STUDY OF CONTROL COMPUTERS FOR CONTROL MOMENT GYRO STABILITY AND CONTROL SYSTEMS

VOLUME I (ENGINEERING)

N67-30155
(ACCESSION NUMBER)

146
(PAGES)

CR-66373
(NASA CR OR TMX OR AD NUMBER)

FACILITY FORM 602

(THRU)

/
(CODE)

08
(CATEGORY)

STUDY OF CONTROL COMPUTERS FOR CONTROL MOMENT GYRO STABILITY AND CONTROL SYSTEMS

By D.P. Bedford, H. Markarian, N.L. Pleszkoch

VOLUME I (ENGINEERING)

ABSTRACT

This 6 months' contractual study effort was performed under NASA Contract NAS1-6698 awarded to the IBM Corporation on 2 September 1966, in response to bids on the Langley Research Center's Statement of Work L-7035, dated 22 June 1966

The study has two objectives. The first objective was to perform an analysis of the control computer computational requirements. This included preliminary hardware sizing for a control moment gyro (CMG) stability and control system used in an Apollo Applications Spacecraft to perform typical control tasks required for experiments such as horizon spectrometry, earth mapping, and solar astronomy. The second objective was to develop a fixed word-length, digital model of the control computer to be incorporated in a sample-data simulation of the integrated CMG control system for manned spacecraft. The final report comprises two volumes. Volume I describes the overall engineering analyses, and Volume II discusses the digital simulation program from a user's viewpoint.

CONTENTS

CONTENTS (Continued)

ILLUSTRATIONS

also included. This simulation has been the primary working tool throughout the study. In addition to verifying analytical results, its application has been in the evaluation of system accuracy as a function of word length and sampling period, thereby providing definition for the digital control computer requirements.

SUMMARY

Three 2-degree-of-freedom control moment gyros (CMGs), nominally aligned with the spacecraft principal axes, form the nucleus of the attitude control system analyzed in this report. The spacecraft considered is representative of an Apollo CSM docked with the LM configuration. The basic control system elements (sensor package, control computer, controls and displays, etc.) are those required to perform the four specified orbital experiments: (1) earth mapping, (2) inertial mode, (3) horizon spectrometry, and (4) microwave transmission.

The primary objective of this study is to derive and define the functional requirements of the on-board digital control computer. These functions are: (1) computation of attitude commands (which require on-board navigation for two experiments), (2) computation of spacecraft attitude, which in two experiments requires updating a direction cosine matrix, (3) stability compensation, (4) control law computation, (5) command CMG gimbal desaturation as required, and (6) monitor CMG parameters and process data for controls and display. This last function requires a preliminary design for the controls and displays to provide realistic computer requirements.

Control law development is the most significant analytical work required for the performance of the contract task. Control law is here defined as the computational process of determining which of the six CMG gimbals will be commanded, and the associated proper rates, to reduce any existing attitude errors in three spacecraft control channels. Since the CMG torque output axis may be oriented anywhere with respect to the spacecraft axes, undesirable cross-channel control torques result unless the control law is capable of compensating for them. The report contains the developments of two control laws which essentially eliminate the undesirable cross-channel control torques. This results in a more efficient system providing better overall system response, and prolongs the gimbal desaturation intervals.

To verify analytical results and accurately assess the control computer requirements, a detailed digital simulation was developed. It solves the complete rigid body equations of motion of the CMGs and spacecraft in floating point arithmetic; the control computer equations are simulated in fixed point with variable word length option. All defined nonlinearities and error models are

$A_{a,b,g}$	element of CMG inertia tensor, lb-ft-sec^2
A	telescope azimuth angle (earth mapping experiment), or star tracker azimuth angle to boresight when the vehicle has the desired attitude
a	star tracker azimuth angle to boresight given the vehicle actual attitude
a_{ij}	element of total system generalized inertia tensor, lb-ft-sec^2
$\underline{a}_i$	CMG torque produced by $\alpha_i = 1 \text{ rad/sec}$, ft-lb
$B_{a,b,g}$	element of CMG inertia tensor, lb-ft-sec^2
$[B]$	matrix $[\underline{a}_1, \underline{a}_2, \underline{a}_3, \underline{b}_1, \underline{b}_2, \underline{b}_3]$
$\underline{b}_i$	CMG torque produced by $\dot{\beta}_i = 1 \text{ rad/sec}$, ft-lb
b_i	the left hand side of the ith equation of motion, ft-lb
$C_{a,b}$	element of CMG inertia tensor, lb-ft-sec^2
E	star tracker elevation angle to boresight when the vehicle has the desired attitude
e	star tracker elevation angle to boresight given the actual attitude of the vehicle
$F(z)$	sampled data compensation
f_c	cost function for iterative control law
$\underline{F}$	resultant external forces, lb
f	coefficient of the navigation f and g series

$G_\alpha, (G_\beta)$	outer (inner) CMG gimbal drive gear ratio
g	coefficient of the navigation f and g series
H	CMG spin momentum magnitude, lb-ft-sec
h	normalized CMG spin momentum vector
$\underline{i}, \underline{j}, \underline{k}$	unit vectors along vehicle x, y, z axes
$\underline{I}, \underline{J}, \underline{K}$	unit vectors along inertial X, Y, Z axes
$J_{m\alpha}, (J_{m\beta})$	outer (inner) gimbal torquer rotor moment of inertia, lb-ft-sec^2
K_c	control law gain
K_e, K_r	multipliers in the baseline control law
K_x, K_y, K_z	attitude control system gain terms, sec^{-1}
K_x'	rate feedback gain
K_s	algebraic control law parameter
$K_\alpha, K_\beta, K_{\alpha\beta}$	cost function constants
$K_{TOLERANCE}$	iterative control law parameter
k_i	linear combination of attitude and rate errors for the baseline control law
L	longitude of spacecraft
$[M_{ij}]$	direction cosine matrix, commanded body to inertial coordinates
$\underline{M}$	external torque vector, ft-lb
M_α	G_α times the torque produced by the outer gimbal torquer, ft-lb
M_β	G_β times the torque produced by the inner gimbal torquer, ft-lb
$M_{T\alpha}$	total torque interaction between CMG and the vehicle along the outer gimbal axis, ft-lb

$M_{T\beta}$	total torque interaction between the inner gimbal assembly and the outer gimbal ring along the inner gimbal axis, ft-lb
m	mass, slugs
$[m_{ij}]$	direction cosine matrix, body to inertial coordinates
Q	constant defining C.M. change, $m_M m_S / m_M + m_S$, slugs
$\underline{q}$	position vector of mass element measured from C.M. of the main vehicle, feet
R	gimbal rate $\dot{\alpha}$ or $\dot{\beta}$ in the iterative control law, rad/sec
$\underline{R}$	position vector of target from the geocenter, feet
$\underline{S}$	first moment of mass of the total system about the C.M. of the main vehicle, lb-sec^2
s	Laplace transform variable, rad/sec
$\underline{s}$	slant range vector, feet
$\underline{\dot{s}}_E$	slant range rate vector with respect to the earth, ft/sec
T	sampling interval, seconds
$\underline{T}_b$	torque due to CMG precession at vehicle rates, ft-lb
$\underline{T}_c$	remainder torque, $\underline{T}_d - T_b$, ft-lb
$\underline{T}_d$	desired torque, ft-lb
$\underline{u}$	normalized position vector of the vehicle
V_0	circular velocity at earth's radius, ft/sec
$\underline{V}$	slant range rate vector with respect to the earth, $\underline{\dot{s}}_E$, ft/sec
X, Y, Z	inertial coordinates
x, y, z	vehicle principal axes coordinates
Z_{EI}	local ellipsoid vertical

