertical. Each coordinate system consists of a set of mutually orthogonal axes exhibiting right-handedness.

An inertial (with respect to rotation only) coordinate system is a system which retains its orientation with respect to the celestial sphere, although the origin may be moving along any general curvilinear path in space. Similarly, a vehicle fixed system retains its orientation with respect to the vehicle.

Orbital Coordinate System (O)

The Orbital Coordinate System (X_O, Y_O, Z_O) is a precessing coordinate system with its origin at the Earth center of mass. The rate of precession about the Earth's north pole is approximately -5 degrees/day. The Z_O axis lies in the orbital plane, positive through the ascending node of the orbit. The X_O axis also lies in the orbital plane 90 degrees ahead of the Z_O axis. Since the Skylab orbit was in the $X_O Z_O$ plane at all times, the Y_O axis was parallel to the orbital angular momentum vector, completing the right-handed system (Fig. 2).

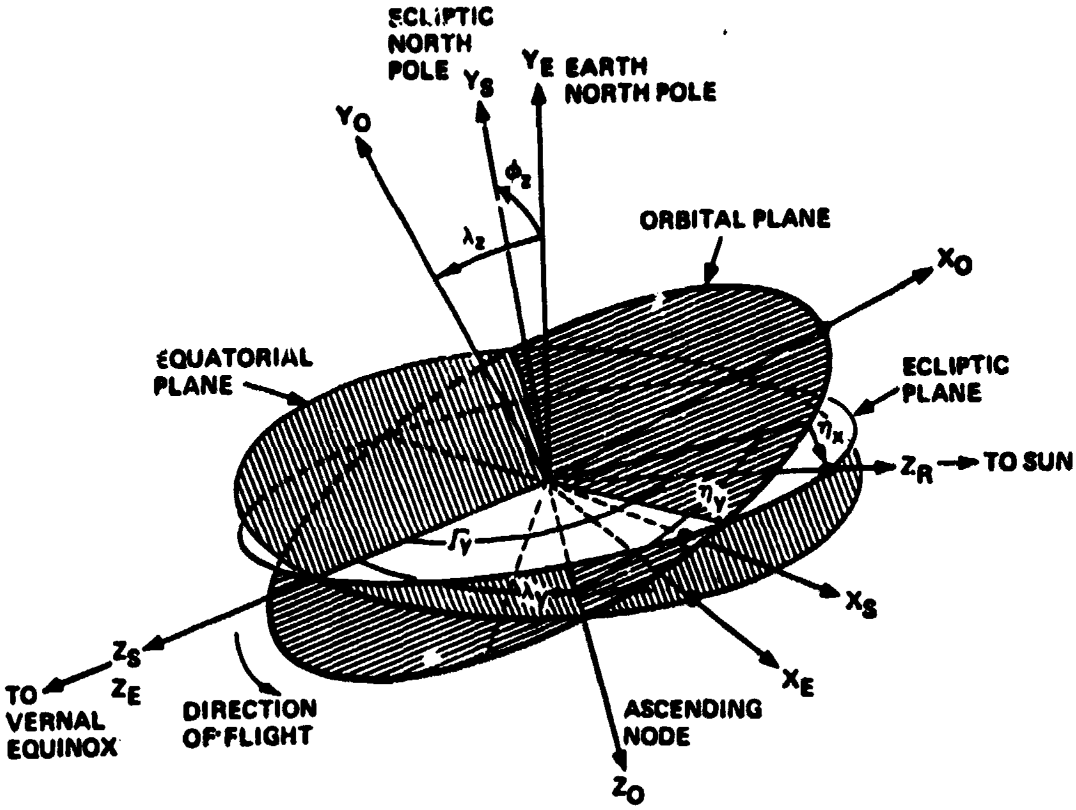


Figure 2. Coordinate systems.

Vehicle Coordinate System (V)

The Vehicle Coordinate System (X_V, Y_V, Z_V) is a vehicle-fixed system with its origin at the center of mass. The X_V axis lies along the long axis of Skylab and is positive in the direction of the Multiple Docking Adapter (MDA). The Z_V axis is positive toward the Apollo Telescope Mount (ATM) and the Y_V axis completes the right-handed system.

Attitude Reference Coordinate System (A)

The Attitude Reference Coordinate System (X_A, Y_A, Z_A) is a movable system with its origin coincident with the Vehicle Coordinate System origin. The axes of this system represent the instantaneous desired orientation of the Vehicle Coordinate System axes.

Solar Inertial Coordinate System (I)

The Solar Inertial Coordinate System (X_I, Y_I, Z_I) is only a pseudo-inertial system since it makes one revolution per year. It was used during the Skylab mission to point the instruments in the desired direction. The origin is coincident with the origin of the Vehicle Coordinate System origin. The Z_I axis is positive toward the center of the Sun. The X_I axis lies at an angle ν_z from the orbital plane (this angle is calculated on-board such that the principal X axis is in the orbital plane to minimize the build-up of angular momentum) and is positive toward the sunset terminator.

Z-Local Vertical Coordinate System (L)

The Z-Local Vertical Coordinate System (X_L, Y_L, Z_L) is a rotating system with its origin at the center of mass of Skylab (the rate of rotation is one revolution per orbit). The X_L axis is positive in the direction of flight and lies in the orbital plane. The Z_L axis is parallel to the local vertical direction and is positive outward, away from the Earth. The Y_L axis is parallel to the orbit normal and is positive toward orbital North.

SKYLAB ATTITUDE AND POINTING CONTROL SYSTEM (APCS)

The actual control of the Skylab attitude to the attitude reference was done exactly as in the original mission [3]. However, only the pointing control system (PCS) of the APCS was used; the experiment pointing control system (EPCS) was disabled. The major parts of the APCS were the rate gyros, the sun sensors, the star tracker (it had failed during the original Skylab mission), the Apollo Telescope Mount Digital Computer (ATMDC), the Workshop Computer Interface Unit (WCIU), double-gimbaled CMG's, and cold-gas (compressed nitrogen) Thruster Attitude Control System (TACS).

Six control modes were addressable: (1) STANDBY, (2), SOLAR INERTIAL (SI), (3) EXPERIMENT POINTING, (4) ATTITUDE HOLD/CMG, (5) ATTITUDE HOLD/TACS, and (6) ZLV. TEA control was programmed to be a substate of the ZLV mode. The basic ZLV (for Z axis along the local vertical) attitude was with the Z axis along the local vertical, pointing up, and the X axis in the orbital plane, pointing in the direction of the velocity vector. Any angular offset from the basic ZLV attitude (offset identified by the quaternion Q_{AL}) could be commanded via a set of three Euler angles (χ) with a 2-3-1 rotation sequence. Changes in the TEA attitude reference were achieved by commanding changes to Q_{AL},

$$[AL] = [\chi_x]_1 \, [\chi_z]_3 \, [\chi_y]_2 \quad .$$ [1]

Since none of the original APCS capabilities were eliminated by the addition of the TEA control, the torque equilibrium attitude control could also be held by TACS only (this option was contemplated for low-drag attitude at low altitudes in case low-drag was needed for reentry control, but active TEA control had proven impossible).

The TACS consisted of software plus six cold-gas thrusters, two uncoupled ones for control about the Y axis (one would be fired for +Y, the other for -Y control); four other thrusters for coupled X and Z control [4].

CMG CONTROL SYSTEM

The CMG control system was composed of three orthogonally mounted, double-gimbaled CMG's with angular momentum magnitude H of 3050 Nms (2280 ft-lb-sec) as shown in Figure 3. The CMG control law utilized three normalized torque commands and the CMG momentum status to generate proper CMG gimbal rate commands [5]. The CMG control law consisted of three parts: CMG steering law, rotation law, and gimbal stop avoidance logic. There also were some other routines for specialized situations like caging the CMG's to a desired momentum state [4].

---

1. [ ]-quantities are usually 3 × 3 rotation matrices, with the further definitions (where χ_i represents any angle; s = sin, c = cos):

$$[\chi_x]_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\chi_x & s\chi_x \\ 0 & -s\chi_x & c\chi_x \end{bmatrix} ; \quad [\chi_y]_2 = \begin{bmatrix} c\chi_y & 0 & -s\chi_y \\ 0 & 1 & 0 \\ s\chi_y & 0 & c\chi_y \end{bmatrix} ;$$

$$[\chi_z]_3 = \begin{bmatrix} c\chi_z & s\chi_z & 0 \\ -s\chi_z & c\chi_z & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

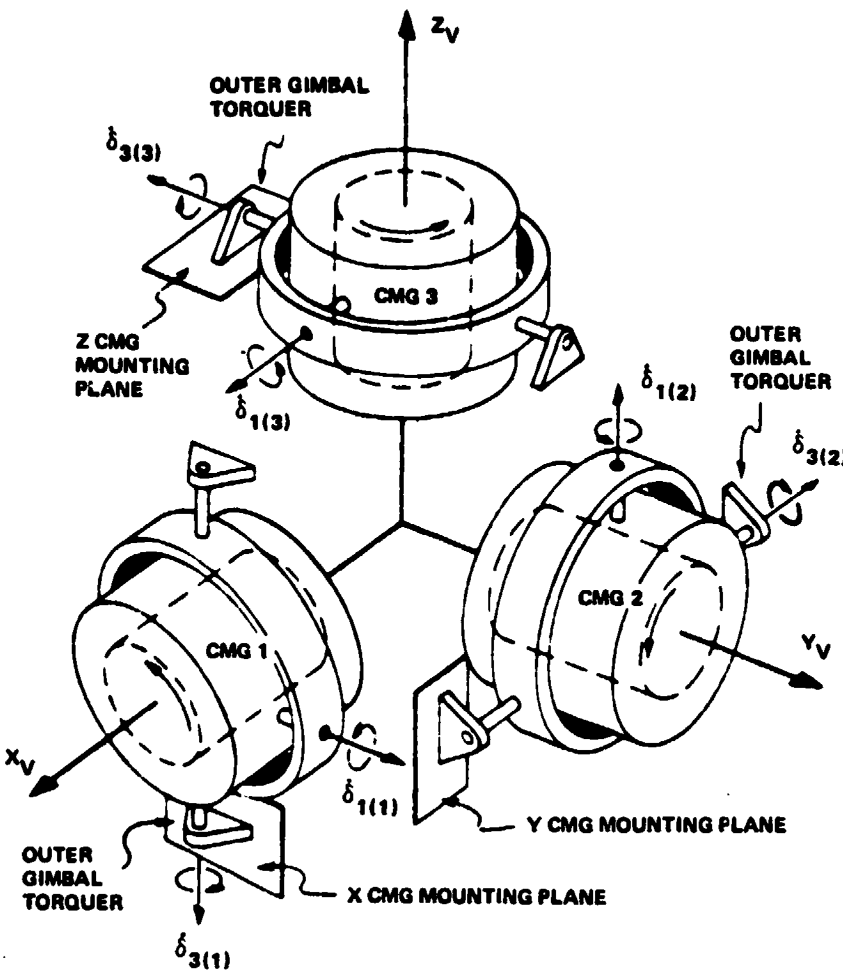


Figure 3. CMG mounting arrangement.

The CMG control law had the ability to operate with either three or two CMG's for redundancy. Since CMG No. 1 had failed during the original Skylab mission, the CMG control law was always in the two-CMG option.

BANK, ATTACK, AND ROLL ANGLES

The aerodynamic forces and torques are dependent on the aerodynamic angle of attack and roll angles. To complete the specification of an arbitrary attitude, a third rotation angle is required. This angle is the so-called bank angle. The sequence is

$$[VL] = [\rho]_1 \, [\alpha]_2 \, [\beta]_1 \quad . \tag{1}$$

These angles are obtained from the following relations:

$$\alpha = \text{arc cos } (VL_{11}) \tag{2}$$

$$\left.\begin{aligned} \beta &= \text{arc tan } (VL_{12}/-VL_{13}) \quad (3) \\ \rho &= \text{arc tan } (VL_{21}/VL_{31}) \quad (4) \end{aligned}\right\} \quad \text{four quadrant} \quad .$$

The Skylab onboard computer works in quaternions so that the matrix [VL] must be computed from the given quaternions Q_{VI}, Q_{VA}, Q_{AL}:

$$Q_{VL} = \bar{\bar{Q}}_{VA} \, Q_{AL} \quad .^2 \tag{5}$$

The matrix [VL] is derived from Q_{VL} by taking the upper left 3 × 3 matrix from

$$[VL] = (\text{upper left } 3 \times 3 \text{ of}) \; \bar{\bar{Q}}_{VL}^{-1} \, \tilde{\tilde{Q}}_{VL} \quad .^2 \tag{6}$$

The physical bank, attack, and roll (BAR) angles will differ slightly from those above, because the orbit of Skylab is not truly circular and the air tends to move with the Earth surface so that the velocity of Skylab relative to the air is not exactly just the orbital velocity. Thus when equations (A2) and (A3) are used, values slightly different from those in equations (2) and (3) must be used. These values were only used for evaluation purposes and proved to be quite useful, so the differences were insignificant.

---

2. See Appendix E for nomenclature.

TORQUE EQUILIBRIUM ATTITUDES

As mentioned previously the aerodynamic coefficients are only a function of the roll angle and the angle-of-attack. Figure 4 shows the zero-moment curves for the three components with respect to these angles (see Table 1 for mapping of the vehicle axes). As can be seen, there is no set of angles, where all three curves intersect. However, they come close in several areas. To get a true three-axis equilibrium, other external torques are required. Gravity gradient (GG) torques and gyroscopic torques were found to be sufficiently large for all altitudes of concern to create 12 TEA's altogether. Gyroscopic torques come into play, since the aerodynamic and GG torques are constant with respect to the rotating local vertical coordinate system, and the total angular momentum of the Skylab was selectable within certain limits (imposed by the finite storage capacity of the CMG system). Since the aerodynamic torques do not change (relative to body-fixed axes) when the Skylab is rotated about the relative wind velocity vector and GG torques do not change in body-fixed axes for a rotation of 180 degrees about any axis perpendicular to the local vertical, there is always a pair of TEA's with the same angle-of-attack/roll angle combination, but with bank angles differing by 180 degrees, i.e., there were actually only six basically different TEAs with respect to the aerodynamic torques (they are indicated in Fig. 4 by asterisks). The TEA's are shown for an altitude of 200 km (108 n.mi.) and zero total angular momentum (Table 2).

From solar panel power considerations a sun-pointing, inertially fixed attitude, as given in the SI mode, is the best and TEA is rather bad. Nine of the TEAs were completely hopeless when it was established that 28 percent of full Sun illumination was required, on the average, to supply the needed power (100 percent is the power received when the Sun is perpendicular to the solar panels and the vehicle is in an all-daylight orbit, as is the case for high Sun elevation angles with respect to the orbital plane). Only the remaining three TEA's were usable, and these only when the optimum angular momentum was used (each of the TEA's exists in a volume of the BAR angle space when the total angular momentum is varied within its available volume). Even then, some of the TEAs did not have enough power for certain Sun elevation angles. The three TEA's were named T121P, T121G, and T275, and they are shown in Figures 5, 6, and 7 where the point of view is slightly south of the orbital plane and the vehicle is moving from the front lower left to the back upper right. T275 was a low-drag attitude (the ballistic coefficient was approximately 275) with the MDA trailing; T121G and T121P were high-drag attitudes (the ballistic coefficients were about 121). In T121G the Skylab was approximately in a GG equilibrium attitude with the MDA pointing upward (the same attitude which Skylab had been left in when it was deactivated in early 1974); in T121P the MDA was almost perpendicular to the orbital plane and pointing South. For T121G and T121P

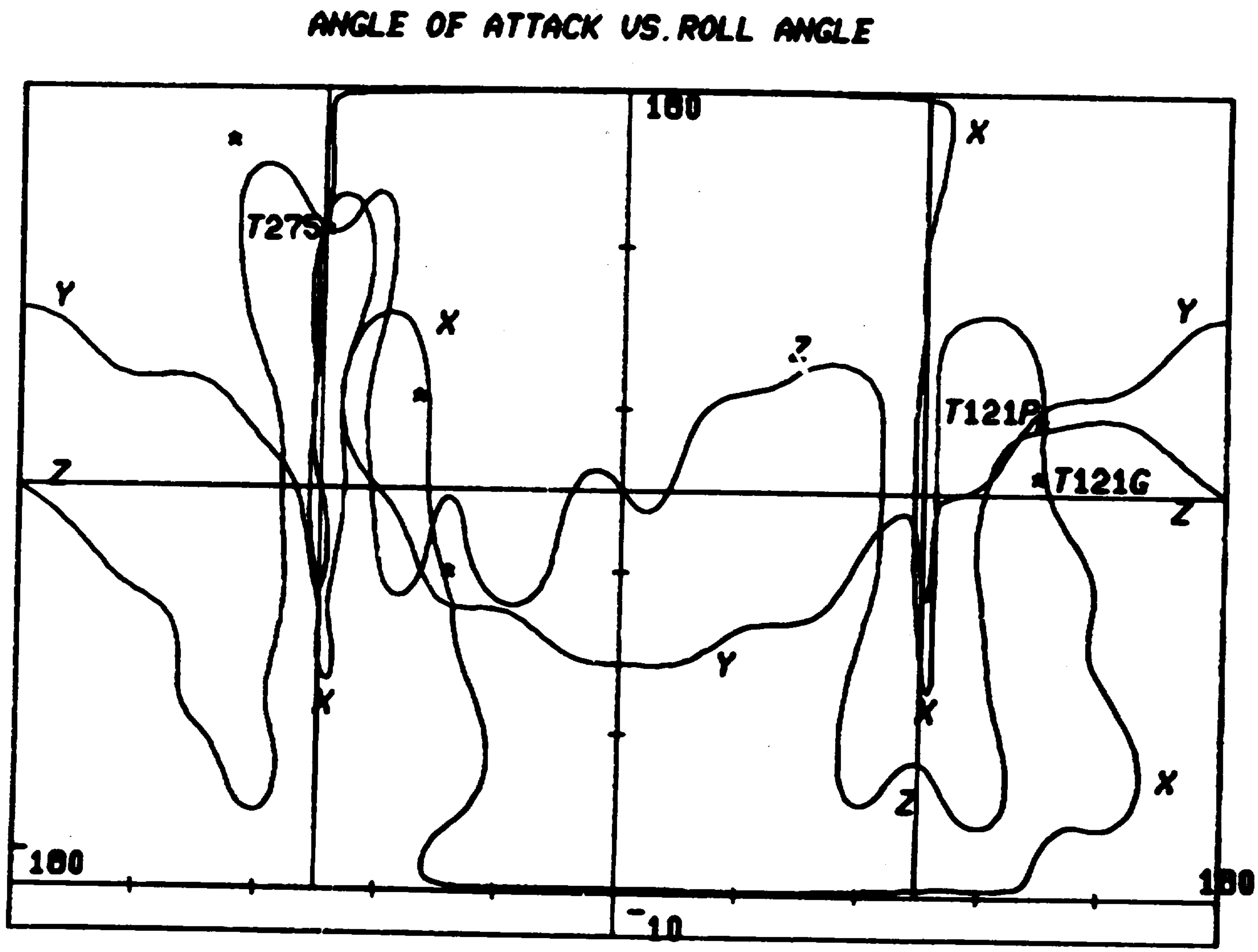


Figure 4. Combined zero torque contours.

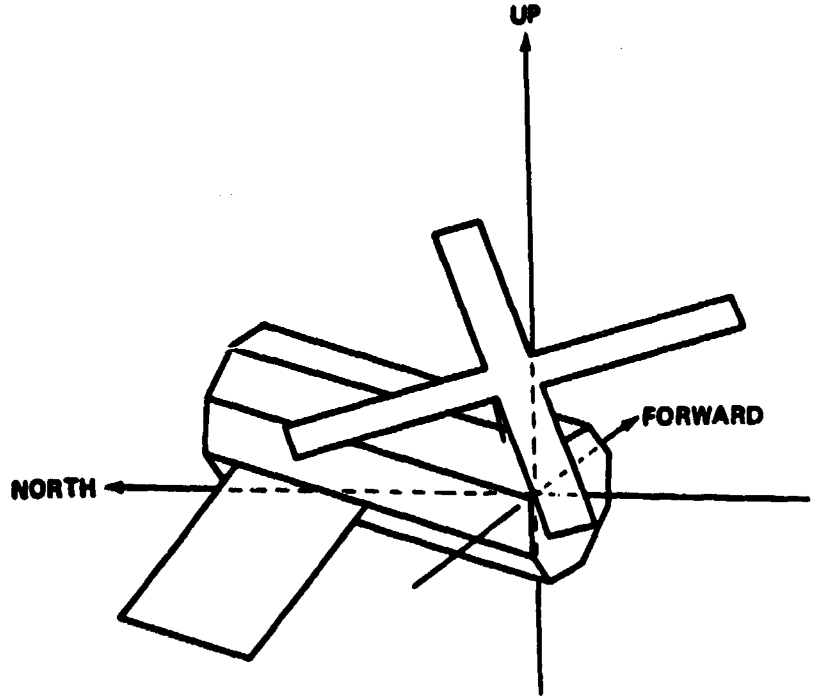
TABLE 1. MAPPING OF VEHICLE AXES

Axis	Angle of Attack (deg)	Roll Angle (deg)
+x	0	any
-x	180	any
+y	90	90
-y	90	-90
+z	90	0
-z	90	±180

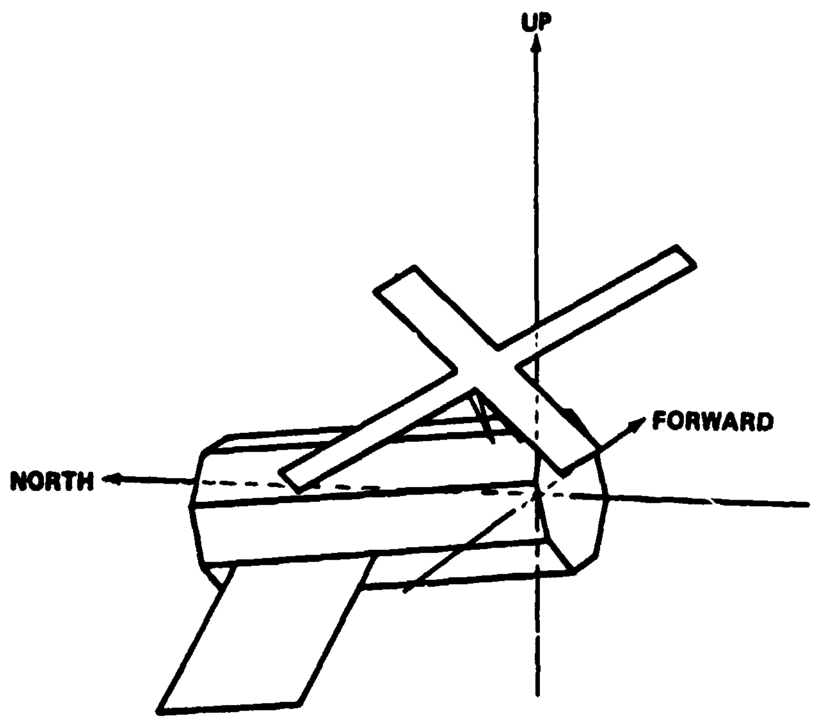
TABLE 2. TEA'S FOR δ = 3.11E-10 kg/m^3
(~200 km or 108 n.mi.)

No.	ID	Bank Angle (deg)	Angle of Attack (deg)	Roll Angle (deg)	CD	BC
1	T121P/SU	-78.6	106.0	+124.9	6.66	132.5
2	ND	+101.4				
3	T121G/UN	-173.2	93.8	+124.0	6.94	127.3
4	DS	+6.8				
5	T275/BU	+70.2	147.8	1-89.2	3.52	250.5
6	BD	-109.8				
7	BN	-51.4	168.1	-117.6	2.72	324.9
8	BS	+128.6				
9	DN	-11.9	110.6	-60.7	6.37	138.9
10	US	+168.1				
11	NU	+86.4	71.8	-51.2	7.11	124.2
12	SD	-93.6				

Note: These TEA's are for zero angular momentum and a CP offset of $[+0.06, -0.07, -0.11]^T$ m. See Appendix G for nomenclature.

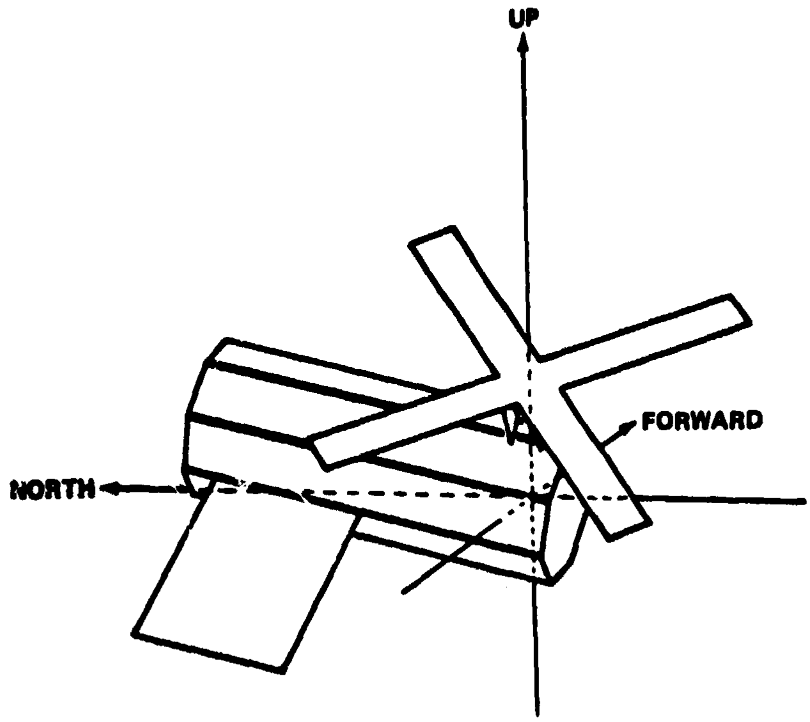


a. No angular momentum.



b. -1H angular momentum in the Z axis .

Figure 5. T121P.



c. +0.5H angular momentum in the X axis.

Figure 5. (Concluded).

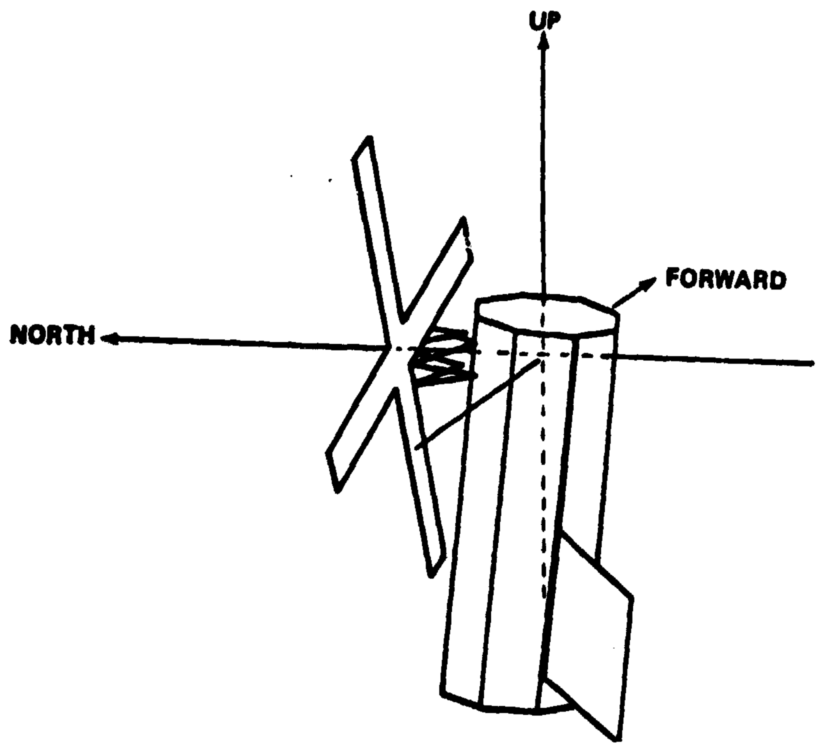


Figure 6. T121G (no angular momentum).

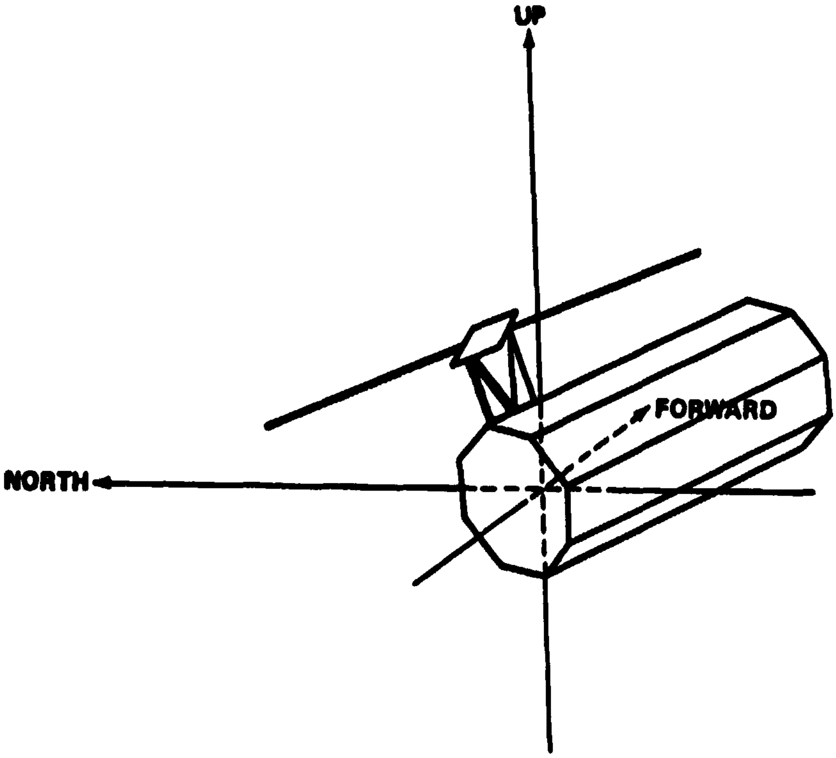


Figure 7. T275 (no angular momentum).

the solar panels were trailing and they were statically stable with respect to aerodynamic torques. T121G was also statically stable with respect to the GG torques (therefore stable in all axes) whereas T121P was in an unstable GG equilibrium.

TEA SEEKING METHOD

The assumption is made that the total external torque, T_{ext}, acting on Skylab is changing linearly with the attitude offset, $\Delta\phi_{off}$, and that the partial derivatives of the torque with respect to the offset are known.

$$\underline{T}_{ext} = \left[\frac{\partial T_i}{\partial \phi_j}\right] \underline{\Delta\phi}_{off} \quad ^3 \tag{7}$$

3. An underlined quantity (as $\underline{T}_{ext}$) is a vector with three components.

where

$$\left[\frac{\partial T_i}{\partial \phi_j}\right] \triangleq \begin{bmatrix} \frac{\partial T_1}{\partial \phi_1} & \frac{\partial T_1}{\partial \phi_2} & \frac{\partial T_1}{\partial \phi_3} \\ \frac{\partial T_2}{\partial \phi_1} & \frac{\partial T_2}{\partial \phi_2} & \frac{\partial T_2}{\partial \phi_3} \\ \frac{\partial T_3}{\partial \phi_1} & \frac{\partial T_3}{\partial \phi_2} & \frac{\partial T_3}{\partial \phi_3} \end{bmatrix} .$$

Evaluation of the on-board total angular momentum change over the desaturation interval, T_{des}, results in an estimate of the total external torques,

$$\underline{T}_{ext} = (\underline{H} - \underline{H}_p)/T_{des} , \qquad (8)$$

where $\underline{H}$ is the present total angular momentum and $\underline{H}_p$ is its past value. The attitude offset from the torque equilibrium attitude, assuming the offset is constant, is then

$$\underline{\Delta\phi}_{off} = \left[\frac{\partial T_i}{\partial \phi_j}\right]^{-1} \underline{T}_{ext} . \qquad (9)$$

Changing the attitude reference by $-\underline{\Delta\phi}_{off}$ would, ideally, eliminate a further angular momentum change. However, the previously accumulated angular momentum away from a desired momentum state, $\underline{H}_{nom}$, has to be eliminated during the next desaturation interval by

$$\underline{\Delta\phi}_{mom} = -\left[\frac{\partial T_i}{\partial \phi_j}\right]^{-1} (\underline{H} - \underline{H}_{nom})/T_{des} . \qquad (10)$$

The total required attitude change is therefore

$$\underline{\Delta\phi} = \underline{\Delta\phi}_{mom} - \underline{\Delta\phi}_{off}$$

$$\underline{\Delta\phi} = \frac{-1}{T_{des}} \left[\frac{\partial T_i}{\partial \phi_j}\right]^{-1} [(\underline{H} - \underline{H}_{nom}) + (\underline{H} - \underline{H}_p)] \quad . \tag{11}$$

This method for attitude change eliminates, ideally, any initial condition within two desaturation intervals. [The parenthetical expressions in equation (11) were not combined since these quantities had to be limited separately.]

In the Skylab software, all angular momentum quantities were normalized by the nominal angular momentum magnitude, H, of one CMG, and they were called $\underline{e}$. Equation (11) then becomes

$$\underline{\Delta\phi} = [\text{SLOPE}] (\underline{\Delta e}_m + \underline{\Delta e} - \underline{\Delta e}_p) \quad , \tag{12}$$

where

$$[\text{SLOPE}] = \frac{-H}{T_{des}} \left[\frac{\partial T_i}{\partial \phi_j}\right]^{-1}$$

$$\underline{\Delta e} = (\underline{H} - \underline{H}_{nom})/H$$

$$\underline{\Delta e}_m = \text{the limited value of } \underline{\Delta e}$$

$$\underline{\Delta e}_p = \text{the past unlimited value of } \underline{\Delta e} .$$

The reorientation capability of Skylab was limited and therefore $\underline{\Delta\phi}$ had to be limited also. To avoid the possibility of a large momentum offset overcoming the signal due to a momentum change, $\underline{\Delta e}_m$ is limited to a value which cannot command more than approximately 80 percent of the limit on $\underline{\Delta\phi}$.

The actual reference change is done by generating a quaternion

$$\Delta Q_{AL} = [\underline{\Delta\phi}/2, \sqrt{1 - 0.25\ \underline{\Delta\phi}^2}\ ]^T \tag{13}$$

and updating the reference quaternion[4]

$$Q_{AL}(\text{new}) = \bar{\bar{\Delta Q}}_{AL}\ Q_{AL}(\text{old}) \quad . \tag{14}$$

STRAPDOWN DRIFT CORRECTION

The TEA control mode contains a provision for strapdown updating from the ground. This capability was used extensively to maintain TEA control. The error in the strapdown was estimated by several techniques which gave either the strapdown error components in the orbit plane or along the orbit normal. The only source of data for strapdown error about the orbit normal (or orbital Y axis) is data from the Sun sensor or the solar arrays. These data are available only occasionally from the Sun sensor when its line of sight along the vehicle Z axis comes within approximately 18 degrees of the Sun over a ground station. Strapdown error along orbit normal was estimated from the timing error of solar passage through the vehicle XZ plane. This event happened once per orbit on the daylight side. Passage occurred when the X Sun sensor output passed through null. Null passage was too indistinct to be reliable when the Sun was greater than 18 degrees away from the vehicle Z axis at passage. For those periods, the solar array power angle or the aerodynamic roll angle deviation was used to estimate the strapdown error. These latter techniques were less reliable and were used only when a consistent trend was established or when more than one indicator was telling the same story.

The strapdown is the onboard vehicle attitude reference. The vehicle angular rates are integrated using quaternions as attitude parameters. Since the rate data contains errors which are integrated into these quaternions, they must be corrected periodically to eliminate offsets from the true reference. Such corrections are referred to as updates. These strapdown updates are applied through an update quaternion which is quaternion multiplied onto the strapdown quaternion. The vehicle strapdown quaternion is called Q_{VI}, and the update quaternion is $\underline{\Delta Q}_{VI}$. This quaternion is constructed from the vector angle $\underline{\Delta\theta}_L$:

$$\Delta Q_{VI} = [0.5[VL]\underline{\Delta\theta}_L, \sqrt{1 - 0.25\underline{\Delta\theta}_L^{\,2}}\ ]^T \quad . \tag{15}$$

4. See Appendix E for nomenclature.

The quaternion update multiplication is performed as shown

$$Q_{VI}(\text{new}) = \overline{\overline{\Delta Q}}_{VI}\ Q_{VI}(\text{old}) \quad . \tag{16}$$

The onboard computer computes the quaternion Q_{LI} and from this comes

$$Q_{AI} = \bar{\bar{Q}}_{AL}\ Q_{LI} \tag{17}$$

and

$$Q_{VA} = \bar{\bar{Q}}_{VI}\ Q_{AI}^{-1} \quad . \tag{18}$$

The quaternion Q_{AL} is the commanded attitude of the vehicle relative to the local vertical reference. The quaternion Q_{VA} is the attitude error quaternion which is maintained close to identity by the onboard control system. A method for estimating the strapdown correction angle $\underline{\Delta \theta_L}$ is given in Appendix D.

Originally, an automatic on-board in-the-orbital-plane strapdown update (with only updates perpendicular to the orbital plane from the ground) had been considered. It was assumed that the active TEA control had brought the V system into alignment with the TEA at the start of the desaturation interval. At the end of the next desaturation interval any misalignment about the X_L axis then results in a first order offset from the ideal TEA about the Z_L axis. Using part of the $\underline{\Delta \phi}$ correction about the Z_L axis for a strapdown update should bring the estimate of the orbital Y_L axis into alignment with the actual orbital Y_L axis, since the misalignment is inertially fixed and the X_L misalignment sweeps the whole orbital plane. This scheme worked as long as the assumption was made that the relative wind direction is always colinear with the vehicle velocity vector. However, the atmosphere rotates with the Earth and therefore the relative wind direction oscillates approximately two degrees about the vehicle velocity vector. The result was that the onboard strapdown update settled out at an offset which was furthermore dependent upon the TEA used. Since the ground update method (which did not have these shortcomings) was developed at that time, the onboard update was disabled by setting its gain to zero. (There was not enough time available to take it completely out of the ATMDC software [2]. Therefore the ground update method was used through the already existing ground command capability, eliminating the need for any further software changes).

TEA OPERATION AND PERFORMANCE

Nominal flight parameters had to be continuously generated for several days in advance since the conditions tended to change (for example: estimated density, estimated CP location, etc.). The data were given to the flight controllers in the form of tables (Table 3[5] shows the data for DOY 171.5 through DOY 174.5 or 6/20/79 to 6/23/79 at noon GMT). Since the altitude decreased slowly and everybody needed time to get used to the TEA operation, the SLOPE was not changed during this time. ETLN (with components ETLN1, ETLN2, ETLN3 in the L system) is the normalized nominal angular momentum. Z-AXIS BETA is the elevation angle of the vehicle Z axis with respect to the orbital plane. CD and BC are the drag and the ballistic coefficients, respectively. CMG MOM is the angular momentum in the CMG system and it is in percent of 3H. ETSF is the normalized and filtered total system momentum in vehicle components. QBLNOM is the nominal attitude reference quaternion Q_{AL} (a QBL was existing already and could be used; therefore, the subscrips A and B are equivalent here).

Table 4 shows the Slope Matrix Update Log. The first three columns show the date when the Slope was updated (DOY 171 corresponds to 6/20 and DOY 192 to 7/11). The first entry is the start of TEA control rather than the time when the initial Slope was loaded. The index in the fourth column is added to facilitate correlation with Table 5. Columns 5 through 13 show the components of the Slope as indicated by the heading. Table 5 shows what data were used to generate the slopes. The slope generation parameters for index 2 and 3 are the same; the transpose of Slope 3 had been sent up (Slope 2) by mistake, whereupon control was promptly lost. Slope 3 was sent as soon as attitude control had been regained. Slope 13 was used for a higher altitude than the one for which it was calculated, since a 6 hr gap in the ground coverage eliminated an additional slope change and it was considered better to have the proper slope at a lower altitude and take a reduced gain at the higher altitudes.

Figure 8 shows the actual BAR angles for DOY 172:03:21 to DOY 172:06:12 versus time in seconds and the predicted BAR angles are shown as horizontal lines. It can also be noted that the BAR angle traces zig-zagged, indicating that the gain was too high. As a consequence the slope gain was later reduced to 0.5 from 1.0, resulting in a much smoother trace. The actual BAR angles are only correct to within the strapdown error, which, due to the availability of acquisition Sun Sensor (ACQ SS) information, was less than 1 degree. A very good correspondence between the nominal and the actual BAR angles can be seen, indicating that the prediction of the CM and CP locations, as well

5. In the APL computer language, a minus sign is a superscript minus (to distinguish it from the subtraction operation).

TABLE 3. NOMINAL FLIGHT PARAMETERS

INDEX	ALTITUDE KM	ALTITUDE NM	DOY	MONTH	DAY	HOUR
1	262.925	141.968	171.50	6	20	12
2	260.825	140.834	172.50	6	21	12
3	259.493	140.115	173.50	6	22	12
4	257.671	139.131	174.50	6	23	12

DOY	ALT	BETA	ETLN 1	ETLN 2	ETLN 3	BANK	ATTACK	ROLL
171.50	141.97	18.20	0.0	0.8	¯1.0	¯98.05	93.32	114.06
172.50	140.83	13.40	0.0	0.8	¯1.0	¯98.04	93.22	113.99
173.50	140.11	8.60	0.0	0.8	¯1.0	¯98.04	93.37	114.09
174.50	139.13	4.10	0.0	0.8	¯1.0	¯98.04	93.60	114.24

DOY	ALT	BETA	ETLN 1	ETLN 2	ETLN 3	CHIX	CHIY	CHIZ
171.50	141.97	18.20	0.0	0.8	¯1.0	¯88.22	¯112.52	¯81.38
172.50	140.83	13.40	0.0	0.8	¯1.0	¯87.73	¯111.95	¯81.34
173.50	140.11	8.60	0.0	0.8	¯1.0	¯88.53	¯112.86	¯81.28
174.50	139.13	4.10	0.0	0.8	¯1.0	¯89.70	¯114.19	¯81.19

DOY	ALT	BETA	ETLN 1	ETLN 2	ETLN 3	Z-AXIS BETA	CD	BC
171.50	141.97	18.20	0.0	0.8	¯1.0	8.78	6.55	134.72
172.50	140.83	13.40	0.0	0.8	¯1.0	8.65	6.55	134.79
173.50	140.11	8.60	0.0	0.8	¯1.0	8.71	6.56	134.68
174.50	139.13	4.10	0.0	0.8	¯1.0	8.81	6.56	134.51

TABLE 3. (Concluded)

ALT	DIR COS MATRIX COMPONENTS 1,1	2,1	3,1	1,2	2,2	3,2	1,3	2,3	3,3
141.97	¯0.058	0.912	¯0.407	¯0.988	0.005	0.151	0.140	0.411	0.901
140.83	¯0.056	0.912	¯0.406	¯0.989	0.006	0.150	0.140	0.410	0.901
140.11	¯0.059	0.911	¯0.407	¯0.988	0.004	0.151	0.140	0.412	0.901
139.13	¯0.063	0.910	¯0.410	¯0.988	0.001	0.153	0.140	0.415	0.899

ALT	SLOPE MATRIX COMPONENTS 1,1	2,1	3,1	1,2	2,2	3,2	1,3	2,3	3,3
141.97	2.038	¯0.791	¯1.466	2.171	¯0.410	1.398	4.271	1.119	1.557
140.83	**								
140.11	**								
139.13	**								

ALT	ETLN 1	ETLN 2	ETLN 3	CMG MOM X	CMG MOM Y	CMG MOM Z	GIMBAL ANGLES IG2	OG2	IG3	OG3	ETSF X	ETSF Y	ETSF Z
142.0	0.0	0.8	¯1.0	¯19	¯15	¯40	¯16	89	27	105	¯0.931	¯0.407	¯0.780
140.8	0.0	0.8	¯1.0	¯19	¯15	¯40	¯16	89	27	105	¯0.930	¯0.405	¯0.781
140.1	0.0	0.8	¯1.0	¯19	¯15	¯40	¯16	89	27	105	¯0.930	¯0.409	¯0.779
139.1	0.0	0.8	¯1.0	¯19	¯15	¯40	¯16	89	27	105	¯0.930	¯0.414	¯0.777

ALT	ETLN 1	ETLN 2	ETLN 3	QBLNOM 1	QBLNOM 2	QBLNOM 3	QBLNOM 4
142.0	0.0	0.8	¯1.0	0.0956	¯0.2012	¯0.6989	0.6796
140.8	0.0	0.8	¯1.0	0.0953	¯0.2005	¯0.6985	0.6803
140.1	0.0	0.8	¯1.0	0.0958	¯0.2014	¯0.6992	0.6793
139.1	0.0	0.8	¯1.0	0.0965	¯0.2027	¯0.7002	0.6777

TABLE 4. SLOPE MATRIX UPDATE LOG

DATE DAY	HR	MIN	IND	SLOPE MATRIX COMPONENTS 1,1	1,2	1,3	2,1	2,2	2,3	3,1	3,2	3,3
171	13	17	1	2.030	2.171	4.271	-0.791	-0.410	1.119	-1.466	1.398	1.557
175	16	14	2	1.975	-1.058	-2.125	1.719	-0.249	1.513	3.702	1.244	1.982
175	19	21	3	1.975	1.719	3.702	-1.058	-0.249	1.244	-2.125	1.513	1.982
176	15	57	4	2.598	1.309	2.003	-1.421	0.122	1.782	-2.005	2.066	2.925
177	15	43	5	1.664	0.319	0.711	-0.756	0.215	1.101	-1.478	1.218	1.684
180	13	45	6	1.904	0.004	0.072	-0.759	0.215	0.825	-0.973	0.873	0.716
181	13	32	7	1.522	0.137	0.212	-1.010	0.197	0.800	-1.303	0.842	0.683
183	20	52	8	1.088	0.093	0.130	-0.662	0.050	0.732	-1.167	0.736	0.736
186	13	50	9	0.784	0.053	-0.024	-0.239	-0.080	0.528	-0.555	0.515	0.379
189	20	21	10	0.378	0.073	-0.056	0.006	-0.193	0.351	-0.246	0.328	0.078
190	13	42	11	0.283	0.079	-0.061	0.046	-0.215	0.319	-0.204	0.294	0.022
191	4	9	12	0.208	0.084	-0.065	0.075	-0.232	0.295	-0.175	0.269	-0.020
191	17	57	13	0.189	0.189	-0.145	0.236	-0.531	0.539	-0.288	0.482	-0.165
192	3	33	14	0.186	0.204	-0.157	0.276	-0.578	0.541	-0.279	0.488	-0.219

TABLE 5. SLOPE GENERATION PARAMETERS

INDEX	SIGMA	GAIN	REF ALTITUDE KM	REF ALTITUDE NM	ETLN X	ETLN Y	ETLN Z	CP X	CP Y	CP Z	REF BAR ANGLES BANK	REF BAR ANGLES ATTACK	REF BAR ANGLES ROLL
1	-1	1.0	262.00	141.47	0.0	0.8	-1.0	0.00	0.00	0.00	-98.85	93.32	114.86
2	-1	1.0	255.52	137.97	0.0	0.8	-0.7	-0.06	-0.07	-0.16	-92.17	92.91	114.02
3	-1	1.0	255.52	137.97	0.0	0.8	-0.7	-0.06	-0.07	-0.16	-92.17	92.91	114.02
4	-1	1.0	253.13	136.68	0.0	0.8	-0.4	-0.06	-0.07	-0.16	-86.93	95.97	115.51
5	-1	0.5	250.59	135.31	0.0	0.8	-0.2	-0.06	-0.07	-0.16	-83.51	98.92	118.03
6	-1	0.5	241.78	130.55	0.1	1.2	0.0	0.00	0.00	0.00	-80.15	100.16	126.74
7	-1	0.5	238.33	128.69	0.2	1.2	0.0	0.00	0.00	0.00	-80.09	94.75	126.38
8	0	0.5	229.65	124.00	0.2	1.2	0.0	0.06	-0.07	-0.11	-78.89	94.56	122.36
9	0	0.5	212.33	114.65	0.2	1.2	0.0	0.06	-0.07	-0.11	-79.11	100.03	123.47
10	0	0.5	187.53	101.26	0.2	1.2	0.0	0.06	-0.07	-0.11	-77.38	103.38	124.48
11	0	0.5	179.94	97.16	0.2	1.2	0.0	0.06	-0.07	-0.11	-76.39	104.02	124.71
12	0	0.5	172.64	93.22	0.2	1.2	0.0	0.06	-0.07	-0.11	-75.11	104.58	124.88
13	0	1.0	158.94	86.36	0.2	1.2	0.0	0.06	-0.07	-0.11	-70.73	105.14	125.14
14	0	1.0	152.49	82.34	0.2	1.2	0.0	0.06	-0.07	-0.11	-67.08	105.32	125.22

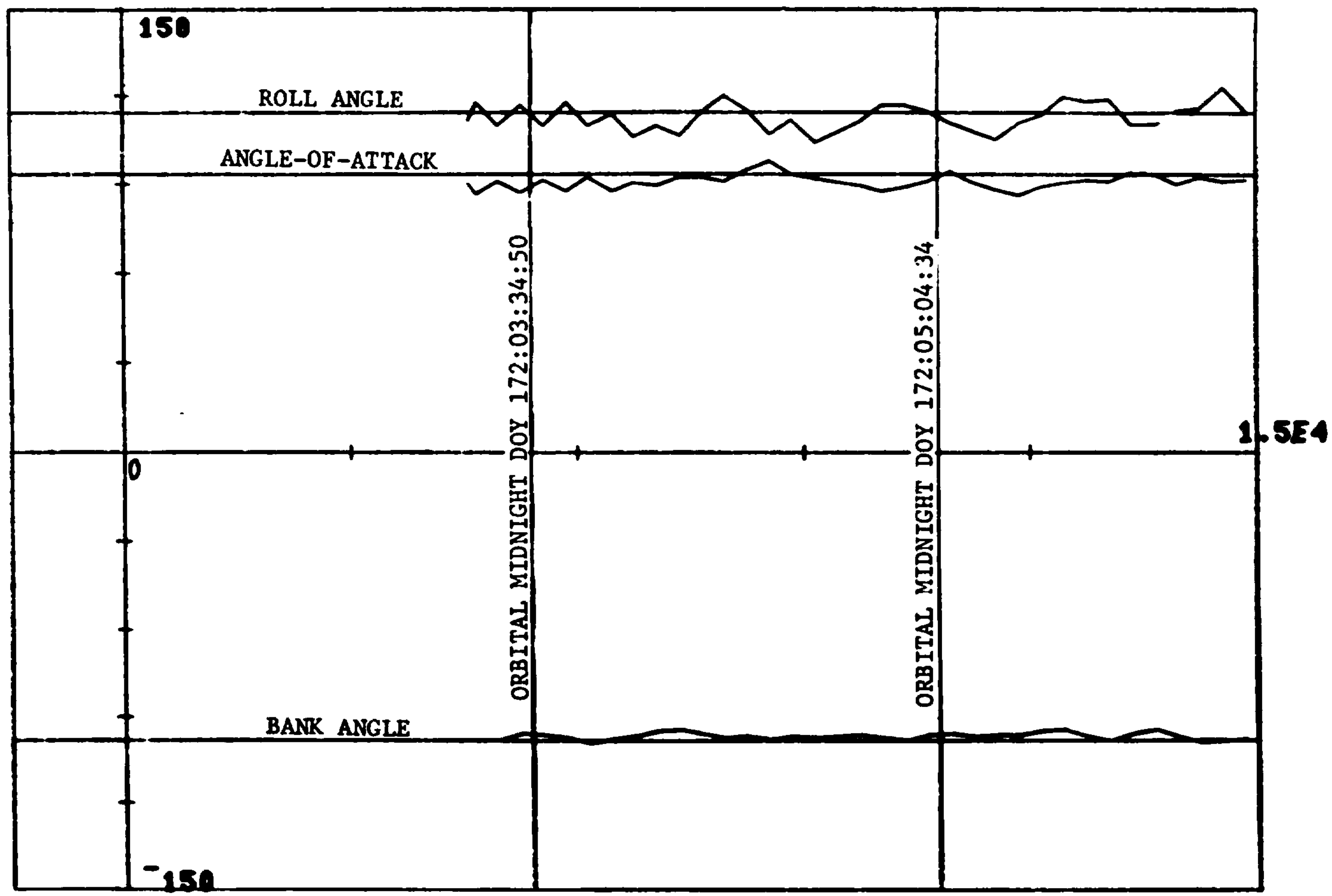
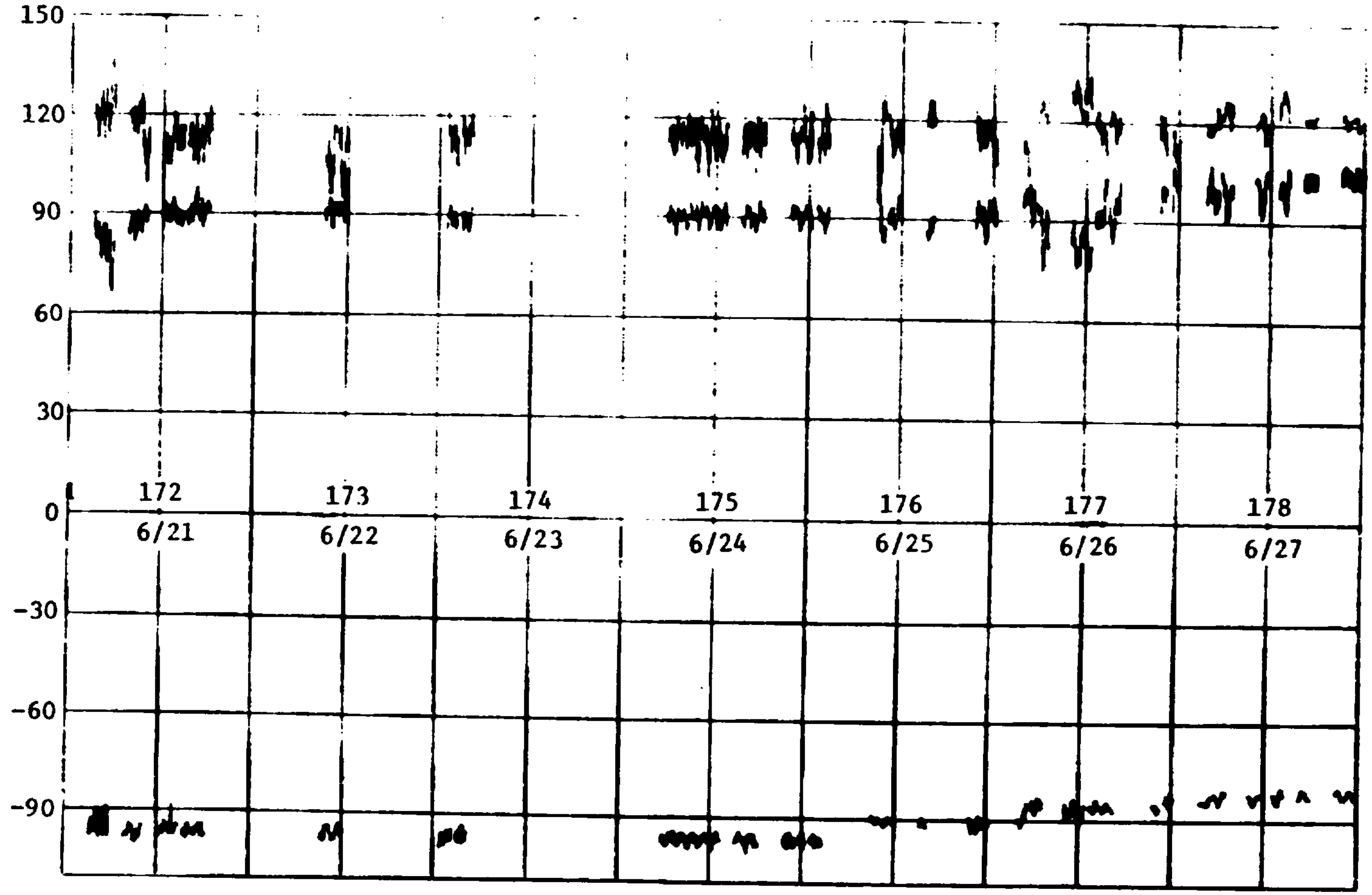


Figure 8. Actual BAR angles (DOY 172:03:21/172:06:12).

as the aerodynamic coefficients, was much better than anticipated. The nominal BAR angles were generated assuming that the relative wind velocity is antiparallel to the vehicle velocity vector, but reduced by an amount appropriate for the Earth rate. The nominal BAR angles are therefore the nominal average of the actual BAR angles. Some of the good correspondence is due to the fact that changes in ETLN1 were used to influence the roll angle. Initially, a roll angle smaller than nominal was very detrimental since the slope matrix changed drastically such that a roll angle of less than 90 degrees was found to be unstable: The actual slope components had changed sign. An initial unfavorable rate gyro bias could have changed the actual roll angle by that much long before the ground would have had enough data to detect and correct for it. To eliminate this possibility, a favorable rate gyro bias compensation was introduced into the software such that if we had had the bad gyro drift, the compensation would have eliminated it. As it turned out, the real rate gyro bias was in a good direction and the compensation had to be changed to the opposite polarity (as always, Skylab seemed to refute Murphy's Law, since anything that could go right, would; also see the last paragraph of this section). The great sensitivity to the rate-gyro introduced drift about orbital North was due to the relatively large negative angular momentum bias in the Z axis (ETLN3 = -1H) which then was resolved into the actual X_L axis thereby changing the equilibrium position drastically. This resolution also suggested the (at least temporary) remedy: Buck it with an appropriate amount of momentum bias in the X_L axis.

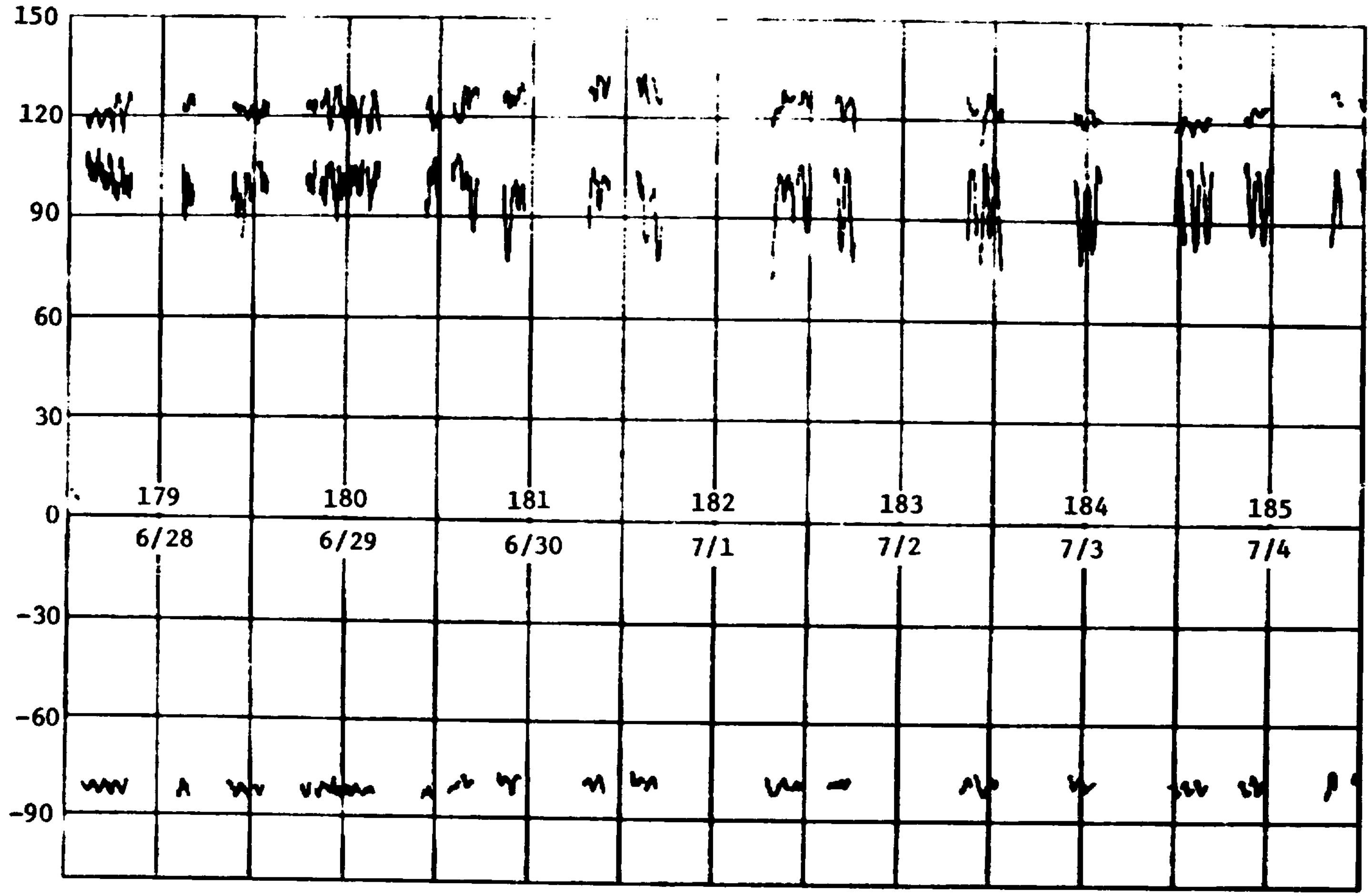
An overview of the BAR angles for the three weeks of TEA operation is given in Figure 9. The bank angle is the lowest trace, the angle of attack is in the middle, and the roll angle is the highest. The horizontal scale is in days, the numbers indicate the start of a day. Only BAR angles from the auxiliary storage and playback recorder (ASAP) tapes are shown, since they were saved in our ground computer.

TEA control was initiated on DOY 171:13:17 GMT (6/20/79) and Skylab impact occurred on DOY 192:16:37 GMT (7/11/79). During the latter half of DOY 171, it can be seen that the average roll angle steadily increased. This was due to the initial rate gyro bias (intentional compensation and basic bias). After corrective action was taken the angles settled close to their predicted values. Figures 9a and 9b show the BAR angles during the shift in ETLN3 from -1H to 0 between DOY 175.5 and DOY 180.5. With the gradual removal of the ETLN3 the sensitivity to rate gyro drift was removed; however, the excellent control capability over the roll angle (and with it, the drag) also vanished. In fact, the actual roll angle could later not be influenced at all (the indicated roll angle also showed the effect of strapdown error about orbit North and this fact in conjunction with the fact that the actual roll angle was steady, allowed the use of the indicated roll angle for strapdown update about orbit North).



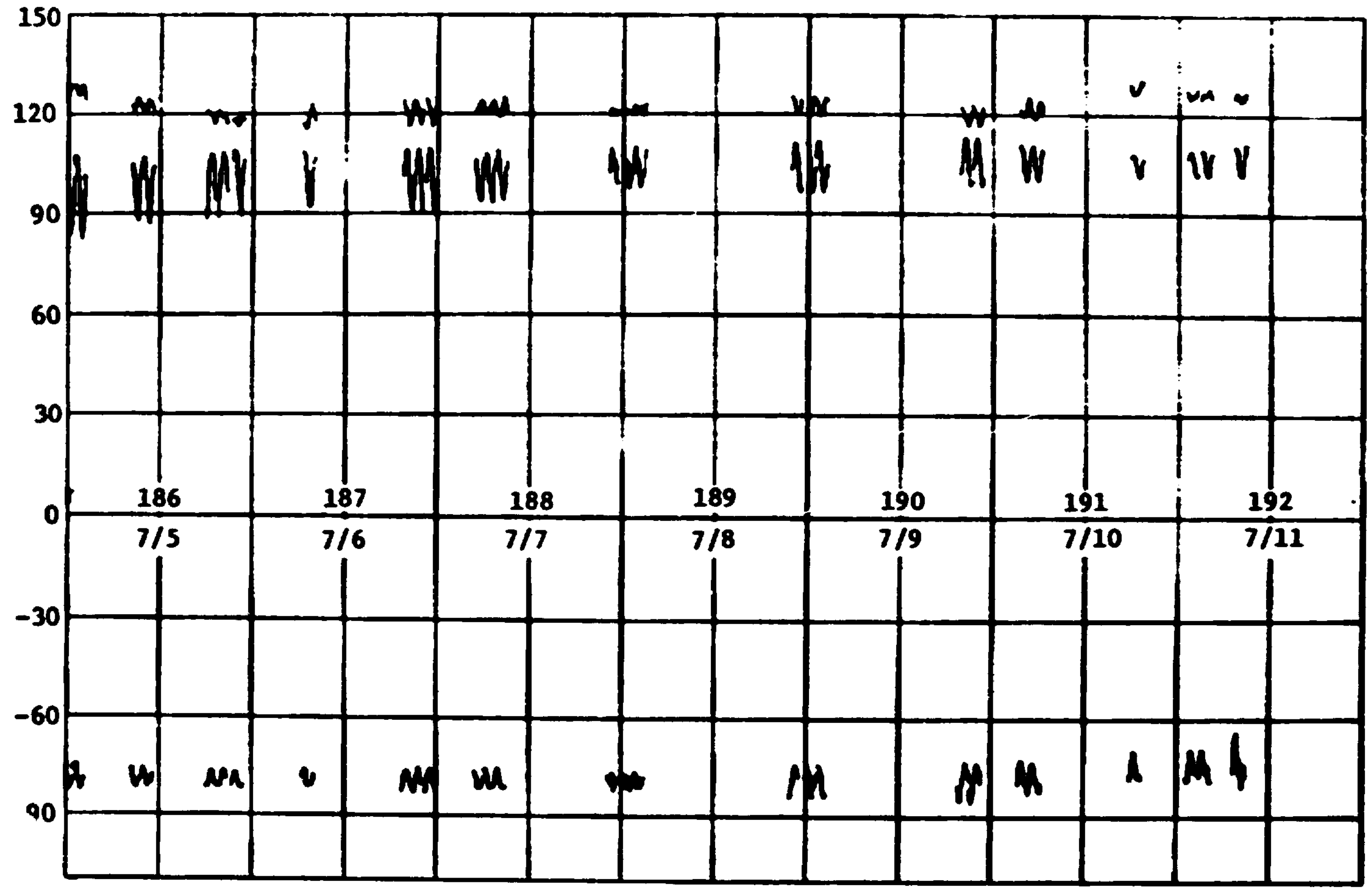
a. First week.

Figure 9. BAR angles.



b. Second week.

Figure 9. (Continued)



c. Third week.

Figure 9. (Concluded)

To illustrate the evaluation of the information on the ASAP tapes, DOY 187:16:56:30 to DOY 187:20:56:30 are taken as an example. Table 6 shows the basic QBL/time (QBL = Q_{AL}) information, the equivalent BAR angles are shown in Table 7 and graphed in Figure 10. The BAR angles were always evaluated for their average and a sinusoid of orbital frequency. The latter was fitted by a least-square fit, after the average had been subtracted. The results for our example are shown in Figures 11, 12, and 13. In the lower left quadrant of each figure there is a table of expected angle values (right column) as a function of the orbital angle (left column). Since the circle charts (Fig. 14) show this orbital angle, the tables were used to check the actual BAR angles (live; from telemetry) against these reference angles.

The sinusoids in the angle-of-attack and roll angle were due to the relative wind direction oscillating about the vehicle velocity direction (since the atmosphere rotates with the Earth). The sinusoid in the bank angle was due to the in the orbital-plane strapdown error and the appropriate strapdown update is shown in the upper left hand corner of Figure 11, both with respect to orbital midnight and with respect to the actual time; the update is to be telemetered. For the latter a resolution of the error with respect to midnight had to be made.

Application of the various methods was severely hampered by lack of data and large delays between the occurrence and the receipt of the data. If the strapdown error had drifted substantially there would have been no way to correct it in time to prevent loss of control. Fortunately, the rate gyro drift components in the orbital plane (if constant) integrate to exactly zero in one revolution and generally TEA control was not very sensitive to the drift components along the orbit normal. In-plane strapdown errors resulted from inaccurate navigation updates and from the Sun motion of about 1 degree/day. For data on strapdown errors, the onboard ASAP tape recorder had to be run for at least one full orbit (more consecutive orbits were desired for noise content reduction) during a quiet state; i.e., the last slope update had to be done two orbits before and no other strapdown correction was permitted. Once the ASAP tape was dumped over a ground station, it took at least another half hour of data processing and parameter extraction. The results were 18 sets of Q_{AL} per orbit, which then had to be fed into our Sigma 5 computer to be evaluated with the appropriate programs. The actual updating of the parameters had to wait until the next ground station. The bottom line was, that the delay in the actual onboard happening and the corrective action amounted to at least two orbits, sometimes more.

The ground support operations were aided by the "circle charts" (Fig. 14) and telemetry formats (Format 8 for TEA parameters is shown in Fig. 15). The data on Format 8 changed every second provided the Skylab was over a ground station; the circle charts were applicable for one orbit.

TABLE 6. REPRESENTATIVE QBL SET (DOY 186:16:56/186:20:56)

1	0.2275000	-0.1183000	-0.7917000	0.5544987	187:16:56:30.00
2	0.2345000	-0.1128000	-0.7684000	0.5846771	187:17: 1:30.00
3	0.2359000	-0.1260000	-0.7687000	0.5914400	187:17: 6:30.00
4	0.2343000	-0.1362000	-0.7540000	0.5973525	187:17:11:30.00
5	0.2324000	-0.1456000	-0.7450000	0.6000038	187:17:16:30.00
6	0.2336000	-0.1523000	-0.7314000	0.6223261	187:17:21:30.00
7	0.2374000	-0.1529000	-0.7211000	0.6326750	187:17:26:30.00
8	0.2422000	-0.1536000	-0.7133000	0.6394914	187:17:31:30.00
9	0.2435000	-0.1506000	-0.7136000	0.6374393	187:17:36:30.00
10	0.2378000	-0.1625000	-0.7361000	0.6125371	187:17:41:30.00
11	0.2252000	-0.1599000	-0.7545000	0.5953543	187:17:46:30.00
12	0.2184000	-0.1566000	-0.7725000	0.5753900	187:17:51:30.00
13	0.2108000	-0.1493000	-0.7792000	0.5710694	187:17:56:30.00
14	0.2115000	-0.1363000	-0.7850000	0.5660963	187:18: 1:30.00
15	0.2152000	-0.1263000	-0.7906000	0.5591059	187:18: 6:30.00
16	0.2213000	-0.1219000	-0.7971000	0.5404500	187:18:11:30.00
17	0.2256000	-0.1183000	-0.7991000	0.5445631	187:18:16:30.00
18	0.2260000	-0.1125000	-0.7947000	0.5516060	187:18:21:30.00
19	0.2351000	-0.1050000	-0.7700000	0.5730705	187:18:26:30.00
20	0.2341000	-0.1103000	-0.7609000	0.5846571	187:18:31:30.00
21	0.2351000	-0.1223000	-0.7599000	0.5935670	187:18:36:30.00
22	0.2318000	-0.1355000	-0.7507000	0.6036208	187:18:41:30.00
23	0.2297000	-0.1469000	-0.7437000	0.6103840	187:18:46:30.00
24	0.2322000	-0.1501000	-0.7320000	0.6217373	187:18:51:30.00
25	0.2373000	-0.1540000	-0.7187000	0.6349771	187:18:56:30.00
26	0.2422000	-0.1533000	-0.7100000	0.6423487	187:19: 1:30.00
27	0.2469000	-0.1529000	-0.7110000	0.6404225	187:19: 6:30.00
28	0.2377000	-0.1594000	-0.7370000	0.6116039	187:19:11:30.00
29	0.2253000	-0.1612000	-0.7612000	0.5863694	187:19:16:30.00
30	0.2169000	-0.1601000	-0.7740000	0.5718456	187:19:21:30.00
31	0.2080000	-0.1526000	-0.7760000	0.5744032	187:19:26:30.00
32	0.2093000	-0.1404000	-0.7820000	0.5689512	187:19:31:30.00
33	0.2150000	-0.1276000	-0.7897000	0.5602385	187:19:36:30.00
34	0.2200000	-0.1203000	-0.7953000	0.5519292	187:19:41:30.00
35	0.2212000	-0.1202000	-0.8015000	0.5424207	187:19:46:30.00
36	0.2252000	-0.1144000	-0.7968000	0.5409147	187:19:51:30.00
37	0.2303000	-0.1033000	-0.7832000	0.5682330	187:19:56:30.00
38	0.2302000	-0.1046000	-0.7686000	0.5876401	187:20: 1:30.00
39	0.2200000	-0.1103000	-0.7581000	0.5994210	187:20: 6:30.00
40	0.2300000	-0.1316000	-0.7501000	0.6059137	187:20:11:30.00
41	0.2311000	-0.1410000	-0.7420000	0.6131244	187:20:16:30.00
42	0.2320000	-0.1494000	-0.7294000	0.6259643	187:20:21:30.00
43	0.2381000	-0.1531000	-0.7202000	0.6333003	187:20:26:30.00
44	0.2451000	-0.1552000	-0.7145000	0.6366543	187:20:31:30.00
45	0.2400000	-0.1612000	-0.7224000	0.6247018	187:20:36:30.00
46	0.2359000	-0.1667000	-0.7376000	0.6103340	187:20:41:30.00
47	0.2244000	-0.1603000	-0.7591000	0.5896743	187:20:46:30.00
48	0.2130000	-0.1567000	-0.7730000	0.5766690	187:20:51:30.00
49	0.2064000	-0.1470000	-0.7820000	0.5683434	187:20:56:30.00

TABLE 7. BAR ANGLES FOR REPRESENTATIVE QBL SET

1	¯76.19	106.35	120.81
2	¯76.50	101.91	120.21
3	¯77.66	100.90	121.15
4	¯78.81	100.17	121.65
5	¯80.14	98.77	121.97
6	¯81.19	96.68	122.34
7	¯81.40	94.98	122.54
8	¯81.41	93.71	122.90
9	¯81.62	93.94	123.44
10	¯81.23	97.85	123.67
11	¯81.25	100.93	122.69
12	¯80.67	104.04	122.25
13	¯80.59	105.00	121.11
14	¯79.36	105.64	120.34
15	¯78.03	106.38	120.13
16	¯76.72	107.48	120.67
17	¯75.92	107.76	120.92
18	¯75.71	106.76	120.41
19	¯75.38	103.45	119.99
20	¯76.34	101.93	119.98
21	¯77.54	100.65	120.75
22	¯79.22	99.43	121.24
23	¯80.55	98.59	121.80
24	¯81.10	96.04	122.05
25	¯81.66	94.65	122.65
26	¯81.51	93.29	122.83
27	¯81.05	93.31	123.22
28	¯80.88	97.98	123.36
29	¯80.94	102.17	122.90
30	¯80.90	104.59	122.45
31	¯81.21	104.68	121.02
32	¯79.97	105.37	120.37
33	¯78.10	106.25	120.17
34	¯76.87	107.09	120.33
35	¯76.34	108.28	120.71
36	¯75.86	107.21	120.48
37	¯75.45	104.37	119.58
38	¯76.36	101.73	119.14
39	¯78.04	100.22	119.69
40	¯79.16	99.20	120.74
41	¯80.17	98.13	121.47
42	¯81.24	96.24	121.91
43	¯81.40	94.83	122.60
44	¯81.20	93.97	123.31
45	¯80.86	95.49	124.30
46	¯81.60	98.26	123.87
47	¯81.09	101.76	122.76
48	¯81.19	104.13	121.73
49	¯80.60	105.59	120.59

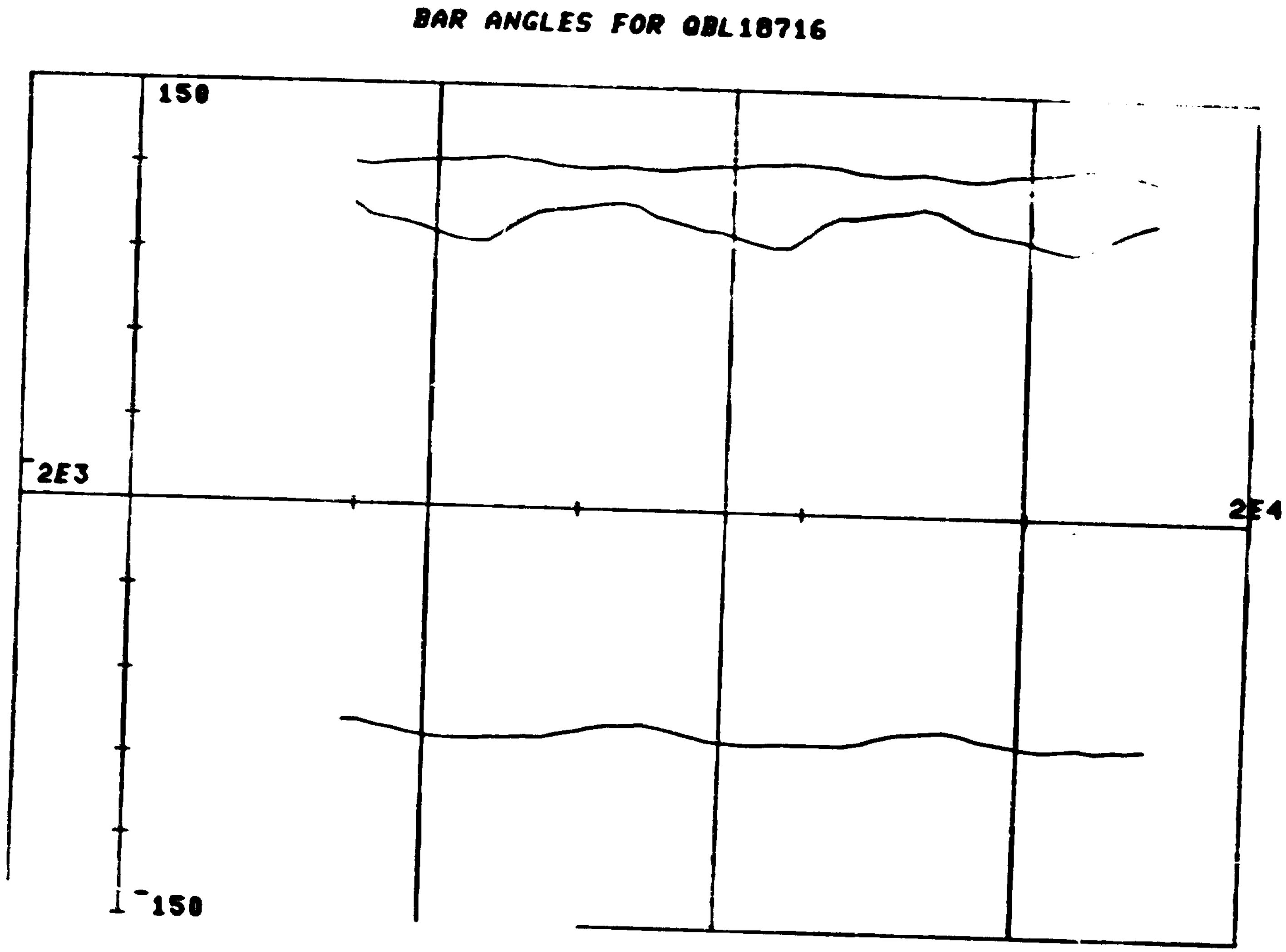


Figure 10. Actual BAR angles (DOY 187:16:56/187:20:56).

DETHL WRT MIDNIGHT 0.99947 0 ¯2.7012

SUGGESTED DETHL 2.0723 0 ¯0.21296

THE AVERAGE BANK ANGLE IS ¯79.186
THE AMPLITUDE IS 2.0002
THE PHASE IS 159.69

DESIRED UPDATE	188: 5:27: 0.00
MIDNIGHT TIME	187:17:21: 0.00
1 REV DELTA TIME	1:28:44.00
FIRST QBL TIME	187:16:56:30.00
LAST QBL TIME	187:20:56:30.00

0	¯80.2
30	¯81.4
60	¯82.0
90	¯81.9
120	¯81.0
150	¯79.7
180	¯78.2
210	¯77.0
240	¯76.3
270	¯76.5
300	¯77.3
330	¯78.7
360	¯80.2

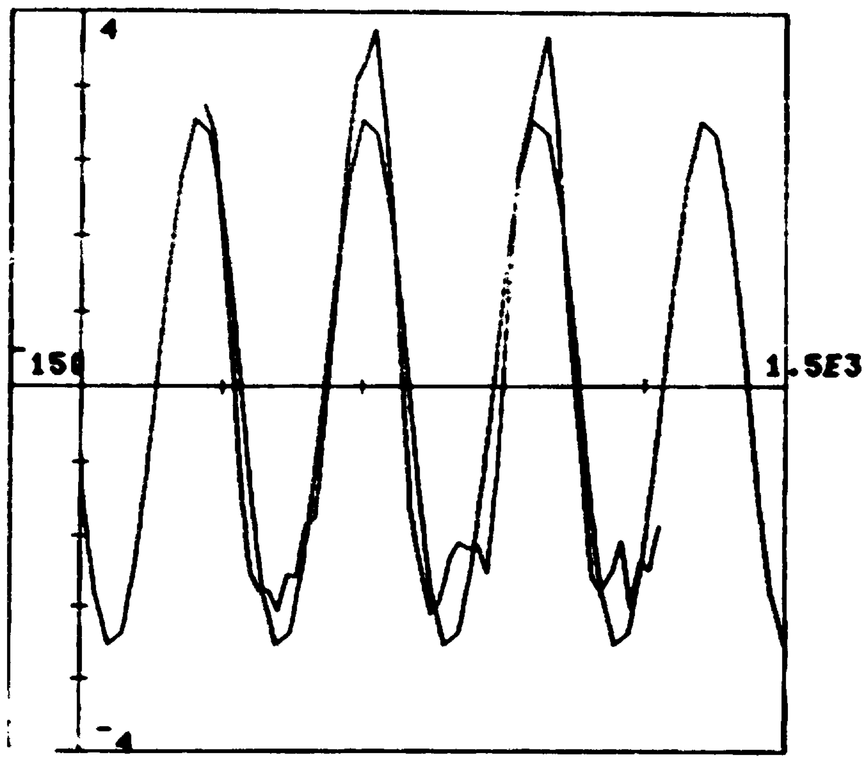


Figure 11. Bank angle evaluation for QBL18716.

THE AVERAGE ANGLE OF ATTACK IS 101.51
THE AMPLITUDE IS 6.41
THE PHASE IS 120.01

DESIRED UPDATE	188: 5:27: 0.00
MIDNIGHT TIME	187:17:21: 0.00
1 REV DELTA TIME	1:28:44.00
FIRST QBL TIME	187:16:56:30.00
LAST QBL TIME	187:20:56:30.00

0	96.0
30	95.1
60	96.0
90	98.3
120	101.5
150	104.7
180	107.1
210	107.9
240	107.1
270	104.7
300	101.5
330	98.3
360	96.0

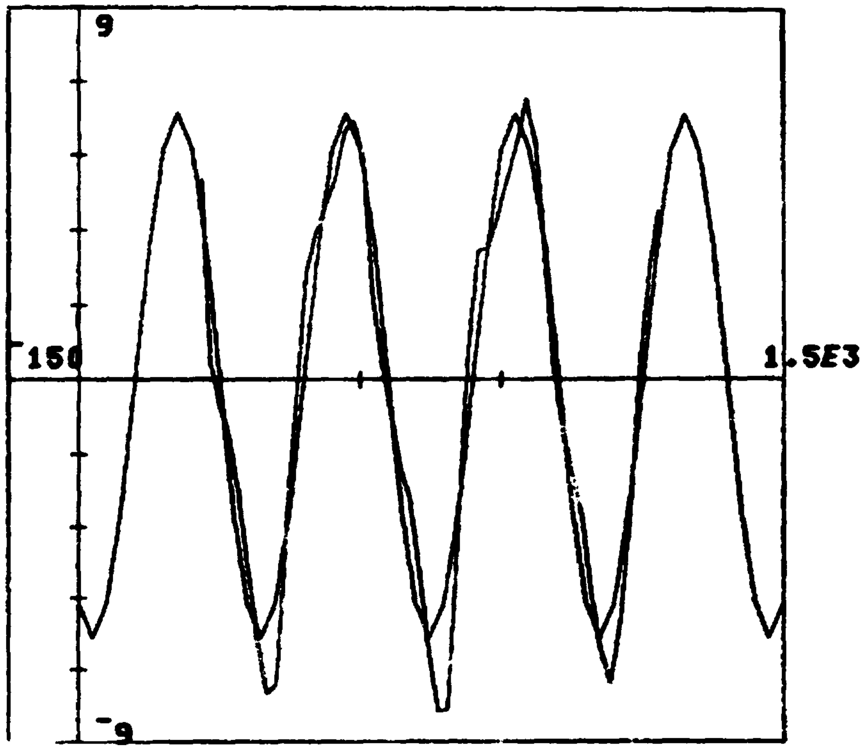


Figure 12. Angle-of-attack evaluation for QBL18716.

```
THE AVERAGE ROLL ANGLE IS 121.47
         THE AMPLITUDE IS 1.6477
             THE PHASE IS 335.18

DESIRED UPDATE       188: 5:27: 0.00
MIDNIGHT TIME        187:17:21: 0.00
1 REV DELTA TIME          1:28:44.00
FIRST QBL TIME       187:16:56:30.00
LAST QBL TIME        187:20:56:30.00
```

0	122.2
30	122.8
60	123.1
90	123.0
120	122.4
150	121.6
180	120.8
210	120.1
240	119.8
270	120.0
300	120.5
330	121.3
360	122.2

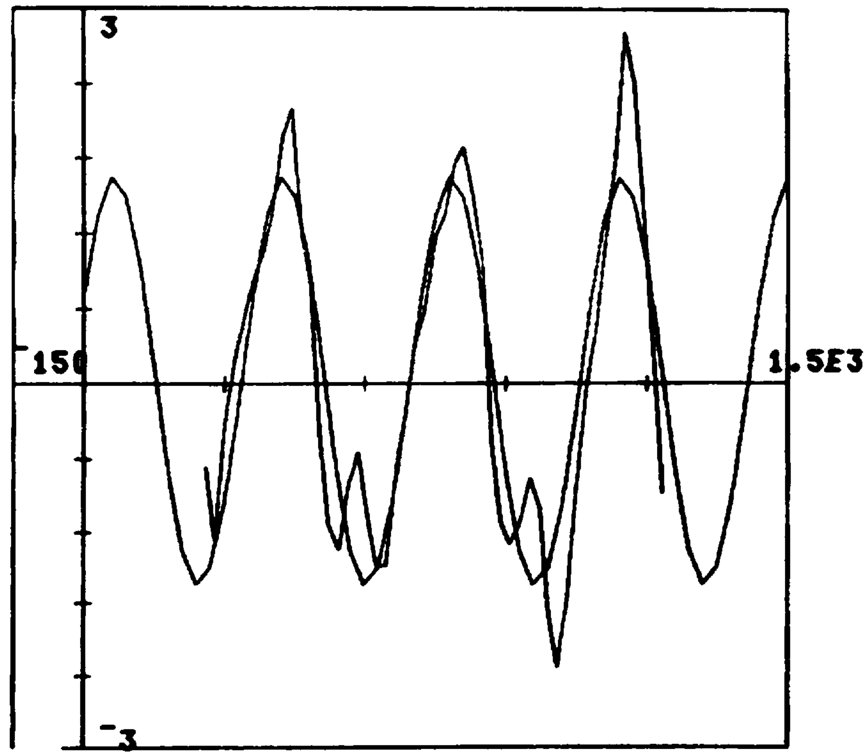


Figure 13. Roll angle evaluation for QBL18716.

STATION ACQUISITION PERIODS FOR SKYLAB

TIME OF NORAD ELEMENT SET 12:09:17 GMT
7- 5-79

DATE	7- 7-79 (FIRST AOS)
ORBIT NUMBER	2
DAY NUMBER	188
BETA ANGLE	-25.4709 DEG

1 REV DELTA TIME	01:28:43.48
ORBITAL MIDNIGHT	05:15:45.59 GMT
ORBITAL SUNRISE	05:33:52.81
NIGHT TO DAY TERM	05:37:54.85
ORBITAL NOON	06:00:06.50
DAY TO NIGHT TERM	06:22:18.33
ORBITAL SUNSET	06:26:20.79
SI DUMP MIDNIGHT	06:41:33.61
SUN SENSOR UPDATE	05:57:11.87

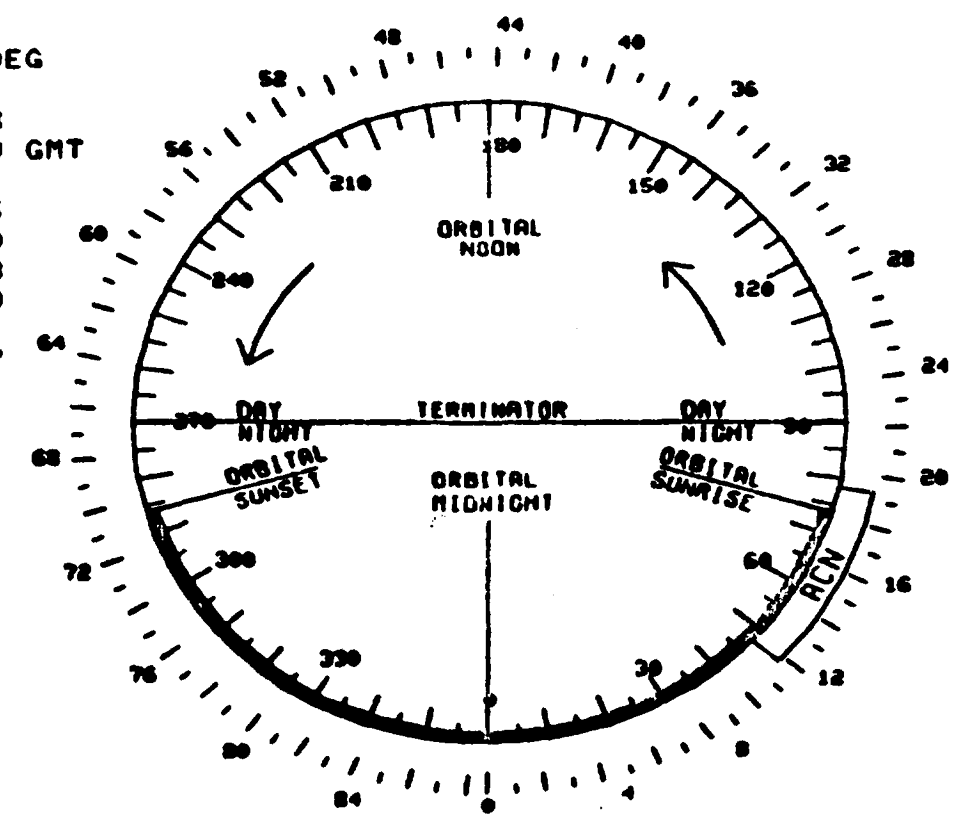
STA	AOS	LOS	MAX ELV
ACN	05:27:28	05:35:00	73.6

TS2 05:35:01 TS6 06:19:22

EOVV SAMPLE TIMES

1 - 05:37:54
2 - 06:00:06
3 - 06:22:18
4 - 05:15:45

NOTE - ER SAMPLE VALID ONLY BETWEEN SAMPLE POINTS 1 & 2.



PROVIDED BY MSFC
SYS ANAL & INTEG LAB
MISSION ANAL DIV, EL-85

Figure 14. Circle chart.

```
07-JUL-79 05:31:40  AM/ATM BLOCKS        158/   1708  AM/ATM/ATMDC */ /    ACN
                    TEA CONTROL PARAMETERS            RSDP GMT 188:  5:31:38.33
                X              Y              Z            TMID-387:23:27:56.75
DELTA E     0.034180       0.054687     -0.025879          DMID         1:28:48.50
DELTA EP    0.023437       0.101562      0.031250          TGMT-387:23:12: 5.00
                                                             DELTA T   TEA   -288
DELTA PH    0.024170      -0.046753     -0.051025
DELTA TH   -0.004761       0.042969     -0.025879          BUS  11 29
                                                     TEMP       12 29  -COMMAND-
ETSF       -1.156250       0.023437     -0.234375      1    17  21 29  CUR 07777
                                                       5    18  22 29  WD1   237
CMG MOM        -27             12           -10        9    28  31 29  WD2     1
                                                      13    28  32 29  WD3    14
          -----CMG 2---    -----CMG 3---              17    17  41 29  WD4  2001
IGA IGR     5   0.6357       4   -0.3828              21    25  42  0  CNTR 0005
OGA OCR    52  -0.7793      48   -0.3486       PRESS  1   153   51 29  SWDA 0096
WH SPD      8960                                      2   181   52 29  SWDL 01C7
          DMIB      0  MIB  3296  JET 123456  TEMP  1    17   61 29  INTRUPT
                       FO    304                      2    27   62 29  -ERRORS--
                                                                       DATA ERR
QBL    0.2416382  -0.1369629  -0.7496948   0.5996094                   MODE ERR
QVA    0.0126342  -0.0229492  -0.0256957  -0.9988989                   UND CMD
QVI    0.6567366  -0.6428816  -0.0582274   0.3894033                   DC REJ
```

Figure 15. Format 8.

The circle chart has midnight at the 6 o'clock position; the outer numbers are minutes from midnight; the inner ones are orbital degrees from midnight. Orbital night time is indicated by the heavy part of the circle. The boxes on the outside of the circle indicate the ground station coverage; the ground stations are Ascension (ACN), Bermuda (BDA), Goldstone (GDS), Madrid (MAD), and Santiago (AGO).

Format 8 showed the present and the past $\underline{\Delta e}$, the present attitude reference change $\underline{\Delta \phi}$, and any impending strapdown update $\underline{\Delta \theta}$. Also shown were the filtered total system momentum (in units of $\overline{H}$), the CMG momentum (in percent of 3H), and the inner and outer gimbal angles (to the nearest degree) plus their gimbal rates.

Of further interest are the quaternions Q_{AL} (QBL), Q_{VA}, and Q_{VI}. The first allowed evaluation of the current BAR angles, the second showed how well the vehicle was following the attitude reference, and the third indicated the attitude with respect to the SI system (useful for determining the closest approach of the Sun line to the center of the ACQ SS which then was used for strapdown update information).

The history of the TEA control parameters as well as the solar elevation angle with respect to the orbital plane are shown in Figures 16 through 20. The biasing history of the rate gyros is shown in Figure 18. Only one rate gyro was still operating in the Y axis (Y3), two each were averaged for the other two axes (X1 and X3; Z1 and Z2). As mentioned before (cf. middle of 3rd paragraph of this section), the initially introduced beneficial bias had to be taken out and replaced with a bias of the opposite sign. The fact that in two axes two rate gyro outputs were averaged allowed us to work effectively with half least significant bits (LSBs) by biasing only one of the averaged gyros with a full LSB. A trimming of the rate gyros to within about 0.25 degree/orbit drift was therefore possible.

The relatively many changes in ETLN1 (Fig. 19) reflect our effort in controlling the roll angle to acceptable values. ETLN2 did not influence the TEA since it was parallel to the orbital velocity, but it affected the CMG gimbal angles and was therefore changed once (in two steps, to minimize the transients) to improve the actual gimbal angle traces.

Simulation predicted that TEA control in T121P was possible down to about 140 km (75 n.mi.) and in T275 down to 130 km (70 n.mi.). At these altitudes the GG and precessional torques were no longer large enough to achieve zero external torques in all axes simultaneously. Very fortunately, Skylab was about to reenter on the most desirable orbit, one with the least population density beneath it. This orbit, however, did go over the United States, the southern part of Canada and over Australia, but most of it was over water. Therefore, only a slight

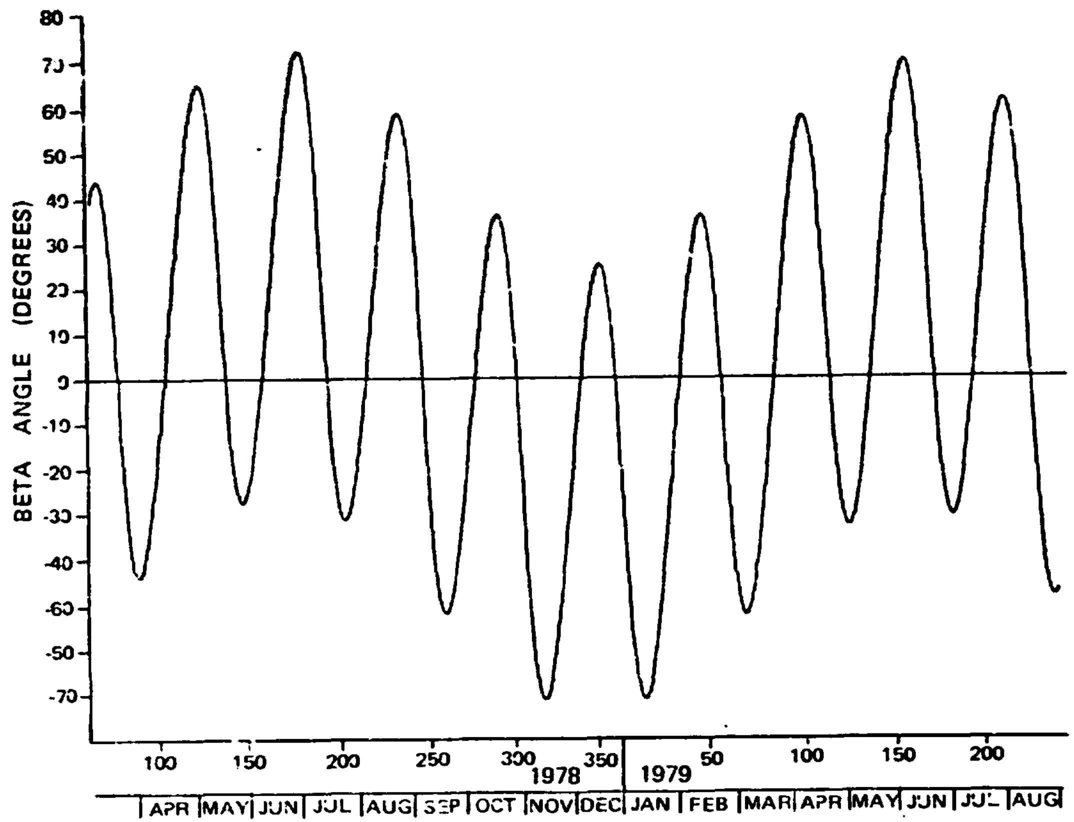


Figure 16. Sun elevation angle beta (April 1978/August 1979).

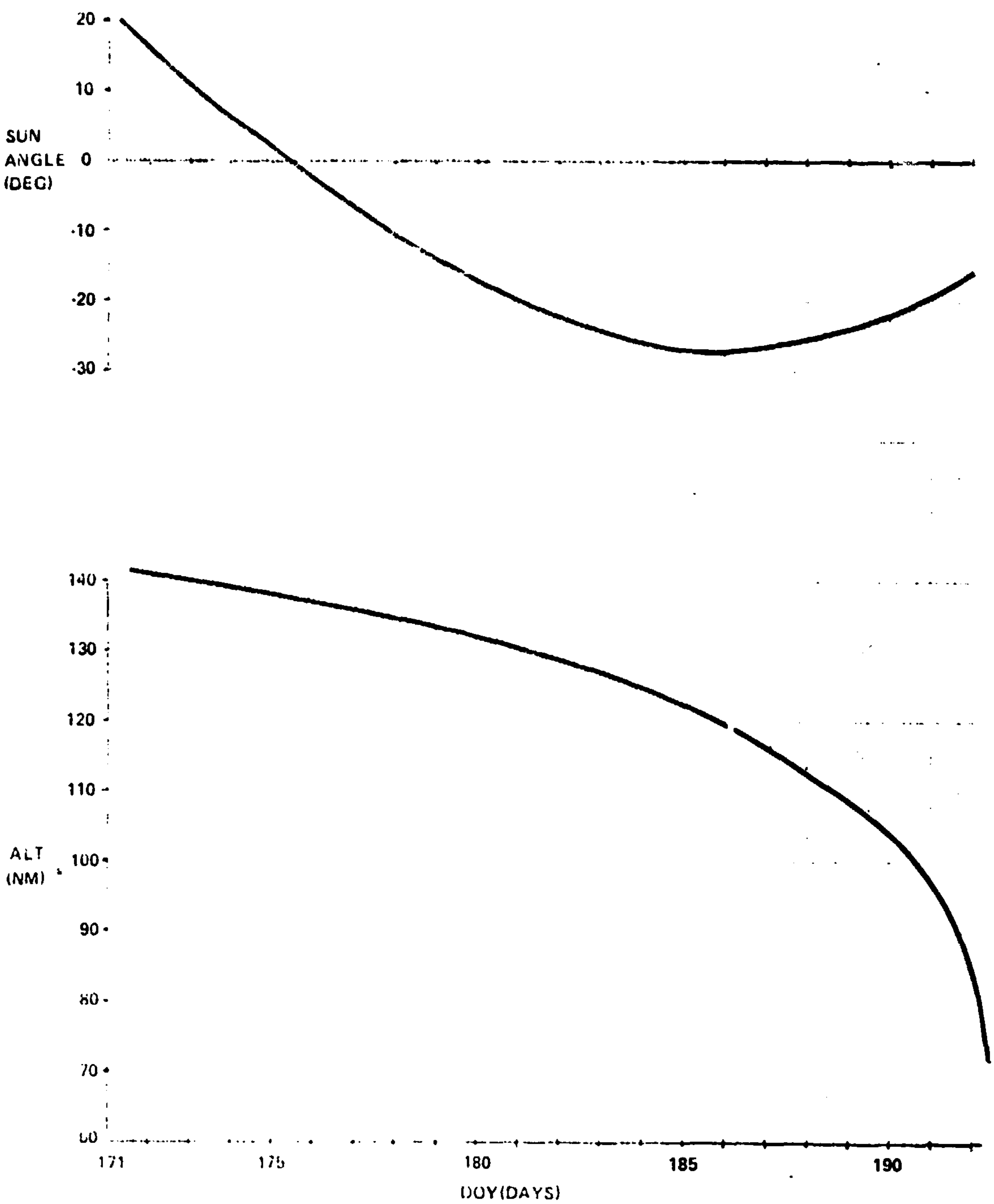


Figure 17. Skylab altitude and Sun elevation angle for DOY 171/192, 1979.