α	CMG outer gimbal angle
β	CMG inner gimbal angle
$\underline{\epsilon}$	negative of small angle attitude error
$\dot{\eta}$	nominal vehicle rate about the z axis in the horizon spectrometry experiment
θ	Euler angle
λ	vehicle geodedic latitude
λ_R	vehicle geocentric latitude
$\underline{\rho}$	position vector of the vehicle measured from the geocenter, feet
ρ_E	earth's radius, feet
φ	Euler angle
ψ	Euler angle
Ω	CMG spin rate, rad/sec
Ω_{EI}	earth's rate in inertial coordinates, rad/sec
$\underline{\omega}$	vehicle angular rate vector, rad/sec
$\underline{\omega}_i$	inertial rate of the ith CMG inner gimbal ring resolved along inner gimbal axes, rad/sec
$\underline{\omega}_i'$	inertial rate of the ith CMG outer gimbal ring resolved along outer gimbal axes, rad/sec

Superscripts:

$'$ (prime)	indicated
$''$ (double prime)	signal at the output of the stabilization compensation

Subscripts:

a	outer gimbal
b	inner gimbal
c	command
d	desired
g	gyro rotor
i	ith element
M	crew member
n	step of iteration
r	remainder in the iterative control law
S	total system excluding the moving mass
t	telescope

Section 1

INTRODUCTION

The use of momentum exchange devices for active spacecraft attitude control has evolved because of limitations of the simpler flight-proven reaction jet systems. The on-off or "bang-bang" nature of reaction jet systems precludes their use in many present-day missions with high pointing accuracy requirements. Secondly, the fuel weight penalty for long term missions becomes prohibitive. The continuous torque capability of momentum exchange devices allows much higher accuracy, and they are operable from energy supplied by solar panels or fuel cells.

The simplest momentum exchange concept, and the first to evolve, was the momentum wheel or "flywheel." The logical extension is the control moment gyro (CMG). The attending increased complexity is overshadowed by its advantages of considerably lower electrical power and input torquing requirements for spacecraft maneuvering, and greater pointing accuracy due to its faster system response. These mark the CMG as the next generation spacecraft attitude control actuator for missions which are either long term or require high attitude control accuracy.

The complexity of the equations of motion for systems employing CMGs presents a rather difficult systems analysis problem. This has resulted in the proposal and analysis of numerous CMG configurations. In most cases, one of the objectives has been to simplify the analytical problem by the choice of hardware configuration.

Perhaps the most demanding, from a systems analysis standpoint, is the configuration of three double-gimbaled CMGs. This configuration, which has been chosen by NASA's Langley Research Center for use in the Apollo Applications Program, results in a rather unique system in that there are six degrees of command freedom with which to control three degrees of spacecraft rotational freedom. This inherent redundancy provides higher system reliability, since attitude control can be maintained with only two CMGs operable, and presents a wide range of control law possibilities.

In support of Langley's CMG development program, under contract No. NAS1-6698, IBM undertook the task of determining the digital control computer functional requirements for the CMG attitude control system. One of the objectives of the study is to make use of the computational versatility of the digital computer to more effectively use the CMG's six degrees of control freedom. This phase of the study resulted in the development of two control laws. One of these, the "Iterative Control Law," has the flexibility of a built-in cost function which can be tailored to meet specific mission requirements.

The control computer functions are defined for the basic modes of operation, namely: attitude hold, maneuver, manual, and for four specified orbital experiments. These functions include stability and control law computation, on-board navigation, determination of spacecraft attitude via the solution of the direction cosine differential equations, CMG gimbal desaturation, and monitoring of critical CMG parameters. The computer also processes the necessary data for controls and displays. A preliminary design of the controls and display panel and hardware recommendations are included in this report.

To verify analytical results and accurately assess the control computer computational requirements, a detailed digital simulation was developed. It solves the complete equations of motion of the spacecraft and CMGs, and includes all nonlinearities and error models for which information is available. The control computer is simulated by programming its arithmetic operations in fixed-point language with word length being an input data option. It also simulates the quantization of sensor data, and the time delays incurred by the control computer execution time and input/output operations. These are also specifiable input data options.

Perhaps the most important result obtained from this accurate simulation is the evaluation of the system accuracy as a function of the control computer word length and the sampling interval.

The report is organized in two volumes. Following a description of the entire system, Volume I contains the analytical work consisting of stabilization of the sampled data attitude control loops and two sections on control law development and their comparison. Simulation results, control and display, and computer sizing sections complete Volume I. The work done on the experiment equations, the CMG gimbal servo analysis, the equations of motion of the entire system, and control and display trade offs comprise the appendixes.

Volume II of the report is entirely devoted to a detailed discussion of the simulation developed as part of the study.

Section 2

SYSTEM DESCRIPTION

The spacecraft considered for this study is representative of a vehicle consisting of the CSM docked with the LM. The specified moments of inertia are 151000, 150000 and 30000 slug-feet^2, respectively. The spacecraft control moments are the precession torques of three double-gimballed control moment gyros. Overall system control is provided by the digital-control computer. This allows a highly integrated system as shown in Figure 2-1. Choice of operating mode and/or specific experiment is supplied to the control computer via the controls and display panel.

Control Moment Gyros

Each of the three CMGs has a nominal rotor angular momentum of 1000 foot-pound-seconds. The gimbal angular rates are limited to a maximum of 10 degrees per second, providing a maximum output torque of approximately 175 foot-pounds. While the outer gimbal angular range is unlimited, the inner gimbals have limit stops at less than ±90 degrees. The deployment of the three CMGs relative to the vehicle axes is shown in Figure 2-2.

When gimbal angles are at their reference value of zero degrees, each gimbal axis is aligned with a vehicle axis with the result that each gimbal rate produces a precession torque about a single body axis. However, during operation the gimbals move away from their reference values and the precession torque corresponding to a single gimbal angle rate is no longer about a single body axis. For instance, to generate a torque about the vehicle x axis only, it is not sufficient to command a rate about the inner gimbal axis of the #3 CMG if the inner and outer gimbal angles (α_3 and β_3) are not zero. A gimbal rate, $\dot{\beta}_3$, produces a torque not only about the vehicle x axis but also about the y and z axes. Thus, it is desirable to command more than one gimbal rate in such a way that the summation of the resulting individual torques about the x axis give the required value of torque, while their contributions along the y and z axes add up to zero.