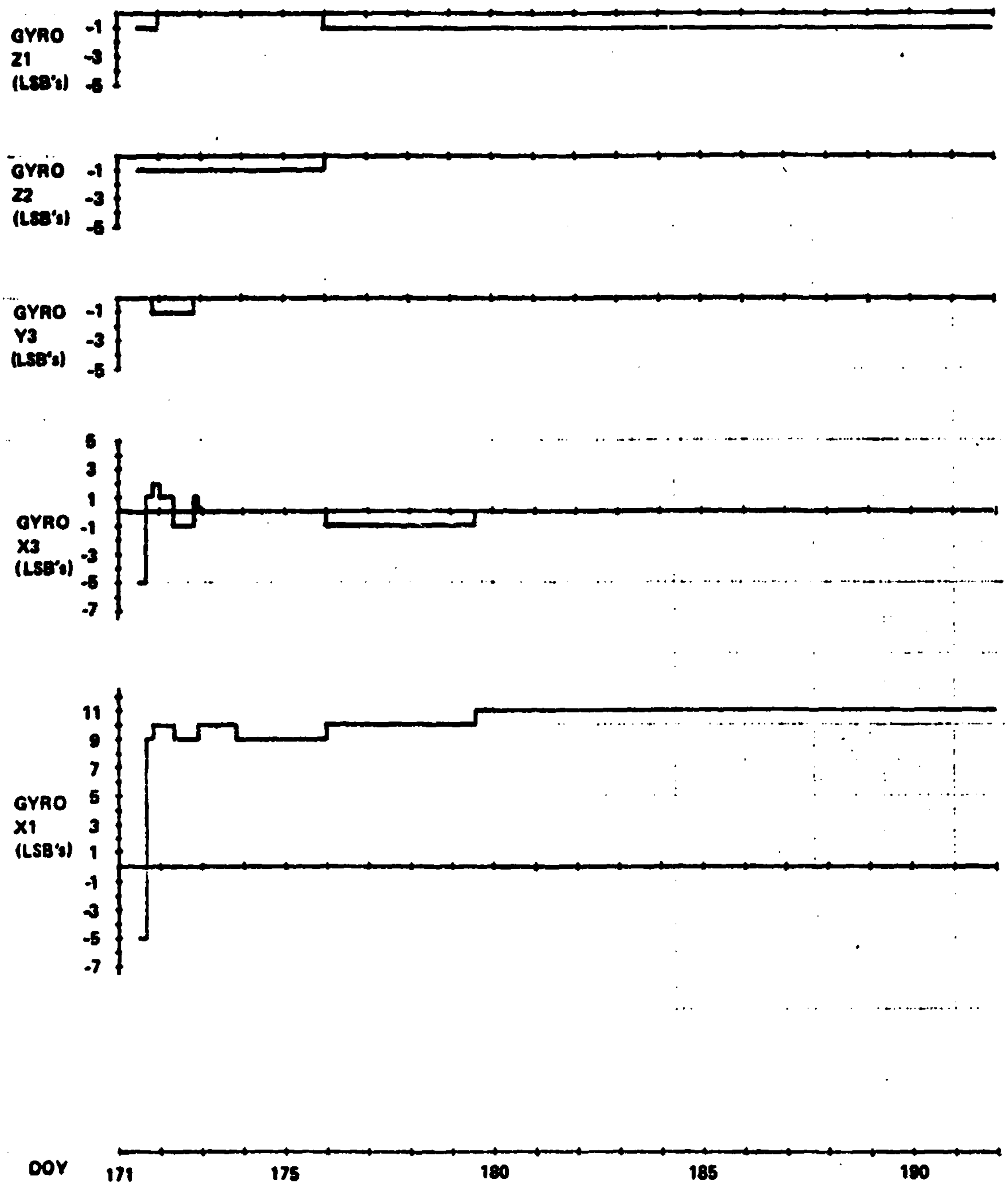


Figure 18. History of rate gyro biasing.

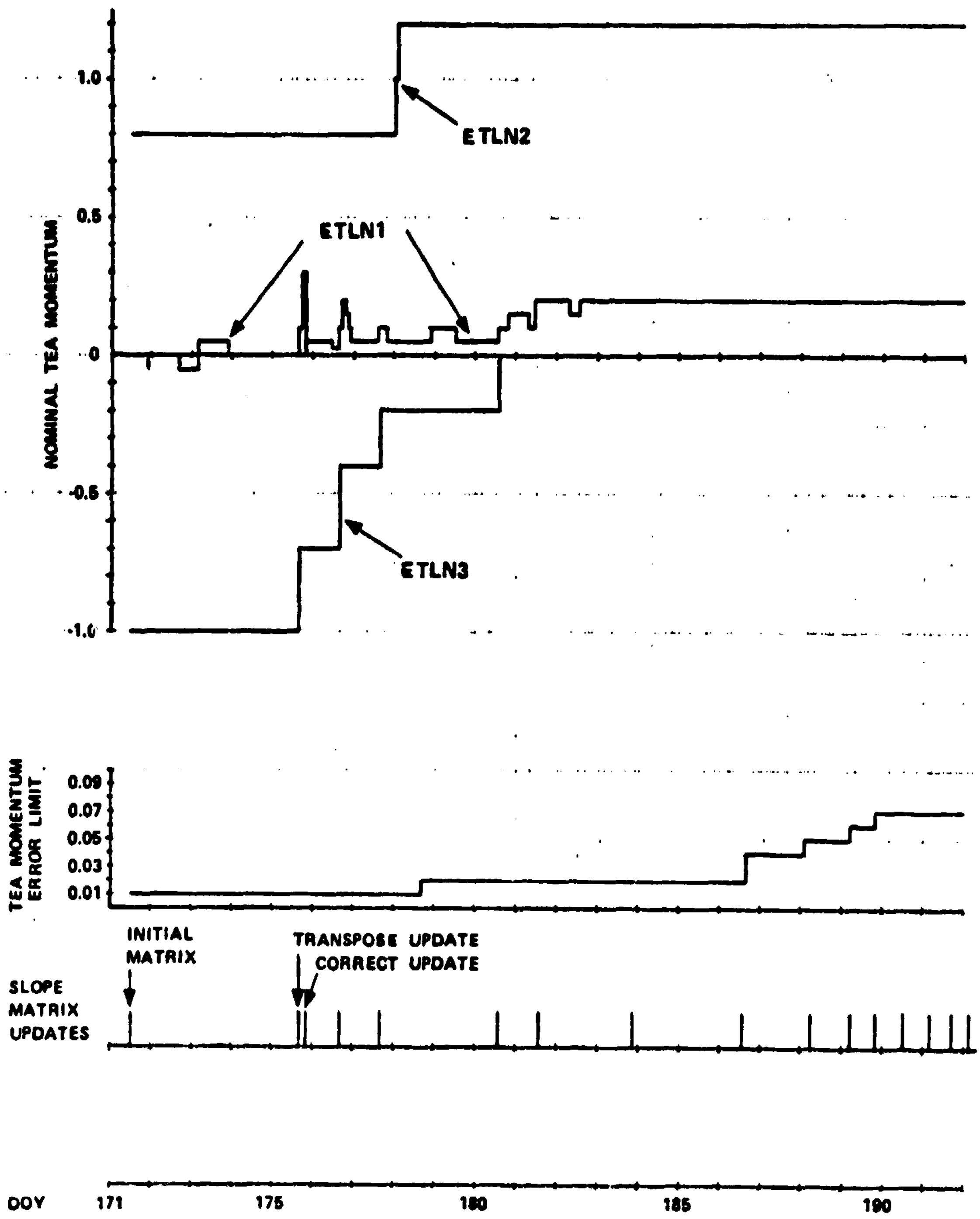


Figure 19. History of slope matrix updates, momentum error limit changes, and ETLN1,2,3 changes.

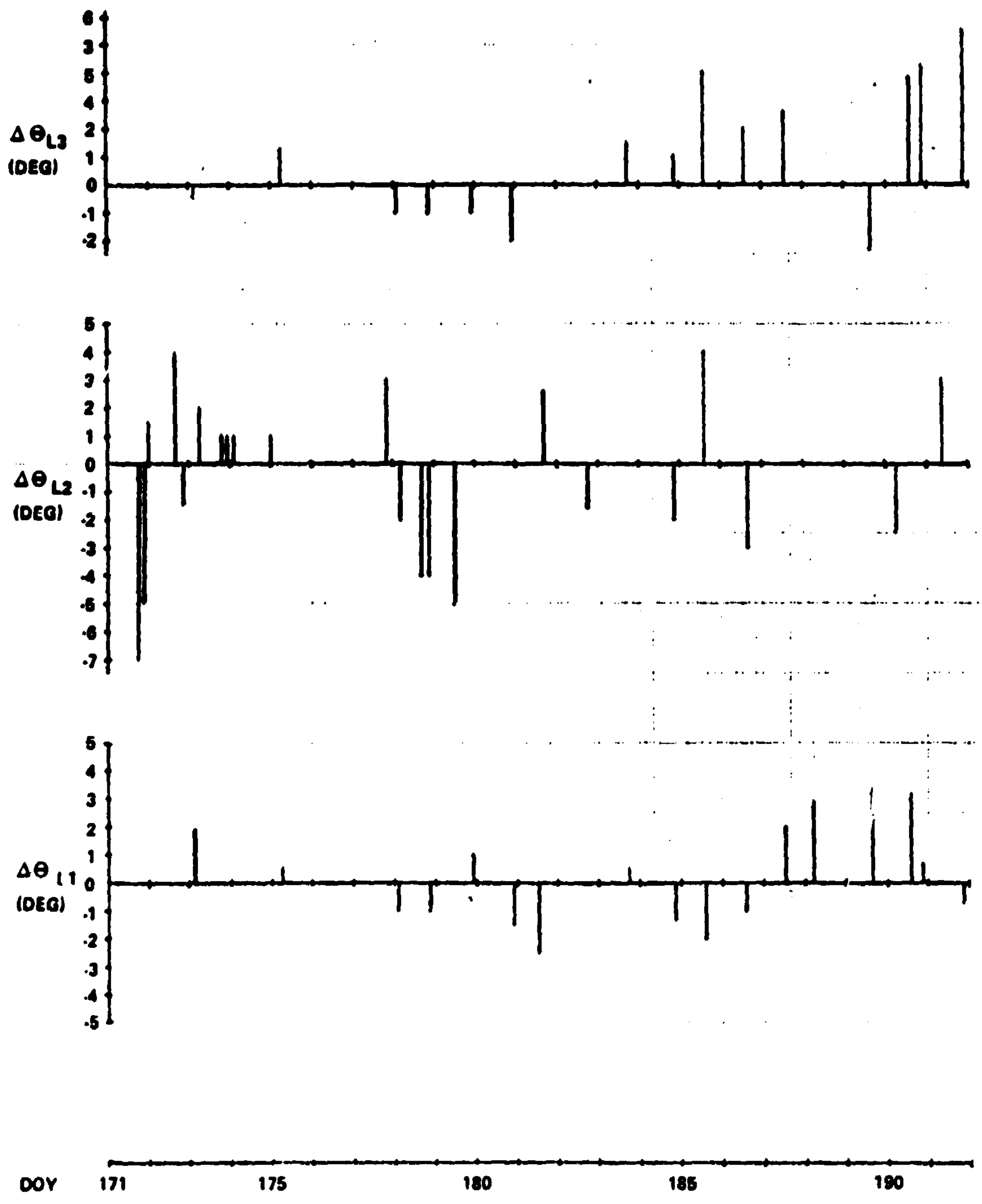


Figure 20. History of strapdown updates.

adjustment of the nominal impact point (which was in the Atlantic Ocean) was indicated, and there was no need to maneuver to the low-drag T275 attitude (this attitude was on an unstable equilibrium in all axes, and the equilibria were much more precarious than the ones in T121P). The slight impact point adjustment could be achieved by going to a random tumble (with somewhat less average drag than T121P) at an altitude of 150 km (81 n.mi.). This was done by commanding a delayed maneuver from the T121P attitude to the solar inertial attitude. The latter, due to the high aerodynamic torques, could not be held, and a random tumbling resulted. On the average this random tumbling had the desired drag, which then lengthened the lifetime sufficiently to place the nominal impact at the desired location. The vehicle broke apart later and, therefore, lower than expected. As a consequence the drag was reduced, shifting the reentry point farther downrange.

CONCLUSIONS

The TEA control was most successful in performing the mission that was originally laid out for it. The experience gained from the TEA control effort in particular and the Skylab Reactivation in general will be valuable in all future space activities. We found out that we could, in fairly short order, come up with new control methods, new procedures, and new tools to perform tasks and accomplish goals that would have been considered impossible prior to the original mission. However, this fast reaction capability was due to the fact that we were able to use APL on a Sigma V computer with interactive terminals. We can flatly state that without APL the development of the new Skylab control methods (EOVV and TEA control) would not have been possible in the extremely short time available (EOVV control [1] was conceived, developed, simulated, implemented, and flown in less than three months). In many respects TEA control was more successful and went more smoothly than the previous phase with EOVV. This may have been due in part to the greater efforts in preparation and time available for TEA. It seems also likely that the experience gained from EOVV carried over to TEA and sustained our confidence. The demonstrated use of aerodynamic torques for spacecraft attitude control seems destined to be of value for design of future, low altitude spacecraft or tethered vehicles.

APPENDIX A

AERODYNAMIC TORQUE MODEL

A set of six coefficients were developed to express forces and torques in body coordinates as a function of the velocity of the vehicle relative to the air. If this velocity is $\underline{V}$, then the forces and torques relative to the origin of coordinates are

$$\underline{F} = 0.5\ \delta\ \underline{V}^2\ A\ \underline{C}_F \tag{A1a}$$

$$\underline{T} = 0.5\ \delta\ \underline{V}^2\ A\ D\ \underline{C}_T \tag{A1b}$$

where δ [kg/m^3] is the density and

$A = 79.46\ m^2$ (855.3 ft^2) reference area

$D = 10.06$ m (396 in.) reference diameter

$\underline{C}_F = [-CA, CY, -CN]^T$ force coefficients

$\underline{C}_T = [CL, CM, CEN]^T$ moment (torque coefficients).

The force and moment coefficients had been calculated by Northrop [6] with a computer program generated by Lockheed [7] and are given in Table A1. These coefficients depend only on the direction of $\underline{V}$ (it is to be noted that the density is assumed to be small enough that free molecular flow assumptions apply). This direction is customarily expressed as aerodynamic roll angle and angle of attack, ρ and α respectively. These are defined in the Skylab vehicle as (cf. Eqs. 3 and 4):

$$\alpha = \arccos (V_x / |\underline{V}|) \tag{A2}$$

$$\rho = \arctan (V_y / V_z) \quad \text{(four quadrant)} \quad . \tag{A3}$$

TABLE A1. FORCE AND MOMENT COEFFICIENTS

RHO = 0.

RHO = 360. ALPHA	CA	CY	CN	CL	CM	CEN
0	2.401	-.002	.002	.000	-.332	-.298
20.	3.166	.000	1.314	.169	-1.578	-.421
40.	4.124	.000	3.770	.650	-3.628	-.719
60.	3.366	.000	6.235	.958	-5.041	-.522
80	1.433	.000	8.578	1.458	-6.132	-.244
90	.001	.000	9.330	1.702	-6.423	-.001
100	-1.529	.000	9.153	1.745	-5.767	.291
120.	-4.142	.000	7.605	1.430	-4.302	.776
140.	-5.136	.000	4.644	.860	-2.161	.943
160.	-4.106	.000	1.703	.310	-.487	.731
180.	-2.590	-.001	.000	.000	.304	.436

RHO = 20. ALPHA	CA	CY	CN	CL	CM	CEN
0.	2.401	-.002	.002	.000	-.332	-.298
20	3.129	-.408	1.225	.249	-1.492	-.143
40	3.919	-1.143	3.372	.865	-3.294	.115
60.	3.251	-1.947	5.657	1.315	-4.677	.879
80.	1.347	-2.640	7.626	2.002	-5.639	1.625
90.	-.004	-2.857	8.237	2.242	-5.807	2.041
100.	-1.445	-2.821	8.132	2.275	-5.251	2.253
120.	-3.905	-2.335	6.757	1.836	-3.842	2.418
140	-4.899	-1.426	4.181	1.095	-1.885	1.944
160.	-3.956	-.505	1.552	.381	-.419	1.112
180.	-2.590	-.001	.000	.000	.304	.436

RHO = 40. ALPHA	CA	CY	CN	CL	CM	CEN
0.	2.401	-.002	.002	.000	-.332	-.298
20	2.964	-.724	.960	.270	-1.249	.121
40	3.330	-1.840	2.366	.750	-2.490	.910
60.	2.989	-3.375	4.279	1.593	-3.778	1.971
80.	1.220	-4.490	5.647	2.057	-4.170	2.925
90.	-.001	-4.670	5.874	2.162	-4.045	3.273
100	-1.226	-4.495	5.659	2.095	-3.564	3.345
120	-3.332	-3.752	4.746	1.711	-2.578	3.301
140.	-4.366	-2.388	3.073	1.028	-1.326	2.537
160.	-3.491	-.840	1.143	.357	-.199	1.261
180	-2.590	-.001	.000	.000	.304	.436

RHO = 60. ALPHA	CA	CY	CN	CL	CM	CEN
0.	2.401	-.002	.002	.000	-.332	-.298
20.	2.787	-.929	.609	.199	-.937	.527
40.	3.036	-2.280	1.419	.519	-1.632	1.583
60	2.560	-3.931	2.415	.991	-2.263	2.744
80.	.997	-4.959	3.026	1.296	-2.280	3.434
90.	.003	-5.103	3.115	1.350	-2.124	3.606
100	-.993	-4.973	3.040	1.319	-1.858	3.599
120	-2.725	-4.169	2.578	1.096	-1.353	3.459
140	-3.591	-2.663	1.695	.688	-.693	2.638
160	-3.201	-1.043	.716	.265	-.089	1.356
180.	-2.590	-.001	.000	.000	.304	.436

TABLE A1. (Continued)

RHO = 80. ALPHA	CA	CY	CN	CL	CM	CEN
0.	2.401	-.002	.002	.000	-.332	-.298
20.	2.600	-.994	.227	.126	-.497	.728
40.	2.669	-2.295	.468	.256	-.695	2.049
60.	2.110	-3.711	.733	.475	-.874	3.243
80.	.812	-4.615	.903	.640	-.829	3.899
90.	.004	-4.715	.922	.664	-.730	3.984
100.	-.801	-4.600	.900	.639	-.593	3.862
120.	-2.111	-3.703	.734	.474	-.329	3.255
140.	-2.760	-2.348	.491	.265	-.155	2.373
160.	-2.700	-1.011	.248	.141	.091	1.225
180.	-2.590	-.001	.000	.000	.304	.436

RHO = 90. ALPHA	CA	CY	CN	CL	CM	CEN
0.	2.401	-.002	.002	.000	-.332	-.298
20.	2.707	-1.048	-.001	.117	-.317	.706
40.	2.687	-2.346	-.001	.227	-.269	2.062
60.	1.988	-3.551	-.001	.332	-.192	3.221
80.	.741	-4.280	.000	.386	-.068	3.865
90.	.004	-4.348	.000	.391	.000	3.948
100.	-.730	-4.269	.000	.385	.068	3.844
120.	-1.988	-3.545	.000	.333	.193	3.257
140.	-2.742	-2.372	.000	.223	.265	2.378
160.	-2.813	-1.066	-.001	.112	.305	1.350
180.	-2.590	-.001	.000	.000	.304	.436

RHO = 100. ALPHA	CA	CY	CN	CL	CM	CEN
0.	2.401	-.002	.002	.000	-.332	-.298
20.	2.517	-.966	-.224	.110	-.150	.788
40.	2.710	-2.330	-.472	.248	.086	2.061
60.	2.154	-3.778	-.751	.433	.355	3.280
80.	.832	-4.721	-.933	.567	.652	4.032
90.	.004	-4.857	-.961	.575	.785	4.181
100.	-.827	-4.734	-.938	.552	.872	4.104
120.	-2.186	-3.825	-.768	.395	.904	3.503
140.	-2.810	-2.392	-.500	.199	.735	2.422
160.	-2.630	-.988	-.246	.084	.518	1.125
180.	-2.590	-.001	.000	.000	.304	.436

RHO = 120. ALPHA	CA	CY	CN	CL	CM	CEN
0.	2.401	-.002	.002	.000	-.332	-.298
20.	2.952	-.985	-.663	.133	-.030	.663
40.	3.522	-2.637	-1.655	.297	.606	1.926
60.	2.854	-4.367	-2.691	.654	1.525	3.279
80.	1.122	-5.576	-3.422	.722	2.423	4.279
90.	.004	-5.826	-3.578	.668	2.755	4.572
100.	-1.128	-5.637	-3.465	.626	2.875	4.550
120.	-2.943	-4.494	-2.778	.445	2.722	3.977
140.	-3.513	-2.613	-1.643	.204	1.923	2.634
160.	-3.044	-.998	-.678	.063	1.035	1.218
180.	-2.590	-.001	.000	.000	.304	.436

TABLE A1. (Continued)

RHO = 140. ALPHA	CA	CY	CN	CL	CM	CEN
0.	2.401	-.002	.002	.000	-.332	-.298
20.	3.340	-.821	-1.096	.095	.075	.413
40.	4.202	-2.322	-2.953	.303	1.117	1.379
60.	3.423	-3.876	-4.853	.352	2.804	2.552
80.	1.315	-4.840	-6.029	.363	4.102	3.427
90.	.004	-5.072	-6.320	.264	4.593	3.735
100.	-1.317	-4.871	-6.069	.223	4.740	3.798
120.	-3.298	-3.731	-4.661	.129	4.323	3.388
140.	-3.981	-2.194	-2.777	.048	3.011	2.316
160.	-3.269	-.783	-1.041	-.023	1.376	1.096
180.	-2.590	-.001	.000	.000	.304	.436

RHO = 160. ALPHA	CA	CY	CN	CL	CM	CEN
0.	2.401	-.002	.002	.000	-.332	-.298
20.	3.630	-.473	-1.434	-.021	.169	.038
40	4.713	-1.384	-4.022	-.119	1.598	.449
60	3.857	-2.320	-6.658	-.365	3.719	1.121
80.	1.471	-2.876	-8.215	-.582	5.471	1.839
90.	.005	-2.923	-8.339	-.683	5.990	2.121
100.	-1.399	-2.754	-7.853	-.549	5.938	2.216
120.	-3.566	-2.148	-6.146	-.453	5.358	2.146
140.	-4.308	-1.264	-3.668	-.324	3.712	1.662
160.	-3.313	-.429	-1.298	-.122	1.648	.846
180.	-2.590	-.001	.000	.000	.304	.436

RHO = 180. ALPHA	CA	CY	CN	CL	CM	CEN
0.	2.401	-.002	.002	.000	-.332	-.298
20.	3.860	-.001	-1.618	-.231	.335	-.536
40.	4.873	-.001	-4.416	-.632	1.784	-.686
60.	4.017	-.001	-7.368	-1.252	4.023	-.674
80.	1.518	-.001	-8.985	-1.573	5.782	-.264
90.	.005	-.001	-9.236	-1.671	6.378	.000
100.	-1.446	-.001	-8.640	-1.488	6.346	.248
120.	-3.515	-.001	-6.438	-1.028	5.334	.556
140.	-4.308	-.001	-3.886	-.706	3.821	.767
160.	-3.344	-.002	-1.390	-.225	1.731	.519
180.	-2.590	-.001	.000	.000	.304	.436

RHO = 200. ALPHA	CA	CY	CN	CL	CM	CEN
0.	2.401	-.002	.002	.000	-.332	-.298
20.	3.554	.460	-1.404	-.237	.116	-.665
40.	4.672	1.368	-3.989	-.885	1.568	-1.590
60	3.840	2.307	-6.629	-1.704	3.678	-2.284
80.	1.436	2.807	-8.015	-2.171	5.210	-2.234
90.	.005	2.875	-8.203	-2.230	5.757	-2.043
100.	-1.359	2.674	-7.625	-1.968	5.740	-1.687
120.	-3.321	1.999	-5.715	-1.352	4.938	-.970
140.	-3.882	1.136	-3.295	-.826	3.383	-.230
160.	-3.246	.418	-1.271	-.280	1.630	.136
180.	-2.590	-.001	.000	.000	.304	.436

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

TABLE A1. (Continued)

RHO = 220. ALPHA	CA	CY	CN	CL	CM	CEN
0.	2.401	-.002	.002	.000	-.332	-.298
20.	3.212	.779	-1.058	-.260	.009	-.824
40.	3.909	2.137	-2.759	-.832	.832	-1.830
60	3.187	3.585	-4.532	-1.591	2.383	-3.015
80.	1.224	4.482	-5.615	-2.097	3.532	-3.302
90.	.004	4.714	-5.902	-2.244	4.025	-3.252
100.	-1.225	4.510	-5.655	-2.148	4.192	-2.918
120	-2.767	3.114	-3.928	-1.397	3.662	-1.969
140.	-3.255	1.778	-2.281	-.771	2.585	-.983
160	-3.029	.733	-.982	-.302	1.318	-.138
180.	-2.590	-.001	.000	.000	.304	.436

RHO = 240. ALPHA	CA	CY	CN	CL	CM	CEN
0	2.401	-.002	.002	.000	-.332	-.298
20.	2.798	.920	-.632	-.222	-.086	-.908
40.	3.063	2.264	-1.456	-.580	.346	-1.769
60.	2.473	3.748	-2.357	-1.014	1.082	-2.788
80.	.976	4.809	-2.993	-1.344	1.828	-3.403
90.	.005	5.065	-3.149	-1.423	2.166	-3.531
100.	-.974	4.832	-3.020	-1.373	2.317	-3.321
120.	-2.419	3.659	-2.312	-1.022	2.292	-2.657
140.	-2.962	2.179	-1.406	-.587	1.727	-1.575
160.	-2.901	.940	-.649	-.246	1.013	-.501
180	-2.590	-.001	.000	.000	.304	.436

RHO = 260. ALPHA	CA	CY	CN	CL	CM	CEN
0.	2.401	-.002	.002	.000	-.332	-.298
20.	2.370	.898	-.212	-.126	-.175	-.836
40.	2.480	2.109	-.440	-.281	.037	-1.836
60.	1.980	3.445	-.702	-.500	.277	-2.874
80.	.786	4.423	-.893	-.667	.594	-3.664
90.	.005	4.626	-.935	-.701	.747	-3.858
100.	-.779	4.431	-.900	-.671	.833	-3.698
120.	-1.999	3.464	-.717	-.507	.884	-3.070
140.	-2.595	2.186	-.468	-.290	.732	-2.037
160.	-2.556	.951	-.239	-.143	.518	-.745
180.	-2.590	-.001	.000	.000	.304	.436

RHO = 270. ALPHA	CA	CY	CN	CL	CM	CEN
0.	2.401	-.002	.002	.000	-.332	-.298
20.	2.625	1.005	.006	-.121	-.339	-1.134
40.	2.628	2.269	.010	-.234	-.292	-2.149
60.	1.988	3.516	.009	-.333	-.203	-3.251
80.	.763	4.360	.004	-.370	-.069	-4.011
90.	.005	4.515	.001	-.362	-.001	-4.164
100.	-.754	4.359	.003	-.367	.061	-3.969
120.	-2.049	3.615	.007	-.317	.177	-3.281
140.	-2.829	2.424	.006	-.212	.246	-2.117
160.	-2.858	1.074	.002	-.110	.297	-.659
180.	-2.590	-.001	.000	.000	.304	.436

TABLE A1. (Concluded)

RHO = 280. ALPHA	CA	CY	CN	CL	CM	CEN
0	2.401	-.002	.002	.000	-.332	-.298
20.	2.576	.977	.235	-.106	-.516	-1.050
40.	2.706	2.307	.493	-.226	-.729	-2.227
60	2.164	3.775	.780	-.428	-.935	-3.410
80	.847	4.776	.976	-.553	-.925	-4.168
90.	.005	4.945	1.009	-.556	-.846	-4.286
100	-.839	4.780	.981	-.542	-.708	-4.078
120.	-2.204	3.834	.800	-.378	-.426	-3.291
140.	-2.868	2.421	.526	-.166	-.201	-2.080
160.	-2.760	1.027	.257	-.070	.076	-.690
180	-2.590	-.001	.000	.000	.304	.436

RHO = 300. ALPHA	CA	CY	CN	CL	CM	CEN
0.	2.401	-.002	.002	.000	-.332	-.298
20.	2.842	.943	.624	-.106	-.951	-1.00[illegible]
40.	3.286	2.457	1.548	-.292	-1.774	-2.271
60.	2.828	4.321	2.678	-.536	-2.577	-3.671
80.	1.125	5.568	3.436	-.659	-2.845	-4.481
90.	.004	5.806	3.584	-.640	-2.772	-4.579
100.	-1.130	5.628	3.479	-.606	-2.469	-4.287
120.	-2.987	4.548	2.838	-.420	-1.721	-3.314
140.	-3.769	2.784	1.785	-.202	-.824	-1.846
160.	-3.299	1.070	.739	-.049	-.122	-.519
180.	-2.590	-.001	.000	.000	.304	.436

RHO = 320. ALPHA	CA	CY	CN	CL	CM	CEN
0.	2.401	-.002	.002	.000	-.332	-.298
20.	2.945	.720	.951	-.045	-1.235	-.827
40.	3.699	2.033	2.625	-.161	-2.778	-1.954
60.	3.187	3.589	4.558	-.231	-4.119	-3.120
80.	1.290	4.740	5.971	-.295	-4.627	-3.656
90.	-.001	4.990	6.291	-.186	-4.629	-3.723
100.	-1.323	4.846	6.114	-.167	-4.171	-3.398
120.	-3.535	3.973	5.037	-.073	-2.977	-2.472
140.	-4.533	2.476	3.189	.013	-1.462	-1.191
160.	-3.646	.874	1.189	.041	-.260	-.137
180.	-2.590	-.001	.000	.000	.304	.436

RHO = 340. ALPHA	CA	CY	CN	CL	CM	CEN
0.	2.401	-.002	.002	.000	-.332	-.299
20.	3.099	.405	1.208	.054	-1.468	-.651
40.	4.145	1.209	3.566	.268	-3.533	-1.522
60.	3.407	2.037	5.923	.393	-5.011	-1.972
80.	1.371	2.686	7.762	.493	-5.830	-2.132
90.	-.003	2.933	8.460	.709	-6.051	-2.125
100.	-1.492	2.911	8.393	.755	-5.535	-1.814
120.	-3.971	2.374	6.867	.672	-4.008	-1.034
140.	-4.956	1.442	4.224	.433	-1.933	-.192
160.	-4.007	.510	1.566	.178	-.439	.270
180.	-2.590	-.001	.000	.000	.304	.436

The aero data provided us consisted of six coefficients $\underline{C}_F$ and $\underline{C}_T$ for each of 231 directions defined by $\rho = 0, 20, \ldots, 360, 90, 270$ and $\alpha = 0, 20, \ldots 180$. To get forces and torques for other directions we had to interpolate and initially we used a linear interpolation scheme. This was satisfactory at first but later caused problems in the calculation of TEA control coefficients. Ultimately a Fourier fit of the data was used. This model was set up such that each table value was reproduced and since the in-between values were now produced by a sinusoidal series, the variations were extremely smooth and computation problems were eliminated. The coefficients $C_i\,(\alpha_j, \rho_k)$ can be expressed as a finite series of sines and cosines in a two-step process.

$$C_i(\alpha_j, \rho_k) = \sum_{m=0}^{10} a_{im}(\alpha_j) \cos m\rho_k + \sum_{m=1}^{10} b_{im}(\alpha_j) \sin m\rho_k \quad . \tag{A4}$$

It can be observed that the given set of angles for ρ is equivalent to the following more convenient set: $\rho = 0, \pm 20, \ldots, \pm 180, \pm 90$. We can simplify our task by defining two auxiliary functions G and H such that

$$G(\alpha, \rho) = 0.5\,[C(\alpha, \rho) + C(\alpha, -\rho)] \tag{A5}$$

$$H(\alpha, \rho) = 0.5\,[C(\alpha, \rho) - C(\alpha, -\rho)] \quad . \tag{A6}$$

These functions can be expressed as

$$G_i(\alpha_j, \rho_k) = \sum_{m=0}^{10} a_{im}(\alpha_j) \cos m\rho_k \tag{A7}$$

$$H_i(\alpha_j, \rho_k) = \sum_{m=1}^{10} b_{im}(\alpha_j) \sin m\rho_k \quad . \tag{A8}$$

The functions $G_i(\alpha_j, \rho_k)$ and $H_i(\alpha_j, \rho_k)$ can be computed for each α_j and equations (A7) and (A8) inverted to obtain a_{im} and b_{im}. In turn these coefficients can be expressed as a fourier series in α. Since the given set of α's only range from 0 to 180, we can make the assumption that a_{im} and b_{im} are symmetric in α. This means we can express a_{im} and b_{im} as cosine series:

$$a_{im}(\alpha_j) = \sum_{n=0}^{10} \bar{a}_{im} \cos n\alpha_j \tag{A9}$$

$$b_{im}(\alpha_j) = \sum_{n=1}^{10} \bar{b}_{im} \cos n\alpha_j \quad . \tag{A10}$$

Using equations (A7) through (A10) the coefficients $\bar{a}_{im}$ and $\bar{b}_{im}$ can be determined. Inspection of the above will reveal that the solution procedure only requires inversion of the 11×11 matrix ($\cos m\alpha_j$), where $m = 0,1,\ldots,10$ and $\alpha_j = 0,20,\ldots,180,90$; and the 10×10 matrix ($\sin m\alpha_j$), where m and α_j are as before except the 0 is removed. The final form of the resulting expansion is

$$C_i(\alpha,\rho) = \sum_{n=0}^{10} \cos n\alpha \left(\sum_{m=0}^{10} \bar{a}_{imn} \cos m\rho + \sum_{m=1}^{10} \bar{b}_{imn} \sin m\rho \right) \quad . \tag{A11}$$

The subscript i ranges over the values xyz for torques and xyz for forces. Thus there are six components of C_i and for each component there are 121 $\bar{a}_{imn}$ and 110 $\bar{b}_{imn}$ for a total of 231 per axis which makes 1386 total Fourier coefficients.

The uncertainty in the aero torques are represented by a center of pressure (CP) offset. The nominal set of coefficients are computed for the given coordinate axes and origin. Torques are computed about the origin and the total force is assumed to act through it. The torque about the center of mass is

$$\underline{T}_n = \underline{T}_o - \underline{R}_{cm} \times \underline{F}_o \tag{A12}$$

in the nominal case. If the center of pressure is moved by $\underline{\Delta}$, the torque becomes

$$\underline{T}_{ae} = \underline{T}_n + \underline{\Delta} \times \underline{F}_o \quad . \tag{A13}$$

The vector $\underline{T}_{ae}$ is the torque vector required for the aero model we used. The nominal center of mass location, $\underline{R}_{cm}$, was taken from Reference 8. Details are given in Tables A2 and A3. The units are m(in.).

TABLE A2. NOMINAL CENTER OF MASS LOCATION

	Moment Reference Point, m (in.)	Nominal CM Location, m (in.)	CM, m (in.)
X	90.037 (3544.765)	81.892±0.155 (3224.1±6.1)	-8.145±0.155 (-320.665±6.1)
Y	0	-0.107±0.053 (-4.2±2.1)	-0.107±0.053 (-4.2±2.1)
Z	0	+0.838±0.064 (33.2±2.5)	+0.838±0.064 (33.2±2.5)

TABLE A3. CENTER OF PRESSURE LOCATION

	ΔCP_{est}, m(in.)	ΔCP_{act}, m(in.)	ΔCM_{eff}, m(in.)
x	0±1.524(0±60)	0.06(+2.36)	-8.205(-323.03)
Y	0±0.762(0±30)	-0.07(-2.76)	-0.037(-1.46)
Z	0±0.762(0±30)	-0.11(-4.33)	+0.948(+37.32)

As can be seen in Table A3 the tolerance on the CP location is an order of magnitude larger than the uncertainty in the center of mass location. Both have the same effect on the predicted aerodynamic torques and as a consequence, the CM tolerance can be ignored. Before actual flight data were available, the large CP tolerance was used to assess the sensitivity of the TEA seeking method (see TEA Seeking Method). After flight data were available, the actual estimate given in column 2 of Table A3 was used to give the effective ΔCM given in column three.

Zero contours for the three torque components are shown in Figure 4. Asterisks indicate all TEA's (the feasible ones are also labeled). Individual contour maps for the torque components are given in Figures A1, A2, and A3. The contour numbers signify actual aerodynamic torques in Nm for a density of 3.11E-10 kg/m^3, corresponding roughly to an altitude of 200 km (108 n.mi.) and nominal solar activity.[7] Table A4 shows contour number multiplication factors for other altitudes (contour number times multiplication factor equals torque).

Stereo doubles (see Appendix F for an explanation) of Figures A1, A2, and A3 are given in Figures A4, A5, and A6. Stereo projections of the contours on unit spheres are shown in Figures A7 through A12.

TABLE A4. CONTOUR NUMBER MULTIPLICATION FACTORS

Altitude (km)	Altitude (n.mi.)	Density (kg/m^3)	Multiplication Factor
278	150	5.310E-11	0.17
259	140	7.855E-11	0.25
241	130	1.197E-10	0.38
222	120	1.885E-10	0.61
204	110	3.110E-10	1.00
185	100	5.485E-10	1.80
167	90	1.065E-09	3.40
148	80	2.505E-09	8.10
130	70	8.920E-09	28.70

7. Nominal data for 6/15/79 with sun spot activity number 136.2, F10 = 181.66 and A_p = 17.96.

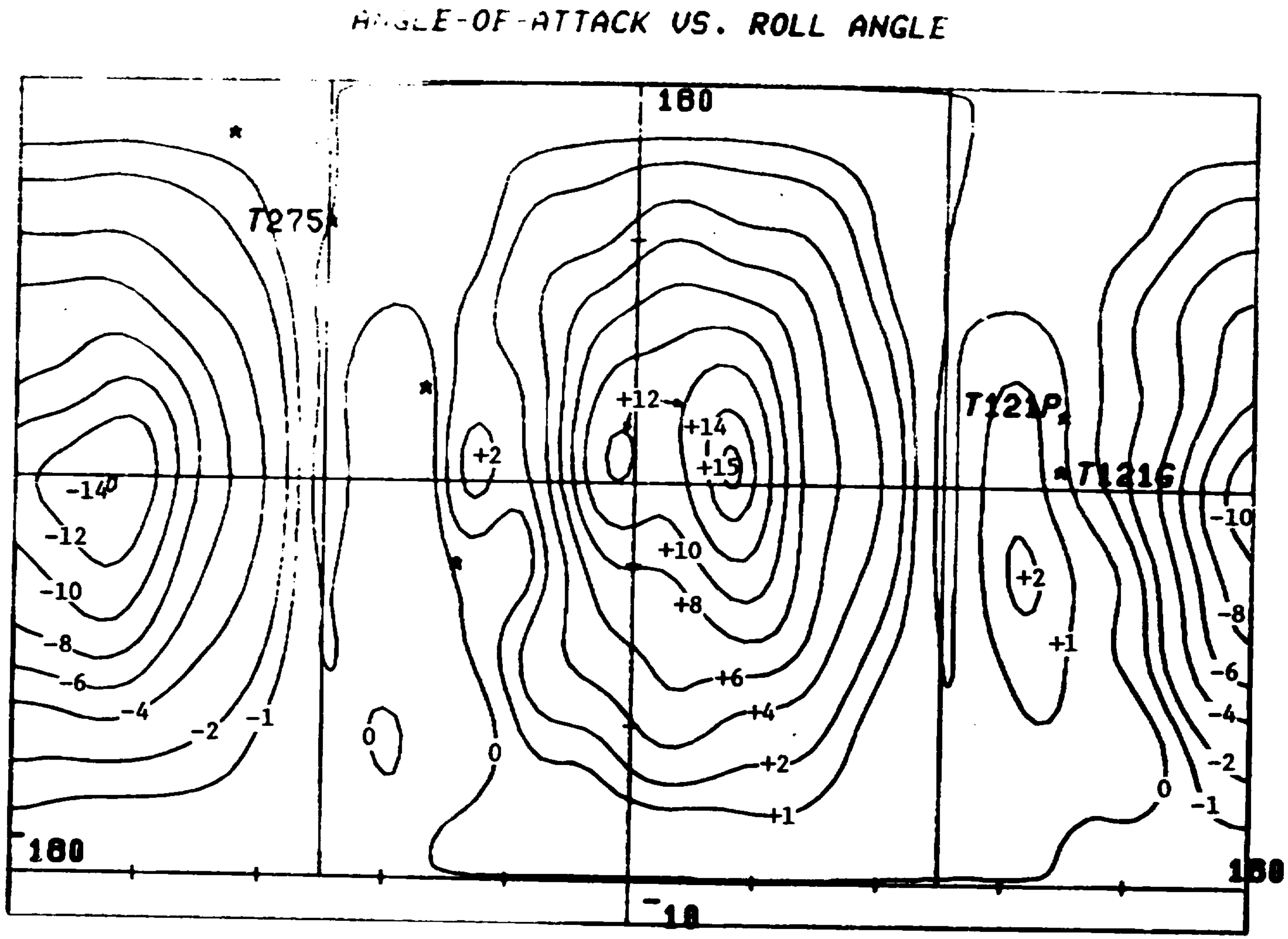


Figure A1. X torque contours.

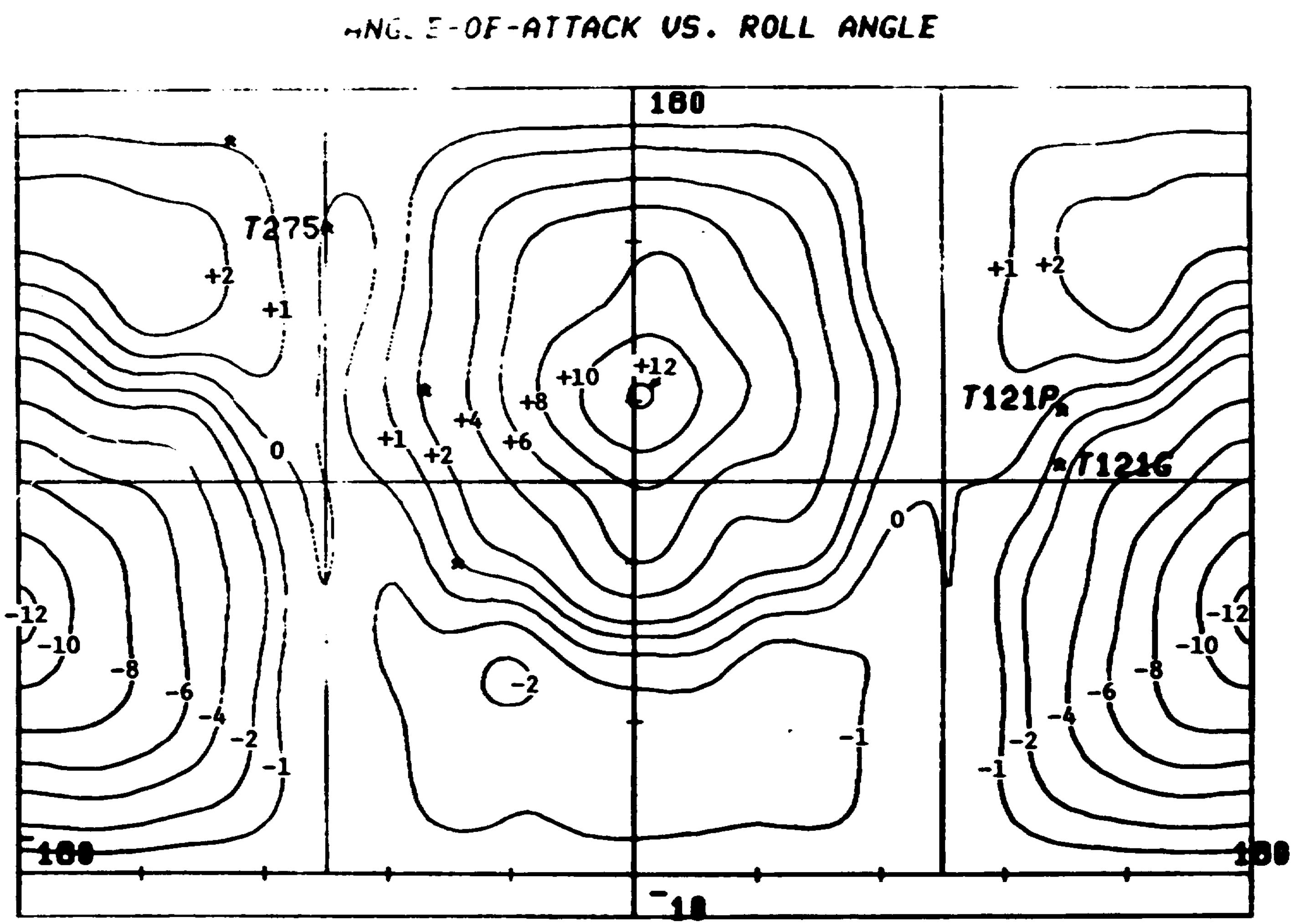


Figure A2. Y torque contours.

ANGLE-OF-ATTACK VS. ROLL ANGLE

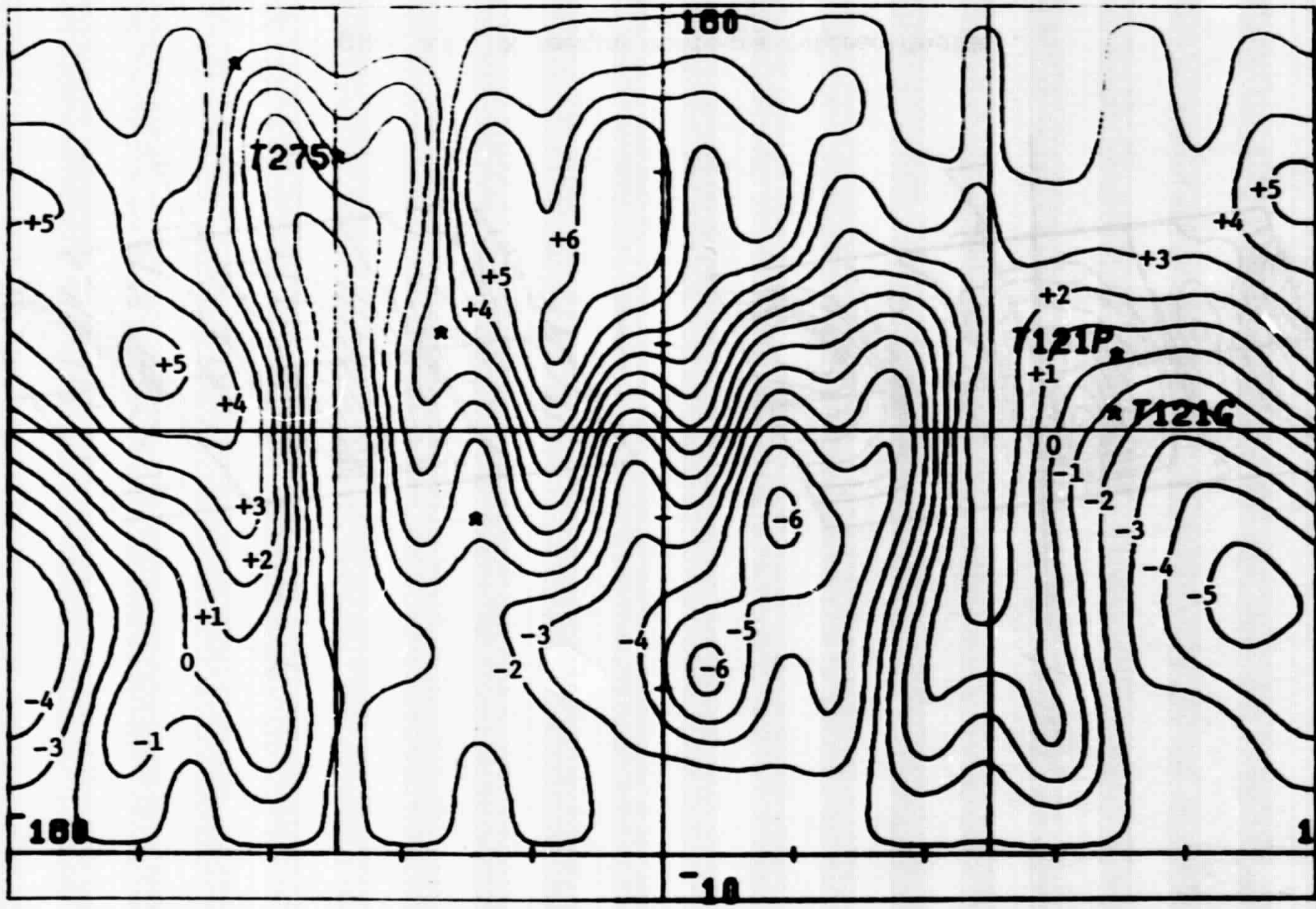


Figure A3. Z torque contours.

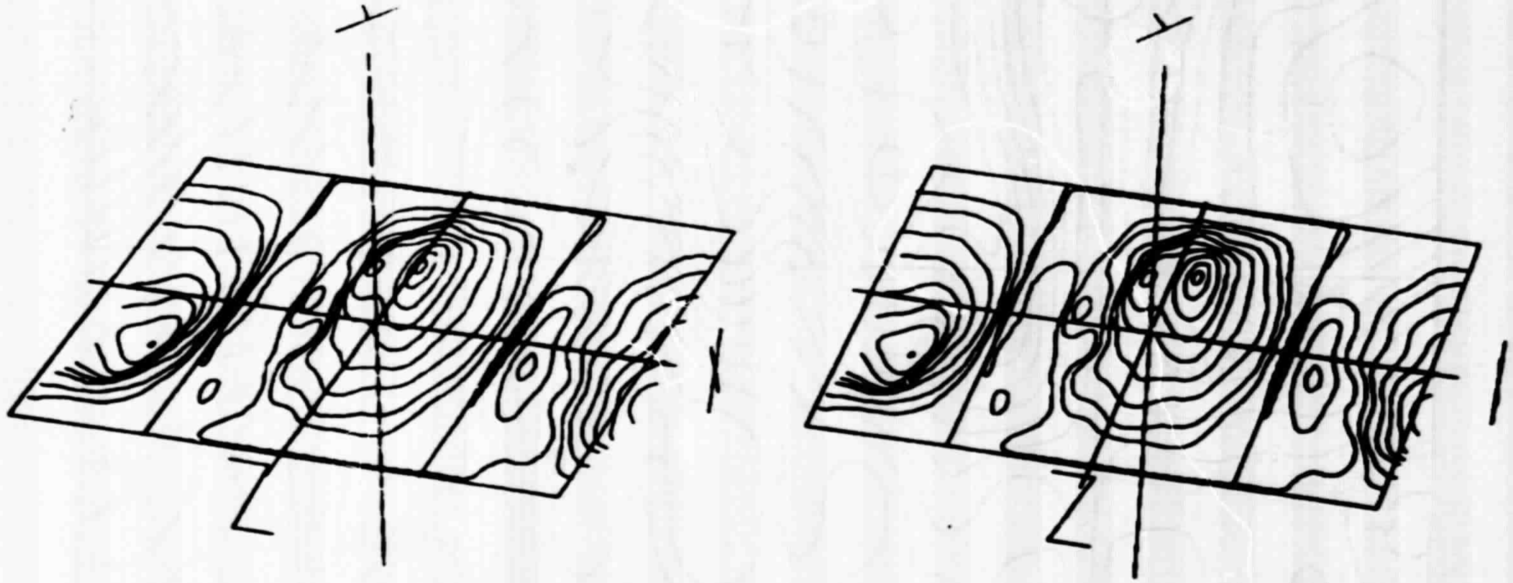


Figure A4. X torque contours (stereo double).

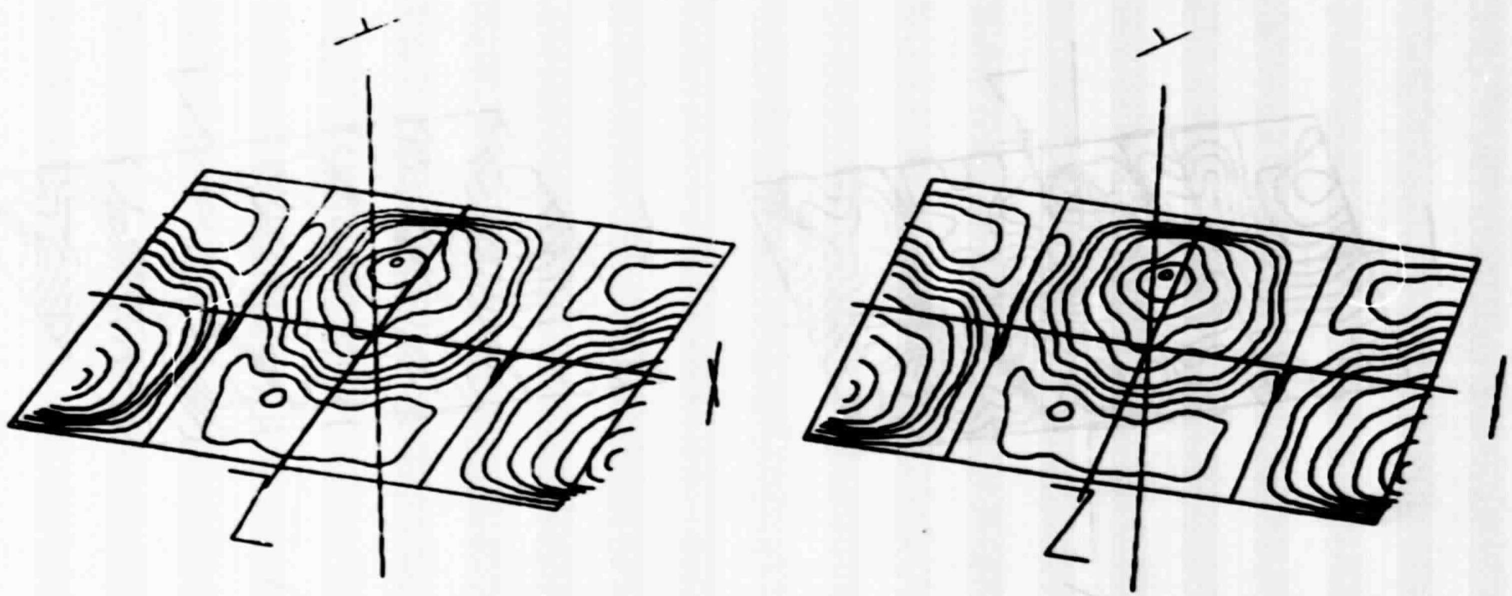


Figure A5. Y torque contours (stereo double).

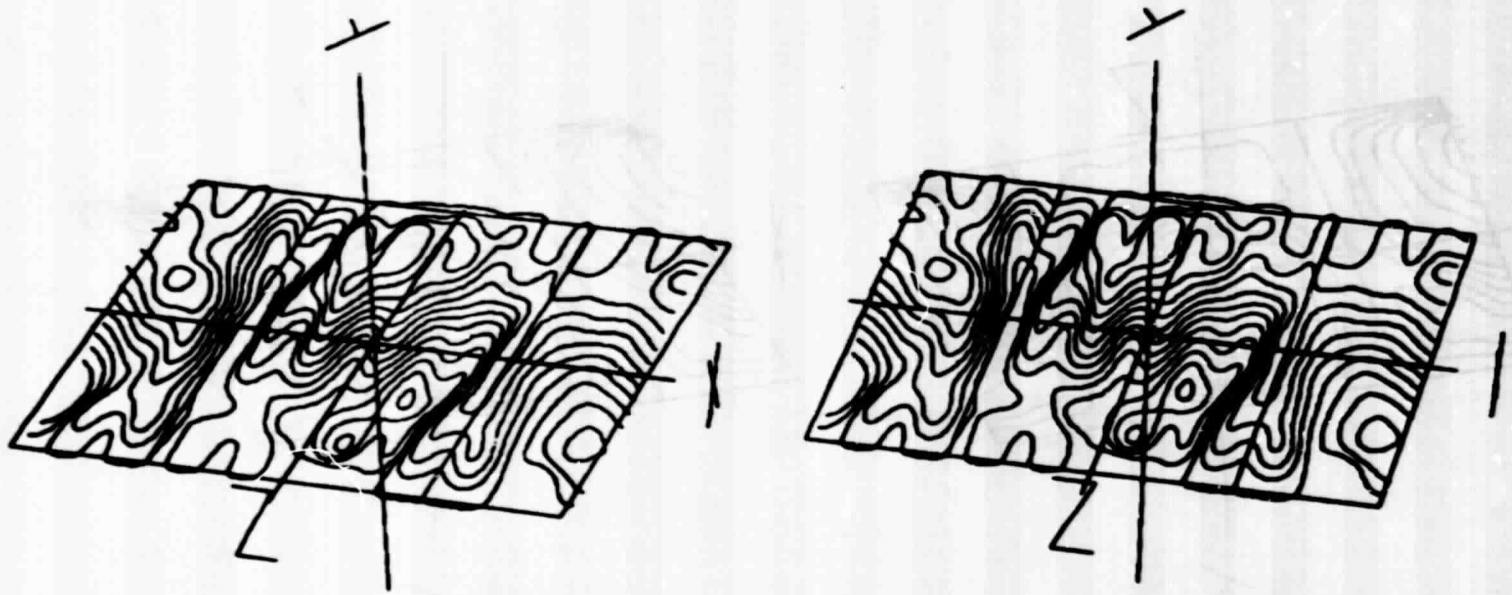


Figure A6. Z torque contours (stereo double).

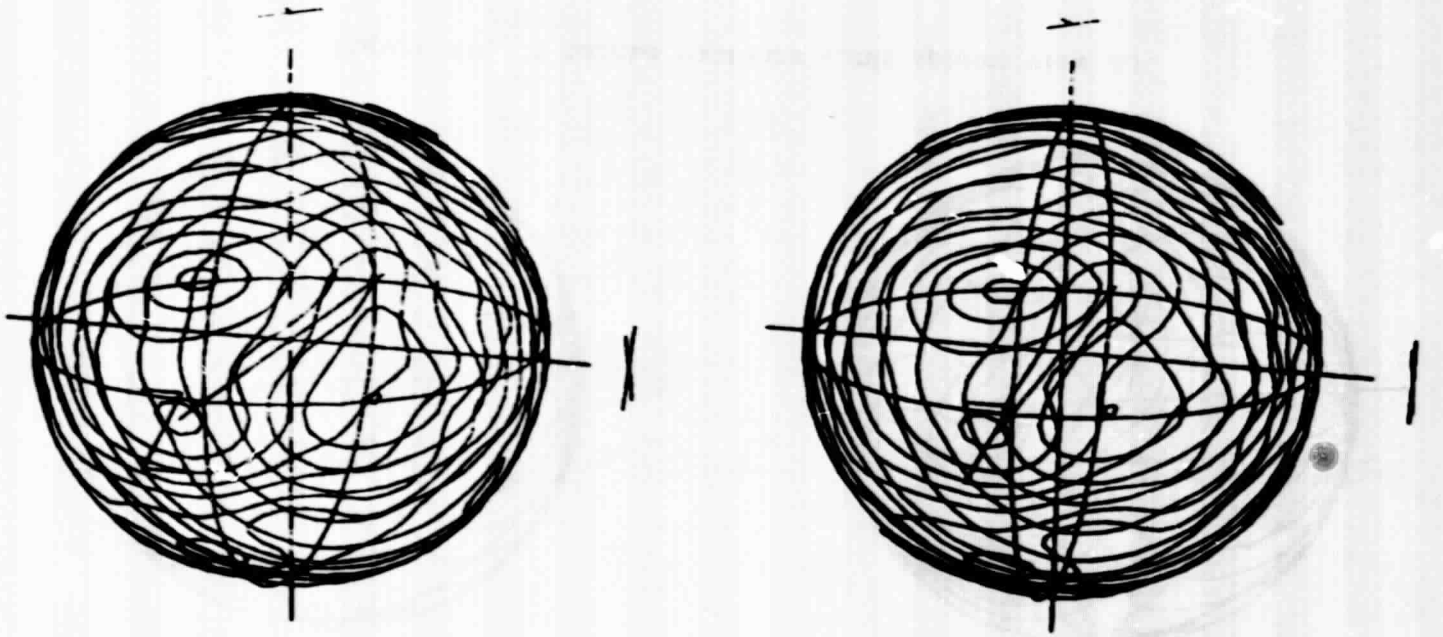


Figure A7. X torque contours (unit sphere view A).

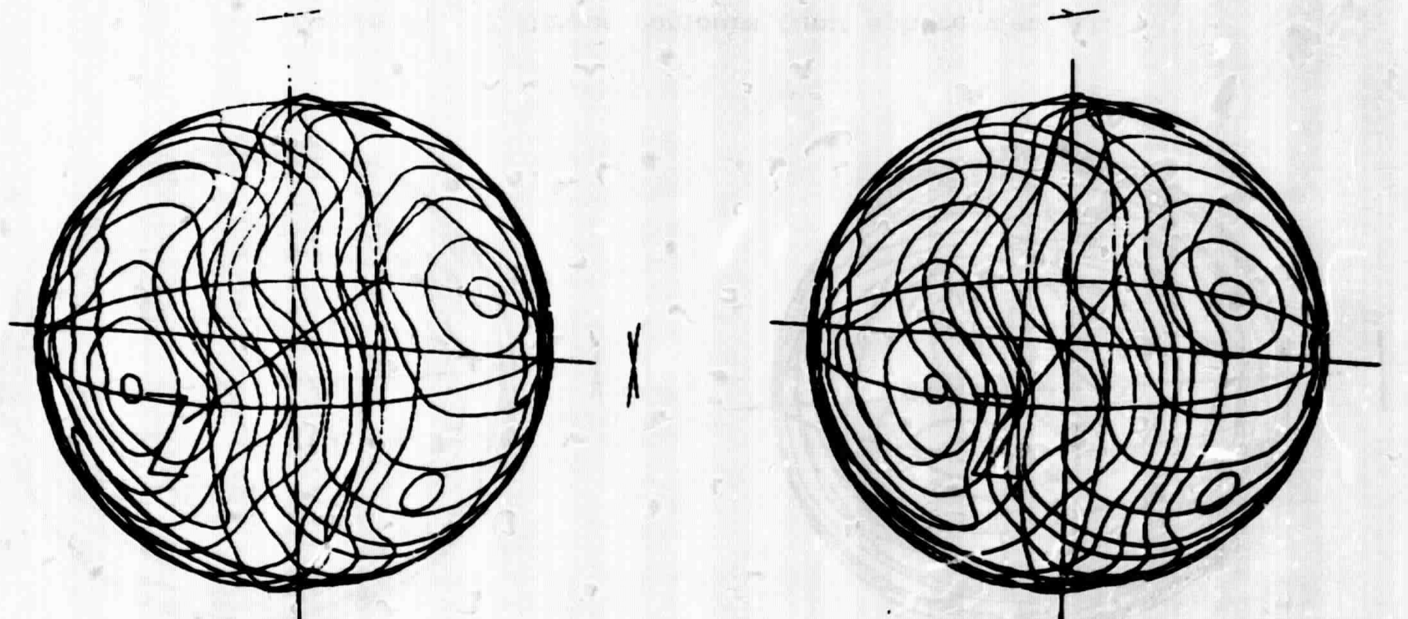


Figure A8. Y torque contours (unit sphere view A).

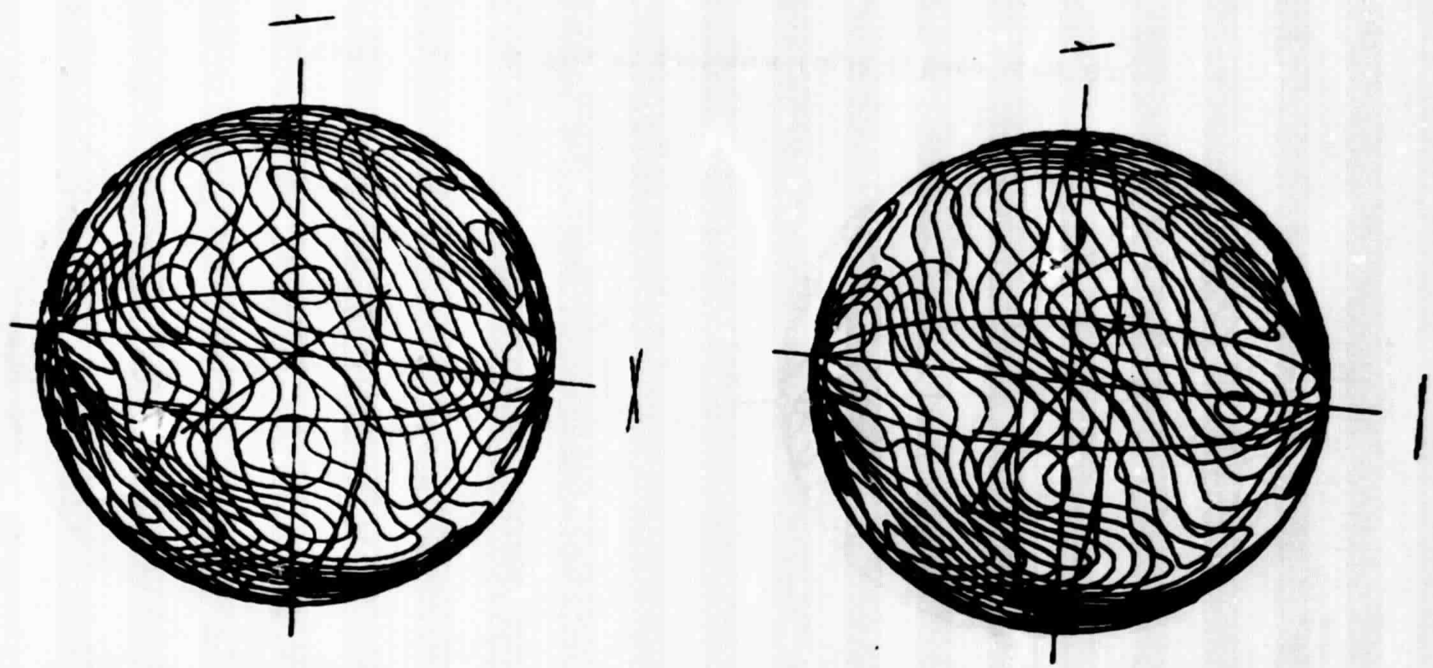


Figure A9. Z torque contours (unit sphere view A).

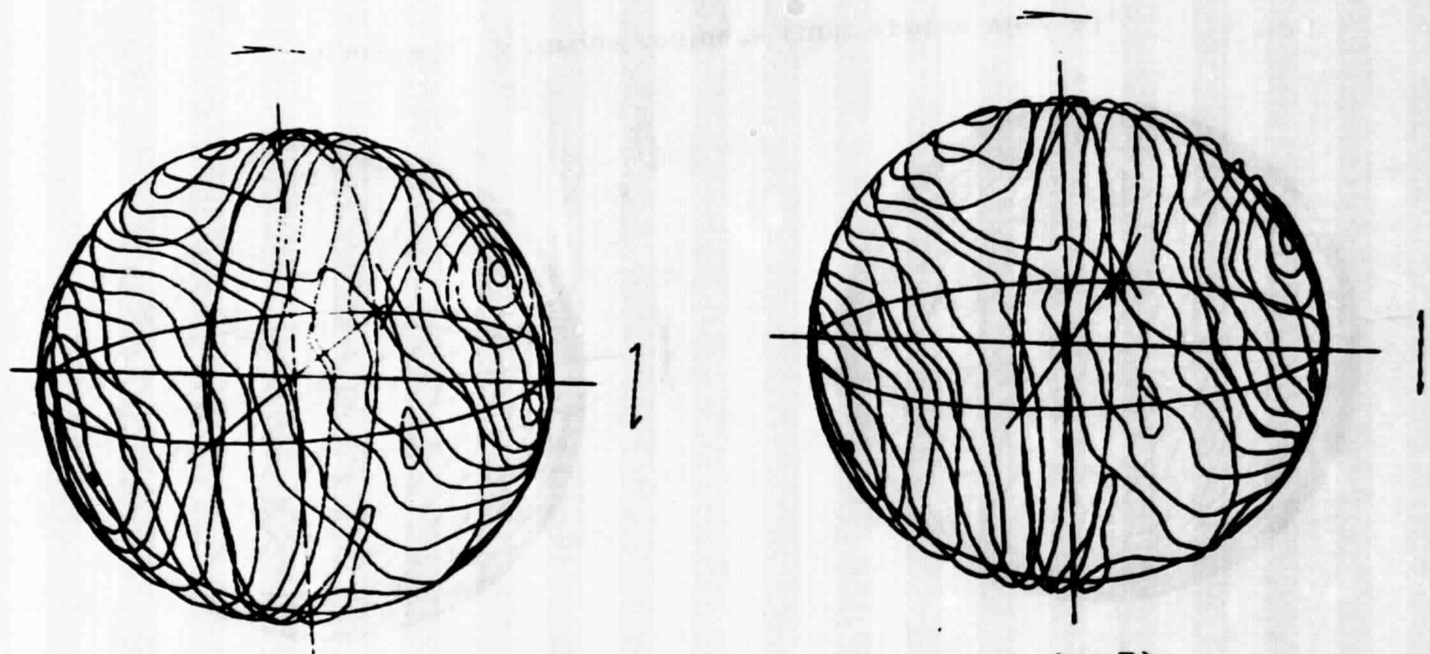


Figure A10. X torque contours (unit sphere view B).

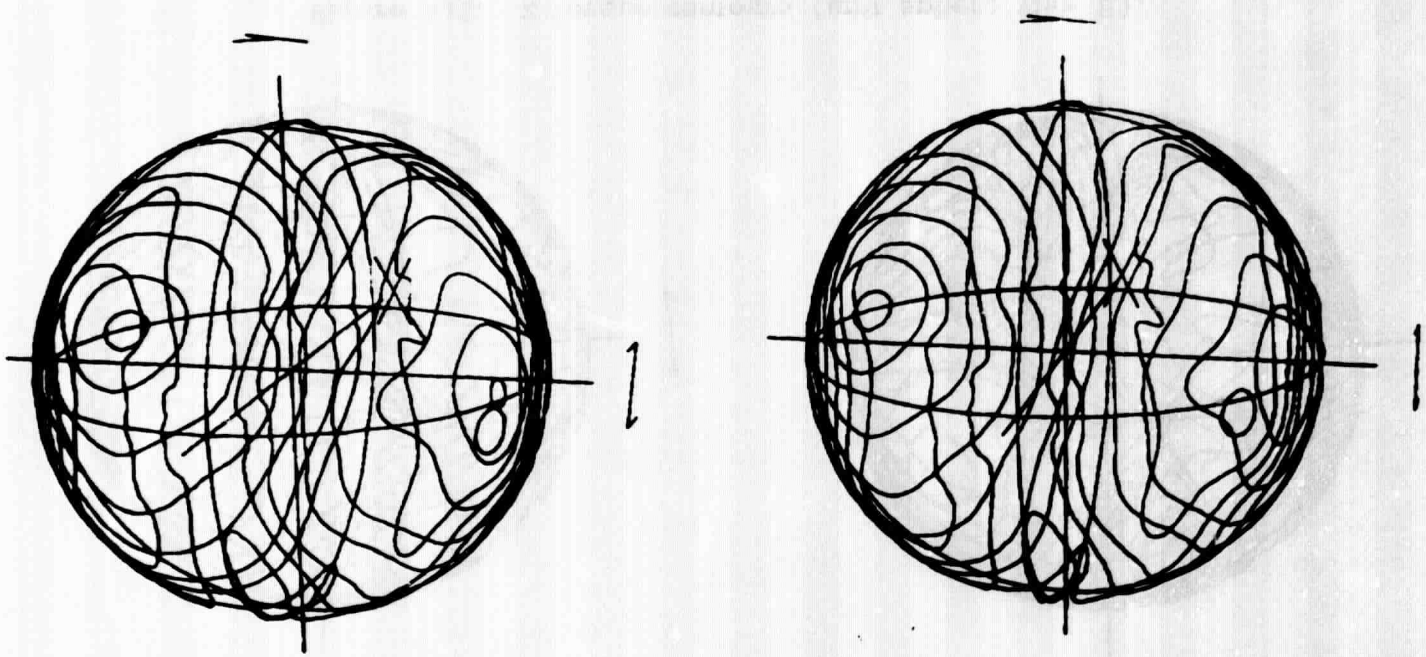


Figure A11. Y torque contours (unit sphere view B).

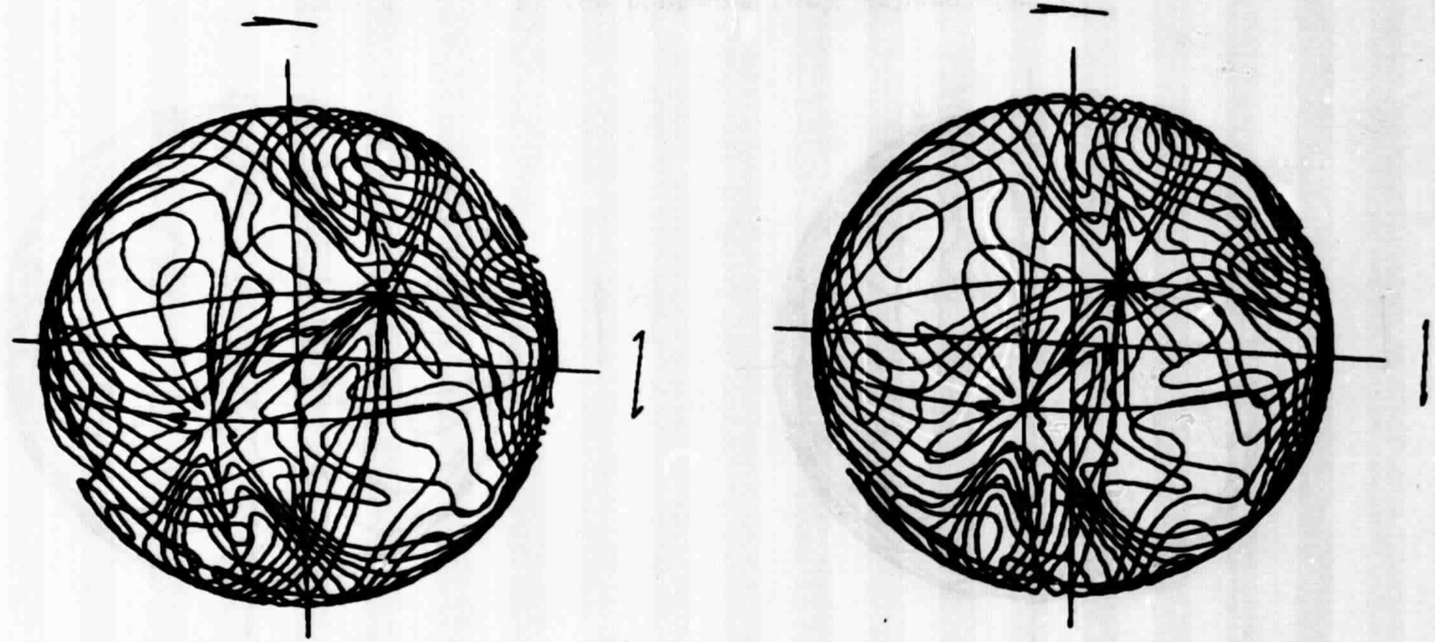


Figure A12. Z torque contours (unit sphere view B).

APPENDIX B

GRAVITATIONAL TORQUE MODEL

The gravitational torques on a satellite produced by a large, spherical primary body are important contributors to its rotational dynamics. The force on a point mass m exerted by the primary M is

$$\underline{F}_i = -\frac{GMm}{|R_i|^3}\underline{R}_i \; ; \; G = 6.672E\text{-}11 \ Nm^2/kg^2 \quad . \tag{B1}$$

The satellite can be viewed as a collection of point masses. The net torque on this collection of masses about the origin of satellite coordinates is

$$\underline{T}_g = -\sum_i \underline{r}_i \times \underline{F}_i \quad . \tag{B2}$$

The vector $\underline{r}_i$ is the position of m_i relative to the origin which is at $\underline{R}_o$ relative to the primary center. Thus

$$\underline{R}_i = \underline{R}_o + \underline{r}_i \tag{B3}$$

and

$$\underline{T}_g = -\sum_i \underline{r}_i \times GMm_i \frac{\underline{R}_o + \underline{r}_i}{|\underline{R}_o + \underline{r}_i|^3} \quad . \tag{B4}$$

In general $|\underline{r}_i| << |\underline{R}_o|$ and hence an expansion of $\underline{T}_g$ keeping only low order terms becomes sufficient for most purposes. Now

$$\frac{\underline{R}_o + \underline{r}_i}{|R_o + r_i|^3} = \frac{\underline{R}_o}{|R_o|^3} + \frac{\underline{r}_i}{|R_o|^3} - 3\underline{R}_o \frac{\underline{R}_o \cdot \underline{r}_i}{|R_o|^3} + \text{h.o.t.} \tag{B5}$$

Using the expansion of equation (B5) in (B4),

$$\underline{T}_g = - \underline{r}_{cm} \times \frac{GMm}{|\underline{R}_o|^3} \underline{R}_o + 3 \frac{GM}{|\underline{R}_o|^3} \sum_i m_i \underline{r}_i \times \underline{R}_o \frac{\underline{R}_o \cdot \underline{r}_i}{|\underline{R}_o|^3} + \text{h.o.t.} \quad . \tag{B6}$$

We have used the definition $m \underline{r}_{cm} = \sum_i m_i \underline{r}_i$. Rearranging and grouping equation (B6) we obtain

$$\underline{T}_g = - \underline{r}_{cm} \times \frac{GMm}{|\underline{R}_o|^3} \underline{R}_o + \frac{3GM}{|\underline{R}_o|^3} \frac{\underline{R}_o}{|\underline{R}_o|^3} \times \left(- \sum_i m_i \underline{r}_i \underline{r}_i\right) \cdot \frac{\underline{R}_o}{|\underline{R}_o|} + \text{h.o.t.} \quad . \tag{B7}$$

The term in parentheses in equation (B7) occurs in the definition of the moment of inertia dyadic (or tensor)

$$\underline{\underline{I}} = \sum_i m_i \left(\underline{r}_i^{\,2} \, \underline{\underline{1}} - \underline{r}_i \, \underline{r}_i\right) \quad . \tag{B8}$$

Using this definition in equation (B7) and dropping the higher order terms yields the gravity gradient torque expression

$$\underline{T}_{gg} = - \underline{r}_{cm} \times \frac{GMm}{|\underline{R}_o|^3} \underline{R}_o + 3 \frac{GM}{|\underline{R}_o|^3} \frac{\underline{R}_o}{|\underline{R}_o|} \times \underline{\underline{I}} \cdot \frac{\underline{R}_o}{|\underline{R}_o|} \quad . \tag{B9}$$

As can be seen, if the origin is positioned at the center of mass the more familiar gravity gradient torque expression results:

$$\underline{T}_{gg} = 3 \frac{GM}{|\underline{R}_o|^3} \frac{\underline{R}_o}{|\underline{R}_o|} \times \underline{\underline{I}} \cdot \frac{\underline{R}_o}{|\underline{R}_o|} \quad . \tag{B10}$$

For an orbiting body m, the orbital angular velocity magnitude is given by

$$\omega_o^2 = GM/|\underline{R}_o|^3 \quad . \tag{B11}$$

Noting that $\underline{R}_o/|\underline{R}_o|$ is a unit vector $\underline{u}_R$, we finally write

$$\underline{T}_{gg} = 3\,\omega_o^2\,\underline{u}_R \times \underline{\underline{I}} \cdot \underline{u}_R \quad . \tag{B12}$$

The torque of equation (B12) is what is commonly referred to as the gravity gradient torque.

APPENDIX C

DETERMINATION OF TORQUE EQUILIBRIUM ATTITUDES

The torque equation for a rotating vehicle in vehicle fixed coordinates is

$$\dot{\underline{H}} - \underline{\omega} \times \underline{H} = \underline{T}_{gg} + \underline{T}_{ae} \quad . \tag{C1}$$

The equilibria we seek are those which satisfy $\dot{\underline{H}} = 0$. In the case of Skylab, the momentum $\underline{H}$ is not only from vehicle rotation at orbital velocity but also includes vehicle fixed CMG momentum. Since the disturbance torques on the right hand side of equation (C1) are smooth functions, we can use Newton's method to determine equilibrium positions. The technique starts from the assumption that

$$\dot{\underline{H}} = \underline{F}([VL]) \quad ; \tag{C2}$$

i.e., the rate of change of angular momentum in body axes depends upon the relative attitude only. The symbol [VL] is the direction cosine matrix or transformation which takes a vector from the L (local vertical) coordinate system to the V (vehicle fixed) system:

$$\underline{F} \approx \underline{F}_o([VL]) + \underline{\Delta\theta} \cdot \nabla \underline{F}_o \quad ; \quad |\underline{\Delta\theta}| \text{ small} \quad . \tag{C3}$$

The basic algorithm then is

1) Select $[VL]_p$

2) $\underline{F}_p = \underline{F}([VL]_p)$

3) $\underline{\Delta\theta} = \lim [-(\nabla \underline{F}_p)^{-1} \underline{F}_p]$ (C4)

4) $[VL]_n = [\Delta\theta_z]_3 [\Delta\theta_y]_2 [\Delta\theta_x]_1 [VL]_p$

5) If $|\underline{\Delta\theta}| < \varepsilon$, set $[VL] = [VL]_n$ and terminate.

6) Otherwise, set $[VL]_p = [VL]_n$ and loop to step 2).

PRECEDING PAGE BLANK NOT FILMED

The limit in step 3) is a component limit which limits the changes of attitude permitted in the iteration procedure above and improves convergence. With this procedure 12 equilibrium attitudes were found. Others may exist and an exhaustive search was not conducted. These equilibria could be shifted somewhat by varying the momentum stored in the CMG's. The total momentum $\underline{H}$ was held constant in the L frame to $\underline{H}_{nom}$. The gradient of the vector function $\underline{F}$ was computed numerically.

APPENDIX D

STRAPDOWN ERROR ESTIMATION

The strapdown correction angle $\underline{\Delta\theta}_L$ was estimated from Skylab attitude history over one or more orbits in TEA. The assumption is that the strapdown error changes slowly over an orbit, especially the components of the error in the orbital plane. This assumption is justified by the observation that, since the vehicle orientation stays nearly fixed relative to the local vertical, the gyro drift resolved into the orbital plane is nearly sinusoidal and hence averages to near zero over an orbital period. The out-of-plane drift remains essentially constant and hence out-of-plane strapdown error grows linearly with time. The three angles θ_{e1}, θ_{e2}, and θ_{e3} represent the strapdown offset error. The error matrix is

$$[OO_k] = [\theta_{e1}]_1 \, [\theta_{e2}]_2 \, [\theta_{e3}]_3 \tag{D1}$$

where a k subscript signifies the true quantity and O is the estimate of the real orbital coordinate system O_k. The coordinate system O is defined such that O coincides with L when the vehicle crosses the ascending node and is inertially fixed except for orbit regression. We shall assume orbit regression is too small to have a significant effect on strapdown error determination. We further assume θ_{e1} and θ_{e3} are small angles. The position of the vehicle is then

$$[VO_k] = [VL] \, [\omega_o t]_2 [\theta_{e1}]_1 \, [\theta_{e2}]_2 \, [\theta_{e3}]_3 \quad . \tag{D2}$$

Using the small angle assumption, we can write

$$[VO_k] = [VL] \, [1 - \tilde{\theta}'_e] \, [\omega_o t + \theta_{e2}]_2 \quad , \tag{D3}$$

where

$$\underline{\theta}'_e = [\omega_o t]_2 [\theta_{e1}, 0, \theta_{e3}]^T \quad . \tag{D4}$$

We can also write

$$[VO_k] = [VL_k]\,[\omega_o t]_2 \quad . \tag{D5}$$

In this expression $[VL_k]$ is assumed to remain constant. We can now equate equations (D3) and (D5) to obtain

$$[VL] = [VL_k]\,[-\theta_{e2}]_2\,[1 - \tilde{\theta}_e'] \quad . \tag{D6}$$

We must also assume that the gyro drift and other drift rates are small so θ_{e2} does not change significantly over an orbit. With this assumption $[VL_k][1-\hat{\theta}_e']$ remains essentially constant. Since the location of the TEA is not known precisely, $[VL_k]$ is unknown and of course at this point θ_{e2} is also unknown so that we may as well set

$$[VL_k'] = [VL_k]\,[-\theta_{e2}]_2 \quad . \tag{D7}$$

Thus all that is known is that $[VL_k']$ is constant, that $\underline{\theta}_e'$ varies sinusoidally, and that $[VL]$ is available over one or more orbits from telemetry. This matrix is used to compute BAR angles. Intuitively it would seem that the bank angle would contain the most direct information on the in plane strapdown error, so let us look first at it.

$$\tan\beta = \frac{(VL_{k12}'/-VL_{k13}') - \theta_{e1}' + (VL_{k11}'/VL_{k13}')\,\theta_{e3}'}{1 + (VL_{k12}'/VL_{k13}')\,\theta_{e1}'} \quad . \tag{D8}$$

For high drag TEA, VL_{k11}' is small (the angle of attack is near 90 degrees). Thus the inverse tangent of equation (D8) yields

$$\beta = \beta_k' - \theta_{e1}' = \beta_k' - \theta_{e1}\cos(\omega_o t) + \theta_{e3}\sin(\omega_o t) \tag{D9}$$

$$= \beta_o + \beta_C\cos(\omega_o t) + \beta_S\sin(\omega_o t) \quad . \tag{D10}$$

We can extract the strapdown in-plane errors from equation (D10) and construct the update vector:

$$\underline{\theta}_e' = [\theta_{e1}, 0, \theta_{e3}]^T = [-\beta_c, 0, \beta_s]^T \quad . \tag{D11}$$

The strapdown correction $\underline{\Delta\theta}_L$ is developed from $\underline{\theta}_e'$ as

$$\underline{\Delta\theta}_L = [\omega_o t]_2 \, \underline{\theta}_e' \quad . \tag{D12}$$

The time t is the time of the update relative to the time of data.

The out-of-plane strapdown error could only be determined from timing and Sun passage data. The basic procedure for this was the same whether Sun sensor or solar array data were used. In either case it was assumed the strapdown error was along the orbit normal, i.e., $\theta_{e1} = \theta_{e3} = 0$. The timing events used occurred when the Sun made a known angle from a known body-fixed direction. This can be expressed mathematically as

$$\underline{U}_s \cdot \underline{U}_v = C_{sra} \quad . \tag{D13}$$

This was most easily solved in the L coordinate system:

$$\underline{U}_s = [LI] \, [0, 0, 1]^T \tag{D14a}$$

$$\underline{U}_v = [VL]^T \, \underline{U}_F \tag{D14b}$$

where $\underline{U}_F$ is a vehicle-fixed unit vector (see below, after equation D17c). The matrices [LI] and [VL] are computed at the timing event instant. At this instant we can determine $\underline{\theta}_{e2}$:

$$C_{sra} = [U_{sx}, U_{sy}, U_{sz}] \, [-\theta_{e2}]_2 \, \underline{U}_v \quad . \tag{D15}$$

From equation (D15) we obtain

$$\theta_{e2} = \text{arc sin}\left[(C_{sra} - U_{sy}\ U_{vy}) / \sqrt{S\phi^2 + C\phi^2}\right] - \phi$$

or (D16)

$$\theta_{e2} = \pi - \text{arc sin}\left[(C_{sra} - U_{sy}\ U_{vy}) / \sqrt{S\phi^2 + C\phi^2}\right] - \phi$$

with

$$S\phi = U_{sx}\ U_{vx} + U_{sz}\ U_{vz} \tag{D17a}$$

$$C\phi = U_{sx}\ U_{vz} - U_{sz}\ U_{vx} \tag{D17b}$$

$$\phi = \text{arc tan}\ (S\phi / C\phi) \quad . \tag{D17c}$$

The Sun sensor null crossing occurred when $\underline{U}_F = [0, 1, 0]^T$ and $C_{sra} = 0$. The power angle was determined from sunrise on the solar arrays so that $\underline{U}_F = [0, 0, 1]^T$ and for various values of C_{sra} depending on which array was chosen.

APPENDIX E

QUATERNIONS — A BRIEF EXPOSITION

Complex numbers of the form $z = a + ib$ have proven to be a valuable concept in the study of many physical phenomena. A generalization of this concept which proves useful in the study of rotational motion is the quaternion. Recall that the imaginary unit $i = \sqrt{-1}$. Let us define additional units j and k together with the product operation o:

$$
\begin{aligned}
i \circ i &= j \circ j = k \circ k = -1 \\
i \circ j &= -j \circ i = k \\
j \circ k &= -k \circ j = i \\
k \circ i &= -i \circ k = j
\end{aligned}
\qquad \text{(E1)}
$$

We shall define a quaternion as any quantity of the form

$$Q = Q4 + i\,Q1 + j\,Q2 + k\,Q3 \quad . \qquad \text{(E2)}$$

(See Note 1 at end of Appendix E.) By analogy to the complex number terminology Q4 is referred to as the real or scalar part of Q. It will also be convenient to think of the remaining part of Q as the imaginary or vector part. The reason for this will become clear as we proceed. Let $R = R4 + iR1 + jR2 + kR3$. The sum of quaternions Q and R is defined as

$$
\begin{aligned}
S = Q + R &= (Q4 + R4) + i\,(Q1 + R1) \\
&\quad + j\,(Q2 + R2) + k\,(Q3 + R3) \quad .
\end{aligned}
\qquad \text{(E3)}
$$

From this definition we can see that the sum operation is commutative and associative. We can now give the complete definition of the product operation o:

$$P = Q \circ R = (Q4\ R4 - Q1\ R1 - Q2\ R2 - Q3\ R3)$$
$$+\ i\ (Q4\ R1 + Q1\ R4 + Q2\ R3 - Q3\ R2)$$
$$+\ j\ (Q4\ R2 + Q2\ R4 + Q3\ R1 - Q1\ R3)$$
$$+\ k\ (Q4\ R3 + Q3\ R4 + Q1\ R2 - Q2\ R1) \quad . \qquad \text{(E4)}$$

With this definition we can show that o is associative and distributive but not commutative, i.e., $Q \circ R \neq R \circ Q$. We shall call any quaternion having zero imaginary part a scalar and, obviously, the algebra of scalars is just the algebra of real numbers. Thus, multiplication of a quaternion by a scalar simply results in a quaternion whose elements are multiplied by that scalar according to definition E4. We can now define the difference operation as

$$D = Q - R = Q + (-1) \circ R \quad . \qquad \text{(E5)}$$

For convenience, we shall always omit the o when multiplying a quaternion by a scalar so that $(-2) \circ Q = -2\ Q$.

By analogy to complex algebra, let us define the conjugate quaternion to Q. The conjugation operation will be denoted by ( )*. Thus

$$Q^* = Q4 - i\ Q1 - j\ Q2 - k\ Q3 \quad . \qquad \text{(E6)}$$

So far all of our definitions have been extensions of those for complex numbers as can be seen by assuming $Q2 = Q3 = R2 = R3 = 0$. Thus the complex number system is a subset of the quaternions. It can be easily seen that

$$Q^* \circ Q = Q \circ Q^* = Q1\ Q1 + Q2\ Q2 + Q3\ Q3 + Q4\ Q4 \quad . \qquad \text{(E7)}$$

Note that $Q^* \circ Q$ is a pure real number or scalar. With this observation we can define the inverse:

$$Q^{-1} = (1/(Q^* \circ Q))\ Q^* = Q^*/(Q^* \circ Q) \quad ; \quad Q^* \circ Q \neq 0 \quad . \text{(E8)}$$

Finally then, we can define a division operation:

$$Q \div R = Q \circ R^{-1} \quad . \qquad \text{(E9)}$$

We can see that $Q \div Q = Q \circ Q^{-1} = Q^{-1} \circ Q = 1$ so that Q^{-1} satisfies the necessary properties of an inverse as long as $Q \neq 0$. This will be useful later.

We need some additional results and definitions. First we can show that

$$(Q \circ R)^* = R^* \circ Q^* \quad . \tag{E10}$$

If the quaternion $Q = Q^*$, then Q is necessarily a scalar. Also, if $Q = -Q^*$, Q is purely imaginary or a vector quaternion. If V is a vector quaternion, we shall designate this by an underline as is also used to designate a 3-space vector, i.e., (i, j, and k will not be underlined)

$$\underline{V} = i\ V1 + j\ V2 + k\ V3 \quad . \tag{E11}$$

For compactness of our notation we shall let

$$Q = Q4 + \underline{Q} \quad , \tag{E12}$$

where

$$\underline{Q} = i\ Q1 + j\ Q2 + k\ Q3.$$

Thus

$$Q \circ R = (Q4\ Q4 - \underline{Q} \cdot \underline{R}) + Q4\ \underline{R} + R4\ \underline{Q} + \underline{Q} \times \underline{R} \quad . \tag{E13}$$

The operations $\cdot$ and $\times$ are defined as for 3-space vectors so that

$$\underline{Q} \cdot \underline{R} = Q1\ R1 + Q2\ R2 + Q3\ R3 \tag{E14}$$

and

$$\underline{Q} \times \underline{R} = i\ (Q2\ R3 - Q3\ R2) + j\ (Q3\ R1 - Q1\ R3)$$
$$+ k\ (Q1\ R3 - Q3\ R1) \quad . \tag{E15}$$

For vectors $\underline{A}$ and $\underline{B}$,

$$\underline{A} \circ \underline{B} = -\underline{A} \cdot \underline{B} + \underline{A} \times \underline{B} \quad . \tag{E16}$$

In general the product of quaternions mixes scalar and vector parts together so that this produce is not very interesting in the study of rotational motion in 3-space. The triple product

$$V' = Q^* \circ V \circ Q \tag{E17}$$

is more interesting since it does preserve scalar and vector parts of V without mixing them. This property is trivial for the scalar part of V and follows for the vector part since

$$\underline{V}^* = (Q^* \circ \underline{V} \circ Q)^* = -Q^* \circ \underline{V} \circ Q = -\underline{V} \quad . \tag{E18}$$

Hence as noted the triple quaternion product (E17) takes a scalar into a scalar and a vector into a vector for any quaternion Q. Furthermore, the length of the vector $|\underline{V}| = \sqrt{\underline{V} \cdot \underline{V}}$ and

$$\underline{V}' \cdot \underline{V}' = \underline{V}' \circ \underline{V}'^* = Q^* \circ \underline{V} \circ Q \circ (Q^* \circ \underline{V} \circ Q)$$

$$= (Q^* \circ Q)\ \underline{V} \circ \underline{V}^* \quad . \tag{E19}$$

Equation (E19) indicates the triple product (E17) multiplies vector length by the factor $Q^* \circ Q$ which is a real number. We note that if $Q^* \circ Q = 1$, vector length is preserved and the vector mapping $\underline{V} \rightarrow \underline{V}'$ looks like a rotation operator. It is a linear operator in that $a\underline{A} + b\underline{B} \rightarrow a\underline{A}' + b\underline{B}'$. Restricting ourselves to normalized quaternions which preserve length, we see that E17 is equivalent to a rotation of vector $\underline{V}$ into $\underline{V}'$. Since we are looking only at normalized quaternions, we can without loss of generality, represent Q as

$$Q = \cos \phi/2 + \sin \phi/2\ \underline{u} \quad ; \quad \text{where} \quad \underline{u} \cdot \underline{u} = 1 \quad . \tag{E20}$$

The triple product equation (E17) can be combined with equation (E20) to give

$$\underline{V}' = \cos\phi\, \underline{V} - \sin\phi\, \underline{u} \times \underline{V} + (1 - \cos\phi)\, \underline{u}\, \underline{u} \cdot \underline{V} \quad . \tag{E21}$$

Equation (E21) is the general form of the rotation of a vector $\underline{V}$ about axis $\underline{u}$ through the angle $-\phi$. That this is true is seen by examining the rotation operation. Clearly any vector along the rotation axis $\underline{u}$ is not changed by the rotation so that if the vector $\underline{V}$ is broken into parts parallel to and normal to $\underline{u}$; i.e.

$$\underline{V} = \underline{V}_{\parallel} + \underline{V}_{\perp} \quad , \text{ where } \underline{V}_{\parallel} = \underline{V} \cdot \underline{u}\, \underline{u} \text{ and } \underline{V}_{\perp} \cdot \underline{u} = 0 \ . \tag{E22}$$

We must also have $\underline{V}' = \underline{V}_{\parallel} + \underline{V}'_{\perp}$; where $\underline{V}'_{\perp} \cdot \underline{u} = 0$ and $\underline{V}_{\parallel} = \underline{V}'_{\parallel}$. Since $\underline{V}_{\perp}$ and $\underline{V}'_{\perp}$ are normal to $\underline{u}$, we can express $\underline{V}'_{\perp}$ as

$$\underline{V}'_{\perp} = x\, \underline{u} \times \underline{V}_{\perp} + y\, \underline{u} \times (\underline{u} \times \underline{V}_{\perp}) \quad . \tag{E23}$$

Now $\underline{V}'_{\perp} \cdot \underline{V}_{\perp} = |\underline{V}_{\perp}|^2 \cos\alpha = -y\, |\underline{V}_{\perp}^2| \rightarrow y = \cos\alpha$ and $\underline{V}_{\perp} \times \underline{V}'_{\perp} = |\underline{V}_{\perp}|^2 \sin\alpha\, \underline{u}$ and thus $x = \sin\alpha$; where α is the rotation angle. Combining, we obtain

$$\underline{V}'_{\perp} = \sin\alpha\, \underline{u} \times \underline{V}_{\perp} - \cos\alpha\, \underline{u} \times (\underline{u} \times \underline{V}_{\perp}) \tag{E24}$$

and

$$\underline{V}' = \underline{u}\, \underline{u} \cdot \underline{V} + \sin\alpha\, \underline{u} \times \underline{V} - \cos\alpha\, \underline{u} \times (\underline{u} \times \underline{V}) \quad . \tag{E25}$$

The equivalence between equations (E21) and (E25) for $\alpha = -\phi$ is established. Thus, the mapping (E17) is equivalent to a rotation operator in a vector 3-space. Since the coordinate directions are also vectors, we can rotate the coordinate system instead of the vector. Rotating the vector through $-\phi$ yields the same components $\underline{V}'$ as rotating the coordinate axes through ϕ. Thus we can look at $\underline{V}'$ as a new vector formed from $\underline{V}$ by rotation and expressed in the old axis system or as the old vector expressed in new axes rotated relative to the old. We have now demonstrated that any coordinate system rotation can be represented by a quaternion. Note that if Q satisfies equation (E17), then so does -Q. Looking back at equation (E20) tells us that -Q corresponds to ϕ + 360 degrees which represents the same attitude ϕ does.

We now look at the time variations of Q. Since Q is constrained to be normalized, we necessarily have

$$d/dt(Q^* \circ Q) = 0 = \dot{Q}^* \circ Q + Q^* \circ \dot{Q} \quad . \qquad \text{(E26)}$$

We see from equation (E26) that $Q^* \circ \dot{Q} = -(Q^* \circ \dot{Q})^*$ and hence must be a vector. Let us define this vector by

$$Q^* \circ \dot{Q} = 1/2\ \underline{\omega} \qquad \text{(E27)}$$

$$\rightarrow \dot{Q} = 1/2\ Q \circ \underline{\omega} \quad (\text{since } Q \circ Q^* = Q^* \circ Q = 1) \quad . \qquad \text{(E28)}$$

We shall see the reason for the 1/2 factor later. When we evaluate the rate of change of a vector with time in two reference frames, we find

$$\underline{\dot{V}} = Q \circ \underline{\dot{V}}' \circ Q^* + \dot{Q} \circ \underline{V}' \circ Q^* + Q \circ \underline{V}' \circ \dot{Q}^* \quad . \qquad \text{(E29)}$$

Using equations (E28) in (E29),

$$\underline{\dot{V}} = Q \circ [\underline{\dot{V}}' + 1/2\ (\underline{\omega} \circ \underline{V}' - \underline{V}' \circ \underline{\omega})] \circ Q^* \qquad \text{(E30)}$$

$$\underline{\dot{V}} = Q \circ [\underline{\dot{V}}' + \underline{\omega} \times \underline{V}] \circ Q^* \quad . \qquad \text{(E31)}$$

Now the reason for the factor 1/2 becomes clear. It is so that we can identify $\underline{\omega}$. Equation (E31) is exactly like the corresponding equation for 3-space vectors if we identify $\underline{\omega}$ as the angular velocity of the primed reference frame expressed in primed coordinates. This identification follows from the fact that equation (E31) holds for an arbitrary vector $\underline{V}$. Thus, $\underline{\omega}$ is identified as the relative angular velocity of the primed axes with respect to the unprimed.