A second point to be considered with the use of three double-gimballed CMGs is the redundancy they provide. In general, to produce a desired torque vector it is sufficient to command rates about three gimbal axes. Thus, there exists the manner of selection of those gimbal axes whose use is in some sense advantageous. This inherent redundancy also allows operation of the system in a failure mode, where only two of the CMGs are available for attitude control.
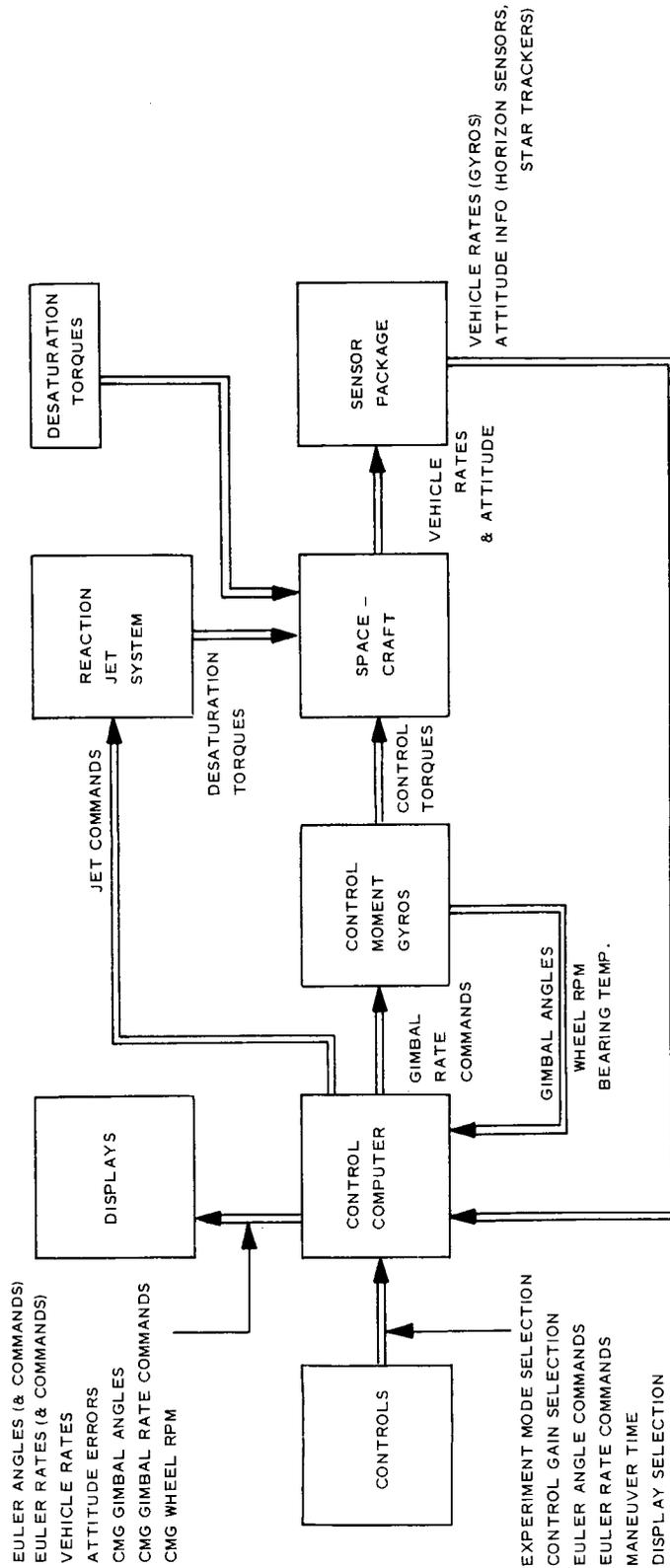
Figure 2-1.   Attitude Control System
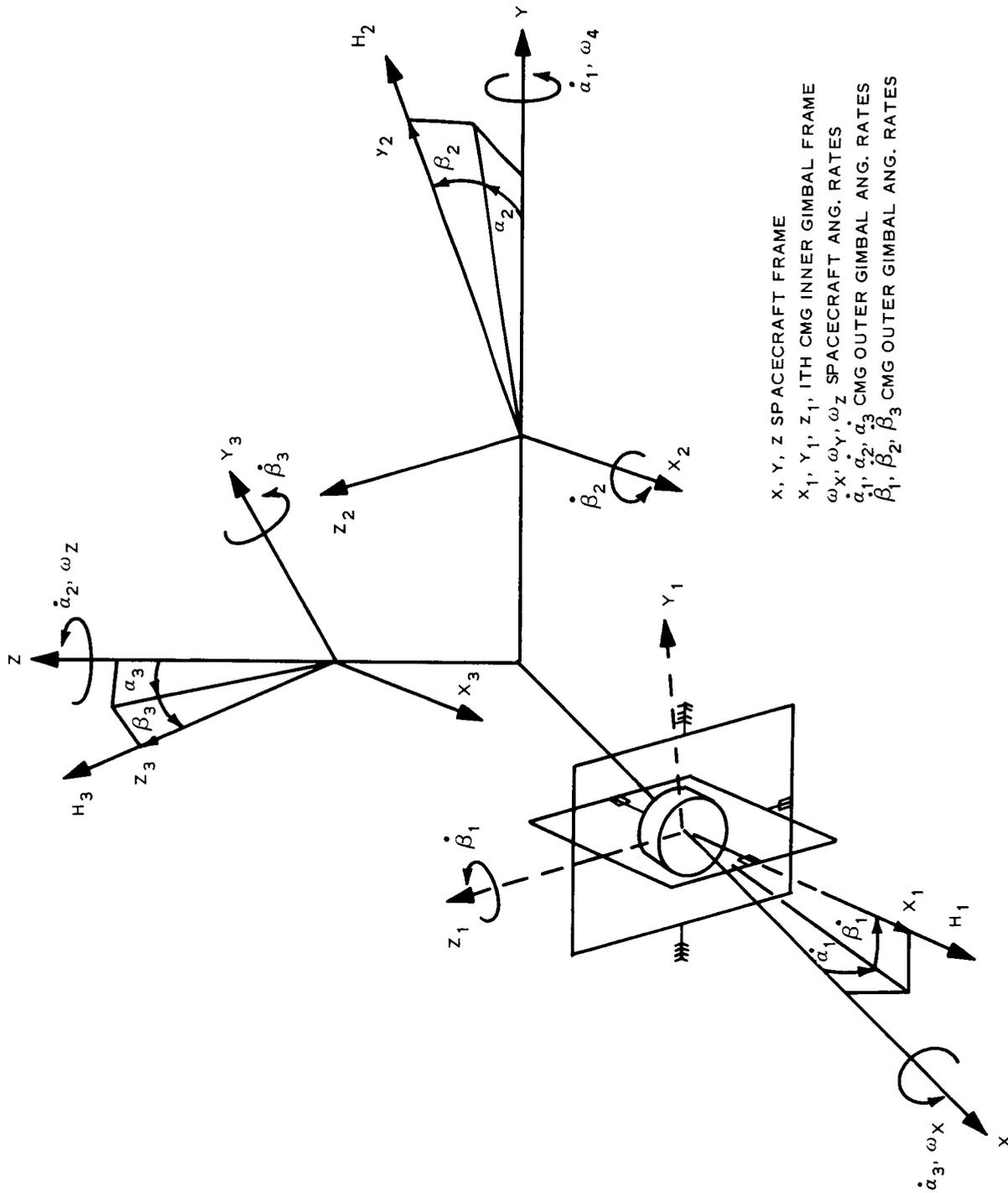
Figure 2-2. Reference Frames

X, Y, Z SPACECRAFT FRAME

x_1, Y_1, z_1, ITH CMG INNER GIMBAL FRAME

$\omega_X, \omega_Y, \omega_Z$ SPACECRAFT ANG. RATES

$\dot{\alpha}_1, \dot{\alpha}_2, \dot{\alpha}_3$ CMG OUTER GIMBAL ANG. RATES

$\dot{\beta}_1, \dot{\beta}_2, \dot{\beta}_3$ CMG OUTER GIMBAL ANG. RATES

Consideration of the CMG system redundancy to provide gimbal rate commands which result in the desired aggregate CMG precession torque vector is the task of the control law algorithm. This is discussed in the control laws section.

Control Computer

To control the attitude of the vehicle, the control computer determines the desired precession rate of each gyro and issues these rates as gimbal rate commands each sampling interval. To accomplish this, in a representative case, the computer performs the following tasks.

1. Determine the modes of operation from information received from the controls and display panel.

2. Compute the desired attitude and rate appropriate for the particular experiment or mode of operation.

3. Process data from appropriate sensors.

4. Compute indicated attitude errors: ϵ'_x, ϵ'_y, ϵ'_z.

5. Process attitude errors according to stability compensation equations to obtain the inputs to the control law algorithm.

6. Compute gimbal rate commands ($\dot{\alpha}_{ic}$ and $\dot{\beta}_{ic}$) via the control law algorithm.

7. Determine if CMG desaturation is required, and, if so, issue reaction jet on and off discretes.

8. Monitor CMG rotor RPM and bearing temperatures and compare to preset tolerances; if tolerances are exceeded, issue warnings via the display panel.

9. Process the necessary data for display purposes.

Figure 2-3 shows how various functions and components of the control loops are related. For conceptual simplicity, rate gyros are indicated as the sensors.

Control and Display

The control and display unit is designed to operate in conjunction with the CMG control computer to provide the astronaut with a command and monitor capability. The unit's major performance capabilities are herein summarized.
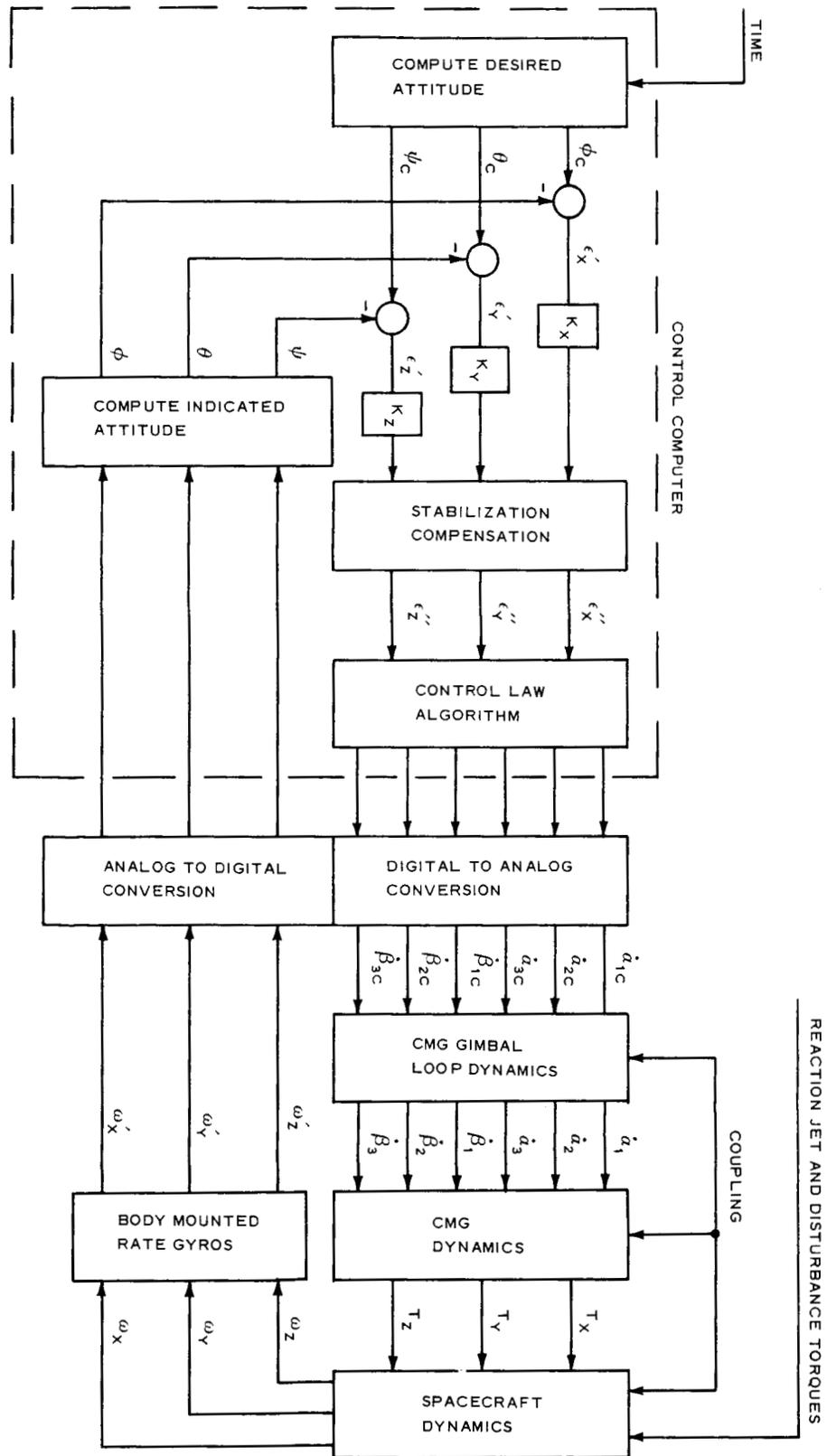
Figure 2-3. Spacecraft Control Loops

1. Special purpose numeric display — three rows of address and data with display mode select capability.

2. CMG monitor — a 9-light panel to indicate limit excesses in bearing temperature and/or rotor RPM.

3. Control and command mode switches — manual selection of the control system operational mode and control loop gains.

4. General purpose insert and display — address and data thumbwheel switches and insert/display switches, in conjunction with a single row display, provides data insert/data display access to 99 control computer locations.

Experiments

To establish control computer requirements, adequacy of control-loop compensation and sampling period, the performance of the total system in its execution of four experiments is studied. Certain additions, modifications, and corrections made to the experiment equations provided in the RFP are given in Appendix A. For easy reference, the basics of the four experiments are listed as follows.

Experiment #1 - Earth Mapping. — In this experiment the z axis of the vehicle is normal to the orbital plane, and the y axis points at the projection of the target on the orbit plane. This makes it possible to point the optical axis (y_t) of a telescope at the target through the action of a single gimbal about the vehicle x axis.

To start the experiment it is necessary to provide the control computer with the vehicle position and inertial velocity and with the target position vectors resolved in inertial coordinates. The actual vehicle attitude at the start of the experiment must also be known. With this information, the computer initializes the commanded vehicle attitude matrix M_{BcI} and the actual vehicle attitude matrix m_{BI}. Alternately, to reduce the computational load of the control computer, the initial values of the matrices M_{BcI} and m_{BI} may be entered into the computer.

After initialization, the vehicle position and velocity are computed as a function of time by means of a Taylor series expansion of the coordinates and velocity of a body in Keplerian motion (f and g series). This information is needed to compute the desired telescope gimbal angle, A_c, and the desired rate about the vehicle z axis, ω_{zc}. The matrix m_{BI} is updated using sensed-body rate information from 3-body-mounted rate gyros. From matrices M_{BcI} and m_{BI}, the computer determines the small angle indicated attitude error vector $\underline{\epsilon}'$, with components in vehicle coordinates.