The above discussion completes the basic development of our quaternion tools. We now turn to the problem of developing a more convenient notation. The most logical choice which comes to mind is a matrix representation. The quaternion Q would logically become

$$Q = \begin{bmatrix} Q1 \\ Q2 \\ Q3 \\ Q4 \end{bmatrix} \quad . \tag{E32}$$

Looking back to the definition equation (E4) of the quaternion product o, we see that for P = Q o R we have

$$P = \begin{bmatrix} Q4 & -Q3 & Q2 & Q1 \\ Q3 & Q4 & -Q1 & Q2 \\ -Q2 & Q1 & Q4 & Q3 \\ -Q1 & -Q2 & -Q3 & Q4 \end{bmatrix} \begin{bmatrix} R1 \\ R2 \\ R3 \\ R4 \end{bmatrix} \triangleq \tilde{\tilde{Q}}\, R \quad . \tag{E33}$$

(See Notes at end of Appendix E.) Similarly, if D = A o B o C, then

$$D = \widetilde{\widetilde{A \circ B}}\; C = \widetilde{\widetilde{\tilde{\tilde{A}}\, B}}\; C = \tilde{\tilde{A}}\, \tilde{\tilde{B}}\, C \quad . \tag{E34}$$

This result shows that the set of matrices of the form $\tilde{\tilde{Q}}$ have the properties of the quaternions and in fact comprise a matrix representation of quaternion algebra with matrix multiplication corresponding to o. We can also express the quaternion product in the alternate form

$$P = \begin{bmatrix} R4 & R3 & -R2 & R1 \\ -R3 & R4 & R1 & R2 \\ R2 & -R1 & R4 & R3 \\ -R1 & -R2 & -R3 & R4 \end{bmatrix} \begin{bmatrix} Q1 \\ Q2 \\ Q3 \\ Q4 \end{bmatrix} \triangleq \bar{\bar{R}}\, Q \tag{E35}$$

(See Notes at end of Appendix E.) Here the mapping also yields a matrix representation except the order of the factors must be reversed. Thus, we have

$$P = \tilde{\tilde{Q}}\, R = \bar{\bar{R}}\, Q$$

$$D = \tilde{\tilde{A}}\, \tilde{\tilde{B}}\, C = \bar{\bar{C}}\, \bar{\bar{B}}\, A \quad . \tag{E36}$$

$$\overline{\overline{B \circ C}} = \bar{\bar{C}}\, \bar{\bar{B}}$$

According to these definitions and results, we have

$$A \tilde{\tilde{B}} C = \tilde{\tilde{A}} \bar{\bar{C}} B = \bar{\bar{C}} \bar{\bar{B}} A = \bar{\bar{C}} \tilde{\tilde{A}} B \quad . \tag{E37}$$

Thus, an interesting and sometimes useful result is that $\tilde{\tilde{A}} \bar{\bar{C}} = \bar{\bar{C}} \tilde{\tilde{A}}$. Let us now look at our previous work and make use of these new definitions:

$$\underline{V}' = Q^* \circ \underline{V} \circ Q \rightarrow \underline{V}' = \tilde{\tilde{Q}}^* \tilde{\tilde{\underline{V}}} Q = \tilde{\tilde{Q}}^* \bar{\bar{Q}} \underline{V} \quad . \tag{E38}$$

Note that we now have a matrix formed from Q which rotates coordinate axes and produces $\underline{V}'$ from $\underline{V}$. Linear vector spaces are also represented by matrices. The 3-space vectors, $\underline{V}$ and $\underline{V}'$ are related as

$$\underline{V}' = M \underline{V} \quad . \tag{E39}$$

Here M is a 3 × 3 matrix which transforms components of $\underline{V}$ to primed coordinates. Referring back to equation (E25) and replacing α by $-\phi$ we see that

$$M = \underline{u}\, \underline{u}^T - \sin\phi\, \tilde{u} + \cos\phi\, (1 - \underline{u}\, \underline{u}^T) \quad . \tag{E40}$$

(See Notes at end of Appendix E.) The matrix $\tilde{u}$ is the so-called cross product matrix and just happens to be the upper 3 × 3 formed by dropping the final row and column of $\tilde{\tilde{u}}$. The matrix 1 is the identity matrix of the appropriate size to fit the current application. We have already shown the equivalence between M and $\tilde{\tilde{Q}}^* \bar{\bar{Q}}$. Let us look more closely at the latter since it is 4 × 4. We can partition the double-tilde or double-bar matrices as

$$\tilde{\tilde{Q}} = Q_4\, 1 + \left[\begin{array}{c|c} \tilde{Q} & Q \\ \hline -Q^T & 0 \end{array}\right] \; ; \quad \bar{\bar{Q}} = Q_4\, 1 + \left[\begin{array}{c|c} -\tilde{Q} & \underline{Q} \\ \hline -Q^T & 0 \end{array}\right] \quad . \tag{E41}$$

Since $Q^* \circ Q = 1$, $Q^* = Q^{-1}$ so that $Q^* Q = 1$. Also, $Q^* = Q$(transpose) so that

$$\tilde{\tilde{Q}}^* \bar{\bar{Q}} = \left[\begin{array}{c|c} Q4^2\, 1 - 2\, Q4\, \tilde{Q} + \tilde{Q}\, \tilde{Q} + \underline{Q}\, \underline{Q}^T & 0 \\ \hline 0 & 1 \end{array}\right] . \qquad (E42)$$

Hence $M = Q4^2\, 1 - 2\, Q4\, \tilde{Q} + 2\, \underline{Q}\, \underline{Q}^T - Q^2\, 1$. In expanded form

$$M = \left[\begin{array}{c|c|c} Q1^2 - Q2^2 - Q3^2 + Q4^2 & 2(Q1\,Q2 + Q3\,Q4) & 2(Q1\,Q3 - Q2\,Q4) \\ 2(Q2\,Q1 - Q3\,Q4) & -Q1^2 + Q2^2 - Q3^2 + Q4^2 & 2(Q2\,Q3 + Q1\,Q4) \\ 2(Q3\,Q1 + Q2\,Q4) & 2(Q3\,Q2 - Q1\,Q4) & -Q1^2 - Q2^2 + Q3^2 + Q4^2 \end{array}\right] \qquad (E43)$$

From equation (E28)

$$\dot{Q} = \frac{1}{2}\, Q \circ \underline{\omega} = \frac{1}{2}\, \bar{\bar{\underline{\omega}}}\, Q = \frac{1}{2} \begin{bmatrix} 0 & \omega 3 & -\omega 2 & \omega 1 \\ -\omega 3 & 0 & \omega 1 & \omega 2 \\ \omega 2 & -\omega 1 & 0 & \omega_3 \\ -\omega 1 & -\omega 2 & -\omega 3 & 0 \end{bmatrix} \begin{bmatrix} Q1 \\ Q2 \\ Q3 \\ Q4 \end{bmatrix} . \qquad (E44)$$

Equations (E43) and (E44) summarize the useful results from our discussion.

We are now ready to consider the question of successive rotations applied to a coordinate reference. A coordinate frame rotation is a rigid displacement of all the points in the system with a fixed axis passing through the origin. Thus, it would seem that several successive rotations should displace every point except the origin. Let us now consider the coordinate frame as a rigid body and determine the most general displacement of it which keeps one point fixed. We must first explain what is meant by a rigid body displacement. A rigid body displacement is one which preserves distances between every possible pair of points in the body. The displacement is mathematically represented as a vector function $\underline{f}$. This function then has two basic properties:

1) $\underline{f}(0) = 0$

2) $|\underline{f}(\underline{r}_A) - \underline{f}(\underline{r}_B)| = |\underline{r}_A - \underline{r}_B|$. (E45)

To study this in more detail, we define two additional points $\underline{r}_1$ and $\underline{r}_2$ together with their images under $\underline{f}$, $\underline{f}(\underline{r}_1)$ and $\underline{f}(\underline{r}_2)$. Let $\underline{f}_1 = \underline{f}(\underline{r}_1)$ and $\underline{f}_2 = \underline{f}(\underline{r}_2)$. Let us define unit vectors .

$$i = \underline{r}_1 / |\underline{r}_1| \quad ; \quad \underline{u}_1 = \underline{f}_1 / |\underline{f}_1|$$

$$j = \frac{\underline{r}_2 - \underline{r}_2 \cdot \underline{i}\,\underline{i}}{|\underline{r}_2 - \underline{r}_2 \cdot \underline{i}\,\underline{i}|} \quad ; \quad \underline{u}_2 = \frac{\underline{f}_2 - \underline{f}_2 \cdot \underline{u}_1\underline{u}_1}{|\underline{f}_2 - \underline{f}_2 \cdot \underline{u}_1\underline{u}_1|} \quad . \tag{E46}$$

The vectors i, j, k and $\underline{u}_1$, $\underline{u}_2$, $\underline{u}_3$ each form orthonormal bases for 3 dimensional space. An arbitrary vector $\underline{r}$ can be expressed as

$$\underline{r} = x\, i + y\, j + z\, k \quad . \tag{E47}$$

The corresponding $\underline{f}(\underline{r}) = f_x\, \underline{u}_1 + f_y\, \underline{u}_2 + f_z\, \underline{u}_3$. Condition 2 of equation (E45) can only be satisfied if

$$\underline{f}(\underline{r}) = x\, \underline{u}_1 + y\, \underline{u}_2 \pm z\, \underline{u}_3 \quad . \tag{E48}$$

Thus we can define two functions $\underline{f}+$ and $\underline{f}-$ that both satisfy equation (E45) and map $\underline{r}_1$ into $\underline{f}_1$ and $\underline{r}_2$ into $\underline{f}_2$. The function $\underline{f}-$ can be viewed as the reflection $(x,y,z) \rightarrow (x,y,-z)$ followed by $\underline{f}+$. We are only interested in continuous transitions from an initial position to a final position and thus reflections must be eliminated since it is not possible to go from (x,y,z) to $(x,y,-z)$ continuously without violating condition 2 of equation (E45). Thus continuous rigid displacements can only occur in the form $\underline{f}+$. This function can be written out as

$$\underline{f}(\underline{r}) = x\, \underline{u}_1 + y\, \underline{u}_2 + z\, \underline{u}_3 = \alpha_x i + \alpha_y j + \alpha_z k \tag{E49}$$

$$\begin{aligned} \alpha_x &= x\, \underline{u}_1 \cdot i + y\, \underline{u}_2 \cdot i + z\, \underline{u}_3 \cdot i \\ \alpha_y &= x\, \underline{u}_1 \cdot j + y\, \underline{u}_2 \cdot j + z\, \underline{u}_3 \cdot j \\ \alpha_z &= x\, \underline{u}_1 \cdot k + y\, \underline{u}_2 \cdot k + z\, \underline{u}_3 \cdot k \quad . \end{aligned} \tag{E50}$$

Equation (E50) can be rewritten in the matrix form

$$\underline{\alpha} = M \underline{r} \quad . \tag{E51}$$

The vectors $\underline{\alpha}$ and $\underline{r}$ are of the same length and since this must hold for all pairs $\underline{\alpha}$ and $\underline{r}$ we must have that

$$M^T M = 1 \quad . \tag{E52}$$

Equation (E52) also implies all eigenvalues of M are of unit magnitude. The eigenvalues and eigenvectors of M may be complex so that if M $\underline{x}$ = λ $\underline{x}$, then $\underline{x}^{T*} M^T M \underline{x} = 1 = \lambda^* \lambda x^{T*} x$. For 3-dimensional space M must have at least one real eigenvalue. Since M is real, its eigenvalues must occur in complex pairs. Therefore at least one eigenvalue of M must be equal to 1. The value -1 could not be acceptable since it would imply M $\underline{u}$ = -$\underline{u}$ which would be a reflection and already ruled out. Thus, we have that det M = 1.

The matrix M is now looking very much like a rotation since the eigenvector $\underline{u}$ is an eigenaxis. All we must do now is to determine the angle of rotation. Along with the eigenvector $\underline{u}$, let us define unit vectors $\underline{v}$ and $\underline{w}$ such that $\underline{u}$, $\underline{v}$, $\underline{w}$ is an orthonormal basis set. Also, we assume $\underline{w} = \underline{u} \times \underline{v}$. With these definitions we can express the matrix M as

$$M = M_{11} \underline{u}\, \underline{u}^T + M_{12} \underline{u}\, \underline{v}^T + M_{21} \underline{v}\, \underline{u}^T + \ldots + M_{33} \underline{w}\, \underline{w}^T \ . \tag{E53}$$

From the fact that M $\underline{u}$ = $\underline{u}$ and that M^T M = 1 equation (E53) reduces to

$$M = \underline{u}\, \underline{u}^T + p(\underline{v}\, \underline{v}^T + \underline{w}\, \underline{w}^T) + q(\underline{v}\, \underline{w}^T - \underline{w}\, \underline{v}^T) \ ; \ p^2 + q^2 = 1 \ . \tag{E54}$$

We can now eliminate the vectors $\underline{v}$ and $\underline{w}$ from this equation by use of the proper function of $\underline{u}$. Thus

$$M = \underline{u}\, \underline{u}^T + p(1 - \underline{u}\, \underline{u}^T) - q\, \tilde{u} \quad . \tag{E55}$$

This completes the proof that the matrix M is a rotation matrix. This is now obvious from inspection of equation (E55) by comparing it to equation (E40) with $p = \cos \phi$ and $q = \sin \phi$. Thus the most general displacement of a rigid body (or transformation of a coordinate system) in which at least 1 point remains fixed is a rotation about a fixed axis i.e., the final orientation can be obtained from the original by a single rotation about the axis $\underline{u}$ through the angle ϕ ($\underline{u}$ and ϕ are determined from M) even though the actual motion from initial to final may have been more complex.

What all the previous discussion boils down to is that the product of a pair of rotations is itself a rotation. Thus, if M_1 and M_2 are rotations about $\underline{u}_1$ and $\underline{u}_2$ respectively, then $M_1 M_2 = M_3$ is also a rotation through some angle ϕ_3 about some axis $\underline{u}_3$. In fancier terms the set of rotations forms a group under matrix multiplication.

The results of our previous discussions now suggest some new notation that may aid us in keeping up with the multiplicity of coordinate systems that must usually be dealt with in analysis of spacecraft rotational dynamics. To remain completely general, let us consider three coordinate frames A, B, C. We shall let the symbol [BA] represent the rotation matrix which transforms a vector expressed in the A frame to a vector expressed in the B frame.

$$\underline{V}^{(B)} = [BA]\, \underline{V}^{(A)} \quad . \tag{E56}$$

For convenience we use the notation superscript (A) or (B) etc. to indicate which coordinate frames the vectors are being expressed in. If the superscripts are not specified, it means that the coordinate frame is implicit in the definition of the symbol or that it doesn't matter as long as all vectors are in the same frame. We are here more interested in the rotations [BA] etc. There are three rotations between pairs: AB, BC, CA. From our previous work

$$[CA] = [CB]\,[BA] \quad . \tag{E57}$$

Corresponding to equation (E57) is a quaternion relation of similar form. First, since $\underline{V}^{(C)} = [CA]\, \underline{V}^{(A)}$, we have

$$\underline{V}^{(C)} = Q_{CA}{}^{*} \circ \underline{V}^{(A)} \circ Q_{CA} \quad . \tag{E58}$$

Here, Q_{CA} is the quaternion corresponding to [CA]. Thus analogous to equation (E57) we have

$$\underline{V}^{(C)} = Q_{CB}{}^* \circ Q_{BA}{}^* \circ \underline{V}^{(A)} \circ Q_{BA} \circ Q_{CB} \quad . \tag{E59}$$

Interestingly, we see that $Q_{CA} = Q_{BA} \circ Q_{CB}$ so that the factors occur in reverse order from equation (E57). However, if we use the double-bar operator we can multiply in the same order, i.e.

$$Q_{CA} = \bar{\bar{Q}}_{CB}\, Q_{BA} \quad . \tag{E60}$$

This result is the one which we wish to use analogous to equation (E57). Equations (E57) and (E60) have an easily remembered form and in fact behave as if multiplication cancelled the terms appearing on the inside. This makes it quite easy to construct chains of transformations to any desired system. In this notation we see that

$$[BA] = [AB]^T = [AB]^{-1} \quad ; \quad \text{also}$$

$$Q_{BA} = Q_{AB}{}^* = Q_{AB}{}^{-1} \quad . \tag{E61}$$

Finally, there are some useful tricks with the new notation we have defined. Referring to equation (E28) and adding the subscripts we have defined, we have $\dot{Q}_{BA} = 0.5\, Q_{BA} \circ \underline{\omega}_{BA}^{(B)}$. The vector $\underline{\omega}_{BA}^{(B)}$ is the angular velocity of B relative to A with components in B. Consider the quaternion Q_{CB}.

$$\begin{aligned}
\dot{Q}_{CB} &= \frac{1}{2} Q_{CB} \circ \underline{\omega}_{CB}^{(C)} = \frac{1}{2} Q_{CB} \circ \left(\underline{\omega}_{CA}^{(C)} - \underline{\omega}_{BA}^{(C)} \right) \\
&= \frac{1}{2} Q_{CB} \circ \underline{\omega}_{CA}^{(C)} - \frac{1}{2} Q_{CB} \circ \left(Q_{CB}{}^* \circ \underline{\omega}_{BA}^{(B)} \circ Q_{CB} \right) \\
&= \frac{1}{2} Q_{CB} \circ \underline{\omega}_{CA}^{(C)} - \frac{1}{2} \underline{\omega}_{BA}^{(B)} \circ Q_{CB}
\end{aligned}$$

or in matrix form

$$\dot{Q}_{CB} = \frac{1}{2}\left(\tilde{\tilde{\underline{\omega}}}^{(C)}_{CA} - \tilde{\tilde{\underline{\omega}}}^{(B)}_{BA}\right) Q_{CB} \quad . \tag{E62}$$

The utility of equation (E62) is most apparent when we use it to compute the attitude error of a spacecraft relative to a moving or moveable reference. Note that the components of $\underline{\omega}^{(C)}_{CA}$ and $\underline{\omega}^{(B)}_{BA}$ are expressed in different frames. Normally, $\underline{\omega}^{(C)}_{CA}$ would come from rate sensors which are body fixed while $\underline{\omega}^{(B)}_{BA}$ is a commanded maneuver rate which is naturally defined in the moveable reference. This equation then allows us to use both quantities directly without either being transformed.

It often becomes necessary to compute the quaternion corresponding to a given rotation matrix, i.e., find Q given [BA]. We have developed a computer algorithm to do this.

1) Define matrix

$$S = \left[\begin{array}{ccc|c} & & & A_{32} \\ & A & & A_{13} \\ & & & A_{21} \\ \hline A_{32} & -A_{13} & -A_{21} & \mathrm{tr}A \end{array}\right] \quad ;$$

A is the given rotation.

2) $S' = S + S^T + (1 - \mathrm{tr}\, A)\, 1$.

3) $I = \max S'_{ii}$ (index of largest element along diagonal of S').

4) $Q_j = S'_{Ij}/2\sqrt{S'_{II}}$.

5) $Q'_j = Q_j \,\mathrm{sgn}\, Q_4 \; ; \; \mathrm{sgn} = \begin{cases} -1 \text{ for } Q_4 < 0 \\ +1 \text{ for } Q_4 \geq 0 \end{cases}$.

Another useful and perhaps obvious technique is the expression of the quaternion resulting from a sequence of Euler rotations (rotations about coordinate axes):

$$Q_{BA} = \left(C\,\frac{\phi_1}{2} + \underline{u}_1\,S\,\frac{\phi_1}{2}\right) \circ \left(C\,\frac{\phi_2}{2} + \underline{u}_2\,S\,\frac{\phi_2}{2}\right) \circ \ldots$$

$$\circ \left(C\,\frac{\phi_n}{2} + \underline{u}_n\,S\,\frac{\phi_n}{2}\right) \quad ,$$

where $S \triangleq \sin$ and $C \triangleq \cos$. The corresponding rotation is [BA] and is given by

$$[BA] = [\phi_n]_{i_n} \cdots [\phi_2]_{i_2}\,[\phi_1]_{i_1} \quad .$$

The vectors $\underline{u}$ can be any of the three coordinate axes $[1,0,0]^T$, $[0,1,0]^T$ or $[0,0,1]^T$. If $\underline{u} = [1,0,0]^T$, then $i_1 = 1$, etc. We have added the convention that a rotation bracket with a subscript is an Euler rotation about the indicated axis. As an example consider the quaternion formed when $i_1 = 1$, $i_2 = 2$, $i_3 = 3$:

$$Q_{BA} = \left(C\,\frac{\phi_1}{2} + i\,S\,\frac{\phi_1}{2}\right) \circ \left(C\,\frac{\phi_2}{2} + j\,S\,\frac{\phi_2}{2}\right) \circ \left(C\,\frac{\phi_3}{2} + k\,S\,\frac{\phi_3}{2}\right)$$

$$= C\,\frac{\phi_1}{2}\,C\,\frac{\phi_2}{2}\,C\,\frac{\phi_3}{2} - S\,\frac{\phi_1}{2}\,S\,\frac{\phi_2}{2}\,S\,\frac{\phi_3}{2}$$

$$+ i\left(S\,\frac{\phi_1}{2}\,C\,\frac{\phi_2}{2}\,C\,\frac{\phi_3}{2} + C\,\frac{\phi_1}{2}\,S\,\frac{\phi_2}{2}\,S\,\frac{\phi_3}{2}\right)$$

$$+ j\left(C\,\frac{\phi_1}{2}\,S\,\frac{\phi_2}{2}\,C\,\frac{\phi_3}{2} - S\,\frac{\phi_1}{2}\,C\,\frac{\phi_2}{2}\,S\,\frac{\phi_3}{2}\right)$$

$$+ k\left(C\,\frac{\phi_1}{2}\,C\,\frac{\phi_2}{2}\,S\,\frac{\phi_3}{2} + S\,\frac{\phi_1}{2}\,S\,\frac{\phi_2}{2}\,C\,\frac{\phi_3}{2}\right) \quad ;$$

the corresponding [BA] is

$$\left[\begin{array}{c|c|c} C\phi_2\, C\phi_3 & S\phi_1\, S\phi_2\, C\phi_3 + C\phi_1\, S\phi_3 & -C\phi_1\, S\phi_2\, C\phi_3 + S\phi_2\, S\phi_3 \\ -C\phi_2\, S\phi_3 & -S\phi_1\, S\phi_2\, S\phi_3 + C\phi_1\, C\phi_3 & C\phi_1\, S\phi_2\, S\phi_3 + S\phi_2\, C\phi_3 \\ S\phi_2 & -S\phi_1\, C\phi_2 & C\phi_1\, C\phi_2 \end{array}\right] .$$

In this brief exposition, we have developed a number of useful quaternion results and notations. This by no means exhausts the possibilities. The available quaternion literature does not present the material in an easily applicable form and thus this short development is presented to fill that gap.

NOTES: The following notes apply to the previous discussion:

1) We use Q_4 rather than Q_0 for convenience. Since these quaternion equations will be adapted for the computer and since 0 is not usually allowed as a subscript it becomes necessary to use something else. We desire to use 1, 2, 3 for the vector components, hence Q_4 is the real part;

2) The symbol $\approx$ is called double tilde and the symbol $=$ is called double bar;

3) We shall define

$$\ddot{Q} = \begin{bmatrix} 0 & -Q_3 & Q_2 \\ Q_3 & 0 & -Q_1 \\ -Q_2 & Q_1 & 0 \end{bmatrix}$$

which is the tilde or cross product matrix for the 3-vector $\underline{Q}$.

APPENDIX F

STEREO VISUALIZATION WITHOUT OPTICAL AIDS

(Cross-eyed Stereo)

On many occasions in engineering and physical analysis it would be useful to be able to sketch in three dimensions. This would be especially helpful in the study of the rotational motion of spacecraft. To fulfill this wish in many cases, we have used a convenient technique which requires no optical devices other than one's eyes and some graph paper. All that is required is a stereoscopic pair of images. One additional capability is necessary. The observer must be able to cause the lines of sight of his eyes to converge; i.e., one must cross one's eyes. The stereo projections are formed as shown in Figure F1. The images are reversed and viewed as in Figure F2. With a little practice, one can easily learn to reconstruct mentally, the 3-dimensional scene from the reversed stereo pairs. In Appendix G there are several stereo views of the various equilibrium attitudes for the Skylab vehicle. The Skylab is represented as an idealized "wire-frame" model which is transparent. The interested reader should try several viewing distances (the farther away the page the less crossing of the eyes is required and the easier it becomes to focus the images). Squinting may also help as it increases one's depth of focus. When one first looks at a stereo pair, one focuses on the page and sees two similar but separate images. As one begins to cross his eyes, the two images become four. Continue crossing the eyes until the interior pair of images come together. Since the line connecting corresponding points on the images must be at the same angle about the line of sight as the line connecting the eyes, it may be necessary to rotate the page or rock the head until these two images become superimposed and seem to merge into a stereo image. This technique does require some practice but once mastered it can be very useful for easy visualization in 3 dimensions.

If a computer with plot capability is available to us, we can construct the necessary stereo projections from a set of points and lines that represent the object of interest. We have referred to such a representation as a wire frame model because of the appearance of the image. Let P be a representative point of the model. Each point P is projected into the picture plane S as shown in Figure F3. The point P is projected to the eyepoint E and the line PE intersects the picture plane S at P'. P' is the projection of P onto S. The set of all points P' projected from object points P together with the connecting lines form the desired projection. We set up a reference frame in the plane S. To do this, we must specify which way is up (so to speak). Let $\underline{u}_u$ be a unit vector in this direction and $\underline{u}_r = \underline{u}_u \times \underline{\ell}$ is a unit vector in S pointing to the right. We place the origin of the S coordinate reference at O. Observe that $\underline{r}_o = \underline{r}_E + d\,\underline{\ell}$; where $\underline{r}_o$ and $\underline{r}_E$ are position vectors of O and E respectively. From the geometry shown in Figure F3, we can see that

$$\underline{r}_{P'} = \underline{r}_E + (\underline{r}_P - \underline{r}_E)\, d/\underline{\ell} \cdot (\underline{r}_P - \underline{r}_E) \quad .$$

From this we can compute

$$x_{P'} = (\underline{r}_{P'} - \underline{r}_O) \cdot \underline{u}_r$$

$$y_{P'} = (\underline{r}_{P'} - \underline{r}_O) \cdot \underline{u}_u \quad .$$

Since $\underline{\ell} \cdot \underline{u}_u = \underline{\ell} \cdot \underline{u}_r = 0$,

$$x_{P'} = (\underline{r}_{P'} - \underline{r}_E) \cdot \underline{u}_r$$

$$y_{P'} = (\underline{r}_{P'} - \underline{r}_E) \cdot \underline{u}_u \quad .$$

The set of points $(x_{P'}, y_{P'})$ plotted conventionally forms the desired projection. Size can be altered by scale adjustments. These projections are then placed as desired. Also, the values used for d and eye separation s are arbitrary and can be adjusted for convenience or eye comfort. In real life $s \doteq 65$ mm and $d \doteq 250$ mm for comfortable reading; however, it may be more comfortable for d to be larger. Some initial experimentation with this technique should establish desireable settings.

We have developed some APL software to produce such images. This software was used on our Sigma V computer and is presented for the benefit of the interested reader with a background in APL (See listing at end of Appendix H.).

The software consists of three functions: 1. STSETUP; 2. STROT; and STEREO. STSETUP allows the user to specify the various parameters for the observer such as d, s, $\underline{\ell}$, etc. STROT allows the user to rotate the line of sight and the observer position to view the object from a different angle without changing the viewing distance or eye separation. STEREO is the function which takes a set of points connected by straight lines and forms the stereoscopic projections. Usually these functions are driven by another function which is set up to draw the object of interest. A function called SKYLAB does this for us in this case.

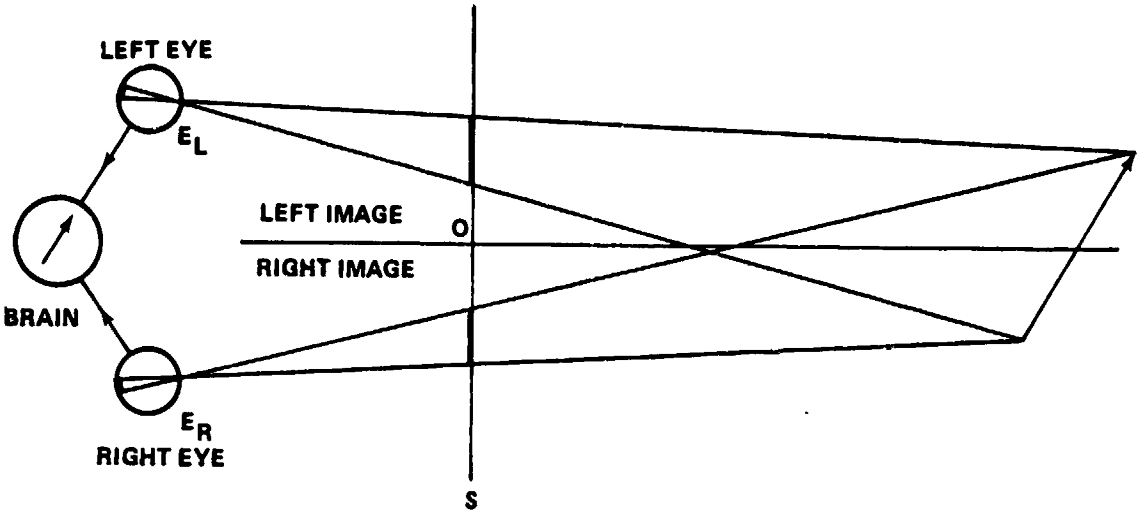


Figure F1. Stereo projection.

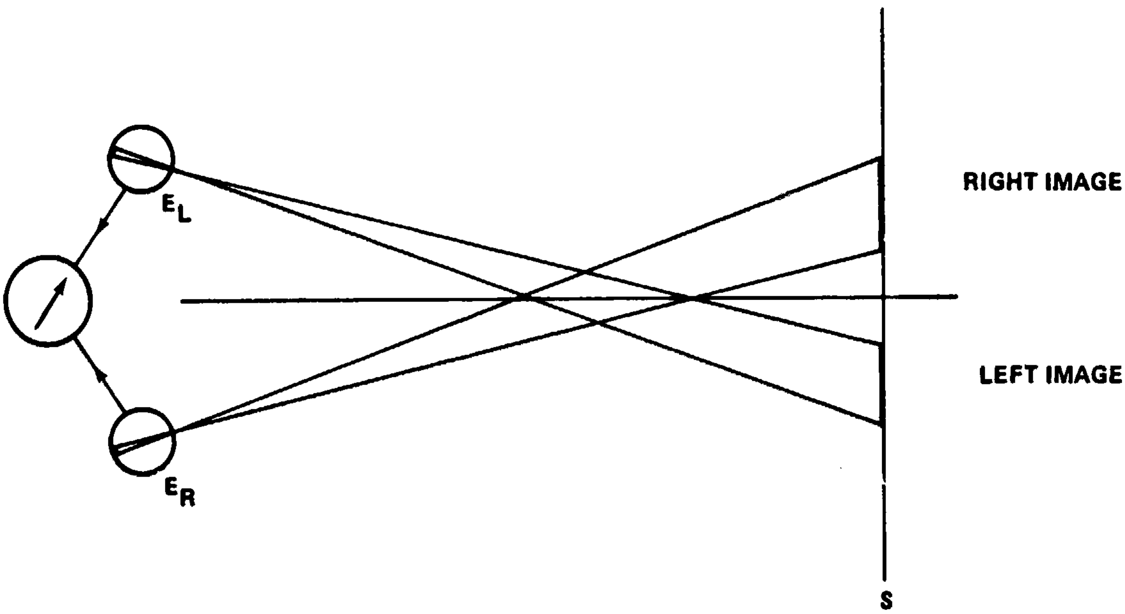


Figure F2. Stereo reconstruction by cross-eyed viewing.

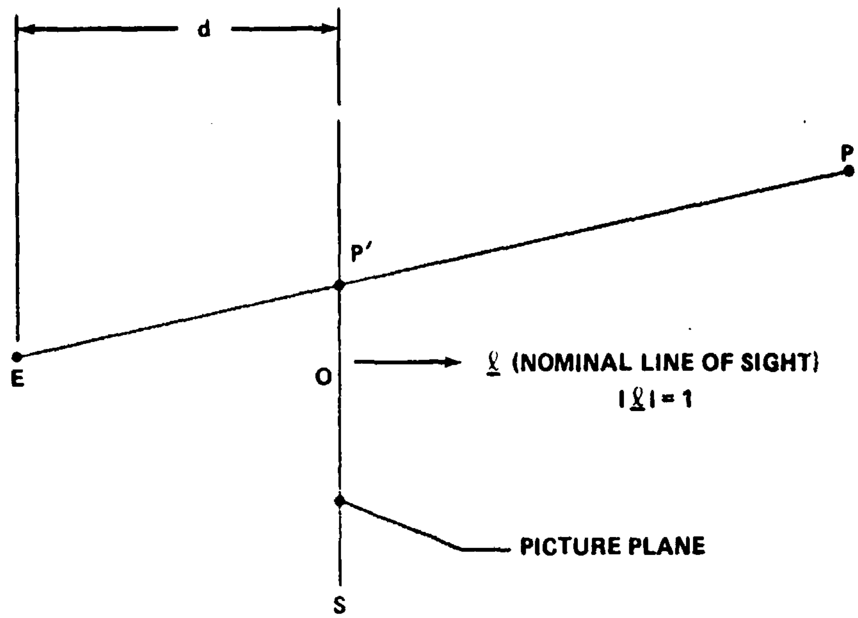


Figure F3. Projection geometry.

APPENDIX G

COMPLETE TEA SET

(Cross-eyed Stereo Doubles)

Figures G1 through G12 show all the available TEA's for an altitude of about 200 km (108 n.mi.). The nomenclature is derived from the general directions of the vehicle X axis (first letter) and the vehicle Z axis (second letter) with respect to the L system.

The axes can point

N north

S south

U up

D down

F forward

B backward

The top set of 3 × 3 numbers[7] in each figure is the direction cosine matrix from the L system to the A system. Further the BAR angles and the χ's are given. The vehicle Z axis beta is the elevation angle of the vehicle Z axis with respect to the orbital plane.

The point of view is from behind the vehicle, slightly south of the orbital plane. The axes shown are these of the L system. The Y axis points to the north, the Z axis points up, and the X axis points forward (into the paper plane). The vehicle is moving from the lower left to the upper right.

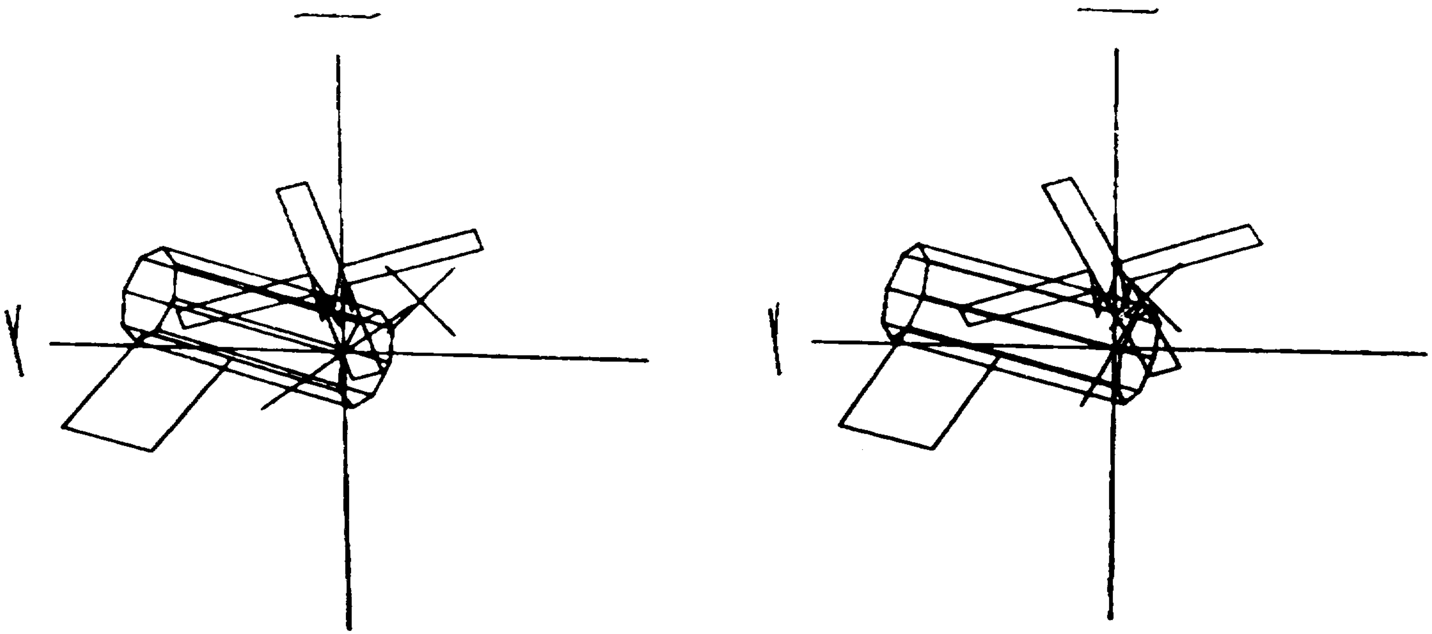
---

7. In APL computer language the minus sign is a superscript minus (to distinguish it from the subtraction operation).

ALSU

¯0.275404	¯0.942404	¯0.189811
0.788208	¯0.334398	0.516629
¯0.550346	¯0.00732893	0.834905

BANK, ATTACK, ROLL ANGLE ¯78.6123 105.986 124.924
CHIX, CHIY, CHIZ 178.744 145.425 ¯78.4592
VEH Z-AXIS BETA ¯0.41992



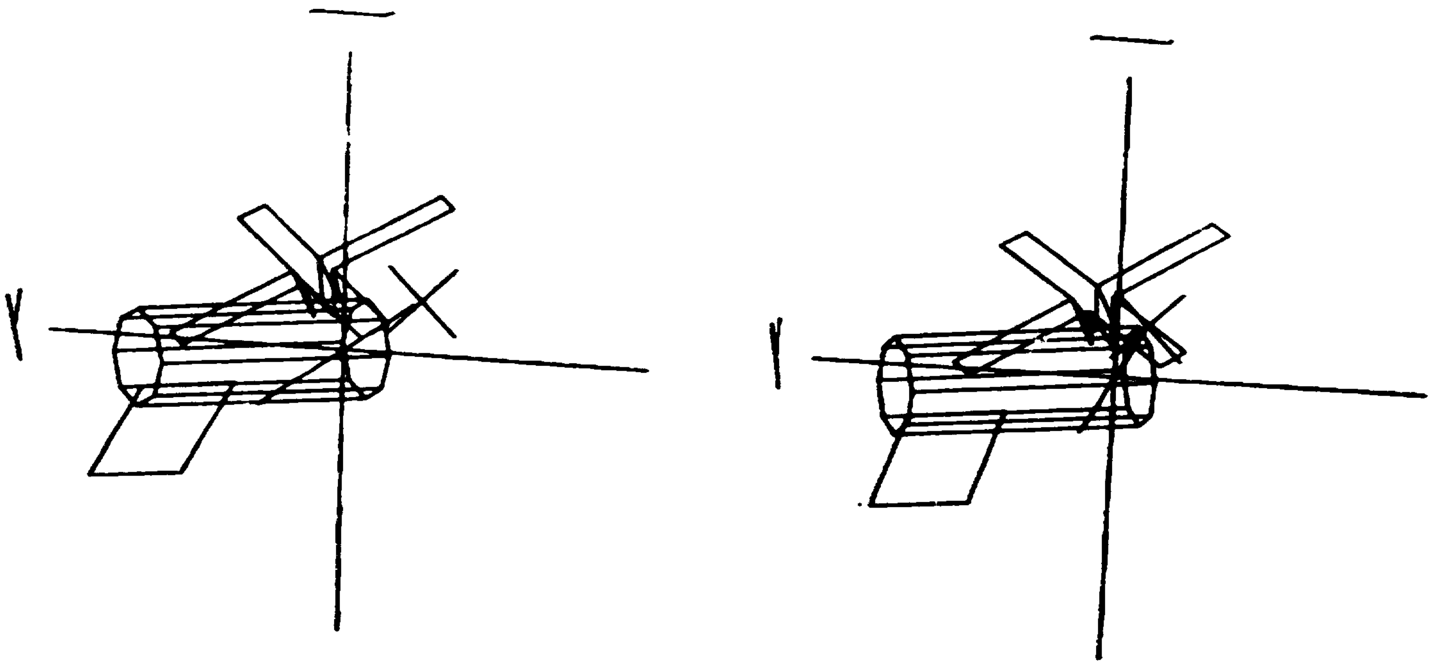
a. No momentum.

Figure G1. TEA - SU or T121P.

ALSUZM1

-0.194598	-0.97163	0.134413
0.870013	-0.107689	0.481125
-0.453001	0.210567	0.866286

BANK, ATTACK, ROLL ANGLE -97.8762 181.221 117.505
CHIX, CHIY, CHIZ -117.086 -145.366 -76.3195
VEH Z-AXIS BETA 12.1556



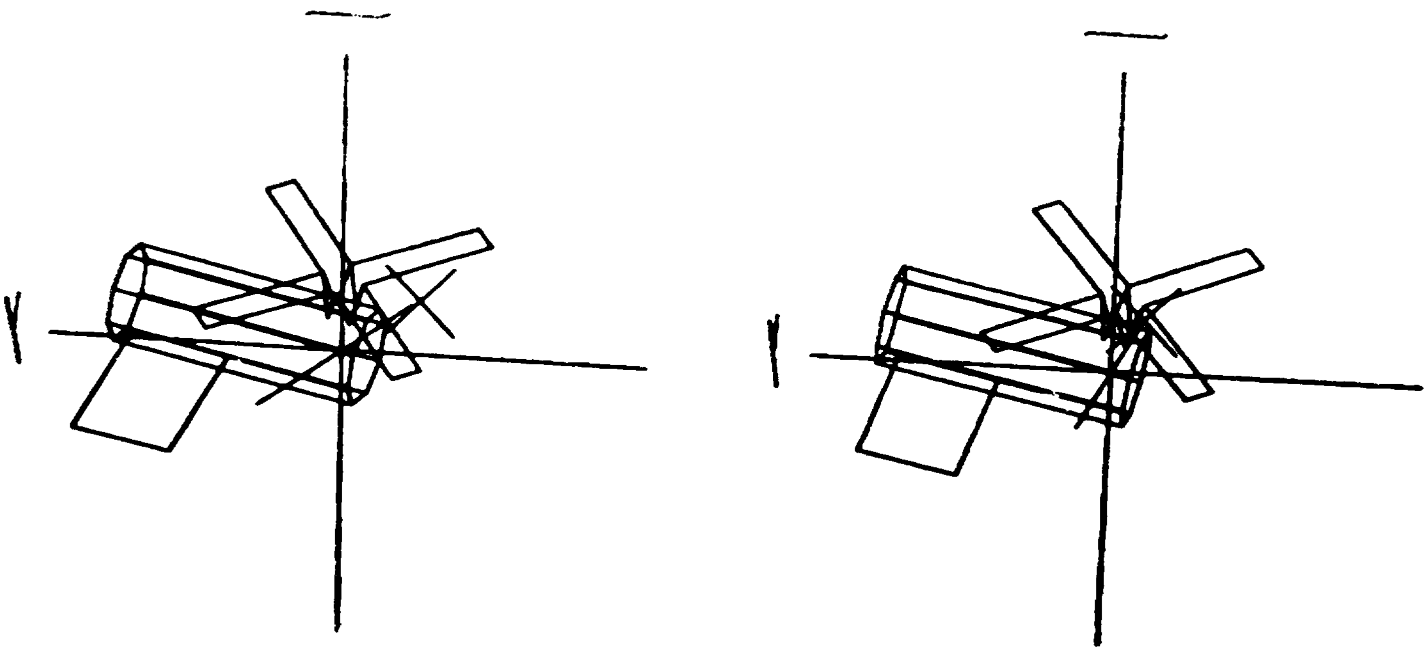
b. -1H momentum in Z axis.

Figure G1. (Continued)

ALSUPPS

```
-0.0545167   -0.975970   -0.196544
-0.036035    -0.151579    0.324052
-0.542886    -0.136141    0.028698

BANK, ATTACK, ROLL ANGLE  -72.648   93.1251   122.936
CHIX, CHIY, CHIZ    138.126   115.593   -75.2311
VEH Z-AXIS BETA   -7.8246
```



c. +0.5H momentum in X axis.

Figure G1. (Concluded)

ALND

¯0.275484	0.942484	¯0.189811
¯0.788288	0.334398	¯0.516629
¯0.558346	0.00732093	¯0.834905

BANK, ATTACK, ROLL ANGLE 101.388 105.906 124.924
CHIX, CHIY, CHIZ ¯1.25554 ¯145.425 70.4592
VEH Z-AXIS BETA 0.41992

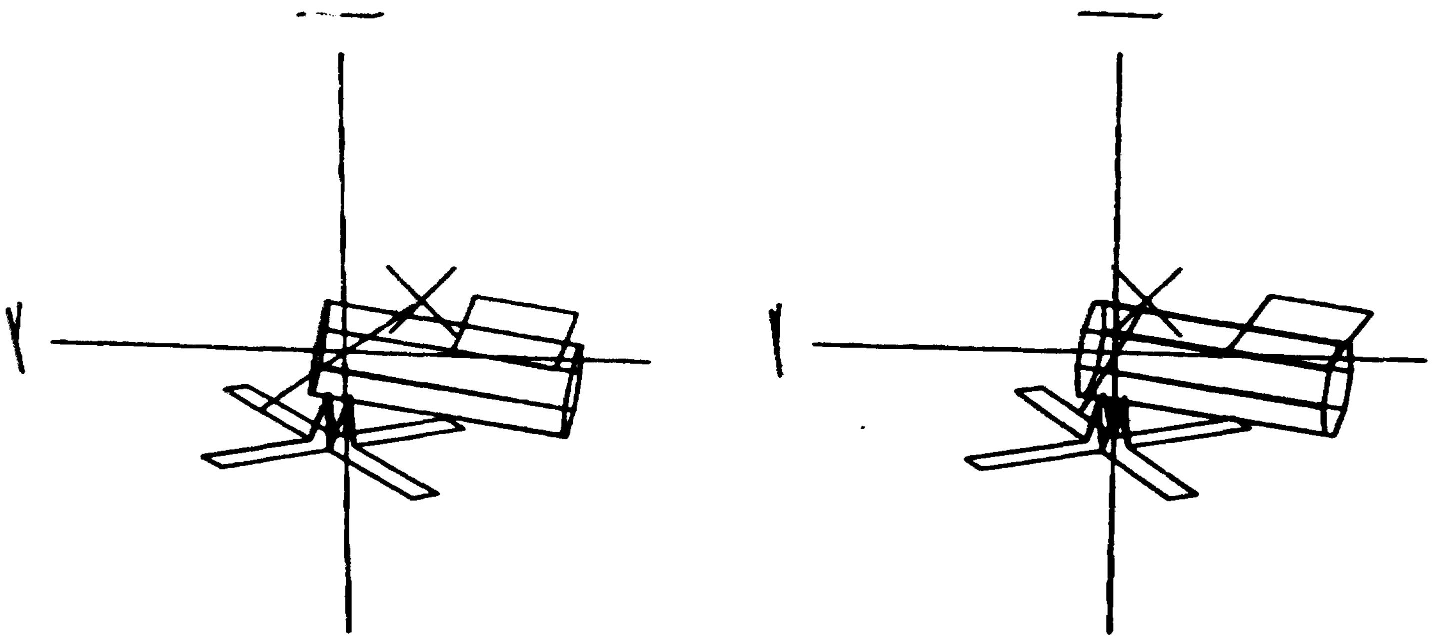


Figure G2. TEA - ND.

ALUN

¯0.0668563	¯0.117329	0.99084
¯0.82704	0.549007	0.120814
¯0.558153	0.827542	0.0603313

BANK, ATTACK, ROLL ANGLE ¯173.247 93.8334 124.015
CHIX, CHIY, CHIZ ¯56.4389 ¯93.8601 ¯6.73799
VEH Z-AXIS BETA 55.847

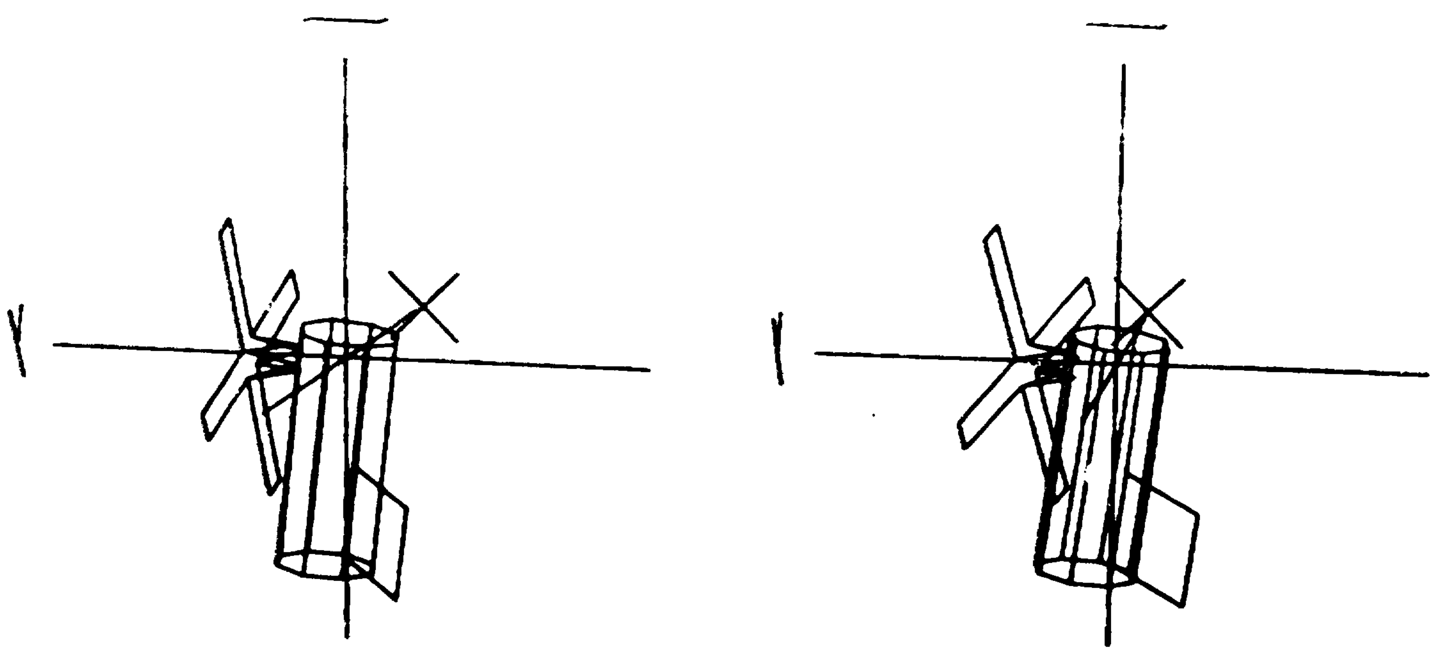


Figure G3. TEA - UN or T121G.

ALDS

-0.0668563	0.117329	-0.99004
0.02704	-0.549007	-0.120814
-0.550153	-0.027542	-0.0603313

BANK, ATTACK, ROLL ANGLE 6.75317 93.0334 124.015
CHIX, CHIY, CHIZ 123.561 93.0601 6.73799
VEH Z-AXIS BETA -55.047

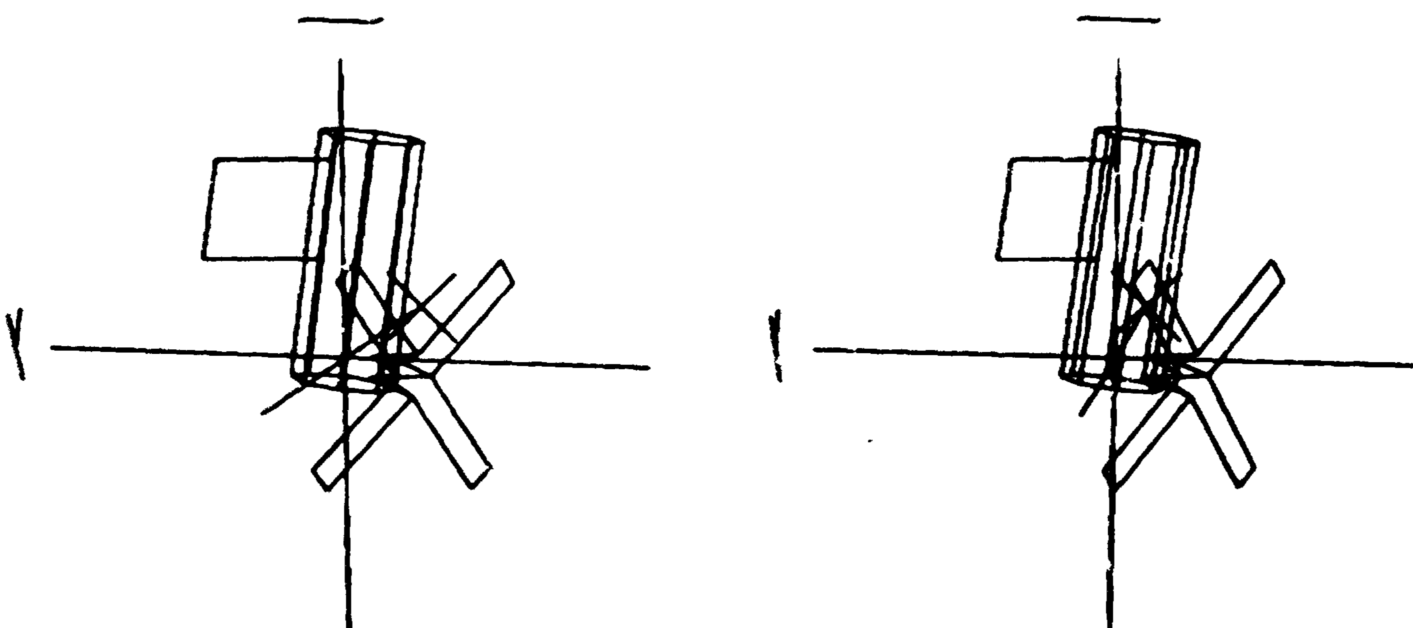


Figure G4. TEA - DS.