The specified accuracy for this experiment is 0.001 degree per second per axis for rate and 0.01 degree per axis for attitude.

Experiment #2 - Inertial Mode — The desired attitude frame of the vehicle for Experiment #2 is the same as the inertial reference frame. The desired vehicle rates are equal to zero, and two star trackers are used for attitude sensing. From the known elevation and azimuth of two selected stars, and from the corresponding two indicated elevation and one azimuth gimbal angles of the star trackers, the control computer computes the small angle indicated attitude error vector.

Experiment #3 - Horizon Spectrometry — To enable the experimental package to sweep the horizon, the commanded vehicle z axis points toward the earth along the local ellipsoid vertical. The vehicle is commanded to rotate about the z axis at a constant rate, $\dot{\eta}$. The commanded rates about the x and y axes are equal to the components of the orbital rate vector along these axes.

The vehicle-indicated attitude frame is computed by updating the direction cosine matrix using the indicated vehicle rates from a set of body-mounted rate gyros. This matrix is periodically corrected using information from two star trackers. In this case two elevation and two azimuth gimbal angles are used.

The f and g series method is used to determine vehicle position, from which its longitude φ and geocentric latitude λ_R are obtained. The geocentric latitude is corrected to yield the geodetic latitude λ. From φ and λ, the I-frame components of the local ellipsoid vertical vector Z_{EI} are calculated. Finally, the control computer obtains the small attitude error components about the vehicle x and y axes by transforming Z_{EI} into indicated vehicle coordinates by means of the m_{BI} direction cosine matrix. The attitude error about the vehicle z axis is arrived at through integration of the rate error about the z axis.

Experiment #4 - Microwave Transmission — The commanded vehicle attitude has the commanded vehicle x axis normal to the orbit plane, the y axis pointing away from the earth and along the local ellipsoid vertical, and the z axis nominally in the orbit plane. Horizon sensors provide the indicated attitude errors about the vehicle x and z axes. The desired rate about the vehicle x axis is the average orbital rate. The desired rates are zero about the vehicle y and z axes, which are nominally in the orbital plane. Thus, a rate gyro along the vehicle z axis senses a component of the orbital rate if there exists an attitude error about the vehicle y axis. The output of the rate gyro along the z axis is used in a gyro compassing mode to supply the attitude error about the vehicle y axis.

Section 3

STABILIZATION

Introduction

Stability compensation design must be performed early in the study to allow system simulation to be used for verification of analytical results. The design must provide stable system response which is adequate for the system accuracy requirements. Thus, the final compensation design is dependent upon simulation results to ensure that accuracy requirements are met; i.e., the system operating points provide sufficiently "tight loops" for the sampling interval chosen.

To apply standard control system analysis techniques, a linear model must first be derived. With small angle approximations, the system block diagram in Figure 2-3 can be reduced to the linear version of the roll channel sampled-data control loop shown in Figure 3-1.
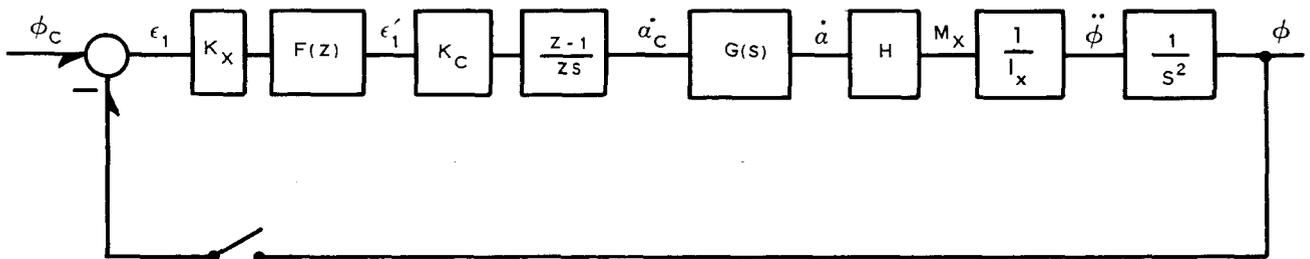
Figure 3-1. Control System Linear Model

Because of the significant difference in natural frequencies, the dynamical coupling from spacecraft to CMG is very small and, therefore, negligible. For the same reason the sensor dynamics have been treated as unity.

The linear transfer function for the CMG outer gimbal loop dynamics, G(s), is given in Equation (3-1). This transfer function is as derived in Appendix B, *except*

$$\frac{\alpha}{\alpha_c} = \frac{(28.94)\,(34760)\,(s+150)\,(s+52.8)\,(s^2 + 98s + 10720)}{(s^2 + 80s + 14820)\,(s^2 + 74s + 2458)\,(s^2 + 148s + 7157)\,(s + 327.5)} \qquad (3\text{-}1)$$

that a gain factor has been included to provide unity steady-state gain. From the location of the poles and zeros it is obvious that the gimbal loop dynamics will have little effect on the much slower spacecraft response. The small difference between the transfer functions for the outer and inner gimbal control loops is even less significant.

The gain factor K_c represents the control law transfer function. In this discussion it is treated as unity and is covered in a later section.

<div align="center">Error Channel Compensation</div>

Z transforms of the continuous portion of the system, including the zero-order-hold, were generated with the use of digital computer programs. * The same programs were used to compute root locus data from the z transforms for various F(z) functions. Since the linear models for the three control channels differ only by the moments of inertia, the same z transforms and root locus can be used for all three channels by proportionately adjusting the gains.

The root locus plot in Figure 3-2 is for a sampling interval of 0.1 second and F(z) equal to unity. It is plotted in the s-plane showing the actual location of the
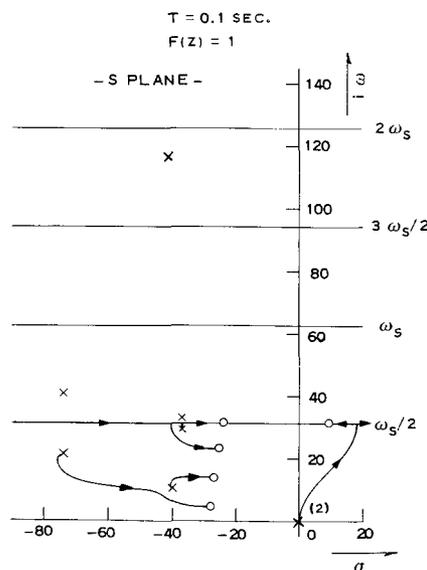
Figure 3-2. Root Locus of Uncompensated System

*M. E. Fowler, "Numerical Methods for the Synthesis of Linear Control Systems", Automatica, 1963, Vol. 1, Pages 207-225; or IBM Report No. 62-907-417.

gimbal loop complex poles and the resulting "folded pole" locations in the primary strip. The root locus has been plotted only in the primary strip since it is identical in all others. This plot has been included to verify the small effect of the gimbal loop dynamics on the spacecraft response; and to indicate the amount of phase lead required to stabilize the spacecraft dynamics.