ALBU

¯0.846361	0.501074	¯0.18055
¯0.532552	¯0.791181	0.3007
0.0078248	0.350653	0.936473

BANK, ATTACK, ROLL ANGLE 70.1846 147.818 ¯89.1582
CHIX, CHIY, CHIZ ¯156.097 167.958 30.0711
VEH Z-AXIS BETA 20.5273

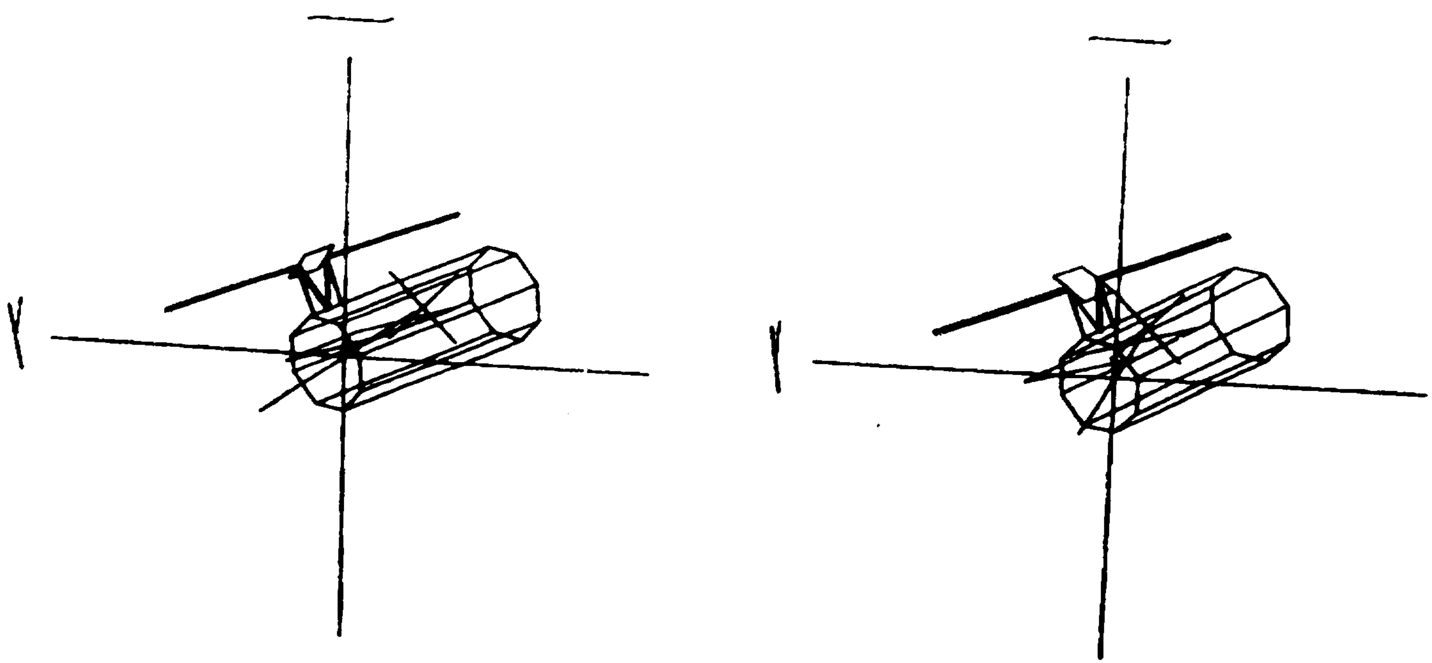


Figure G5. TEA - BU or T275.

ALBD

```
¯0.846361   ¯0.501074    0.18055
¯0.532552    0.791181   ¯0.3007
 0.0078248  ¯0.350653   ¯0.936473

BANK, ATTACK, ROLL ANGLE  ¯109.815  147.818  ¯89.1582
CHIX, CHIY, CHIZ    23.903  ¯167.958  ¯30.0711
VEH Z-AXIS BETA  ¯20.5273
```

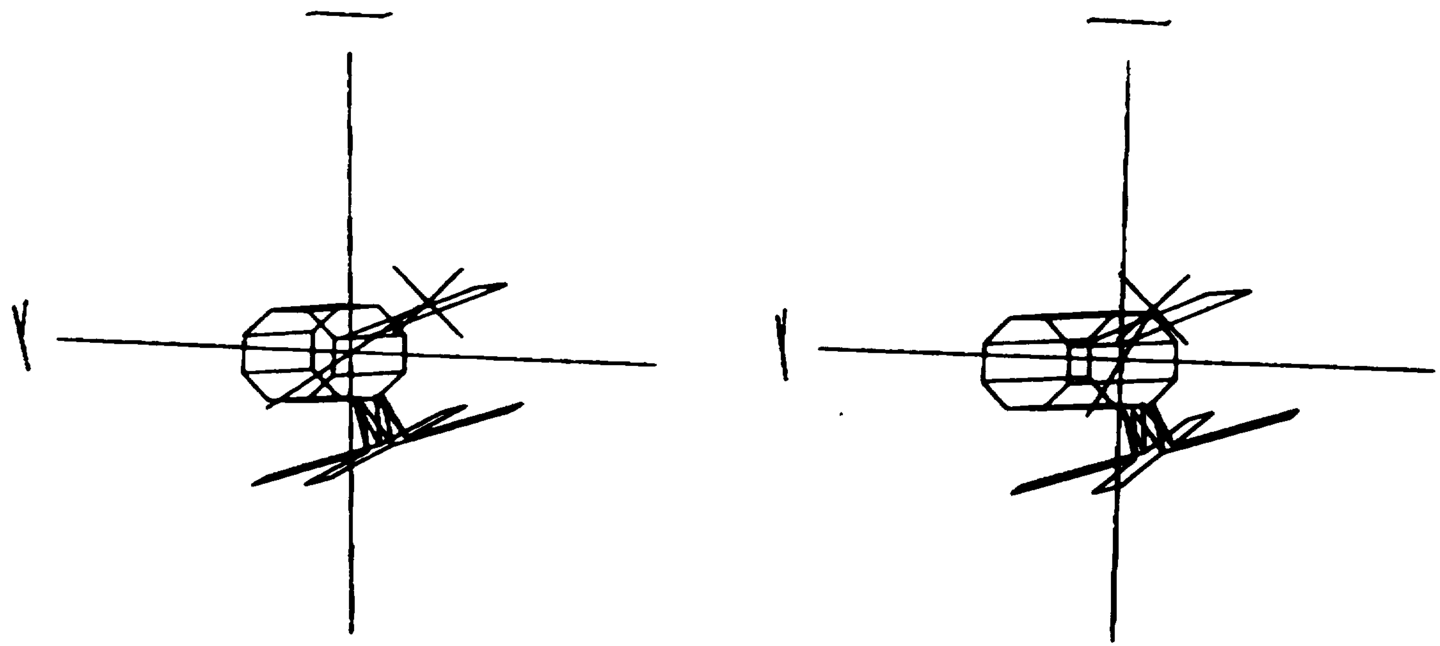


Figure G6. TEA - BD.

A!.BN

¯0.978475	¯0.161294	¯0.12873
¯0.182825	0.388182	0.903266
¯0.0957202	0.907358	¯0.409315

BANK, ATTACK, ROLL ANGLE ¯51.4062 168.09 ¯117.635
CHIX, CHIY, CHIZ ¯66.838 172.505 ¯9.28199
VEH Z-AXIS BETA 65.1428

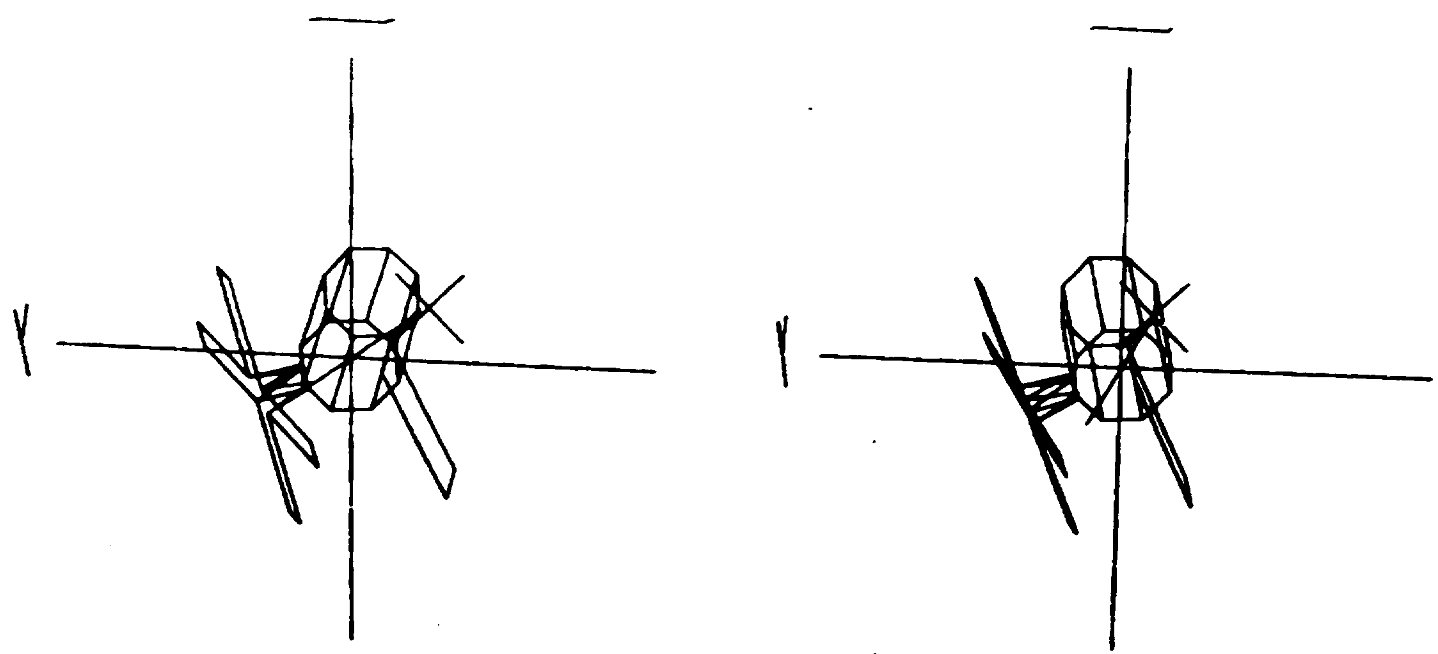


Figure G7. TEA - BN.

```
                  ALBS

  ¯0.978475     0.161294     0.12873
  ¯0.182825    ¯0.388182    ¯0.903266
  ¯0.0957202   ¯0.987358     0.409315

BANK, ATTACK, ROLL ANGLE  128.594  168.09  ¯117.635
CHIX, CHIY, CHIZ    113.162  ¯172.505  9.28199
VEH Z-AXIS BETA   ¯65.1428
```

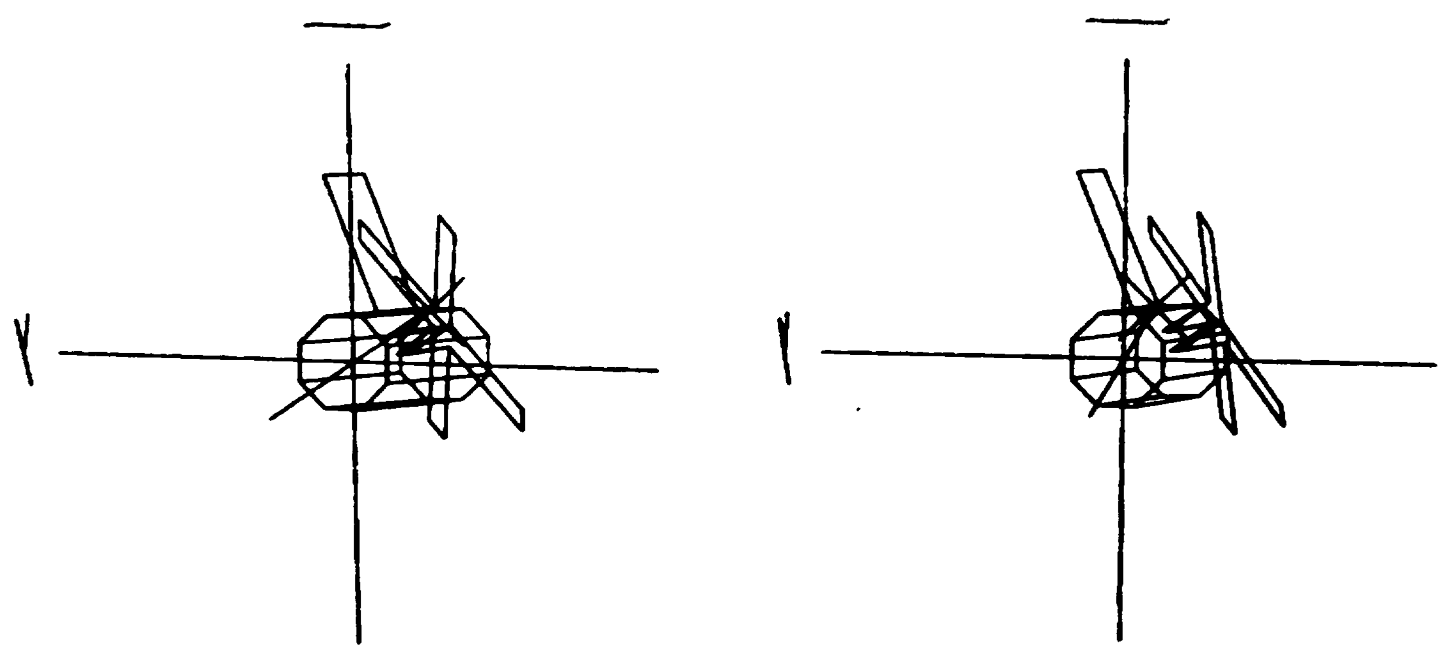


Figure G8. TEA - BS.

ALDN

¯0.351053	¯0.193692	¯0.916103
¯0.816755	0.541789	¯0.198432
0.4579	0.817892	¯0.348396

```
BANK, ATTACK, ROLL ANGLE  ¯11.9382  110.552  ¯60.7236
CHIX, CHIY, CHIZ     ¯56.4787  110.967  ¯11.1683
VEH Z-AXIS BETA  54.8744
```

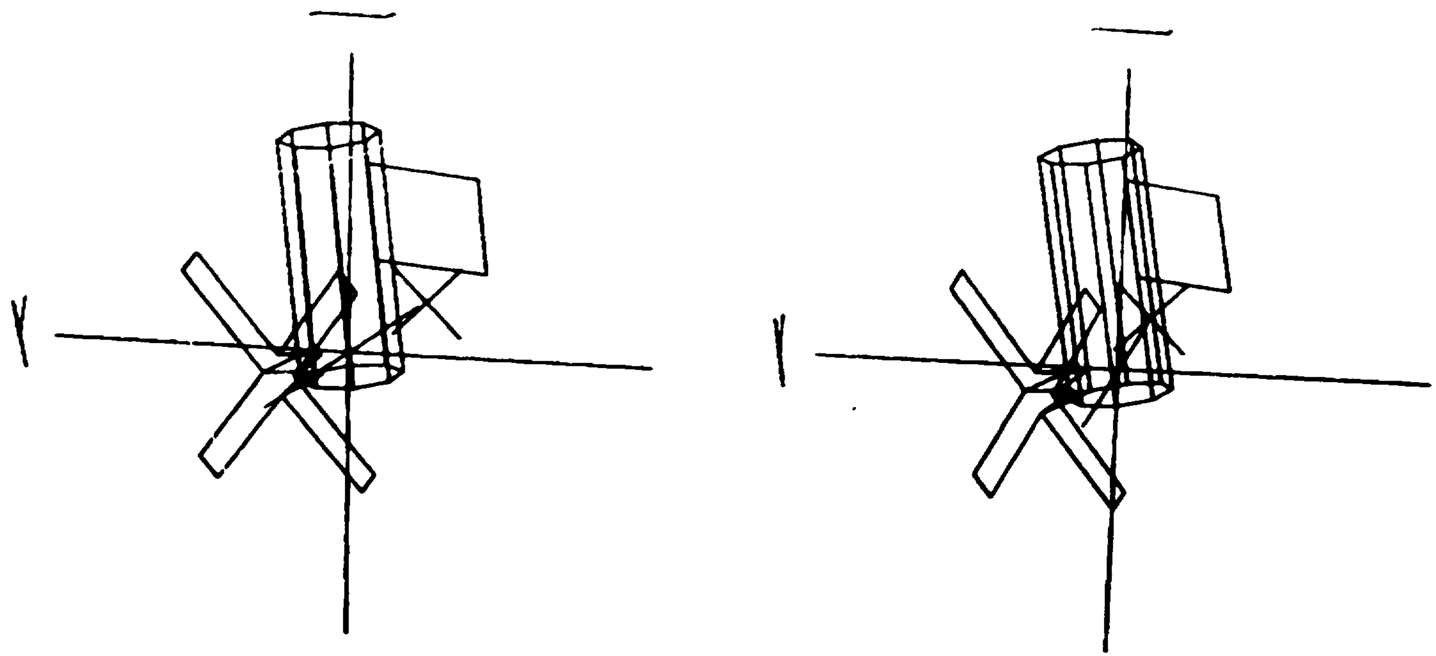


Figure G9. TEA - DN.

ALUS

¯0.351053	0.193692	0.916103
¯0.816755	¯0.541789	¯0.198432
0.4579	¯0.817892	0.348396

BANK, ATTACK, ROLL ANGLE 168.062 110.552 ¯60.7236
CHIX, CHIY, CHIZ 123.521 ¯110.967 11.1683
VEH Z-AXIS BETA ¯54.8744

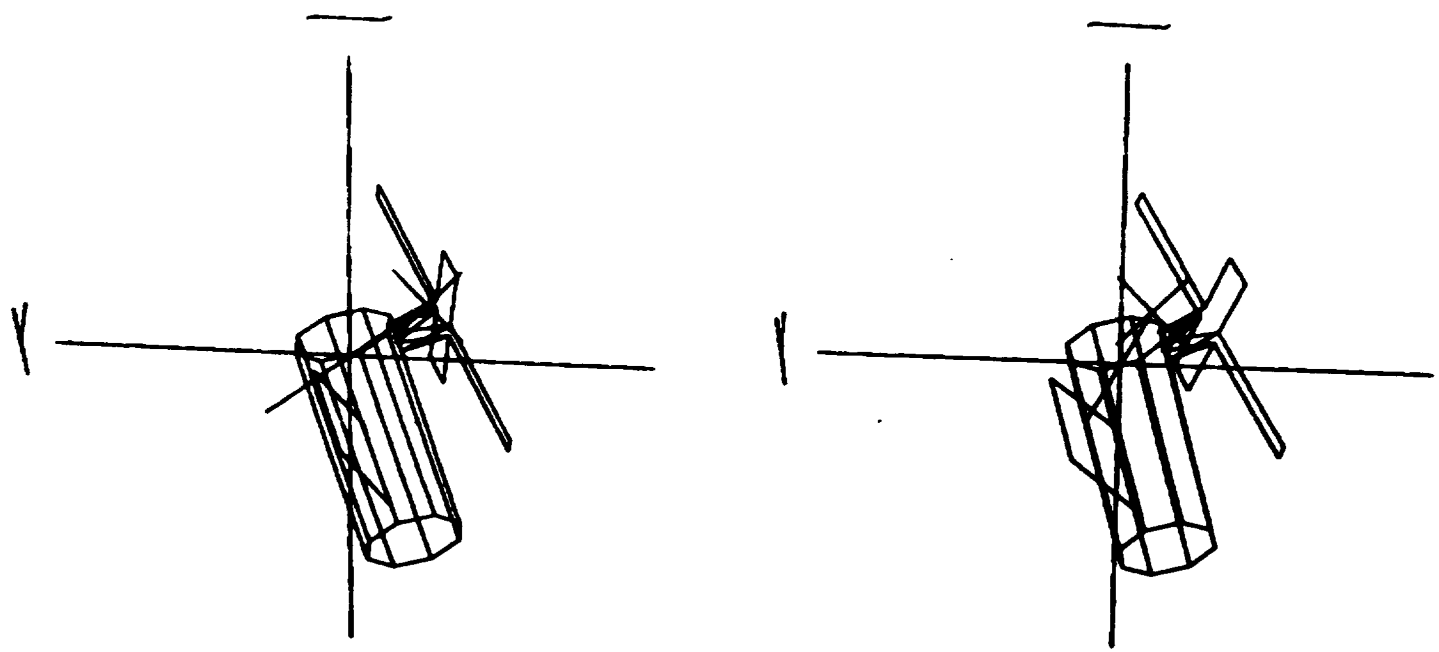


Figure G10. TEA - US.

```
                  ALNU

 0.312325      0.94811      ¯0.0595029
¯0.740696      0.282261      0.609671
 0.594831     ¯0.146342      0.790418

BANK, ATTACK, ROLL ANGLE  86.4089  71.8006  ¯51.2331
CHIX, CHIY, CHIZ    27.405  10.7865  71.4615
VEH Z-AXIS BETA   ¯8.41498
```

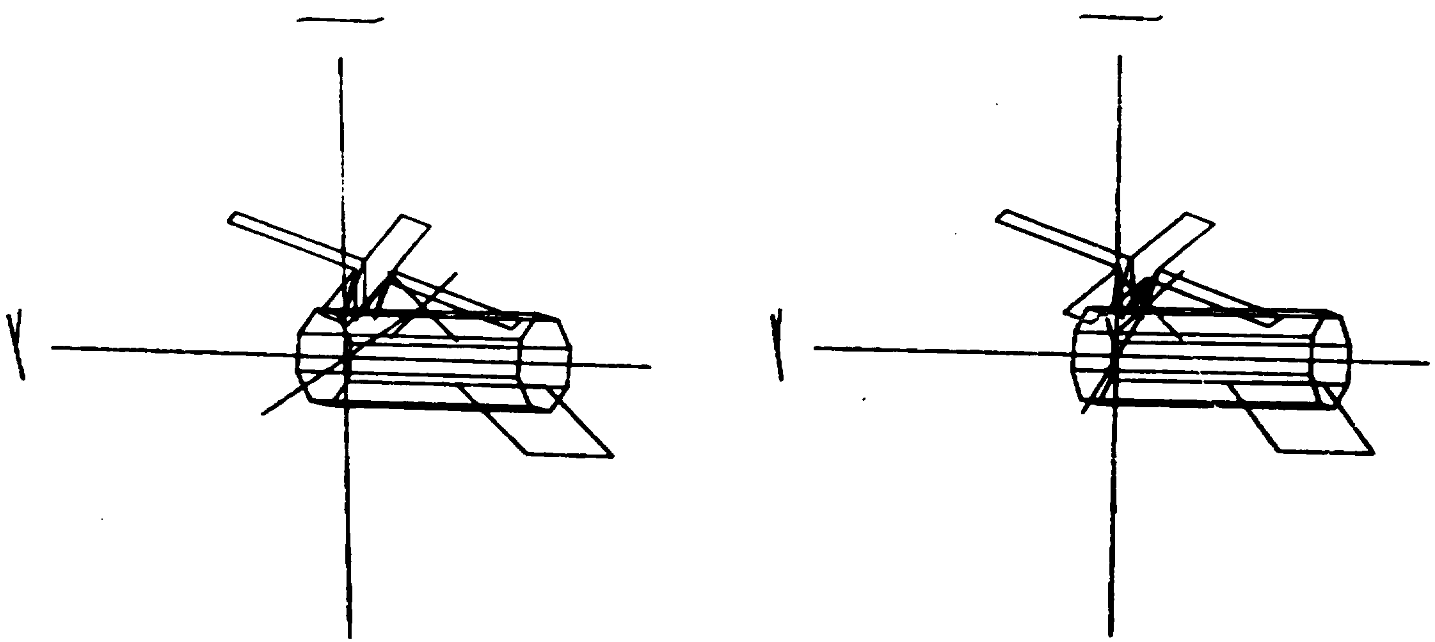


Figure G11. TEA - NU.

ALSD

```
 0.312325    ¯0.94811     0.0595029
¯0.740696    ¯0.202261   ¯0.609671
 0.594031     0.146342   ¯0.790410

BANK, ATTACK, ROLL ANGLE  ¯93.5911  71.0006  ¯51.2331
CHIX, CHIY, CHIZ    ¯152.595  ¯10.7065  ¯71.4615
VEN Z-AXIS BETA  0.41490
```

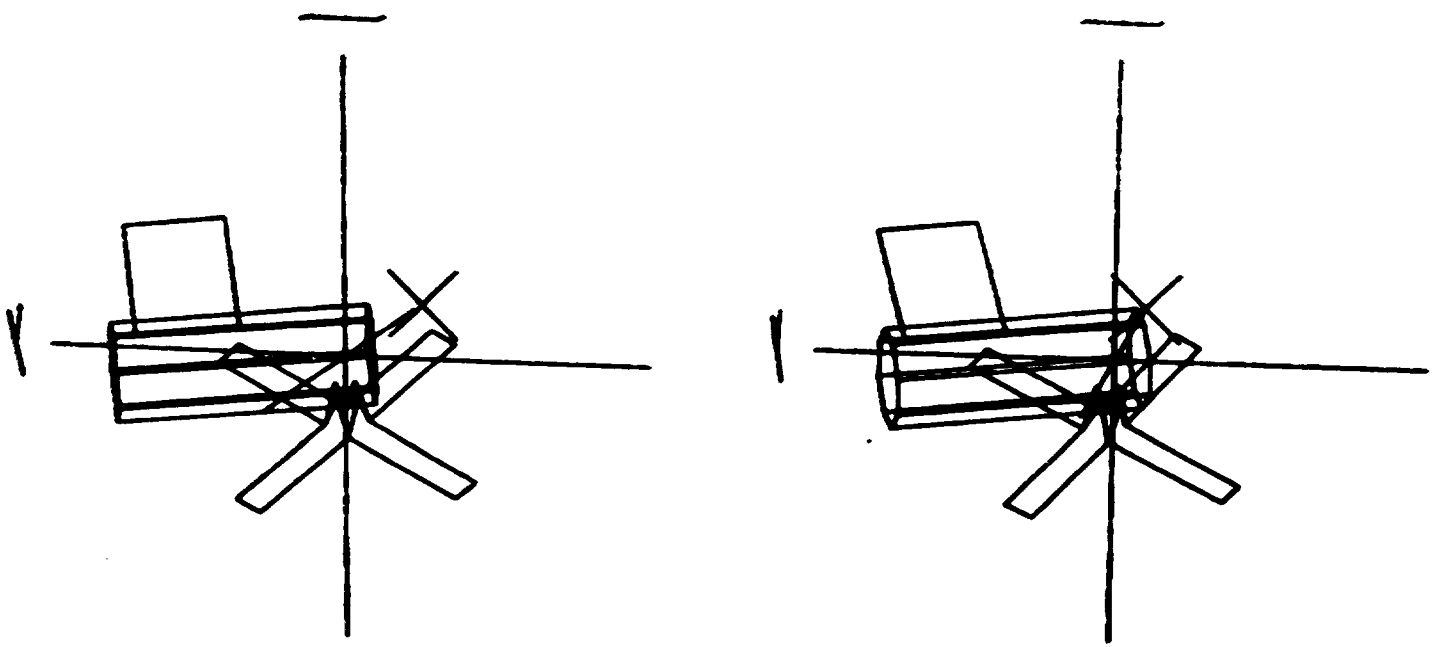


Figure G12. TEA - SD.

APPENDIX H

TEA SIMULATION AND GROUND SUPPORT

To assess the feasibility and refine the techniques of TEA control, we had to develop a simulation. Simulation tools already existed to some extent as a result of our work on EOVV control. These were modified and extended for TEA. The details of the theoretical background for most of the development of TEA control are presented elsewhere. We shall concentrate on a few additional areas.

IBM also had Skylab simulations left over from original Skylab days. These were used to verify the control schemes we developed and also to verify the resulting new flight computer code. In addition, they were used to confirm our ground procedures and to predict TACS propellant usage [2].

The computer and the programming language are of fundamental importance to the success of this effort. The computer used was a Xerox/Honeywell Sigma V. We were most fortunate to have ready access to this machine together with timesharing hardware and software. The language used was APL, a high order, interpretive language which offers high computing power together with a relatively simple syntax. Since the language operates on the tensor or array level, it is a highly compressed language, allowing one or two lines of APL code to be equivalent to dozens of lines of Fortran. TEA control and EOVV control could not have been successfully accomplished in such a short time and low manpower effort without the efficient use of APL and the Sigma V. As an example, we were able to assemble virtually from scratch within one week a complete dynamic simulation of Skylab EOVV with 3-axis rotational dynamics and CMG control.

For TEA we modified the Skylab EOVV simulation primarily by redefining the momentum desaturation technique. Since we had to simulate many orbits of TEA operation to assess its behavior, we had to adopt a somewhat modified approach. To speed running time of the simulation functions, it was desireable to lengthen the integration step being used by an order of magnitude. This was done by sacrificing the body rotational dynamics. A slow running version of the TEA simulation was maintained with rotational dynamics in it to spot check this fast version but was not used much because it proved too slow. We assumed the vehicle control was perfect in maintaining the vehicle at its commanded attitude and that small maneuvers were made instantaneously. These changes in attitude were made at the beginning of the integration period and afterwards only gyro drift caused attitude changes relative to the local vertical reference. To improve accuracy the vehicle torque was computed twice during each cycle: at the beginning and at the end of each step. The attitude maneuver made to desaturate the angular momentum was assumed to be made at the beginning of each step. The torque was calculated for the new attitude at the beginning. Since the position may have changed at the end of the interval it is calculated again and averaged. With this technique we were able to obtain good results using a 300-sec integration step which was also the TEA desaturation interval. This allowed us to simulate 1 day of orbits in about 5 min, if we had priority demand on machine time. This was provided in the month prior to initiating TEA control and allowed us to develop and refine our operating procedures including those used to initiate TEA control.

In addition to the simulation, we developed many analysis tools. As can be noted by examining the listings at the end of this appendix, there are many functions including simulation functions, support functions, plotting functions and auxilliary functions. This set constitutes the library that was used in developing the TEA control schemes and in maintaining TEA control once it was established. Various subsets of these functions were used, depending on the application.

Attitude dynamics in the slow simulation was simulated using the Newton-Euler rigid body rotational equations

$$\dot{H}_{Tk} = \underline{T}_{Ek} - \underline{\omega}_{Vk} \times \underline{H}_{Tk} \quad .$$

The total system angular momentum $\underline{H}_{Tk}$ was made from the sum of vehicle and CMG momentum. The CMG momentum was computed from the gimbal angles which were determined by integration of the rate commands coming from the CMG steering law of Reference 5. These rate commands were assumed to be provided by the CMG gimbal dynamics except for gimbal rate limits and stops on the angles. The vehicle position was determined by integrating the vehicle angular rate. This rate was obtained from the vehicle angular momentum by use of the inverse moment of inertia tensor or in dyadic form,

$$\underline{\omega}_{Vk} = \underline{\underline{I}}_{Vk}^{-1} \cdot (\underline{H}_{Tk} - \underline{H}_{CMGk}) \quad .$$

In the fast simulation, measured body rate was assumed controlled to orbital rate and the true rate $\underline{\omega}_{Vk}$ was then determined by subtracting the gyro drift and dividing by the scale factors as in

$$\underline{\omega}_{Vk} = \underline{\underline{F}}_{Sk}^{-1} \cdot (\underline{\omega}_o - \underline{\omega}_{Dk}) \quad .$$

The CMG momentum was then

$$\underline{H}_{CMGk} = \underline{H}_{Tk} - \underline{\underline{I}}_{Vk} \cdot \underline{\omega}_{Vk} \quad .$$

The total torque from external sources, $\underline{T}_{Ek}$, is made up of aerodynamic and gravity gradient torques as developed in previous appendices. The position variables used were quaternions as explained in Appendix E.

After the TEA control was developed but before it was initiated in flight, the simulation was modified to convert it to a tool for the training of ourselves and the JSC and MSFC flight controllers who would be interfacing with the vehicle when the time came. We were able to run many practice sessions prior to actual TEA initiation. We had not been able to do this for EOVV and paid the price for this on several occasions. As a result of these training and practice sessions, we learned much on how to maintain TEA even in the face of large simulated errors between data in the flight computer models and the real vehicle. As a result we put ourselves in a much more favorable position to enhance the probability of successful initiation of TEA control. We learned how to prebias the rate gyros so that the strapdown would intentionally drift in a favorable direction. This gain in experience was crucial to overall success.