For this sampling interval, sufficient phase lead is provided with a first order difference equation of the form

$$\epsilon'(nT) = K \left[\epsilon(nT) - z_0 \epsilon(nT - T) \right] \tag{3-2}$$

which in z transform notation is

$$\frac{\epsilon'}{\epsilon} = K\, F(z) = K\, \frac{z - z_0}{z} \tag{3-3}$$

The root locus shown in Figure 3-3a, where only the area of interest of the s-plane primary strip is shown, is for $F(z) = (z - 0.9048)/z$. This is equivalent to a lead zero at $s = -1.0$. A desirable operating point is for a gain (K_x) of 4740, which gives a gain margin of 4.4. The roll channel response for this operating point is shown in Figure 3-4. For the 0.01 degree step command in this channel, there is initial gimbal rate limiting which results in the slower response. For commands or disturbances that do not cause gimbal rate limiting, the system will respond with a very fast rise time as shown by the response curve with about 26 percent oversheet.
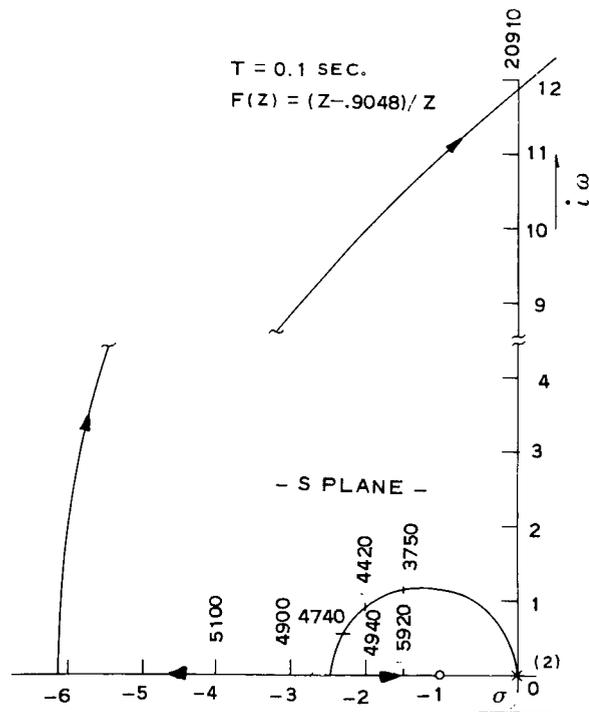
Figure 3-3a. Compensated System Root Loci for a Set of Lead Zero Locations
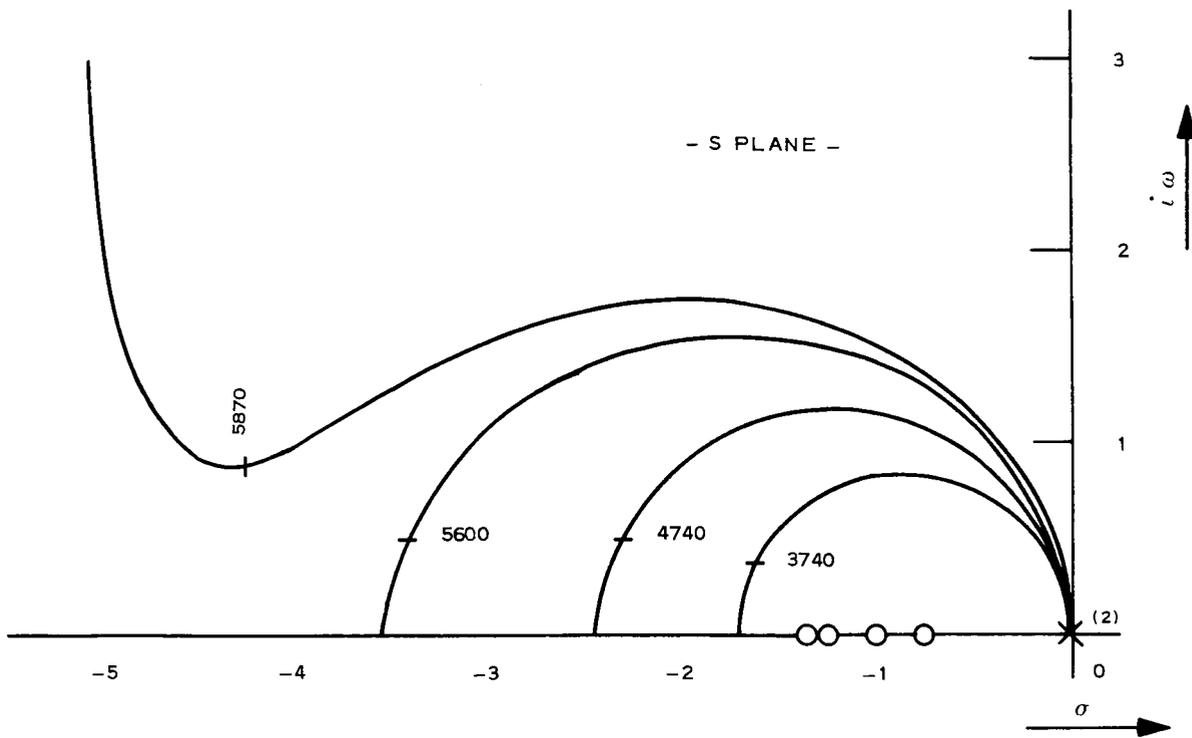
Figure 3-3b. *Compensated System Root Loci*
for a Set of Lead Zero Locations
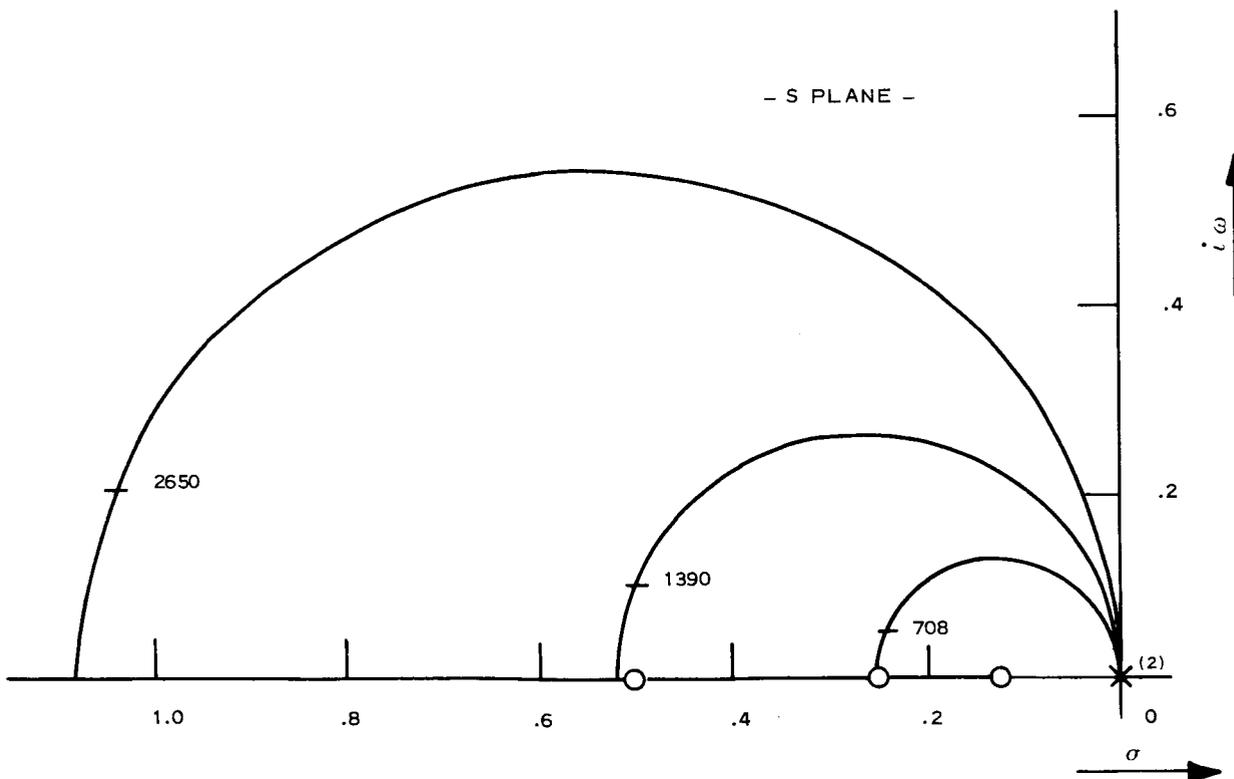
Figure 3-3c. *Compensated System Root Loci*
for a Set of Lead Zero Locations

3-4

$$T = 0.1$$
$$F(z) = (z-9048)/z$$
$$K_x = 4740$$
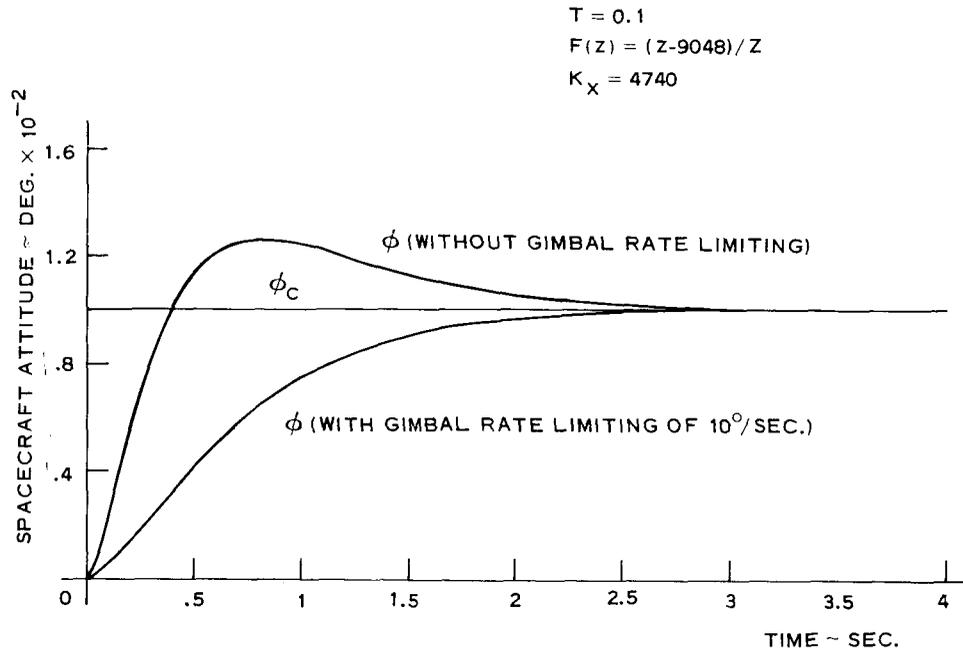
Figure 3-4. System Response

The above operating point was chosen over those with lower gain and damping ratio because of overshoot considerations. With the lead zero in the forward loop, a high damping ratio is desirable to prevent large values of overshoot.

As discussed in the Simulation Results section, this operating point (with the sampling interval of 0.1 second) is adequate for the system accuracy requirements of the experiments considered. To provide a range of operating points for different control modes of operation, a set of root loci for different lead zero locations is plotted in Figures 3-3b and 3-3c. The suggested operating points and corresponding F(z) are listed below.

No.	K_x	K_y	K_z	$F(z)$
1	5870	5830	1165	$(z - 0.8737)/z$
2	5600	5560	1110	$(z - 0.8825)/z$
3	4740	4710	942	$(z - 0.9048)/z$
4	3740	3715	743	$(z - 0.9277)/z$
5	2650	2630	526	$(z - 0.9512)/z$
6	1390	1380	276	$(z - 0.9753)/z$
7	708	703	141	$(z - 0.9876)/z$

No. 1 is obviously the most accurate operating point of the set, and as previously stated, No. 3 meets the accuracy requirements of the experiments considered.

While the sampling interval of 0.1 second is sufficient for the computational load considered in this study, a larger sampling interval may be desirable if it becomes necessary to include additional computational functions. The root locus in Figure 3-5 is for a sampling interval of 0.15 second and $F(z) = (z - 0.8607)/z$. This provides a lead zero in the s-plane at s = -1, the same location as in Figure 3-3a for the sampling interval of 0.1 second. From a comparison of the two figures, the lower sampling rate has obviously introduced phase lag into the system. Actually it is not due to the sampling process, but results from the zero-order-hold, as is evident from the phase/gain plot in Figure 3-6. According to the root locus plot of Figure 3-5, a good operating point is for $K_X = 2850$. At this gain the closed loop complex poles are almost directly above the closed loop pole on the real axis. Simulation results have shown that this case also meets the specified accuracy requirements. Note that when comparing gains for a different $F(z)$ or sampling interval, the steady-state value of $F(z)$ must be considered.

If the sampling interval is lowered to 0.2 second, a first order difference equation for $F(z)$ does not provide sufficient phase lead to allow a desirable operating point that is adequate for system accuracy requirements. This is evident from Figure 3-7 where $F(z) = (z - 0.8607)/z$, and $K_X = 1635$. At this gain the complex closed loop poles and real closed loop pole both have approximately the same time constant. While the complex poles will move away from the origin with increased gain, the real pole will move closer to the origin defeating the objective of an overall faster loop response. To some extent this will not be too detrimental since the real pole residue will become smaller with increased loop gain.

A better solution is to use a higher order compensator. The root locus in Figure 3-8 is for the same sampling interval of 0.2 second and $F(z) = (z - 0.6065)(z - 0.4966)/z(z - 0.2894)$. The control loop response for $K_X = 3250$ is shown in Figure 3-9. While this operating point provides a fast loop response, it may be desirable to adjust the compensation coefficients to provide a higher damping ratio which will result in less overshoot. Actually, the gimbal rate limiters will eliminate the overshoot except for very small commands or distrubances. In any case, should a sampling interval larger than 0.1 second become necessary it is obviously feasible.