On the following pages, the simulation functions are listed. The first four (∆CR, ∆FMT, ∆GRF, ∆WM) are proprietary and cannot be listed. Only ∆GRF and ∆FMT are used by the simulation to support graphics and data formatting. ∆CR and ∆WM are used only for listing all functions in an APL workspace and are not otherwise required.

```
      DATE

    WORKSPACE GLAESE  TEASUP
     TIME= 10   39.7 DATE= 11   9   79


      )FNS
ΔCR ΔFMT    ΔGRF    ΔWM X   ADEG    AECONST AETORQUE    AREA
AFQ ALTFDOY ANYROOT ATAN    ATMDC   AXES    BAIRSTOW    BAR
BARANGS BARAV   BARS    BOX BPRBANK CHI CHRPOL  CP  CPA CPASUP
DΔE DA  DARROW  DATE    DBANK   DBANKPAR    DBANKSUP    DBAR
DBARA   DBARAPAR    DBARASUP    DCD DDDDER  DECMG   DEGA    DEGQ
DELG    DENS    DESAT   DET DFDOY   DFHEX   DFOCT   DFS DH
DOYFALT DPH DPWR    DPWRA   DR  DRAW    DRAWN   DRAWNS  DRIFT
DRO DTQ EAFA    EDITQBL EIGVEC  ENC ERASE   EUL EUL1    EUL2
EUL3    FCF FILLTABS    FINDQR  FIX FORMAT8 FPWR    GACCEL  GBP
GBPA    GBPASUP GETSLOPE    GGTORQUE    HBARS   HELP    HESSEN
HEXFD   HMARK   HOME    IC  IEGEN   JACOBIAN    LIMC    LIMM
LOOP    LSTALL  MAG MARK    NAVUP   NOM NORM    NRMLIZ  OCTFD
ORBMARK OUTPUT  PAR PFR POWER   PRIC    PUT PWRANGLE    QBLCHECK
QBLSET  QBLSETSH    QDB QDEG    QDT QFA QIN QRALG   QUAD    QUADM
QUANTIZE    RANIC   RANPAR  RDTAPE  REF REFRESH REPRIC  RUN RUNON
SCALE   SEEK    SETSTATIONS SETUP   SFD SKYLAB  SLUPDATE    SPIN
SPLIT   SSANGLE SSANGLE0    STAXES  STEREO  STPAR   STRAPIS STROT
STSETUP TACS    TEAEVAL TEXPERT TOP TRANSLATE   TRQ UPDATE  VS
WHATSCALE   WHATWINDOW  WINDOW  X   XA  XH  XPLOT


      ∇LSTALL[⎕]∇
    ∇ LSTALL NLP;I;FNS;FN;ΔWM;ΔFMT;ΔCR;NLR;NLF;NFL;ΔAV
[1]   ⍝  LISTS AUTOMATICALLY ALL UNLOCKED FUNCTIONS IN A WORKSPACE
[2]   ⍝  ALLOWING NLP LINES OF LISTING ON EACH PAGE.
[3]   SΔLSTALL←⍳0;SΔLSTALL←2 2⍴L5,35,L5,4
[4]   FNS←3 ΔWM 6;ΔWM←14⍕6;ΔFMT←14⍕0;ΔCR←14⍕5
[5]   L:NLR←NLP
[6]   L3:I←FNS⍳2⍕21
[7]   →(1≥I)/0
[8]   FN←FNS[⍳I-1];NLF←0
[9]   NLF←3++/(2⍕13)=1 ΔCR FN
[10]  →(NLF>NLR)/M
[11]  2⍕13
[12]  ⍎'∇',FN,'[⎕]∇'
[13]  L5:FNS←I↓FNS
[14]  →L3;NLR←NLR-NLF-1
[15]  M:'     ',2⍕22
[16]  →L;⍞
    ∇
      LSTALL 50
```

```
     ∇ C←A X B
[1]    ⍝ NORMALIZED CROSS PRODUCT
[2]    C←(A[2 3 1]×B[3 1 2])-A[3 1 2]×B[2 3 1]
[3]    C←C÷(+/C×C)*0.5
     ∇

     ∇ R←ADEG A;HANG;Q;S
[1]    ⍝  MAKES DIRECTION COSINES OUT OF EIGENANGLE IN DEGREES.
[2]    Q←4⍴0
[3]    Q[4]←2○HANG←0.5×RFD×MAG A
[4]    S←1○HANG
[5]    Q[1 2 3]←S×A÷MAG A
[6]    R←AFQ Q
     ∇

     ∇ AECONST
[1]    ⍝  COMPUTES AERODYNAMIC CONSTANTS REQUIRED FOR AERO TORQUE CO
       MPUTATIONS.
[2]    KTAE←0.5×79.46×10.06
[3]    DB←DENS ALT←RMAG-REA
[4]    →(IBM=1)/L
[5]    RHOAV←DB[1];KBULGE←DB[2]×SEEKFL=0
[6]    →0
[7]    L:RHOAV←RHOBASE×1+0.5×AMP×2*3÷2
     ∇

     ∇ AETORQUE;U;CMA;CNB;SNB
[1]    ⍝  COMPUTES AERODYNAMIC TORQUE FROM RELATIVE VELOCITY AND ALT
       ITUDE.
[2]    VAE←1000×VEK+.×RD-OMGEK X R
[3]    U←VAE÷MAG VAE
[4]    AANG←¯2○U[1]
[5]    RANG←U[2]ATAN U[3]
[6]    CMA←2○AANG×0,⍳10
[7]    CNB←2○RANG×0,⍳10
[8]    SNB←1○RANG×⍳10
[9]    AECOEFF←(CNB+.×(CMA+.×AN))+SNB+.×(CMA+.×BN)
[10]   →(IBM=1)/L1
[11]   RHO←(1-KBULGE×2○OMOT-THBULGE+○1)×RHOAV
[12]   →L2
[13]   L1:BULGE←1+AMP×(1+2○OMOT-THBULGE+○1)*3÷2
[14]   RHO←(BULGE×RHOBASE×SEEKFL=0)+RHOAV×SEEKFL=1
[15]   L2:TAEK←KTAE×AECOEFF×RHO×VAE+.×VAE
[16]   TAEK←(3↑TAEK)-(ΔCG+ΔCGNOM)X FAEK←(÷10.06)×3↓TAEK
     ∇

     ∇ R←I AFEA A
[1]    ⍝  ROTATION MATRIX FROM EULER ANGLES IN THE SEQUENCE I GOING
[2]    ⍝  FROM RIGHT TO LEFT FROM THE OLD TO THE NEW.
[3]    R←(I[3]EUL A[3])+.×(I[2]EUL A[2])+.×I[1]EUL A[1]
     ∇
```

```
    ∇ A←AFQ Q
[1]    ⍝ DIR COS MATRIX FROM QUATERNION
[2]    A←3 3↑A÷(A←(⍉QDT Q)+.×QDB Q)[4;4]
    ∇

    ∇ ALT←ALTFDOY DOY;XI;YI;I
[1]    ⍝  INTERPOLATES TABULATED DATA TO ESTIMATE ALTITUDE FROM GIVE
       N DOY.
[2]    YI←TABDOY
[3]    XI←⍟TABALTNM
[4]    I←1⌈(+/DOY∘.>TABDOY)⌊10
[5]    ALT←*XI[I]+(DOY-YI[I])×(XI[I+1]-XI[I])÷YI[I+1]-YI[I]
    ∇

    ∇ A←X ANYROOT Y;I;N;P;Q
[1]    ⍝  YIELDS POLYNOMIAL WITH SPECIFIED ROOTS WITH REAL PART X
[2]    ⍝  AND IMAGINARY PART Y.
[3]    A←⍳1;I←1;N←⍴X;X←,X
[4]    L1:→L2×⍳0=Y[I];Y←,Y
[5]    P←-2×X[I];Q←(X[I]*2)+Y[I]*2
[6]    A←(A,0,0)+(P×(0,A,0))+Q×(0,0,A)
[7]    →L3
[8]    L2:A←(-X[I]×(0,A))+A,0
[9]    L3:I←I+1
[10]   →L1×⍳(I≤N)
    ∇

    ∇ A←S ATAN C
[1]    ⍝ 4 QUADRANT INVERSE TANGENT FUNCTION
[2]    A←(¯2○C÷((S*2)+C*2)*0.5)×S÷|S
    ∇

    ∇ ATMDC;IKEKP;ΔQEKE
[1]    ⍝  SIMULATES THE SKYLAB ATMDC.
[2]    →(T<TMID)/L
[3]    IKEKP←IKEK;ETYKP←ETYK;BETAP←BETA;NUZEKP←NUZEK
[4]    IKGEN
[5]    EKE←(⍉IKEK)+.×(EUL3 NUZEK)+.×(EUL1-RFD×BETA)+.×(EUL1 RFD×
       BETAP)+.×(EUL3-NUZEKP)+.×IKEKP
[6]    TMID←T+((○2)-((○2)|OMOT)-ETYK-ETYKP)÷OMO
[7]    TMID←TMID+TORB×0.5>(TMID-T)÷TORB
[8]    ΔQEKE←QFA EKE
[9]    QVEK←(QEP QVEK)+.×QIN ΔQEKE
[10]   L:THD←THDM×AL[;2]
[11]   ETS←(ET←HCMGK÷H)+EV←IV+.×OMV÷H
[12]   DECAY
[13]   TAUS ET+(÷H)×IV+.×OMV-THD
    ∇
```

```
    ∇ Q̲←AXES
[1]   Q̲←5 ΔGRF''⍝  DRAWS X AND Y AXES FOR GRAPHICS.
    ∇

    ∇ RTS←BAIRSTOW A;N;M;E;P;Q;K;J;C;C̲;ΔQ;B;D;ΔP
[1]   →0×ι1>N←ρA←1↓A÷1↑A,E←1ρ1E¯5,P←1,0ρQ←2+M←0,RTS←ι0
[2]   →(2 3=ρC←1 0,ι¯3+ρB←0 1,ιN)/14 13
[3]   B[2]←A[2]+(-P×1ρB)-Q,0ρB[1]←A[1]-P,0ρK←J←3
[4]   →4×ιN>¯1+K←K+ρ,B[K]←A[K]+(-P×B[K-1])-Q×B[K-2]
[5]   C[2]←B[2]+(-P×1ρC)-Q,0ρC[1]←B[1]-P
[6]   →6×ιN>J←J+ρ,C[J]←B[J]+(-P×C[J-1])-Q×C[J-2]
[7]   →9×ιN≠2+ρ,C̲←C[N-1]-B[N-1]
[8]   →10,P←P+ΔP←((B[N-1]×C[N-2])-B[N])÷D←(C[N-2]*2)-C̲
[9]   P←P+ΔP←((B[N-1]×C[N-2])-B[N]×C[N-3])÷D←(C[N-2]*2)-C̲×C[N-3]
[10]  Q←Q+ΔQ←((B[N]×C[N-2])-B[N-1]×C̲)÷D
[11]  →((200<M←M+1),E<(|ΔQ)+|ΔP)/16 3
[12]  →((1=N),3≤N←N-2)/14 3,0ρA←B,0ρP←3-Q←2+M←0,0ρRTS←RTS,QUAD P,Q
[13]  →0×RTS←RTS,QUAD 2ρA
[14]  ⍝ LINE DELETED HERE
[15]  →0×RTS←RTS,(-A[1]),0
[16]  'SLOW OR NON CONVERGENCE'
[17]  ⍝ 'A' CONTAINS THE COEFFICIENTS OF THE N'TH DEGREE POLYNOMIAL
[18]  ⍝ SUBROUTINE 'QUAD' CALCULATES ROOTS FROM QUADRATIC FACTORS
[19]  ⍝ AN EXAMPLE WOULD BE SOLVING THE ROOTS FOR THE POLYNOMIAL
[20]  ⍝ (X**5)+(2×X**4)-(4×X**3)-(20×X**2)-(33×X)-18 ; INPUT WOULD
      BE
[21]  ⍝ BAIRSTOW 1 2 ¯4 ¯20 ¯33 ¯18
[22]  ⍝ PROGRAMMED BY JOHN RICHARSON, O.C.C.,C.S.7,SPRING'70
    ∇

    ∇ ROUT←BAR RIN;S;C;Q1;Q2;Q3;Q4
[1]   ⍝  COMPUTES THE BANK, ATTACK AND ROLL ANGLES FROM QUATERNION
      SET.
[2]   Q1←S↑RIN;S←(ρRIN←,RIN)÷4
[3]   Q2←S↑S↓RIN
[4]   Q3←S↑(2×S)↓RIN
[5]   Q4←(-S)↑RIN
[6]   ROUT←DFR×¯2○(Q1*2)+(-Q2*2)+(-Q3*2)+Q4*2
[7]   ROUT←ROUT,[0.1](DFR×((Q1×Q2)-Q3×Q4)ATAN(Q1×Q3)+Q2×Q4)
[8]   ROUT←(DFR×((Q1×Q2)+Q3×Q4)ATAN(Q2×Q4)-Q1×Q3),[1]ROUT
    ∇

    ∇ BARANGS
[1]   ⍝  LISTS BAR ANGLES CALCULATED BY BAR FUNCTION IN TABULAR FOR
      M.
[2]   'I4,3F8.2'ΔFMT⍉(ι¯1↑ρQBL),[1]BAR ¯4 0↓QBL
    ∇
```

```
     ∇ BARAV;A
[1]    ⍝  ALLOWS GRAPHICAL DETERMINATION OF AVERAGE BAR ANGLES AND
[2]    ⍝  AUTOMATICALLY CALLS TEXPERT.
[3]    'MARK ANGLES IN BAR ORDER '
[4]    A←⎕[2]
[5]    A←A,⎕[2]
[6]    A←A,⎕[2]
[7]    ⎕←A;QAL←QFA 1 2 1 AFEA RFD×A
[8]    TEXPERT
     ∇

     ∇ BARS INT;T;SC;TSTOP
[1]    ⍝  DRAWS VERTICAL LINES ON A PLOT AT SPECIFIED INTERVAL.
[2]    SC←WHATSCALE
[3]    T←INT×⌈SC[1]÷INT
[4]    TSTOP←INT×⌊SC[2]÷INT
[5]    L:MARK T
[6]    →(TSTOP≥T←T+INT)/L
     ∇

     ∇ Q←BOX
[1]    Q←9 ΔGRF''⍝  DRAWS A BOX AROUND GRAPHICS WINDOW.
     ∇

     ∇ BPRBANK
[1]    ⍝  BUILDS ARRAY PRBANK USED TO SIMULATE ASAP DATA.
[2]    PRBANK←3 ¯18↑BAR PRQAL
[3]    PRATT←PRBANK[2;]
[4]    PRROLL←PRBANK[3;]
[5]    PRBANK←(¯18↑PRT),[0.1]PRBANK[1;]
     ∇

     ∇ RO←CHI RI;S;C;Q1;Q2;Q3;Q4
[1]    ⍝  COMPUTES CHI ANGLES FROM INPUT QUATERNION ARRAY.
[2]    Q1←S↑RI;S←(⍴RI←,RI)÷4
[3]    Q2←S↑S↓RI
[4]    Q3←S↑(2×S)↓RI
[5]    Q4←(-S)↑RI
[6]    RO←(2×(Q1×Q4)-Q2×Q3)ATAN(-Q1×Q1)+(Q2×Q2)+(-Q3×Q3)+Q4×Q4
[7]    RO←RO,[0.1](2×(Q2×Q4)-Q1×Q3)ATAN(Q1×Q1)+(-Q2×Q2)+(-Q3×Q3)+Q4×
       Q4
[8]    RO←DFR×RO,[1]¯1○2×(Q1×Q2)+Q3×Q4
     ∇
```

```
    ∇ P←CHRPOL A;AK;K;N;E;S
[1]   ⍝ THIS FUNCTION USES LEVERRIER'S METHOD TO OBTAIN COEFFICIENTS
[2]   ⍝ OF THE CHARACTERISTIC EQUATION OF MATRIX A.  THE COEFFICIEN
      T
[3]   ⍝ OF THE HIGHEST POWER IS NORMALIZED TO UNITY.
[4]   ⍝ WRITTEN BY HOWARD B. WILSON, U. OF ALABAMA
[5]   P←S←+/,A×E←(⍳N)∘.=⍳N,0ρK←1+0×N←1↑ρAK←A
[6]   L1:P←P,(÷K←K+1)×(¯1↑S)-+/P×⌽¯1↓S←S,+/,E×AK←AK+.×A
[7]   →L1×⍳K<N
[8]   P←1,-P
    ∇

    ∇ RX←CP EA;SEEKFL;R;RD;VLK;OMGEK;FAEK;TAEK;TGGK;T0;F0;ΔCG
[1]   ⍝  ESTIMATES CP OFFSET FROM FLIGHT DETERMINED EQUILIBRIA.
[2]   SEEKFL←1;R←(REA+ALT)×LEK[3;];RD←OMOK×(REA+ALT)×LEK[1;];VLK←1
      2 1 AFEA EA
[3]   OMGEK←OMGK×(2○LAZK)×LEK[2;]
[4]   T0←TRQ VLK;ΔCG←0 0 0
[5]   F0←FAEK
[6]   RX←-ΔCG←(÷F0+.×F0)×F0 X T0
[7]   'NEW SLOPE'
[8]   ⎕←NEWSLOPE←GETSLOPE EA
[9]   ' CP OFFSET  ';RX
    ∇

    ∇ RX←CPA;SEEKFL;R;RD;OMGEK;FAEK;TAEK;TGGK;T0;F0;ΔCG;VLK;EA
[1]   ⍝  ESTIMATES CP OFFSET FROM SIMULATION DATA.
[2]   SEEKFL←1;RMAG←MAG R←(REA+ALT)×LEK[3;];RD←OMOK×(REA+ALT)×LEK[1
      ;];VLK←VL
[3]   OMGEK←OMGK×(2○LAZK)×LEK[2;]
[4]   T0←TRQ VLK;ΔCG←0 0 0
[5]   F0←FAEK
[6]   RX←-ΔCG←(÷F0+.×F0)×F0 X T0
[7]   'NEW SLOPE'
[8]   ⎕←NEWSLOPE←GETSLOPE EA←1 2 1 EAFA VLK
[9]   ' CP OFFSET  ';RX
    ∇
```

```
    ∇ ΔCP←CPASUP;VL;RHOSIG;OMOK;ALT;ETLNOM
[1]   ⍝  GROUND SUPPORT FUNCTION USED TO ESTIMATE CP OFFSETS FROM F
      LIGHT DATA.
[2]   ⍝  THIS FUNCTION CONVERTS DATA TO PROPER FORMAT FOR CPA.
[3]   'INPUT ALTITUDE IN NAUTICAL MILES'
[4]   ALT←KFM×⎕
[5]   'INPUT ASCENDING NODE PERIOD IN HMS'
[6]   OMOK←○2÷24 60 60⊥⎕
[7]   'INPUT ESTIMATED SOLAR SIGMA'
[8]   RHOSIG←⎕
[9]   'INPUT ETLNOM'
[10]  ETLNOM←⎕
[11]  'INPUT ESTIMATED L2 STRAPDOWN OFFSET IN DEG'
[12]  ETLNOM←ETLNOM+.×EUL2 RFD×⎕
[13]  'THE TRUE MOMENTUM BIAS IS ';ETLNOM
[14]  'INPUT QBL'
[15]  VL←AFQ⎕
[16]  'THE BAR ANGLES ARE ';DFR×1 2 1 EAFA VL
[17]  ΔCP←CPA
    ∇

    ∇ DΔE
[1]   1 DRAWN'PRΔE'⍝  DRAWS ΔE VS TIME PLOT.
    ∇

    ∇ DA
[1]   1 DRAWN'PRA'⍝  DRAWS STRAPDOWN OFFSET FROM SIMULATION VS TIME
       PLOTS.
    ∇

    ∇ S DARROW A
[1]   STEREO FP+A+.×ARROW×S⍝  DRAWS A STEREO ARROW FOR A 3-SPACE VE
      CTOR.
    ∇

    ∇ DATE
[1]   ERASE⍝  ERASES PAGE AND LABELS PAGE HEADING.
[2]   '     WORKSPACE ',(⌶29),5 ΔWM 1
[3]   '     TIME= ';24 60⊤(⌊(⌶20)÷360)÷10;' DATE= ';100 100 100⊤⌶25
    ∇
```

```
    ∇ DBANK;PRT;AV;TSTOP;N;S;C;ANP;BNP;F0;FC;FS;S2;C2;SC;CI;SI;
      PRTF
[1]   ⍝ ESTIMATES STRAPDOWN ERROR FROM SIMULATION DATA SIMULATING
      ASAP DUMP.
[2]   N←⍴PRT←PRBANK[1;]
[3]   PRT←DFR×OMO×PRT-(PRT[1]-TORB|PRT[1]-TMID)
[4]   TSTOP←TOP PRT
[5]   SI←1○RFD×PRT;CI←2○RFD×PRT
[6]   F0←+/PRBANK[2;];FC←+/PRBANK[2;]×CI;FS←+/PRBANK[2;]×SI
[7]   C←+/CI;S←+/SI;C2←+/CI×CI;S2←+/SI×SI;SC←+/SI×CI
[8]   MAT←⌹(N,C,S),[1](C,C2,SC),[0.1]S,SC,S2
[9]   C←MAT+.×(F0,FC,FS)
[10]  AV←C[1];ANP←C[2];BNP←C[3]
[11]  1 DRAWNS'PRBANK[2;]-AV'
[12]  0 DRAW(C←(ANP×2○RFD×PRTF)+BNP×1○RFD×PRTF)VS PRTF←30×0,⍳⌊
      TSTOP÷30
[13]  'DETHL WRT MIDNIGHT ';(-ANP),0,BNP
[14]  ' '
[15]  'SUGGESTED DETHL ';(EUL2 OMOT+OMO×ΔT)+.×(-ANP),0,BNP
[16]  ' '
[17]  'THE AVERAGE BANK ANGLE IS ';AV
[18]  '            THE AMPLITUDE IS ';((ANP×ANP)+BNP×BNP)*0.5
[19]  '                 THE PHASE IS ';360|DFR×(-ANP)ATAN BNP
[20]  ' '
[21]  DBANKPAR
[22]  ' '
[23]  'I5,F9.1'ΔFMT 13 2↑PRTF,[1.1]C+AV
[24]  F0←+/PRATT;FC←+/PRATT×CI;FS←+/PRATT×SI;⎕
[25]  C←MAT+.×F0,FC,FS
[26]  AV←C[1];ANP←C[2];BNP←C[3]
[27]  1 DRAWNS'PRATT-AV'
[28]  0 DRAW(C←(ANP×2○RFD×PRTF)+BNP×1○RFD×PRTF)VS PRTF
[29]  'THE AVERAGE ANGLE OF ATTACK IS ';AV
[30]  '                  THE AMPLITUDE IS ';((ANP×ANP)+BNP×BNP)*0.5
[31]  '                       THE PHASE IS ';360|DFR×(-ANP)ATAN BNP
[32]  DBANKPAR
[33]  ' '
[34]  'I5,F9.1'ΔFMT 13 2↑PRTF,[1.1]C+AV
[35]  F0←+/PRROLL;FC←+/PRROLL×CI;FS←+/PRROLL×SI;⎕
[36]  C←MAT+.×F0,FC,FS
[37]  AV←C[1];ANP←C[2];BNP←C[3]
[38]  1 DRAWNS'PRROLL-AV'
[39]  0 DRAW(C←(ANP×2○RFD×PRTF)+BNP×1○RFD×PRTF)VS PRTF
[40]  'THE AVERAGE ROLL ANGLE IS ';AV
[41]  '            THE AMPLITUDE IS ';((ANP×ANP)+BNP×BNP)*0.5
[42]  '                 THE PHASE IS ';360|DFR×(-ANP)ATAN BNP
[43]  DBANKPAR
[44]  ' '
[45]  'I5,F9.1'ΔFMT 13 2↑PRTF,[1.1]C+AV
    ∇
```

```
    ∇ DBANKPAR
[1]    ⍝  PRINTS DATA BLOCK FOR STRAPDOWN ERROR ESTIMATE
[2]    ' '
[3]    ' '
[4]    '⎕DESIRED UPDATE      ⎕,I3,⎕:⎕,I2,⎕:⎕,I2,⎕:⎕,F5.2'ΔFMT 1 4⍴
       DFS T
[5]    '⎕MIDNIGHT TIME       ⎕,I3,⎕:⎕,I2,⎕:⎕,I2,⎕:⎕,F5.2'ΔFMT 1 4⍴
       DFS TMID
[6]    '⎕1 REV DELTA TIME        ⎕,I2,⎕:⎕,I2,⎕:⎕,F5.2'ΔFMT 1 3⍴1↓DFS
       TORB
[7]    '⎕FIRST QBL TIME      ⎕,I3,⎕:⎕,I2,⎕:⎕,I2,⎕:⎕,F5.2'ΔFMT⍉¯4 1↑
       QBL
[8]    '⎕LAST QBL TIME       ⎕,I3,⎕:⎕,I2,⎕:⎕,I2,⎕:⎕,F5.2'ΔFMT⍉¯4 ¯1↑
       QBL
[9]    ' '
    ∇

    ∇ DBANKSUP;NAME;ΔT;OMO;OMOT;TGMT0;GMT0;QAL;T;PRQAL;PRT;PRBANK;
       TMID;TORB;PRATT;PRROLL
[1]    ⍝  GROUND SUPPORT FUNCTION USED TO ESTIMATE STRAPDOWN ERROR F
       ROM
[2]    ⍝  ASAP DATA.  THIS PROGRAM SETS UP DATA IN PROPER FORMAT FOR
        DBANK.
[3]    'HAS QBLSET FUNCTION BEEN EXECUTED WITH LATEST DATA?'
[4]    'ENTER Y OR N FOR YES OR NO'
[5]    →('YN'∊⍞)/LY,LN
[6]    LY:'INPUT GMT OF DESIRED UPDATE IN DHMS'
[7]    T←365 24 60 60⊥⎕
[8]    'OBTAIN FOLLOWING FROM CIRCLE CHART'
[9]    'INPUT MIDNIGHT GMT OF QBL SET'
[10]   TMID←SFD⎕
[11]   'INPUT 1 REV DELTA TIME'
[12]   TORB←SFD 0,⎕
[13]   'TO GET PLOTS OF BANK, ATTACK AND ROLL ANGLES '
[14]   'INPUT CARRIAGE RETURNS'
[15]   PRQAL←¯4 0↓QBL;⎕
[16]   PRT←SFD 4 0↑QBL
[17]   OMO←○2÷TORB
[18]   OMOT←OMO×TORB|T-TMID
[19]   ΔT←0
[20]   PRBANK←PRT,[1](BAR PRQAL)
[21]   PRATT←PRBANK[3;]
[22]   PRROLL←PRBANK[4;]
[23]   PRBANK←¯2 0↓PRBANK
[24]   DBANK
[25]   →0
[26]   LN:'EXECUTE FUNCTION QBLSET BEFORE USING THIS FUNCTION'
    ∇
```

```
    ∇ DBAR;INDEX;IMAX;ST;I;PRBAR
[1]    ⍝  DRAWS BAR ANGLES.  SELECTS PROPER ORBIT TO PLOT.
[2]    1 DRAWN'BAR (4,ILEG)↑PRQAL'
[3]    HMARK 114
[4]    HMARK 93
[5]    HMARK ¯98.5
[6]    I←0
[7]    IMAX←+/ILEG≤PRST[2;]
[8]    ST←(-IMAX)↑PRST[2;]
[9]    L:I←I+1
[10]   INDEX←(ST[I]-1),ST[I],ST[I]+1
[11]   0 DRAW(PRBAR←BAR PRQAL[;INDEX])[1;]VS PRT[INDEX]
[12]   0 DRAW PRBAR[2;]VS PRT[INDEX]
[13]   0 DRAW PRBAR[3;]VS PRT[INDEX]
[14]   →(I<IMAX-1)/L
[15]   I←I+1
[16]   INDEX←¯1+(ST[I]-1),ST[I],ST[I]+1
[17]   0 DRAW(PRBAR←BAR PRQAL[;INDEX])[1;]VS PRT[INDEX]
[18]   0 DRAW PRBAR[2;]VS PRT[INDEX]
[19]   0 DRAW PRBAR[3;]VS PRT[INDEX]
    ∇

    ∇ DBARA
[1]    ⍝  DRAWS BAR ANGLES FOR ALL SIMULATION DATA.
[2]    1 DRAWNS'BAR PRQAL'
[3]    DBARAPAR
[4]    BARANGS
    ∇

    ∇ DBARAPAR
[1]    ⍝  DATA BLOCK PLACED ON BAR ANGLE PLOTS.
[2]    2▼13 13
[3]    '⎕MIDNIGHT TIME       ⎕,I3,⎕:⎕,I2,⎕:⎕,I2,⎕:⎕,F5.2'ΔFMT 1 4⍴
       DFS TMID
[4]    '⎕1 REV DELTA TIME         ⎕,I2,⎕:⎕,I2,⎕:⎕,F5.2'ΔFMT 1 3⍴1↓DFS
       TORB
[5]    '⎕FIRST QBL TIME      ⎕,I3,⎕:⎕,I2,⎕:⎕,I2,⎕:⎕,F5.2'ΔFMT⍉¯4 1↑
       QBL
[6]    '⎕LAST QBL TIME       ⎕,I3,⎕:⎕,I2,⎕:⎕,I2,⎕:⎕,F5.2'ΔFMT⍉¯4 ¯1↑
       QBL
[7]    ' '
    ∇
```

```
    ∇ DBARASUP;NAMEQAL;PRQAL;PRT;TMID;TORB;TSTOP
[1]   ⍝  GROUND SUPPORT FUNCTION FOR PLOTTING ASAP BAR ANGLE DATA.
[2]   'HAS QBLSET FUNCTION BEEN EXECUTED WITH LATEST DATA?'
[3]   'ENTER Y OR N FOR YES OR NO'
[4]   →('YN'∊⍞)/LY,LN
[5]   LY:'OBTAIN FOLLOWING FROM CIRCLE CHART'
[6]   'INPUT MIDNIGHT GMT OF QBL SET'
[7]   TMID←SFD⎕
[8]   'INPUT 1 REV DELTA TIME'
[9]   TORB←SFD 0,⎕
[10]  'TO GET A PLOT OF THE BAR ANGLES'
[11]  'INPUT A CARRIAGE RETURN'
[12]  PRQAL←¯4 0↑QBL;⍞
[13]  PRT←SFD 4 0↑QBL
[14]  TSTOP←TOP PRT←PRT-PRT[1]-TORB|PRT[1]-TMID
[15]  DBARA
[16]  →0
[17]  LN:'EXECUTE FUNCTION QBLSET BEFORE USING THIS FUNCTION'
    ∇

    ∇ DCD
[1]   1 DRAWN'PRCD'⍝  DRAWS DRAG COEFFICIENT VS TIME PLOT.
    ∇

    ∇ X←RD DDDDER R
[1]   ⍝  COMPUTES 3RD DERIVATIVE OF ΔR FOR INTEGRATION OF ORBIT DYN
      AMICS.
[2]   X←(-GME÷R2×RMAG)×RD-OMOK×RMAG×EX
[3]   X←X+((3×GME÷R2×R2×RMAG)×R)+(DELG R)+.×RD
    ∇

    ∇ DECMG
[1]   1 DRAWN'PRECMG'⍝  DRAWS ECMG VS TIME PLOT.
    ∇

    ∇ A←DEGA R;P
[1]   P←QFA R⍝  COMPUTES EIGENANGLE FROM DIRECTION COSINES.
[2]   A←2×DFR×(¯2○P[4])×(3↑P)÷MAG 3↑P
    ∇

    ∇ A←DEGQ R
[1]   ⍝  COMPUTES EIGENANGLE FROM INPUT QUATERNION.
[2]   A←2×DFR×(¯2○R[4])×(3↑R)÷MAG 3↑R
    ∇
```

```
    ∇ DG←DELG R;GNOM;ONOK;KTGG
[1]   ⍝  COMPUTES NUMERICAL GRADIENT OF GRAVITATIONAL ACCELERATION.
[2]   DG←1000000×(GACCEL R+1E¯6,0 0)-GNOM←GACCEL R
[3]   DG←DG,[1.1]1000000×(GACCEL R+0,1E¯6,0)-GNOM
[4]   DG←DG,1000000×(GACCEL R+0 0,1E¯6)-GNOM
    ∇

    ∇ R←DENS H;HL;HU
[1]   R←2⍴0⍝  INTERPOLATES DENSITY FROM TABLE OF DENSITY VS ALTITUD
      E.
[2]   HL←1⌈8⌊+/H≤ALTTAB
[3]   HU←HL+1
[4]   R[1]←RHOTAB[HL]×*((⍟RHOTAB[HU]÷RHOTAB[HL])÷ALTTAB[HU]-ALTTAB[
      HL])×H-ALTTAB[HL]
[5]   R[2]←BULGETAB[HL]+((BULGETAB[HU]-BULGETAB[HL])÷ALTTAB[HU]-
      ALTTAB[HL])×H-ALTTAB[HL]
    ∇

    ∇ DESAT;A;A1;A2
[1]   ⍝  APL VERSION OF TEA SEEKING METHOD USED FOR DESATURATION.
[2]   ΔEP←ΔE
[3]   ΔENP←ΔEN
[4]   ETVNOM←AL+.×ETLNOM
[5]   ETACS←VLK+.×ROT+.×(⍉VLK)+.×ETACS
[6]   ΔE←ETS+ETACS-ETVNOM
[7]   ΔEN←ΔELIM LIMC ΔE
[8]   ΔPHH←0.5×ΔPHLIM LIMC SLOPE+.×ΔEN+ΔE-ΔEP
[9]   ΔTHH←0.5×AL+.×ΔTHL-KΔTH×ΔTHLIM⌊(AL[;3]+.×SLOPE+.×ΔENP+ΔE-ΔEP)
      ⌈-ΔTHLIM
[10]  ΔTHL←0 0 0
[11]  ΔQAL←ΔPT,(1-ΔPT+.×ΔPT←ΔPHH+ΔTHH)*0.5
[12]  VL←AL←AFQ QAL←(QDB ΔQAL)+.×QAL
[13]  ETACS←0 0 0
[14]  ΔQVEK←ΔPHH,(1-ΔPHH+.×ΔPHH)*0.5
[15]  VEK←AFQ QVEK←(QDB ΔQVEK)+.×QVEK
[16]  VLK←VEK+.×⍉LEK
[17]  ⍝          PRINTOUT
    ∇

    ∇ Y←DET X;I;K;V
[1]   I←(⍴X)[K←1]
[2]   →(X[K;K]=0)/7
[3]   V←(K⍴0),(⌽(I-K)↑⌽X[;K])÷X[K;K]
[4]   X←X-V∘.×X[K;]
[5]   →(I>K←K+1)/2
[6]   →0,Y←+×/1 1⍉X
[7]   Y←X[K+⍳I-K;K]≠0
[8]   →(0=+/Y)/10
[9]   →3,X[K;]←X[K;]-(Y/[1]X[K+⍳I-K;])[1;]
[10]  Y←0
[11]  ⍝  COMPUTES DETERMINANT OF ARBITRARY SQUARE MATRIX.
    ∇
```

```
    ∇ B←DFDOY A
[1]   ⍝  COMPUTES DHMS FROM DOY IN DECIMAL FORM.
[2]   B←365 24 60 60⊤86400×A
    ∇

    ∇ R←DFHEX X
[1]   R←16 16 16 16⊥¯4↑0 0 0 0,X⍝  CONVERTS HEX DATA INTO DECIMAL
    ∇

    ∇ R←DFOCT X
[1]   R←8 8 8 8⊥¯4↑0 0 0 0,X⍝  DECIMAL FROM OCTAL CONVERSION.
    ∇

    ∇ Y←DFS X
[1]   Y←365 24 60 60⊤X⍝  CONVERTS SECONDS INTO DHMS.
    ∇

    ∇ DH
[1]   1 DRAWN'OEK+.×PRHTEK'⍝  DRAWS HTEK IN O FRAME VS TIME.
    ∇

    ∇ DOY←DOYFALT ALT;XI;YI;I
[1]   ⍝  INTERPOLATES DOY VS ALT TABLE TO ESTIMATE DOY FROM GIVEN A
      LT.
[2]   YI←⍟TABALTNM
[3]   XI←TABDOY
[4]   I←1⌈(+/ALT∘.≤TABALTNM)⌊10
[5]   DOY←XI[I]+((⍟ALT)-YI[I])×(XI[I+1]-XI[I])÷YI[I+1]-YI[I]
    ∇

    ∇ DPH
[1]   1 DRAWN'PRΔPH'⍝  DRAWS ΔPH VS TIME.
    ∇

    ∇ DPWR
[1]   1 DRAWN'PRENK[1;]'⍝  DRAWS SOLAR ARRAY ENERGY VS TIME PLOT.
    ∇

    ∇ DPWRA;PRT
[1]   ⍝  DRAWS POWER ANGLE VS TIME FOR SIMULATED ASAP DATA.
[2]   1 DRAWN'PRPWRA[2;]';PRT←PRPWRA[1;]
    ∇
```

```
     ∇ DR
[1]    ⍝  MENU OF PLOTS TO GO THROUGH TO ASSESS SIMULATION RUN.
[2]    DTQ
[3]    DH;⎕
[4]    XH;⎕
[5]    DA;⎕
[6]    XA;⎕
[7]    DPH;⎕
[8]    DRO;⎕
[9]    DPWR;⎕
     ∇

     ∇ Q←L DRAW R;C
[1]    →((∨/C←((∧/1 4≠ρL)∨(1≠ρρR)∧∧/~1 2 3∊ρR),(∧/1 2≠ρρR),(1=ρ,R)∨
       C∧~∧/</2 2ρL),(~C←1≠ρL←,L),1)/4 3 2,Q←⍳0
[2]    2 ∆GRF L
[3]    →0,ρ⎕←R
[4]    6⌽' ERRORDRAW ',,(3 6ρ'LENGTH RANK DOMAIN')[C⍳1;]
[5]    ⍝  GRAPHICS FOR DRAWING 2 OR 3 DIMENSIONAL PLOTS.
     ∇

     ∇ P DRAWN NAME;L;R;I;N;NI;S;A;Y1
[1]    ⍝ LABELS, SCALES, AND DRAWS MULTIPLE PLOTS ON A PAGE.
[2]    A←⍎NAME
[3]    I←¯1↑ρA
[4]    A←,A
[5]    N←(ρA)÷I
[6]    NI←⌈I÷P
[7]    S←(P×⍳NI)-P-1
[8]    STRAPIS 5
[9]    DATE
[10]   2▼21 21
[11]   '                    FINAL VALUE OF ',NAME,' = ';A[I×⍳N]
[12]   WINDOW 0.5 7.75 0.5 5.5
[13]   BOX
[14]   SCALE(-TSTOP×0.1),(TOP TSTOP),(-Y1),Y1←TOP((N,I)ρA)[;S]
[15]   AXES
[16]   L1:0 DRAW A[S]VS PRT[S]
[17]   A←I↓A
[18]   →(0≠ρA)/L1
     ∇
```

```
    ∇ P DRAWNS NAME;L;R;I;N;NI;Y1;A;S
[1]   A←⍎NAME⍝  PLOTS NAMED VARIABLE(S) VS TIME ON SINGLE PLOT IN S
      MALL WINDOW
[2]   Y1←TOP A
[3]   I←¯1↑⍴A
[4]   A←,A
[5]   N←(⍴A)÷I
[6]   NI←⌈I÷P
[7]   S←(P×⍳NI)-P-1
[8]   STRAPIS S
[9]   DATE
[10]  '          FINAL VALUE OF ',NAME,' = '
[11]  ,(N,1)⍴A[I×⍳N]
[12]  ' '
[13]  WINDOW 3.8 7.8 0.25 4.25
[14]  BOX
[15]  SCALE(-TSTOP×0.1),(TOP TSTOP),(-Y1),Y1
[16]  AXES
[17]  L1:0 DRAW A[S]VS PRT[S]
[18]  A←I↓A
[19]  →(0≠⍴A)/L1
    ∇

    ∇ D←DRIFT
[1]   ⍝  COMPUTES ACTUAL STRAPDOWN ERROR FROM SIMULATION DATA.
[2]   D←-OBK+.×(DEGA(⍉VLK)+.×VL)+.×LBK
    ∇

    ∇ DRO
[1]   ⍝  DRAWS ALTITUDE VS TIME WITH INITIAL ALTITUDE AS REFERENCE.
[2]   1 DRAWN'-REA+ALTIC-MAG PRR'
    ∇

    ∇ DTQ
[1]   1 DRAWN'PRTGGK+PRTABK'⍝  DRAWS TOTAL EXTERNAL TORQUE VS TIME.
    ∇
```

```
    ∇ EA←I EAFA A;U;S
[1]   ⍝THIS PROGRAM COMPUTES EULER ANGLES FROM DIR COSINE MATRICES.
[2]   ⍝ I READS FROM LEFT TO RIGHT THE ROTATIONS FROM THE OLD TO TH
      E NEW.
[3]   ⍝EA[1] IS ABOUT AXIS I[1] ETC.
[4]   ⍝PROGRAMMED BY JOHN GLAESE 1-23-79.
[5]   U←3 3⍴1 0 0 0
[6]   →(I[1]=I[3])/L
[7]   S←DET U[;I[1],I[2],I[3]]
[8]   EA←¯1○S×A[I[3];I[1]]
[9]   EA←((-S×A[I[3];I[2]])ATAN A[I[3];I[3]]),EA
[10]  EA←EA,(-S×A[I[2];I[1]])ATAN A[I[1];I[1]]
[11]  →0
[12]  L:I[3]←6-I[1]+I[2]
[13]  EA←¯2○A[I[1];I[1]]
[14]  S←DET U[;I[1],I[2],I[3]]
[15]  EA←EA,A[I[2];I[1]]ATAN S×A[I[3];I[1]]
[16]  EA←(A[I[1];I[2]]ATAN-S×A[I[1];I[3]]),EA
    ∇

    ∇ EDITQBL;I
[1]   ⍝  ALLOWS GROUND CONTROLLERS TO EDIT QBL ARRAY AFTER IT HAS B
      EEN ENTERED.
[2]   'ENTER A, I OR D TO ADD, INSERT OR DELETE A COLUMN FROM QBL'
[3]   →('AID'ι⍞)/LA,LI,LD
[4]   LA:'INPUT NEW QUATERNION AND GMT'
[5]   QBL←QBL,⎕
[6]   →0
[7]   LI:'ENTER NUMBER OF COLUMNS AHEAD OF LINE TO BE INSERTED'
[8]   QBL←(I←⎕)⌽[2]QBL
[9]   'ENTER NEW QUATERNION AND GMT'
[10]  QBL←QBL,⎕
[11]  QBL←(-I+1)⌽[2]QBL
[12]  →0
[13]  LD:'ENTER NUMBER OF COLUMN TO BE DELETED'
[14]  QBL←((8,I-1)↑QBL),(0,I←⎕)↓QBL
    ∇
```

```
    ∇ V←R EIGVEC A;N;I;K;X0;Y0;C0;E;L;M
[1]   ⍝  COMPUTES EIGENVECTOR FROM GIVEN MATRIX AND EIGENVALUE.
[2]   I←(N,N)ρ1,NρK←0×N←1↑ρA
[3]   X0←0.002×¯501+?Nρ1001
[4]   →REAL×⍳R[2]=0
[5]   COMPLEX:Y0←0.002×¯501+?Nρ1001
[6]   C0←((N×R[2]*2)++/,A×A←A-R[1]×I)*0.5
[7]   M←⌹(I×R[2]*2)+A+.×A←A-I×1E¯9×C0
[8]   M←R[2]×M,(N,0)ρL←A+.×M
[9]   LOOP1:V←,NRMLIZ((L+.×X0)-M+.×Y0),(M+.×X0)+L+.×Y0
[10]  K←K+1,0ρX0←N↑V,0ρY0←N↓V,0ρE←(+/(V-X0,Y0)*2)*0.5
[11]  →((K>100),(E≥0.0001),1)/FAIL,LOOP1,FINISH1
[12]  FAIL:→0,,V←(2,N)ρ0
[13]  FINISH1:→0,,V←(2,N)ρV
[14]  REAL:M←⌹A-I×0.0001×C0←(+/,A×A←A-R[1]×I)*0.5
[15]  LOOP2:K←K+1,0ρX0←V,0ρE←+/|X0-V←(×V[1↑⍒|V])×V÷(+/V×V←M+.×X0)*
      0.5
[16]  →((K>100),(E≥1E¯6),1)/FAIL,LOOP2,FINISH2
[17]  FINISH2:V←(2,N)ρV,Nρ0
    ∇

    ∇ R←ENC N;I
[1]   I←16⍝  ENCODES DECIMAL DATA INTO UPLINK OCTAL.
[2]   N←N×2*I-5
[3]   →(N≥0)/L
[4]   N←N+2*I
[5]   L:R←(Iρ2)⊤⌊0.49999+N
[6]   R←(6ρ8)⊤(18ρ2)⊥(18ρ0,8ρ1)\R
[7]   R←50000 50000+10 10 10⊥⍉2 3ρR
    ∇

    ∇ Q←ERASE
[1]   Q←8 ΔGRF 1⍝  ERASES PAGE FOR GRAPHICS FUNCTIONS.
    ∇

    ∇ R←I EUL A
[1]   →(I=1 2 3)/L1,L2,L3⍝  CONSTRUCTS EULER ROTATION MATRIX ABOUT
      AXIS I = 1,2,3.
[2]   L1:→0;R←EUL1 A
[3]   L2:→0;R←EUL2 A
[4]   L3:→0;R←EUL3 A
    ∇

    ∇ R←EUL1 A
[1]   R←3 3ρ0⍝  COMPUTES EULER ROTATION MATRIX ABOUT X AXIS.
[2]   R[1;1]←1
[3]   R[2;2]←R[3;3]←2○A
[4]   R[3;2]←-R[2;3]←1○A
    ∇
```

```
     ∇ R←EUL2 A
[1]    R←3 3ρ0⍝  COMPUTES EULER ROTATION MATRIX ABOUT Y AXIS.
[2]    R[2;2]←1
[3]    R[3;3]←R[1;1]←2○A
[4]    R[1;3]←-R[3;1]←1○A
     ∇

     ∇ R←EUL3 A
[1]    R←3 3ρ0⍝  COMPUTES EULER ROTATION MATRIX ABOUT Z AXIS.
[2]    R[3;3]←1
[3]    R[1;1]←R[2;2]←2○A
[4]    R[2;1]←-R[1;2]←1○A
     ∇

     ∇ A←FCF Y;X;C1;S1;CK;SK;K;Q;N
[1]    N←0.5×¯1+ρY⍝  COMPUTES FOURIER COEFFICIENTS FOR DATA Y.
[2]    X←○(0,⍳2×N)÷N+0.5
[3]    S1←1○X;C1←2○X
[4]    CK←1+SK←(1+2×N)ρ0
[5]    K←0;A←2 0ρ0
[6]    L:A←A,(Y+.×CK),-Y+.×SK
[7]    Q←(CK×C1)-SK×S1
[8]    SK←(C1×SK)+S1×CK
[9]    CK←Q
[10]   →(N≥K←K+1)/L
[11]   A←ΔTOUT×(+/[1]A×A)*0.5
[12]   A←((÷ΔTOUT×1+2×N)×0,⍳N),[0.1]A
     ∇

     ∇ FILLTABS;I;N;RTS
[1]    ⍝  DEVELOPS DATA TO FILL STANDARDIZED TABLES OF TEA ATTITUDE,
        BIAS, ETC.
[2]    ⍝  VS TIME.
[3]    I←1;TABSL←0 3 3ρ0;TABSLRTS←TABEAT←3 0ρ0
[4]    N←ρTABALTSU
[5]    L:ALT←ALTIC←KFM×TABALTSU[⎕←I]
[6]    IC
[7]    TABSL←TABSL,[1]SLOPE
[8]    RTS←QRALG SLOPE
[9]    TABSLRTS←TABSLRTS,RTS[⍋RTS[;1];1]
[10]   TABEAT←TABEAT,,BAR QAL
[11]   →(N≥I←I+1)/L
     ∇
```

```
    ∇ A←FINDQR H;I;N;S;LAI;MUI;AS;QI;IDN;XK;XK0
[1]   I←1;IDN←3 3⍴1 0 0 0⍝  PERFORMS QR TRANSFORMATION IN QR ALGORI
      THM.
[2]   N←1↑⍴H←A+.×A←H
[3]   →(N≤2)/0;H←H-(A×2×TR[1])-(+/TR×TR)×(N,N)⍴1,N⍴0
[4]   H←(H,(N,2)⍴0),[1](2,N+2)⍴0;A←(A,(N,2)⍴0),[1](2,N+2)⍴0
[5]   L:→(EP≥⌈/|AS←H[S←I+0 1 2;I])/LPP
[6]   XK0←((+/AS*2)*0.5)×(AS[1]≥0)-AS[1]<0
[7]   MUI←(1↓AS)÷XK0+AS[1]
[8]   LAI←2÷1++/MUI×MUI
[9]   QI←IDN-LAI×(1,MUI)∘.×1,MUI
[10]  A[;S]←A[;S]+.×QI
[11]  A[S;]←QI+.×A[S;]
[12]  H[S;]←QI+.×H[S;]
[13]  LPP:→(I<3)/LP
[14]  A[I;⍳I-2]←0
[15]  LP:→(N>I←I+1)/L
[16]  A[N;⍳N-2]←0
[17]  A←¯2 ¯2↓A
    ∇

    ∇ RX←IN FIX X
[1]   ⍝THIS FUNCTION SUBSTITUTES CHARACTERS IN CHARACTER VECTORS.
[2]   ⍝IN[1] IS THE OLD CHARACTER AND IN[2] IS THE NEW.
[3]   ⍝E.G. '-¯' FIX X SUBSTITUTES AN APL MINUS (¯) FOR THE SUBTRAC
      T
[4]   ⍝OPERATOR MINUS (-) FOR ALL OCCURRENCES OF - IN X.
[5]   ⍝CODED BY JOHN R GLAESE, 1/15/79
[6]   L:→(~IN[1]∊RX←X)/0
[7]   →L;X[X⍳IN[1]]←IN[2]
    ∇
```

```
     ∇ FORMAT8
[1]    ⍝  THIS FUNCTION SIMULATES THE OUTPUTS ON ASCO FORMAT 8.
[2]    'X50,⎕AM/ATM/ATMDC   / /  ⎕,3A1'ΔFMT 1 3ρSTATION[STATIONTAB[2;1];]
[3]    '⎕                                                    GMT    ⎕,I3,⎕:⎕,I2,⎕:⎕,I2,⎕:⎕,F7.4'ΔFMT 1 4ρDFS GMT0+T
[4]    '⎕                                                    TMID   ⎕,I3,⎕:⎕,I2,⎕:⎕,I2,⎕:⎕,F7.4'ΔFMT 1 4ρDFS TGMT0+TMID-TORB
[5]    '⎕                                                    DMID   ⎕,I3,⎕:⎕,I2,⎕:⎕,I2,⎕:⎕,F7.4'ΔFMT 1 4ρDFS TORB
[6]    '⎕                    TEA CONTROL PARAMETERS          TGMT   ⎕,I3,⎕:⎕,I2,⎕:⎕,I2,⎕:⎕,F7.4'ΔFMT 1 4ρDFS TGMT0+T
[7]    '                  X         Y         Z             DELTA T TEA   ';0
[8]    '⎕DELTA E       ⎕,3F10.6'ΔFMT 1 3ρΔE
[9]    '⎕DELTA EP      ⎕,3F10.6'ΔFMT 1 3ρΔEP
[10]   ' '
[11]   '⎕DELTA PH      ⎕,3F10.6'ΔFMT 1 3ρ2×ΔPHH
[12]   '⎕DELTA TH      ⎕,3F10.6'ΔFMT 1 3ρ2×DFR×ΔTHH
[13]   ' '
[14]   '⎕ETSF          ⎕,3F10.6'ΔFMT 1 3ρETS
[15]   ' '
[16]   '⎕CMG MOM       ⎕,3F10.0'ΔFMT 1 3ρ(100÷3)×ETS-(÷H)×IV+.×VL+.×0,OMO,0
[17]   ' '
[18]   '          *****CMG 2***   *****CMG 3***'
[19]   'IGA IGR'
[20]   'OGA OGR'
[21]   'WH SPD'
[22]   'X10,⎕DMIB⎕,I5,⎕   MIB⎕,I5,⎕   JET 123456⎕'ΔFMT 1 2ρ40,2633+MIBSUM
[23]   'X22,⎕FO ⎕,I5'ΔFMT 0
[24]   ' '
[25]   '⎕QBL ⎕,4F12.7'ΔFMT 1 4ρQAL
[26]   '⎕QVA ⎕,4F12.7'ΔFMT 1 4ρ0 0 0 1
[27]   '⎕QVI ⎕,4F12.7'ΔFMT 1 4ρQFA VL+.×LEK+.×⍉IKEK
[28]   ' '
     ∇

     ∇ C0←FPWR P;F0;FP;ΔC
[1]    C0←P⍝  THIS FUNCTION COMPUTES ANGLE OF INCIDENCE OF SUNRISE.
[2]    L:F0←(0.308×C0×C0×C0)+(-0.924×C0×C0)+(1.924×C0)-P+0.308
[3]    FP←(0.924×C0×C0)+(-1.848×C0)+1.924
[4]    ΔC←-F0÷FP
[5]    C0←C0+ΔC
[6]    →(0.0001<|ΔC)/L
     ∇

     ∇ A←GACCEL R
[1]    ⍝  COMPUTES THE ACCELERATION OF GRAVITY VS R.
[2]    ER←R÷RMAG←(R2←R+.×R)*0.5
[3]    A←(GME×J2÷R2×R2)×(ER×(7.5×ER[2]*2)-1.5)-3×ER[2]×0 1 0
[4]    KTGG←3×OMOK×OMOK←(GME÷R2×RMAG)*0.5
     ∇
```

```
    ∇ GBP B
[1]    ⍝  COMPUTES GYRO BIASES REQUIRED TO CANCEL OBSERVED DRIFT.
[2]    B←VL[;2]×RFD×B÷TORB
[3]    B←B÷0.5 1 0.5×0.1×RFD×2*¯10
[4]    GBIAS←1 1 1∘.×B←0.5 1 0.5×⌊B+0.5
[5]    OMBIAS←(0.1×RFD×2*¯10)×⎕←B
[6]    'TRUE BIAS ABOUT ORBITAL Y  ';OMBIAS+.×VL×TORB×DFR
    ∇

    ∇ GBPA;B;I;PWRA;S
[1]    ⍝  COMPUTES REQUIRED GYRO BIASES FROM DATA IN SIMULATION
[2]    ⍝  USING POWER ANGLE OR SUN SENSOR ANGLE DATA.
[3]    I←PRPWRA[1;]⍳¯1↑PRPWRA[1;];PWRA←¯1↑PRPWRA[2;]
[4]    I←(I-1),I
[5]    →(I[1]>0)/L1
[6]    'INSUFFICIENT POWER DATA TO CHANGE BIAS'
[7]    →L2
[8]    L1:B←VL[;2]×RFD×(OMBIAS+.×VL×DFR)[2]+S←(-/PRPWRA[2;I])÷-/
       PRPWRA[1;I]
[9]    'DETHL2 UPDATE = ';DFR×ΔTHL[2]←RFD×PWRA+S×T-PRPWRA[1;I[2]]
[10]   B←B÷0.5 1 0.5×0.1×RFD×2*¯10
[11]   GBIAS←1 1 1∘.×B←0.5 1 0.5×⌊B+0.5
[12]   OMBIAS←(0.1×RFD×2*¯10)×⎕←B
[13]   L2:'TRUE BIAS ABOUT ORBITAL Y  ';OMBIAS+.×VL×TORB×DFR
    ∇

    ∇ GBPASUP;T;PRPWRA;QAL;QVA;VL;TORB;ALT
[1]    ⍝  GROUND SUPPORT FUNCTION TO COMPUTE REQUIRED GYRO BIASES
[2]    ⍝  FROM POWER ANGLE OR SUN SENSOR ANGLE FLIGHT DATA.
[3]    'INPUT TIME OF DESIRED UPDATE'
[4]    T←365 24 60 60⊥⎕
[5]    'INPUT DMID IN HMS'
[6]    TORB←24 60 60⊥⎕
[7]    'INPUT LATEST POWER ANGLE AND GMT'
[8]    PRPWRA←(365 24 60 60⊥1↓A),A[1];A←⎕
[9]    'INPUT PREVIOUS POWER ANGLE AND ITS GMT'
[10]   PRPWRA←((365 24 60 60⊥1↓A),A[1]),[1.1]PRPWRA;A←⎕
[11]   'INPUT CURRENT AVERAGE BIASES TO NEAREST HALF LSBS'
[12]   GBIAS[1 2 3;]←1 1 1∘.×⎕
[13]   OMBIAS←(0.1×RFD×2*¯10)×GBIAS[1;]
[14]   'INPUT QBL'
[15]   QAL←⎕
[16]   VL←AFQ QAL
[17]   GBPA
    ∇
```

```
     ∇ RES←GETSLOPE BAR;M;VLK;R;RD;SEEKFL;ONGEK
[1]    ⍝  COMPUTES SLOPE MATRIX APPROPRIATE TO INPUT BAR ANGLES.
[2]    VLK←1 2 1 AFEA BAR;SEEKFL←1
[3]    ONGEK←(2○LAZK)×ONGK×0 1 0+.×LEK
[4]    R←(REA+ALT)×LEK[3;]
[5]    RD←(REA+ALT)×OMOK×LEK[1;]
[6]    M←1E¯6 JACOBIAN VLK
[7]    RES←-(H÷ΔT)×⌹M
     ∇

     ∇ GGTORQUE;UOV3
[1]    ⍝  COMPUTES GRAVITY GRADIENT TORQUE ON VEHICLE.
[2]    TGGK←KTGG×UOV3 X IVK+.×UOV3←VLK[;3]
     ∇

     ∇ HBARS INT;T;SC;TSTOP
[1]    ⍝  DRAWS HORIZONTAL BARS ON GRAPH AT DESIRED INTERVALS.
[2]    SC←WHATSCALE
[3]    T←INT×⌈SC[3]÷INT
[4]    TSTOP←INT×⌊SC[4]÷INT
[5]    L:HMARK T
[6]    →(TSTOP≥T←T+INT)/L
     ∇

     ∇ HELP
[1]    ⍝  THIS FUNCTION ABORTS ANY SUSPENDED APL FUNCTIONS AND ALLOW
       S THE USER
[2]    ⍝  TO REGAIN CONTROL AND START ANY INTERRUPTED OPERATION OVER
       .
[3]    'REENTER FUNCTION AND START OVER'
[4]    ∊')SI CLEAR '
     ∇

     ∇ H←HESSEN A;N;I;R;RI;IDN;LI;S
[1]    I←1⍝TRANSFORMS INPUT MATRIX TO HESSENBURG FORM.
[2]    N←1↑⍴H←A
[3]    L:LI←(⍳I),I+⍒|H[I+⍳N-I;I]
[4]    H←H[LI;LI]
[5]    R←H[S;I]÷H[I+1;I];S←1+I+⍳N-I+1
[6]    H[S;]←H[S;]-R∘.×H[I+1;]
[7]    H[;I+1]←H[;I+1]+H[;S]+.×R
[8]    H[S;I]←0
[9]    →(N>I←I+1)/L
     ∇

     ∇ R←HEXFD X
[1]    R←16 16 16 16⊤X⍝  CONVERTS DECIMAL TO HEX DATA.
     ∇
```

```
    ∇ HMARK X
[1]   0 DRAW(X,X,X)VS 1 0 1\2↑WHATSCALE
    ∇

    ∇ Q←HOME
[1]   Q←8 ΔGRF 2⍝  HOMES CURSOR FOR GRAPHICS USE.
    ∇
```

```
     ∇ IC
[1]    ⍝  INITIALIZES ALL PARAMETERS FOR START OF SIMULATION RUN.
[2]    TTAPE←MIBSUM←T←0
[3]    OMBIAS←ΔTHL←ETACS←ΔE←ΔEP←ΔEN←ΔEMP←ΔPHH←ΔTHH←0 0 0
[4]    OMOT←OMOTIC
[5]    ΔTD2←ΔT÷2;ETLNOM←ETLNOM0
[6]    OMO←OMOK←(GME÷(REA+ALT+ALTIC)*3)*0.5
[7]    THDM←OMO←(((OMO×OMO)-(LADY×2○LAZK)*2)*0.5)-LADY×1○LAZK
[8]    TORB←(○2)÷OMO
[9]    KTGG←3×OMOK*2
[10]   TMID←TMID-TORB×TMID>0;TMID←-(OMOTIC-ETTM)÷OMO
[11]   EN←(OEK←(EUL3 LAZK)+.×EUL2 LAYK)[2;]
[12]   IEGEN;ENK←ENKIC;USEK←(EUL3-PHZ)+.×(1○GAYK),0,2○GAYK←GAYKIC+
       OMSUN×T
[13]   LEK←(EUL2 OMOT+ETYK←○¯1)+.×OEK
[14]   RMAG←MAG R←(REA+ALT)×LEK[3;];RD←(OMOK+ΔOMOK×SEEKFL=0)×(REA+
       ALT)×LEK[1;]
[15]   ⍝   RANIC
[16]   →(SEEKFL=0)/L2
[17]   VLK0←SEEK VLK0
[18]   SLOPE0←-(H÷ΔT)×⌹M
[19]   L2:VLK←AFQ(QDB QFA VLK0)+.×QVLKOFF
[20]   ER←R÷RMAG←(R2←+/R×R)*0.5
[21]   EN←EN÷MAG EN←R X RD
[22]   EX←EN X ER
[23]   QVEK←QFA VEK←VLK+.×LEK
[24]   POWER
[25]   VLEST←VL←AL←AFQ QAL←(QDB QFA VLK0)+.×QALOFF
[26]   ETS←ETSOFF+VL+.×ETLNOM
[27]   THD←THDM×AL[;2]
[28]   HV←IV+.×OMV←VL+.×0,OMO,0
[29]   OMVK←(OMV÷SF)-OMDK
[30]   HVK←IVK+.×OMVK
[31]   HCMGK←H×ET←ETS-HV÷H
[32]   HTEK←(HTVK←HVK+HCMGK)+.×VEK
[33]   PRIC
[34]   ΔE←ETS-AL+.×ETLNOM
[35]   GGTORQUE
[36]   AETORQUE;AECONST;OMGEK←OMGK×0 1 0;RHOTAB←RHOTAB0+RHOSIGK×
       RHOTAB1-RHOTAB0
[37]   DERDD←(GACCEL R)+(FAEK÷MSKY)+.×VEK
[38]   DERDDD←RD DDDDER R
[39]   TEKP←TEK←(⍉VEK)+.×TAEK+TGGK
[40]   OUTPUT;ΔTHRAW←0
[41]   STATIONTAB←STATIONTABIC
[42]   STATIONTAB←86220|STATIONTAB+((STSH-GMT0),0)∘.×25⍴1
[43]   STATIONTAB←STATIONTAB[;⍋STATIONTAB[1;]]
[44]   GBIAS←3 3⍴0
[45]   SLOPE←SLOPE0
[46]   ILEG←1
     ∇
```

```
    ∇ IEGEN;US1;US2;US3;SBE;CBE
[1]   ⍝  COMPUTES [IE] FROM GIVEN SIMULATION DATA.
[2]   US3←USEK
[3]   US1←US1÷MAG US1←EN X USEK
[4]   US2←US3 X US1
[5]   CBE←US2+.×EN;BETA←DFR×¯1○SBE←US3+.×EN
[6]   NUZEK←(-¯3○K[1;2]÷K[1;1])+¯1○-K[1;3]×SBE÷CBE×((K[1;1]*2)+K[1;
      2]*2)*0.5
[7]   UI1K←(US1×2○NUZEK)+US2×1○NUZEK
[8]   UI2K←USEK X UI1K
[9]   IKEK←UI1K,[1]UI2K,[0.1]USEK
[10]  OEK←OEK÷MAG OEK←0 1 0 X EN
[11]  OEK←EN,[0.1]OEK
[12]  OEK←(EN X OEK[2;]),[1]OEK
[13]  ETYK←(USEK+.×OEK[1;])ATAN USEK+.×OEK[3;]
[14]  DARK←-((((REA+ALT)*2)-REA*2)*0.5)÷REA+ALT
    ∇

    ∇ RES←A JACOBIAN VLK;ΔVLK;T0;T
[1]   ⍝  NUMERICALLY COMPUTES GRADIENT OF AERODYNAMIC + GRAVITY GRA
      DIENT
[2]   ⍝  TORQUE WITH RESPECT TO SMALL ROTATIONS ABOUT BODY AXES.
[3]   T0←TRQ VLK
[4]   ΔVLK←EUL1 A
[5]   T←TRQ ΔVLK+.×VLK
[6]   RES←(÷A)×T-T0
[7]   ΔVLK←EUL2 A
[8]   T←TRQ ΔVLK+.×VLK
[9]   RES←RES,[1.1](÷A)×T-T0
[10]  ΔVLK←EUL3 A
[11]  T←TRQ ΔVLK+.×VLK
[12]  RES←RES,[2](÷A)×T-T0
    ∇

    ∇ R←L LINC A
[1]   R←A÷⌈/1,|A÷L⍝  PROPORTIONAL LIMIT ON LARGEST COMPONENT.
    ∇

    ∇ R←L LIMM A
[1]   R←A÷⌈/1,MAG A÷L⍝  PROPORTIONAL LIMIT ON MAGNITUDE.∇
    ∇
```

```
    ∇ LOOP
[1]   ⍝ MAIN SIMULATION DRIVER FUNCTION.
[2]   L1:GGTORQUE
[3]   AETORQUE
[4]   POWER
[5]   DERDDD←RD DDDDER R
[6]   DERDP←DERD←RD-ONOK×RMAG×EX;DERDD←(GACCEL R)+(FFAKE×LEK[1;]+
      MSKY)+(FAEK+MSKY)+.×VEK
[7]   TEK←(TAEK+TGGK)+.×VEK
[8]   OMV←VL+.×0,ONO,0
[9]   OMVK←(OMV+SF)-ONDK+ONBIAS
[10]  QVEK←OMVK UPDATE QVEK
[11]  ONOT←ONOT+ΔT×ONO
[12]  ⎕←T←T+ΔT
[13]  DERD←DERD+ΔTD2×(2×DERDD)+ΔT×DERDDD
[14]  ⍝ DER←(ΔT×DERD)+(ΔT×ΔTD2×DERDD)+ΔT×ΔT×ΔT×(÷6)×DERDDD
[15]  DER←ΔTD2×DERDP+DERD
[16]  R←(RMAG×(ER×2○ONOK×ΔT)+EX×1○ONOK×ΔT)+DER
[17]  RD←DERD+ONOK×EN X R-DER
[18]  REF
[19]  USEK←(EUL3-PHZ)+.×(1○GAYK),0,2○GAYK←GAYKIC+OMSUN×T
[20]  POWER;ENDKP←ENDK
[21]  ENK←2500⌊ENK+ΔTD2×ENDK+ENDKP-2×PWRDR
[22]  AETORQUE;AECONST
[23]  GGTORQUE
[24]  TEKPP←TEKP
[25]  TEKP←TEK
[26]  TEK←(TAEK+TGGK)+.×VEK
[27]  HTEK←HTEK+ΔTD2×TEK+TEKP-PERC×TEKP-TEKPP
[28]  HTVK←VEK+.×HTEK
[29]  HVK←IVK+.×OMVK
[30]  HCMGK←HTVK-HVK
[31]  ATMDC
[32]  OUTPUT
[33]  →(T<TSTOP⌈STATIONTAB[1;1])/L1
[34]  →(T≥TSTOP)/0
[35]  TTAPE←TTAPE,STATIONTAB[1;1]
[36]  STNAME←STATIONTAB[2;1]
[37]  STATIONTAB←0 1↓STATIONTAB,STATIONTAB[;1]+86220,0
[38]  →((+/PRENK[1;]<0)=0)/0
[39]  'ATTITUDE CONTROL LOST DUE TO LOSS OF POWER'
[40]  →
    ∇
```

```
     ∇ N←MAG V
[1]    N←(+/[1]V×V)*0.5⍝COMPUTES MAGNITUDES OF A SET OF COLUMN VECTO
       RS.
     ∇

     ∇ MARK N
[1]    0 DRAW(1 0 1\¯2↑WHATSCALE)VS N⍝  DRAWS VERTICAL LINE ON GRAPH
        AT N.
     ∇

     ∇ NAVUP
[1]    ⍝  SIMULATES A NAV UPDATE FOR SIMULATIONS OVER SEVERAL DAYS.
[2]    OMO←(((OMOK×OMOK)-(LADY×2○LAZK)*2)*0.5)-LADY×1○LAZK
[3]    TORB←○2÷OMO
     ∇

     ∇ NOM
[1]    ⍝  RETURNS RANDOMIZED PARAMETERS TO THEIR NOMINAL VALUES.
[2]    SF←1 1 1
[3]    QVLKOFF←0 0 0 1
[4]    ∆CG←0 0 0
[5]    RHOSIGK←RHOSIG
[6]    STSH←0
[7]    ETSOFF←0 0 0
[8]    ETLNOM0←ETLNOMNOM
     ∇
```

```
     ∇ R←NORN X
[1]    ⍝  SELECTS SET OF SAME SHAPE AS X FROM NORMAL DISTRIBUTION WI
       TH 0 MEAN
[2]    ⍝  AND 1 STANDARD DEVIATION.
[3]    R←(+/(?Xρ1000),-?Xρ1000)÷912.87;X←(ρX),5
     ∇

     ∇ R←NRMLIZ XY;N;X;Y;J;X0;Y0
[1]    ⍝  NORMALIZES COMPLEX EIGENVECTOR TO UNIT MAGNITUDE.
[2]    J←1↑⍒R←(X×X)+Y×Y←N↓XY,0ρX←N↑XY,0ρN←0.5×ρXY
[3]    R←(2,N)ρ((X0×X)+Y0×Y),-(X×Y0←Y[J])-Y×X0←X[J]
[4]    R←R÷(+/,R×R)*0.5
     ∇

     ∇ R←OCTFD X
[1]    R←8 8 8 8⊤X⍝  CONVERTS DECIMAL DATA TO OCTAL.
     ∇

     ∇ ORBMARK;T
[1]    ⍝  MARKS TIME PLOTS AT 1 ORBIT INTERVALS.
[2]    T←TORB-OMOTIC÷OMO
[3]    L:MARK T
[4]    →(TSTOP≥T←T+TORB)/L
     ∇

     ∇ OUTPUT
[1]    ⍝  SELECTS PARAMETERS FROM SIMULATION TO SAVE FOR PLOTTING.
[2]    PRT←PRT,T
[3]    PRTGGK←PRTGGK,TGGK
[4]    PRTAEK←PRTAEK,TAEK
[5]    PRHTEK←PRHTEK,HTEK
[6]    PRECNG←PRECNG,ECNGK÷H
[7]    PRONVK←PRONVK,ONVK
[8]    PRA←PRA,OEK+.×(⍉LEK)+.×-DEGA(⍉VLK)+.×VL
[9]    PRΔPH←PRΔPH,DFR×2×ΔPHH
[10]   PROMOK←PROMOK,(RD+.×EX)÷RMAG
[11]   PRCD←PRCD,MAG AECOEFF[4 5 6]
[12]   PRR←PRR,R
[13]   PRRD←PRRD,RD
[14]   PRENK←PRENK,ENK,ENDKP
[15]   PRQAL←PRQAL,QAL
[16]   PRΔE←PRΔE,ΔE
     ∇
```

```
    ∇ PAR;LLK
[1]   ⍝  LIST IMPORTANT SIMULATION PARAMETER DATA.
[2]   ' '
[3]   ' '
[4]   '                 ';NAME
[5]   ' '
[6]   ' '
[7]   'ALT     = ';ALT;' KM, ';ALT×MPK;' NM'
[8]   'ETLNOM = ';ETLNOM
[9]   'ΔELIN   = ';ΔELIN
[10]  'KΔTH    = ';KΔTH
[11]  'ΔPHLIN = ';ΔPHLIN;' RAD, ';DPR×ΔPHLIN;' DEG'
[12]  'ΔTHLIN = ';ΔTHLIN;' RAD, ';DPR×ΔTHLIN;' DEG'
[13]  'VLKO    = ';VLKO
[14]  'SLOPE   = ';SLOPE
[15]  'QAL     = ';QAL
[16]  'QVLK    = ';QFA VLK
[17]  'QVLKO   = ';QFA VLKO
[18]  'LLK     = ';LLK,MAG LLK←DEGA(⍉VL)+.×VLK
[19]  'QALOFF = ';QALOFF
[20]  'QVLKOFF= ';QVLKOFF
[21]  'ETVK    = ';HTVK÷H
[22]  'ETLK    = ';(⍉VLK)+.×HTVK÷H
[23]  'ΔCG     = ';ΔCG
[24]  'OMDK    = ';OMDK;' RAD/S    ';DPR×OMDK;' DEG/S'
[25]  'SF     = ';SF
[26]  'ΔT ,TSTOP,OMOTIC =';ΔT,TSTOP,DPR×OMOTIC
[27]  'SEEKFL,IBM =';SEEKFL,IBM
    ∇

    ∇ P←PFR R
[1]   P←MAG R⍝  COMPUTES SPHERICAL POLAR FROM RECTANGULAR REPRESENT
      ATION.
[2]   P←P,-¯1○R[3]÷P
[3]   P←P,R[2]ATAN R[1]
    ∇

    ∇ POWER
[1]   ⍝  THIS FUNCTION SUPPLIED BY ROY LANIER 3-2514
[2]   ⍝  IT COMPUTES THE NORMALIZED SOLAR ARRAY POWER FOR GIVEN SUN
[3]   ⍝  POSITION IN VEHICLE REFERENCE.
[4]   ENDK←USVK[3]×DARK<VLK[;3]+.×USVK←VEK+.×USEK
[5]   ENDK←0⌈ENDK-0.308×(1-ENDK)*3
    ∇
```

```
    ∇ PRIC
[1]   ⍝  INITIALIZES PLOT STORAGE ARRAYS.
[2]   PRONOK←PRCD←PRT←⍳0
[3]   PRΔE←PRΔPH←PRA←PRONVK←PRTGGK←PRTAEK←PRECNG←PRHTEK←3 0⍴0
[4]   PRPWRA←PRST←PRENK←2 0⍴0
[5]   PRRD←PRR←3 0⍴0
[6]   PRQAL←4 0⍴0
    ∇

    ∇ Q←L PUT R
[1]   ⍝  ALLOWS USER TO PUT LABEL R ON GRAPH AT POSITION L.
[2]   Q←4 ΔGRF 0⍴⎕←R,4 ΔGRF L
    ∇

    ∇ PWRANGLE;I;TT;S;I1;I1P;THO;THOE;US;U30;SPH;CPH;AL;LE;NAME;
      SMAX
[1]   ⍝  COMPUTES POWER ANGLE FROM FLIGHT DATA.
[2]   'INPUT QBL'
[3]   AL←⎕
[4]   'INPUT QVA'
[5]   AL←APQ(QDB⎕)+.×AL
[6]   'INPUT QVI'
[7]   LE←(⍉AL)+.×(APQ⎕)
[8]   US←LE[;3]
[9]   U30←AL[3;]
[10]  'INPUT ANGLE OF INCIDENCE IN DEGREES'
[11]  CSRA←2○RPD×⎕
[12]  SPH←(US[1]×U30[1])+US[3]×U30[3]
[13]  CPH←(US[1]×U30[3])-US[3]×U30[1]
[14]  THOE←DPR×(-SPH ATAN CPH)+(0,01)+1 ¯1×¯1○1⌊((CSRA-US[2]×U30[2]
      )÷((CPH×CPH)+SPH×SPH)*0.5)⌈¯1
[15]  THOE←(360×THOE<¯180)+(¯360×THOE>180)+THOE
[16]  '⎕POWER ANGLE ⎕,F8.2'ΔFMT THOE[1↑⍋|THOE]
    ∇

    ∇ QBLCHECK;I
[1]   ⍝  THIS FUNCTION DISPLAYS THE CURRENT QBL SET TO CHECK FOR ER
      RORS.
[2]   'THE CURRENT QBL SET IS AS FOLLOWS:'
[3]   'THE COLUMNS ARE DISPLAYED AS ROWS FOR CONCISENESS'
[4]   ' '
[5]   I←⍳¯1↑⍴QBL
[6]   'I4,4F13.7,I7,⎕:⎕,I2,⎕:⎕,I2,⎕:⎕,F5.2'ΔFMT⍉I,[1]QBL
    ∇
```

```
    ∇ QBLSET;PQALL;PQALI;I
[1]   ⍝  THIS FUNCTION IS USED AS AN AID IN ENTERING ASAP QBL DATA
      USED
[2]   ⍝  IN ESTIMATING STRAPDOWN ERRORS.
[3]   'INPUT QBL AND GMT IN DHMS FOR EACH POINT'
[4]   '8 ENTRIES ARE EXPECTED AT EACH POINT.'
[5]   'IF AN ERROR IS ENCOUNTERED, TYPE →L+2 AND REENTER'
[6]   PQALL←8 0⍴0
[7]   PQALI←⍞
[8]   →N×ι8≠⍴,PQALI←⍎PQALI
[9]   L:→(0.01<|1-+/PQALI[1 2 3 4]*2)/M
[10]  PQALL←PQALL,PQALI
[11]  'ENTER NEXT COLUMN, D TO DELETE LINE JUST ENTERED'
[12]  'OR T TO TERMINATE INPUT'
[13]  →('TD'∊PQALI←⍞)/LT,LD
[14]  →L-1
[15]  LT:'⎕QBL BUILT WITH ⎕,I3,⎕ COLUMNS.⎕'ΔFMT ¯1↑⍴PQALL
[16]  QBL←PQALL
[17]  'DO YOU WISH TO SAVE THIS QBL SET?'
[18]  'IF YES, ENTER Y, OTHERWISE JUST RETURN'
[19]  →(~'Y'∊,⍞)/0
[20]  QBLS←QBLS,QBL
[21]  'QBLS CONTAINS ';¯1↑⍴QBLS;' COLUMNS'
[22]  ⍎')SAVE'
[23]  →0
[24]  M:'INVALID DATA, REENTER'
[25]  →L+2
[26]  LD:'ENTER INDEX OF LINE TO BE DELETED'
[27]  PQALL←((8,I-1)↑PQALL),(0,I←⎕)↓PQALL
[28]  '⎕LINE ⎕,I3,⎕ DELETED⎕'ΔFMT I
[29]  →L+2
    ∇

    ∇ QBLSETSH;T;Q
[1]   ⍝  THIS FUNCTION DOES SAME AS QBLSET WITH ABBREVIATED INPUT.
[2]   'ENTER INITIAL TIME IN D H M S'
[3]   T←¯300+SFD⎕
[4]   QBL←4 0⍴0
[5]   'ENTER FIRST THREE COMPONENTS OF QBL'
[6]   'ENTER 0 WHEN FINISHED'
[7]   L1:→(0=+/Q←⎕)/L2
[8]   QBL←QBL,Q,(1-Q+.×Q)*0.5
[9]   →L1
[10]  L2:T←T+300×ι(⍴QBL)[2]
[11]  T←DFS T
[12]  QBL←⍉(⍉QBL),⍉T
    ∇
```

```
    ∇ P←QDB Q
[1]   ⍝  FORMS Q DOUBLE BAR.
[2]   P←(Q,-Q)[SDB]
    ∇

    ∇ Q←QDEG A;HANG;S
[1]   Q←4⍴0⍝  THIS FUNCTION COMPUTES QUATERNION FROM INPUT EIGENANG
      LE.
[2]   Q[4]←2○HANG←0.5×RPD×MAG A
[3]   S←1○HANG
[4]   Q[1 2 3]←S×A÷MAG A
    ∇

    ∇ P←QDT Q
[1]   ⍝  FORMS Q DOUBLE TILDE.
[2]   P←(Q,-Q)[SDT]
    ∇

    ∇ Q←QFA A;S;TRA;I
[1]   ⍝COMPUTES QUATERNIONS FROM DIRECTION COSINES
[2]   ⍝CODED BY J GLAESE 5-12-78
[3]   S←(A,[1]-A[3;2],A[1;3],A[2;1]),A[2;3],A[3;1],A[1;2],TRA←+/1 1
      ⍉A
[4]   I←(⍒(1 1⍉S←S+⍉S)+1-TRA)[1]
[5]   Q←(S[I;]+(-I)⌽0 0 0,1-TRA)÷2×(1+S[I;I]-TRA)*0.5
[6]   Q←Q×Q[4]÷|Q[4]
    ∇

    ∇ R←QIN Q
[1]   R←Q×¯1 ¯1 ¯1 1⍝  TAKES INVERS OF Q.
    ∇

    ∇ RTS←QRALG A;H;EP
[1]   ⍝  DRIVER FOR QR ALGORITHM.
[2]   ⍝  THIS VERSION OF THE QR ALGORITHM WAS CODED BY JOHN GLAESE
      3-20-79
[3]   ⍝  TO USE THIS ALGORITHM YOU ALSO NEED THE FOLLOWING FUNCTION
      S:
[4]   ⍝  FINDQR, HESSEN, MAG, QUADM, SPLIT.
[5]   H←HESSEN A;EP←1E¯12
[6]   RTS←SPLIT H
    ∇

    ∇ Z←QUAD A;B;W
[1]   →3×⍳0<W←A[2]-A[1]×(-B←-A[1]÷2)÷2,Z←⍳0
[2]   →0,Z←(B+W),0,(B-W←(|W)*0.5),0
[3]   Z←B,(-W),B,W←(|W)*0.5
[4]   ⍝  COMPUTES ROOTS FROM QUADRATIC FACTORS.
    ∇
```

```
     ∇ R←QUADM M;D
[1]    ⍝  COMPUTES EIGENVALUES FOR 2×2 MATRICES.
[2]    R←+/1 1⍉M
[3]    R←R,(M[1;1]×M[2;2])-M[1;2]×M[2;1]
[4]    D←(R[1]*2)-4×R[2]
[5]    →(D≥0)/L
[6]    R←0.5×R[1],D,R[1],-D←(-D)*0.5
[7]    →0
[8]    L:R←0.5×(R[1]-D),0,(R[1]+D←((R[1]≥0)-R[1]<0)×(|D)*0.5),0
     ∇

     ∇ R←LSB QUANTIZE X
[1]    ⍝  QUANTIZES INPUT INTO MULTIPLES OF LSB.
[2]    R←LSB×⌊0.5+X÷LSB
     ∇

     ∇ RANIC;R
[1]    ⍝  COMPUTES RANDOM INITIAL CONDITIONS ON SELECTED PARAMETERS
[2]    ⍝  TO TEST OUR ABILITY TO MAINTAIN TEA CONTROL IN FACE OF
[3]    ⍝  UNCERTAIN DATA.
[4]    SF←1 1 1+0.01×1-0.02×?3⍴100
[5]    QVLKOFF←RFD×0.5×(LEK+.×USEK×NORM 1)+0.5×(LEK+.×IKEK[1;]×NORM
       1)+LEK+.×IKEK[2;]×NORM 1
[6]    QVLKOFF←QVLKOFF,(1-+/QVLKOFF*2)*0.5
[7]    ΔCG←2 1 1×0.0254×15×NORM 1 1 1
[8]    RHOSIGK←NORM 1
[9]    ETSOFF←0.1×1-0.002×?1000 1000 1000
     ∇

     ∇ RANPAR
[1]    ⍝  DISPLAYS VALUES OF RANDOMIZED PARAMETERS.
[2]    '⎕          SF          =    ⎕,3F8.2'ΔFMT 1 3⍴(100×SF-1)
[3]    '⎕          CP          =    ⎕,3F8.2'ΔFMT 1 3⍴-ΔCG
[4]    '⎕          OFF         =    ⎕,3F8.2'ΔFMT 1 3⍴DEGQ QVLKOFF
[5]    '⎕       ETS OFFSET     =    ⎕,3F8.2'ΔFMT 1 3⍴ETSOFF
[6]    '⎕  SUNSPOT SIGMA       =    ⎕,F8.2'ΔFMT RHOSIGK
     ∇
```

```
     ∇ RDTAPE;I;TT;S;I1;I1P;THO;THOE;US;U30;SPH;CPH;AL;LE;NAME;SMAX
[1]    ⍝  SIMULATES READING ASAP TAPE DATA AND COMPUTING SUN SENSOR
       OR
[2]    ⍝  POWER ANGLE DATA AS WELL AS QBL DATA.
[3]    I←+/TTAPE≤T-1800;NAME←1 5⍴'POWER';I'P←0
[4]    →(I>0)/L1
[5]    'NO TAPE DATA AVAILABLE'
[6]    ' '
[7]    →L4
[8]    L1:TT←TTAPE[I]
[9]    I←+/PRT≤TT-TORB
[10]   →(I>0)/L3
[11]   'NO TAPE DATA AVAILABLE YET'
[12]   →L4
[13]   L3:I←I,ILEG←+/PRT≤TT
[14]   S←PRENK[2;I[1]+0,⍳I[2]-I[1]];PRBANK←PRT[I[1]+0,⍳I[2]-I[1]],[
       0.1](BAR PRQAL[;I[1]+0,⍳I[2]-I[1]])[1;]
[15]   I1←S⍳0;I←I-1
[16]   I1←(I1-1)+(((I1-1)⌽S)≠0)⍳1
[17]   →((I1<⍴S)×I1>1)/L2
[18]   'NO SUNRISE DETECTED'
[19]   →L4
[20]   L2:THO←0;CSRA←FPWR S[I1]
[21]   LE←PRR[;I[1]+I1-1]
[22]   LE←LE÷MAG LE
[23]   LE←(LE X PRRD[;I[1]+I1-1]),[0.1]LE
[24]   LE[1;]←LE[1;]÷MAG LE[1;]
[25]   LE←(LE[1;]X LE[2;]),[1]LE
[26]   U30←AL[3;];AL←(EUL1 AL[3])+.×(EUL2 AL[2])+.×EUL1(AL←RFD×,BAR
       PRQAL[;I[1]+I1-1])[1]
[27]   US←LE+.×(EUL3-PHZ)+.×(1○GAYK),0,2○GAYK←GAYKIC+OMSUN×PRT[I[1]+
       I1-1]
[28]   SPH←(US[1]×U30[1])+US[3]×U30[3]
[29]   CPH←-(-US[1]×U30[3])+US[3]×U30[1]
[30]   THOE←DFR×(-SPH ATAN CPH)+(○0 1)+1 ¯1×¯1○(CSRA-US[2]×U30[2])÷(
       (CPH*2)+SPH*2)*0.5
[31]   THOE←THOE[1↑⍋|THOE]
[32]   '5A1,⎕ ANGLE⎕,F6.2,⎕ READ AT GMT ⎕,I3,⎕:⎕,I2,⎕:⎕,I2,⎕:⎕,F6.2'
       ∆FMT(NAME;1 5⍴THOE,DFS GMT0+TLDT+PRT[I[1]+I1-1])
[33]   PRPWRA←PRPWRA,PRT[I[1]+I1],THOE-THO
[34]   L4:→(USVK[3]<0.94)/0
[35]   CSRA←USVK[2]
[36]   LE←LEK
[37]   U30←AL[2;];AL←(EUL1 AL[3])+.×(EUL2 AL[2])+.×EUL1(AL←RFD×,BAR
       PRQAL[;¯1+⍴PRT])[1]
[38]   US←LE+.×USEK
[39]   SPH←(US[1]×U30[1])+US[3]×U30[3]
[40]   CPH←(US[1]×U30[3])-US[3]×U30[1]
[41]   THOE←DFR×(-SPH ATAN CPH)+(0,○1)+1 ¯1×¯1○1⌊((CSRA-US[2]×U30[2]
       )÷((CPH×CPH)+SPH×SPH)*0.5)⌈¯1
[42]   'LIVE SUN SENSOR DATA'
[43]   THOE←(360×THOE<¯180)+(¯360×THOE>180)+THOE
[44]   '⎕SUN SENSOR ANGLE ⎕,F8.2'∆FMT THOE[1↑⍋|THOE]
     ∇
```

```
     ∇ REF
[1]    ⍝  GENERATES ALL REQUIRED ROTATIONAL TRANSFORMATIONS FOR SIMU
       LATION.
[2]    ER←R÷RMAG←(R2←+/R×R)*0.5
[3]    EN←EN÷MAG EN←R X RD
[4]    EX←EN X ER
[5]    LEK←EX,[1]EN,[0.1]ER
[6]    VEK←AFQ QVEK
[7]    VLK←VEK+.×⍉LEK
     ∇

     ∇ REFRESH
[1]    ⍝  ERASES THE DISPLAY AND PUTS LATEST DATA EVALUATION BLOCK U
       P.
[2]    ERASE
[3]    TEAEVAL
[4]    RDTAPE
     ∇

     ∇ REPRIC;N
[1]    ⍝  THROWS AWAY OLD DATA FROM PLOT ARRAYS SO THAT ENOUGH MEMOR
       Y IS
[2]    ⍝  IS AVAILABLE TO CONTINUE.
[3]    N←+/PRT<T-2×TORB
[4]    PRT←N↓PRT
[5]    PRCD←N↓PRCD
[6]    PROMOK←N↓PROMOK
[7]    PRHTEK←N↓PRHTEK;N←0,N
[8]    PRECMG←N↓PRECMG
[9]    PRTAEK←N↓PRTAEK
[10]   PRTGGK←N↓PRTGGK
[11]   PROMVK←N↓PROMVK
[12]   PRA←N↓PRA
[13]   PRΔPH←N↓PRΔPH
[14]   PRENK←N↓PRENK
[15]   PRR←N↓PRR
[16]   PRRD←N↓PRRD
[17]   PRQAL←N↓PRQAL
[18]   PRΔE←N↓PRΔE
[19]   N←+/PRST[1;]<T-2×TORB;PRST[2;]←PRST[2;]-N[2]
[20]   PRST←(0,N)↓PRST
     ∇
```

```
    ∇ RUN;NAME
[1]    ⍝  TO SIMPLIFY RUNNING THE SIMULATION FOR TRAINING PURPOSES
[2]    ⍝  THIS FUNCTION SERVES AS AN EXECUTIVE TO DRIVE ALL THE VARI
       OUS
[3]    ⍝  SUBFUNCTIONS REQUIRED SO THAT THE USER DOES NOT HAVE TO BE
[4]    ⍝  AN APL EXPERT.
[5]    IC
[6]    LOOP
[7]    →(T≥TSTOP)/0
[8]    RDTAPE;TEAEVAL
[9]    ' '
[10]   'DO YOU WISH TO CHANGE ANY PARAMETERS NOW OR TERMINATE RUN?'
[11]   'IF YES, ENTER T, Y OR F TO TERMINATE,CHANGE OR EVAL. A FUNCT
       ION'
[12]   'IF YOU TERMINATE , YOU MAY RESUME BY TYPING RUNON AND RETURN
       .'
[13]   'IF NO CHANGE IS DESIRED, HIT CARRIAGE RETURN TO CONTINUE.'
[14]   'IF AN ERROR IS ENCOUNTERED, TYPE →6 AND RETURN.'
[15]   ' '
[16]   L0:→((1<⍴NAME)∨((NAME←,⍞)∊'TYF')=0)/L3
[17]   →('TYF'∊NAME)/0,L1,L2
[18]   →2
[19]   L1:'ENTER NAME OF VARIABLE TO BE CHANGED'
[20]   NAME←∊⍞
[21]   →(0=(1↑NAME)∊'∆ABCDEFGHIJKLMNOPQRSTUVWXYZ')/ERROR
[22]   ' '
[23]   'ENTER NEW VALUE(S)'
[24]   ' '
[25]   ∊NAME,'←⎕'
[26]   OMBIAS[1]←(0.1×RFD×2*¯10)×0.5×GBIAS[1;1]+GBIAS[3;1]
[27]   OMBIAS[2]←(0.1×RFD×2*¯10)×GBIAS[3;2]
[28]   OMBIAS[3]←(0.1×RFD×2*¯10)×0.5×GBIAS[1;3]+GBIAS[2;3]
[29]   'ANY OTHER CHANGES?'
[30]   →7
[31]   ERROR:'IMPROPER VARIABLE NAME'
[32]   'DO YOU WANT TO STOP OR GO ON?'
[33]   ' '
[34]   →7
[35]   L2:'ENTER FUNCTION WITH INPUT OR OTHER EXPRESSION.'
[36]   'AFTER FUNCTION COMPLETION, HIT CARRIAGE RETURN TO RESUME SIM
       ULATION.'
[37]   'IF AN ERROR IS MADE HERE, TYPE →L2 AND RETURN.'
[38]   'YOU MAY HAVE TO EXECUTE RUNON '
[39]   ∊⍞
[40]   'ANY OTHER CHANGES?';⍞
[41]   →7
[42]   L3:'ENTRIES RESTRICTED TO T, Y OR F.'
[43]   →L0
    ∇
```

```
    ∇ RUNON;NAME
[1]    ⍝  THIS FUNCTION DOES SAME AS RUN EXCEPT THE RANDOMIZED PARAM
       ETERS ARE
[2]    ⍝  NOT CHANGED FROM THEIR CURRENT VALUES.
[3]    ⍝ IC
[4]    LOOP
[5]    →(T≥TSTOP)/0
[6]    RDTAPE;TEAEVAL
[7]    ' '
[8]    'DO YOU WISH TO CHANGE ANY PARAMETERS NOW OR TERMINATE RUN?'
[9]    'IF YES, ENTER T, Y OR F TO TERMINATE,CHANGE OR EVAL. A FUNCT
       ION'
[10]   'IF YOU TERMINATE , YOU MAY RESUME BY TYPING RUNON AND RETURN
       .'
[11]   'IF NO CHANGE IS DESIRED, HIT CARRIAGE RETURN TO CONTINUE.'
[12]   'IF AN ERROR IS ENCOUNTERED, TYPE →6 AND RETURN.'
[13]   ' '
[14]   L0:→((1<⍴NAME)∨((NAME←,⍞)∊'TYF')=0)/L3
[15]   →('TYF'∊NAME)/0,L1,L2
[16]   →2
[17]   L1:'ENTER NAME OF VARIABLE TO BE CHANGED'
[18]   NAME←∊⍞
[19]   →(0=(1↑NAME)∊'∆ABCDEFGHIJKLMNOPQRSTUVWXYZ')/ERROR
[20]   ' '
[21]   'ENTER NEW VALUE(S)'
[22]   ' '
[23]   ∊NAME,'←⎕'
[24]   OMBIAS[1]←(0.1×RFD×2*¯10)×0.5×GBIAS[1;1]+GBIAS[3;1]
[25]   OMBIAS[2]←(0.1×RFD×2*¯10)×GBIAS[3;2]
[26]   OMBIAS[3]←(0.1×RFD×2*¯10)×0.5×GBIAS[1;3]+GBIAS[2;3]
[27]   'ANY OTHER CHANGES?'
[28]   →7
[29]   ERROR:'IMPROPER VARIABLE NAME'
[30]   'DO YOU WANT TO STOP OR GO ON?'
[31]   ' '
[32]   →7
[33]   L2:'ENTER FUNCTION WITH INPUT OR OTHER EXPRESSION.'
[34]   'AFTER FUNCTION COMPLETION, HIT CARRIAGE RETURN TO RESUME SIM
       ULATION.'
[35]   'IF AN ERROR IS MADE HERE, TYPE →L2 AND RETURN.'
[36]   'YOU MAY HAVE TO EXECUTE RUNON '
[37]   ∊⎕
[38]   'ANY OTHER CHANGES?';⍞
[39]   →7
[40]   L3:'ENTRIES RESTRICTED TO T, Y OR F.'
[41]   →L0
    ∇
```

```
    ∇ Q←SCALE R
[1]   →2+∨/Q←(∧/0 4 6≠ρR),~∧/</3 2ρ(R←,R),1,2
[2]   →0,Q←3 ΔGRF R
[3]   Q←0ρ⎕←6⌽' ERRORSCALE ',,(2 6ρ'LENGTHDOMAIN')[Q⍳1;]
[4]   ⍝  ALLOWS USER TO SET SCALE FOR GRAPHICS PROGRAMS.
    ∇

    ∇ RES←SEEK VLK;TO
[1]   ⍝  THIS FUNCTIONS DRIVES THE ITERATIVE PROCESS WHICH SEEKS A
      TEA.
[2]   OMGEK←(2○LAZK)×OMGK×0 1 0+.×LEK
[3]   M←1E¯6 JACOBIAN VLK
[4]   TO←TRQ VLK
[5]   ⎕←ΔTH←-TO⌹M
[6]   RES←VLK←(EUL1 ΔTH[1])+.×(EUL2 ΔTH[2])+.×(EUL3 ΔTH[3])+.×VLK
[7]   →(1E¯8≤MAG ΔTH)/1
    ∇

    ∇ SETSTATIONS;NAME
[1]   ⍝  SETS UP A SERIES OF SIMULATED GROUND STATIONS OVER WHICH
[2]   ⍝  DATA IS AVAILABLE FROM THE SIMULATION.  NO DATA IS AVAILAB
      LE
[3]   ⍝  ELSEWHERE.  THIS IS FOR GROUND CONTROLLER TRAINING.
[4]   'INPUT STATION AOS TIME IN HRS MINS SECS'
[5]   NAME←⎕
[6]   NAME←(365 24 60 60⊥DOY,NAME[1 2],0),NAME[3]
[7]   STATIONTABIC←STATIONTABIC,NAME
[8]   →1
    ∇
```

```
     ∇ SETUP
[1]    ⍝  SIMPLIFIES SETTING UP SIMULATION TO MATCH FLIGHT CONDITION
       S.
[2]    'INPUT LAMBDA Y'
[3]    LAYK←RFD×⎕
[4]    'INPUT GAMMA Y'
[5]    GAYK←GAYKIC←RFD×⎕
[6]    'INPUT GMT TGMT AND TMID CORRESPONDING TO ABOVE DATA'
[7]    'INPUT GMT'
[8]    GMT00←SFD⎕
[9]    'INPUT CURRENT ALTITUDE IN NM'
[10]   ALTIC←KFN×⎕
[11]   'INPUT TGMT0'
[12]   TGMT0←SFD⎕
[13]   'INPUT TMID'
[14]   TMID←SFD⎕
[15]   TMID←TMID+GMT00-TGMT0
[16]   'INPUT REGRESSION RATE DEG/DAY'
[17]   LADY←RFD×⎕÷86400
[18]   'INPUT DESIRED GMT'
[19]   GMT0←SFD⎕
[20]   LAYK←LAYK+LADY×GMT0-GMT00
[21]   GAYKIC←GAYK←GAYKIC+OMSUN×GMT0-GMT00
     ∇

     ∇ R←SFD DHMS
[1]    ⍝  COMPUTES TIME IN SECONDS FROM INPUT DHMS.
[2]    R←365 24 60 60⊥DHMS
     ∇
```

```
    ∇ SKYLAB ANAME;I;A
[1]    ⍝  DRAWS IDEALIZED SKYLAB VEHICLE IN STEREO PROJECTIONS.
[2]    STRAPIS 5
[3]    SCALE SC
[4]    DATE;A←⍉εANAME
[5]    STEREO XAXIS
[6]    STEREO LETTERX÷3 6ρ(6ρ6),12ρ0
[7]    STEREO YAXIS
[8]    STEREO LETTERY÷3 6ρ(6ρ0),(6ρ6),6ρ0
[9]    STEREO ZAXIS
[10]   STEREO LETTERZ÷3 6ρ(12ρ0),6ρ6
[11]   STEREO A+.×PXEND
[12]   STEREO A+.×MXEND
[13]   I←1
[14]   L:STEREO A+.×(EUL1 I×○÷4)+.×SIDE1
[15]   →(8≥I←I+1)/L
[16]   STEREO A+.×ATM
[17]   STEREO A+.×BASE
[18]   STEREO A+.×OWS
[19]   ' '
[20]   'EQUILIBRIUM ATTITUDE ',ANAME;⍉A
[21]   ' '
[22]   'BANK ANGLE, ANGLE OF ATTACK, ROLL ANGLE = ';DFR×1 2 1 EAFA⍉
       A
    ∇

    ∇ KSL SLUPDATE I;S;M;N;SL
[1]    ⍝  GENERATES SLOPE MATRIX UPDATE DATA IN PROPER FORMAT FOR UP
       LINK.
[2]    DATE
[3]    '⎕ETLN1,2,3 = ⎕,3F6.2,⎕  DOY = ⎕,F6.2,⎕  ALT = ⎕,F5.0,⎕  BETA
        = ⎕,F6.2'ΔFMT 1 6ρTABBIAS[I;],TABDOY[I],TABALTNM[I],TABBETA[
       I]
[4]    ' '
[5]    ' KSL = ';KSL
[6]    '                                  SLOPE MATRIX UPDATE DCS COMMAND
       S'
[7]    ' '
[8]    '                                  52002         UPDATED SLOPE'
[9]    ' '
[10]   ' '
[11]   SL←TABSL[I;;]×KSL
[12]   SL←(⌊0.5+1000×SL)÷1000
[13]   M←1
[14]   LM:N←1
[15]   LN:S←ENC SL[M;N]
[16]   'X29,I5,⎕         SLOPE(⎕,I1,⎕,⎕,I1,⎕) = ⎕,F6.3'ΔFMT(S[1];M;N;
       SL[M;N])
[17]   'X29,I5'ΔFMT S[2]
[18]   ' '
[19]   →(3≥N←N+1)/LN
[20]   →(3≥M←M+1)/LM
    ∇
```

```
    ∇ SPIN N;X;I
[1]   I←1⍝  THIS FUNCTION ADJUSTS THE SEED FOR THE RANDOM NUMBER GE
      NERATOR
[2]   X←?1000000
[3]   →(N≥I←I+1)/2
    ∇

    ∇ R←SPLIT H;N;LD;I;TR;TRT;TRP;ILD
[1]   ⍝  SPLITS OFF LINEAR OR QUADRATIC FACTORS FOR THE QR ALGORITH
      M.
[2]   R←0 2⍴0
[3]   TOP:→(0 1 2=N)/0,L3,L1;I←0;N←(⍴H)[1];TR←TRT←0 0
[4]   L0:H←FINDQR H;I←I+1
[5]   →(I>30)/L4
[6]   LD←¯2↑1 1⍉1 0↓H
[7]   →(EP>|LD)/L1,L3
[8]   →(</|LD)/LL1
[9]   TRT←TR←H[N;N],0;TRP←TRT
[10]  →(0.25>(MAG TR-TRP)÷MAG TR)/L0
[11]  TR←0 0
[12]  →L0
[13]  LL1:TR←2 2⍴QUADM ¯2 ¯2↑H;TRP←TRT
[14]  TRT←TR←,TR[1↑⍋|TR[;1]-H[N;N];]
[15]  →(0.25>(MAG TR-TRP)÷MAG TR)/L0
[16]  TR←0 0
[17]  →L0
[18]  L3:→TOP;H←¯1 ¯1↓H;R←R,[1]H[N;N],0
[19]  L1:→TOP;H←¯2 ¯2↓H;R←R,[1]2 2⍴QUADM ¯2 ¯2↑H
[20]  L4:'SLOW OR NONCONVERGENCE'
[21]  'OF THE MATRIX H';H
[22]  R←(N,2)⍴0
    ∇
```

```
    ∇ SSANGLE;S;A
[1]   ⍝  COMPUTES SUN SENSOR ANGLE FROM FLIGHT DATA.
[2]   ⍝  THIS FUNCTION ALLOWS INTERPOLATION BETWEEN MORE THAN ONE T
      IME
[3]   ⍝  POINT FOR INCREASED ACCURACY.
[4]   'INPUT QUATERNION DATA AS REQUIRED'
[5]   'INPUT X SUN SENSOR READING'
[6]   S←⎕
[7]   A←SSANGLE0
[8]   'IF YOU WISH TO INTERPOLATE SUN SENSOR ANGLES,'
[9]   'HIT CARRIAGE RETURN; OTHERWISE ENTER T TO TERMINATE'
[10]  →(~'T'∊,⎕)/LI
[11]  'SUN SENSOR ANGLE ';A
[12]  →0
[13]  LI:'INPUT 2ND SET OF DATA'
[14]  'INPUT X SUN SENSOR READOUT FROM FORMAT 7'
[15]  S←S,⎕
[16]  A←A,SSANGLE0
[17]  A←A[1]+(A[2]-A[1])×-S[1]÷S[2]-S[1]
[18]  'INTERPOLATED SUN SENSOR ANGLE ';A
    ∇

    ∇ THOE←SSANGLE0;I;TT;S;I1;I1P;THO;US;U30;SPH;CPH;AL;LE;NAME;
      SMAX
[1]   ⍝  THIS FUNCTION COMPUTES THE SUN SENSOR ANGLE FROM FLIGHT DA
      TA.
[2]   'INPUT QBL'
[3]   AL←⎕
[4]   'INPUT QVA'
[5]   AL←AFQ(QDB⎕)+.×AL
[6]   'INPUT QVI'
[7]   LE←(⍉AL)+.×(AFQ⎕)
[8]   US←LE[;3]
[9]   U30←AL[2;]
[10]  CSRA←0
[11]  SPH←(US[1]×U30[1])+US[3]×U30[3]
[12]  CPH←(US[1]×U30[3])-US[3]×U30[1]
[13]  THOE←DFR×(-SPH ATAN CPH)+(0,01)+1 ¯1ׯ1○1⌊((CSRA-US[2]×U30[2]
      )÷((CPH×CPH)+SPH×SPH)*0.5)⌈¯1
[14]  THOE←(360×THOE<¯180)+(¯360×THOE>180)+THOE
[15]  THOE←THOE[1+⍋|THOE]
    ∇

    ∇ STAXES
[1]   ⍝  THIS FUNCTION DRAWS THE STEREO AXES AND LABELS THEM WITH S
      TEREO LETTERS.
[2]   STEREO XAXIS
[3]   STEREO LETTERX+3 6⍴(6⍴6),12⍴0
[4]   STEREO YAXIS
[5]   STEREO LETTERY+3 6⍴(6⍴0),(6⍴6),6⍴0
[6]   STEREO ZAXIS
[7]   STEREO LETTERZ+3 6⍴(12⍴0),6⍴6
    ∇
```

```
    ∇ STEREO P;E;XR;XL;Y;N
[1]   ⍝  DRAWS STEREO PROJECTIONS OF POINT SET P USING PREVIOUSLY S
      PECIFIED
[2]   ⍝  EYE LOCATION, SEPARATION, LINE OF SIGHT, ETC.
[3]   P←(RT,[1]UP,[0.1]LOS)+.×P
[4]   N←(P[3;]-EYE[3])÷D
[5]   XL←(P[1;]+S-EYE[1])÷N;XR←(P[1;]-S+EYE[1])÷N
[6]   Y←(P[2;]-EYE[2])÷N
[7]   WINDOW 3.5 8 0.5 5
[8]   0 DRAW Y VS XL
[9]   WINDOW 0 4.5 0.5 5
[10]  0 DRAW Y VS XR
    ∇

    ∇ STPAR
[1]   ⍝  PRINTS OUT CURRENT STEREO PARAMETERS.
[2]   'LINE OF SIGHT           ';LOS
[3]   'OBSERVERS RIGHT         ';RT
[4]   'OBSERVERS UP            ';UP
[5]   'AVERAGE EYE POSITION    ';EYE×LOS
[6]   'EYE SEPARATION          ';S
[7]   'PAGE DISTANCE           ';D
    ∇

    ∇ Q←STRAPIS R
[1]   ⍝  SETS UP PROPER STRAPPING FOR GRAPHICS FUNCTIONS AT AN INST
      ALLATION.
[2]   ⍝  AT MSFC IT SHOULD BE SET TO 5 AND MUST BE EXECUTED ONCE AT
       LEAST
[3]   ⍝  FOR EACH TERMINAL SESSION.
[4]   →ERR×⍳(7<R)∨5>R←1↑R;Q←⍳0
[5]   →0;Q←11 ΔGRF R
[6]   ERR:'NO SUCH STRAPPING'
    ∇

    ∇ STROT;TR;TR1;TH
[1]   ⍝  MOVES LOS AND EYE ON SURFACE OF SPHERE FOR BETTER VIEWING.
[2]   'INPUT DESIRED AZIMUTH AND ELEVATION OF LOS'
[3]   'OBJECT TO OBSERVER, ANGLES IN DEGREES'
[4]   TR←RFD×¯1 1×⌽⎕
[5]   TH←1↓PFR-LOS
[6]   TR1←(EUL2 TH[1])+.×EUL3 TH[2]
[7]   TR←⍉(EUL2 TR[1])+.×EUL3 TR[2]
[8]   TR←TR+.×TR1
[9]   LOS←TR+.×LOS
[10]  UP←TR+.×UP
[11]  RT←TR+.×RT
    ∇
```

```
    ∇ STSETUP
[1]   ⍝  SETS OBSERVER PARAMETERS FOR STEREO VIEWING.
[2]   EYE←⎕;⎕←'OBSERVER LOCATION'
[3]   LOS←⎕;⎕←'OBSERVER LINE OF SIGHT IS?'
[4]   UP←⎕;⎕←'OBSERVER VERTICAL DIRECTION (NEED NOT BE EXACT)'
[5]   LOS←LOS÷MAG LOS
[6]   UP←UP-LOS×LOS+.×UP
[7]   UP←UP÷MAG UP
[8]   EYE←(RT,[1]UP,[0.1]LOS)+.×EYE;RT←LOS X UP
[9]   S←0.5×⎕;⎕←'EYE SEPARATION IN USER UNITS'
[10]  D←EYE[3]⍝EYE-SCREEN DISTANCE
    ∇
```

```
    ∇ TACS ETS
[1]    ⍝  THIS FUNCTION SIMULATES THE SKYLAB TACS FIRING LOGIC
[2]    →(1.92>MAG ETS)/0
[3]    FIRE←6 3⍴0
[4]    →(((|ETS[2])>(|ETS[3])+E3B)∧(|ETS[2])>|ETS[1])/L1
[5]    →((((|ETS[3])+E3B)≥|ETS[2])∨(|ETS[1])≥|ETS[2])/L3
[6]    →0
[7]    L1:→(ETS[2]≥0)/L2
[8]    FIRE[4;]←1 1 1
[9]    ⍝          'JET 4'
[10]   →L7
[11]   L2:FIRE[1;]←1 1 1
[12]   ⍝          'JET 1'
[13]   →L7
[14]   L3:→(ETS[1]<EB[1])/L5
[15]   →(ETS[3]≥0)/L4
[16]   FIRE[6;]←1 1 1
[17]   ⍝          'JET 6'
[18]   →L7
[19]   L4:FIRE[2;]←1 1 1
[20]   ⍝          'JET 2'
[21]   →L7
[22]   L5:→(ETS[3]≥0)/L6
[23]   FIRE[3;]←1 1 1
[24]   ⍝          'JET 3'
[25]   →L7
[26]   L6:FIRE[5;]←1 1 1
[27]   ⍝          'JET 5'
[28]   L7:HTEK←(⍉VEK)+.×HTVK←HTVK+ΔHTVK←+/MIB×LEVS X MIBS×⍉FIRE
[29]   ⍝          'T = ';T;'  ΔHTVK = ';ΔHTVK;'  ETOK = ';MAG HTOK÷H
[30]   ETACS←ETACS-ΔHTVK÷H
[31]   ETS←ETS+ΔHTVK÷H
[32]   ⎕←MIBSUM←MIBSUM+1
[33]   ⍝          'MIBS=';MIBSUM
[34]   →1
    ∇

    ∇ TEAEVAL
[1]    ⍝  THIS FUNCTION PRINTS A STATION BLOCK OF DATA TO HELP EVALU
       ATE
[2]    ⍝  TEA CONTROL PERFORMANCE.
[3]    ' '
[4]    STATION[STNAME;],,('X7,⎕  GMT  = ⎕,I3,⎕:⎕,I2,⎕:⎕,I2,⎕:⎕,F6.2'
       ΔFMT 1 4⍴DFS GMT0+T)
[5]    '⎕TIME TIL NEXT STATION  ⎕,I2,⎕:⎕,I2,⎕:⎕,F6.2,⎕       PRESENT A
       LT⎕,F6.0  'ΔFMT 1 4⍴(24 60 60⊤STATIONTAB[1;1]-T),MFK×ALT
[6]    '⎕ELAPSED SIM TIME = ⎕,I3,⎕:⎕,I2,⎕:⎕,I2,⎕:⎕,F6.2'ΔFMT 1 4⍴
       DFS T
[7]    '⎕BANK, ATTACK, ROLL ANGLE  =  ⎕,3F10.2'ΔFMT⍉BAR 4 ¯1↑PRQAL
[8]    PRST←PRST,T,PRT⍳T
[9]    ⍝  '⎕ORBITAL DRIFT ANGLE  ⎕,F10.2'ΔFMT ¯1↑PRCHI
    ∇
```

```
    ∇ TEXPERT;T;ΔCG;SLP;A;ONGEK;R;RD;SEEKFL
[1]   ⍝  THIS FUNCTION ESTIMATED THE CP OFFSET AND RECOMMENDED A MO
      MENTUM BIAS
[2]   ⍝  TO HELP US ACHIEVE OUR DESIRED ATTITUDE.
[3]   SEEKFL←1
[4]   ONGEK←(2○LAZK)×ONGK×LEK[2;];R←(REA+ALT)×LEK[3;];RD←OMOK×(REA+
      ALT)×LEK[1;]
[5]   ΔCG←-CPASUP
[6]   ⍝ 'INPUT CURRENT SLOPE MATRIX'
[7]   ⍝ SLP←⎕
[8]   ETLNOM←0 0 0
[9]   T←TRQ AL←1 2 1 AFEA RFD×EATNOM
[10]  'SUGGESTED MOMENTUM BIAS'
[11]  ⎕←ETL←-(÷H)×(÷OMO)×0 1 0 X T+.×AL
    ∇

    ∇ R←TOP A;L;P
[1]   ⍝  DETERMINES RANGE OF VARIATION AND ROUNDS OFF TO CONVENIENT
       NUMBER.
[2]   R←P÷L←10*⌊10⍟P←⌈/|,A
[3]   P←1.5 2 2.5 3 4 5 6 7 8 9 10
[4]   R←P[1++/R≥P]×L
    ∇
```

```
    ∇ TRANSLATE
[1]    ⍝  PROVIDES CONVERSION TABLE TO CONVERT SKYLAB TERMINOLOGY TO
[2]    ⍝  SIMULATION TERMINOLOGY.
[3]    'ETLN1 = ',ETLN1
[4]    'ETLN2 = ',ETLN2
[5]    'ETLN3 = ',ETLN3
[6]    'KDETHX= ',KDETHX
[7]    'KDETHZ= ',KDETHZ
[8]    'DETHL1= ',DETHL1
[9]    'DETHL2= ',DETHL2
[10]   'DETHL3= ',DETHL3
[11]   'DEPHL1= ',DEPHL1
[12]   'DEPHL2= ',DEPHL2
[13]   'DEPHL3= ',DEPHL3
[14]   'DEL1  = ',DEL1
[15]   'DEL2  = ',DEL2
[16]   'DEL3  = ',DEL3
[17]   'TDESAT= ',TDESAT
[18]   'DEPHC3L=',DEPHC3L
[19]   'PHD1DX= ',PHD1DX
[20]   'PHD1DY= ',PHD1DY
[21]   'PHD1DZ= ',PHD1DZ
[22]   'PHD2DX= ',PHD2DX
[23]   'PHD2DY= ',PHD2DY
[24]   'PHD2DZ= ',PHD2DZ
[25]   'PHD3DX= ',PHD3DX
[26]   'PHD3DY= ',PHD3DY
[27]   'PHD3DZ= ',PHD3DZ
    ∇

    ∇ RES←TRQ VLK;OMVK;HVK;VEK;RHOTAB
[1]    ⍝  COMPUTES EXTERNAL TORQUES.  USED IN SEEK AND JACOBIAN.
[2]    RHOTAB←RHOTAB0+RHOSIG×RHOTAB1-RHOTAB0
[3]    AETORQUE;GGTORQUE;VEK←VLK+.×LEK;AECONST
[4]    OMVK←VLK+.×0,OMOK,0
[5]    HVK←IVK+.×OMVK
[6]    RES←TGGK+TAEK-OMVK X H×VLK+.×ETLNOM
    ∇

    ∇ QNEW←OMA UPDATE QOLD;QOM;ΔPHI;MAGOM
[1]    ⍝  QUATERNION INTEGRATION FUNCTION.
[2]    UOM←OMA÷MAGOM←MAG OMA
[3]    ΔPHI←MAGOM×ΔTD2
[ ]    QOM←(UOM×1○ΔPHI),2○ΔPHI
[5]    QNEW←(QDB QOM)+.×QOLD
    ∇
```

```
    ∇ Q←L VS R;C;D;E
[1]   →2+∨/C←(∧/C≠1⌽C←1,(E←ρL←,L),D←1↓ρR←(¯2↑1 1,ρR)ρR),(1<ρρL)∨2<ρ
      ρR
[2]   →0×ρQ←(((1↑ρR),D)ρR),[1](D←E⌈D)ρL
[3]   ,(2 37ρ'AN ARGUMENT OF VS IS OF IMPROPER RANKARGUMENTS OF VS
      ARE NOT CONFORMABLE ')[(⌽C)⍳1;]
[4]   ⍝  TAKES 2 VECTORS OF SAME LENGTH AND CONSTRUCTS 2×N MATRIX F
      OR PLOTTING.
    ∇

    ∇ Q←WHATSCALE
[1]   Q←11 ΔGRF 1⍝  DISPLAYS CURRENT SCALE SETTING.
    ∇

    ∇ Q←WHATWINDOW
[1]   Q←11 ΔGRF 2⍝  DISPLAYS CURRENT WINDOW SETTING.
    ∇

    ∇ Q←WINDOW R
[1]   →2+∨/Q←(∧/0 4≠ρR),⍲/</2 2ρ(R←,R),⍳4
[2]   →ρQ←2 ΔGRF R
[3]   Q←0ρ⎕←6⌽' ERRORWINDOW ',,(2 6ρ'LENGTHDOMAIN')[Q⍳1;]
[4]   ⍝  ALLOWS USER TO SET NEW WINDOW SPECIFICATION.
    ∇

    ∇ C←A X B
[1]   ⍝ CROSS PRODUCT
[2]   C←((1⌽[1]A)×(¯1⌽[1]B))-(1⌽[1]B)×¯1⌽[1]A
    ∇

    ∇ XA
[1]   ⍝  CONSTRUCTS CROSSPLOTS OF THE IN PLANE STRAPDOWN ERROR.
[2]   1 XPLOT'PRA[3 1;]'
    ∇

    ∇ XH
[1]   ⍝  CROSSPLOTS IN PLANE COMPONENTS OF HTEK.
[2]   1 XPLOT'(OEK+.×PRHTEK)[3 1;]'
    ∇
```

```
    ∇ P XPLOT NAME;S;N;I;NI;Y1
[1]    ⍝  MAKES CROSS PLOT OF DESIRED DATA ON SQUARE WINDOW.
[2]    STRAPIS 5
[3]    DATE
[4]    I←(ρ∊NAME)[2]
[5]    NI←⌈I÷P
[6]    S←(P×⍳NI)-P-1
[7]    WINDOW 2 7 0.25 5.25
[8]    BOX
[9]    SCALE(-Y1),Y1,(-Y1),Y1←TOP⌈/[1]|∊NAME
[10]   AXES
[11]   0 DRAW(∊NAME)[;S]
[12]   '
'
[13]   'FINAL VALUE ',NAME,' = ';,(∊NAME)[;¯1↑ρ∊NAME]
    ∇
```

REFERENCES

1. Glaese, J. R. and Kennel, H. F.: Low Drag Attitude Control for Skylab Orbital Lifetime Extension. NASA TM X- .

2. Skylab Final Report.

3. Chubb, W. B., et al.: Flight Performance of Skylab Attitude and Pointing Control System. NASA TN D-8003, June 1975.

4. ATMDC Program Definition Document. IBM No. 70-207-0002, 10 May 1973.

5. Kennel, H. F.: A Control Law for Double-Gimbaled Control Moment Gyros Used for Space Vehicle Attitude Control. NASA TM X-64536, August 7, 1970.

6. Gyorfi, R. A.: Orbital Aerodynamic Data for the Updated Skylab I In-Orbit Configuration. M-9230-73-197, Northrop Services, Inc., July 11, 1973.

7. Parrott, J. D. and Warr, J. W., III: Orbital Aerodynamic Data for the Skylab Configuration and the Skylab Configuration and End-Docked Command and Service Module. LMSC-HREC D162857, TN 54/20-136, Lockheed Research and Engineering Center, Huntsville, Alabama, January 1971.

8. Thomason, H. E.: Skylab Center of Gravity, NASA MSFC, Memo EL01(411-78), December 1978.

APPROVAL

TORQUE EQUILIBRIUM ATTITUDE CONTROL FOR SKYLAB REENTRY

By John R. Glaese and Hans F. Kennel

The information in this report has been reviewed for technical content. Review of any information concerning Department of Defense or nuclear energy activities or programs has been made by the MSFC Security Classification Officer. This report, in its entirety, has been determined to be unclassified.

George D. Hopson

GEORGE D. HOPSON
Director, Systems Dynamics Laboratory