Rate Feedback System

If rate feedback is added to Figure 3-1 and $F(z)$ set to unity, the rate feedback system can be reduced to that shown in Figure 3-10, where K_X' is the rate gain in the
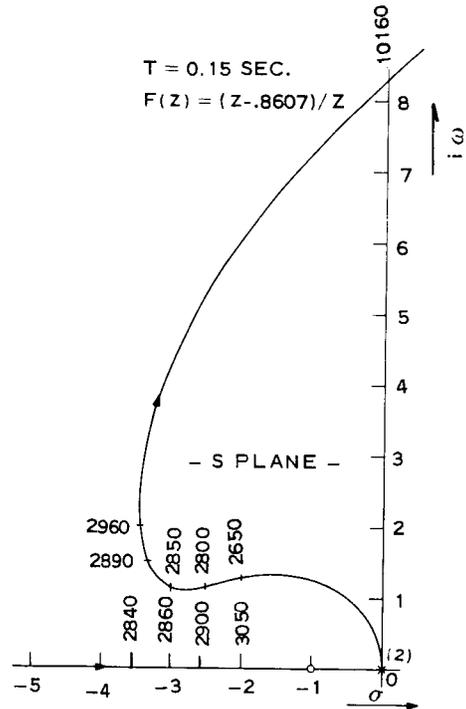
Figure 3-5. Compensated System Root Locus
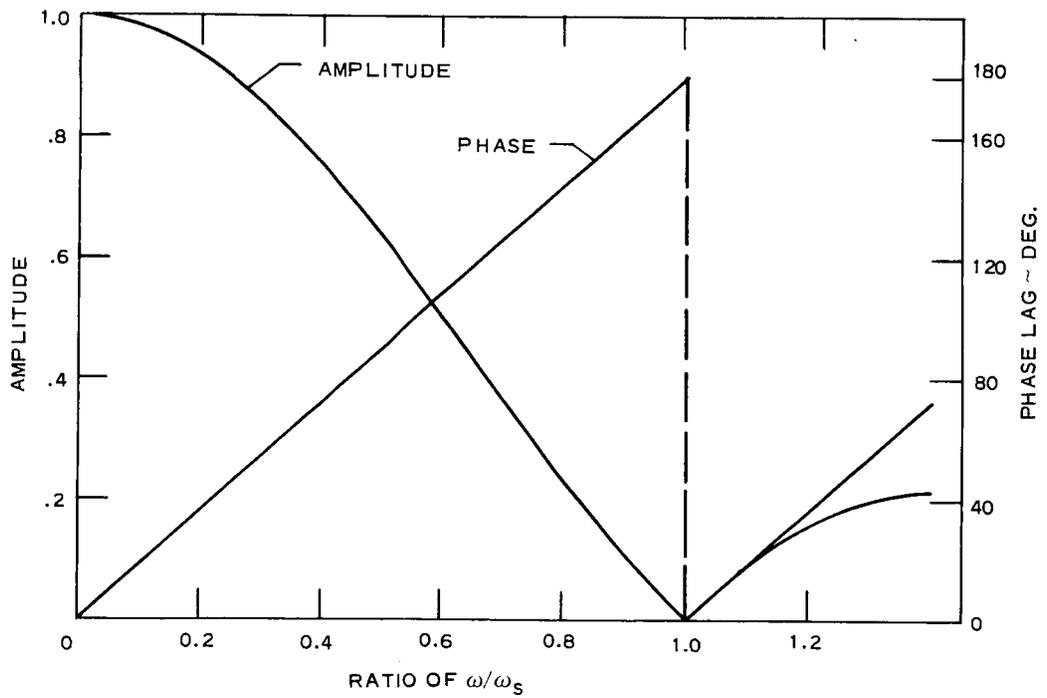
Figure 3-6. Zero-Order-Hold Phase/Gain

3-7

T = 0.2 SEC

$$F(Z) = (Z-.8607)/Z$$
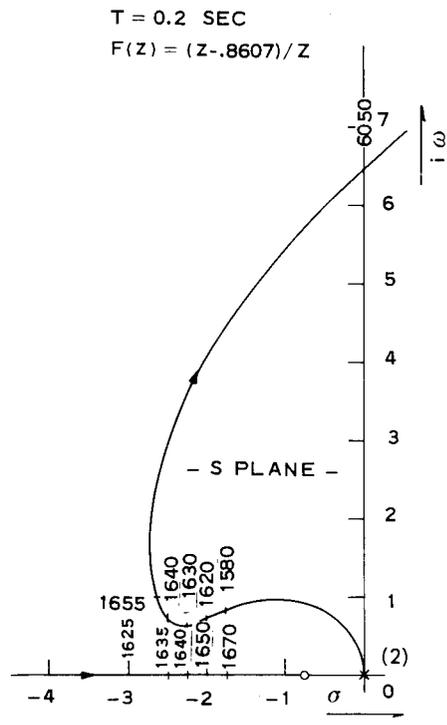
Figure 3-7. Compensated System Root Locus

T = 0.2 SEC.

$$F(Z) = \frac{(Z-.6065)\ (Z-.4966)}{Z(Z-.2894)}$$
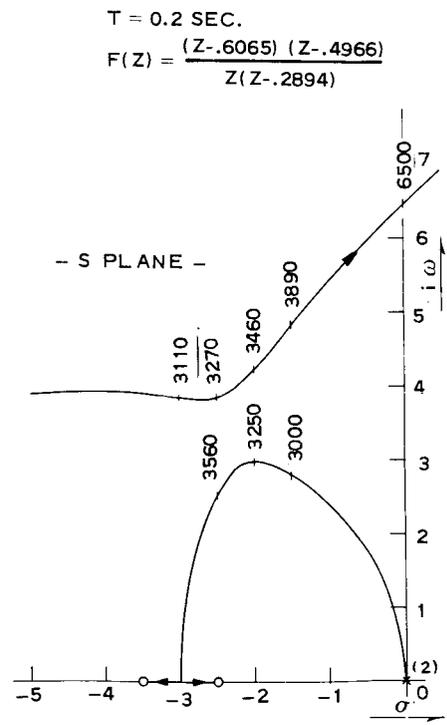
Figure 3-8. Compensated System Root Locus

feedback loop. In this form, root loci can be generated as a function of K_x with the ratio of K_x/K_x' held constant.

The root locus in Figure 3-11 is for a sampling interval of 0.1 second and $K_x/K_x' = 1.5$. Since the lead zero due to rate feedback does not appear in the forward loop of the closed loop transfer function, an operating point with any reasonable damping coefficient will give good response without significant overshoot. A suggested operating point is for $K_x = 758$.

Figures 3-11b and 3-11c show a family of root loci for different ratios of K_x/K_x', providing a range for different modes of operation.
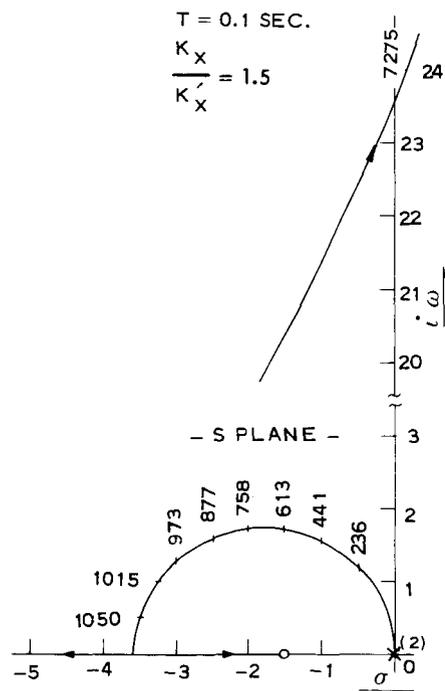
Figure 3-11a. Rate Feedback System Root Locus for a Set of Lead Zero Locations

$T = 0.2$ SEC.

$K_x = 3250$

$$F(Z) = \frac{(Z-.6065)(Z-.4966)}{Z(Z-.2894)}$$

ϕ (WITHOUT GIMBAL RATE LIMITING)

ϕ_C

SPACECRAFT ATTITUDE ~ DEG. $\times 10^{-2}$

TIME $\approx$ SEC.
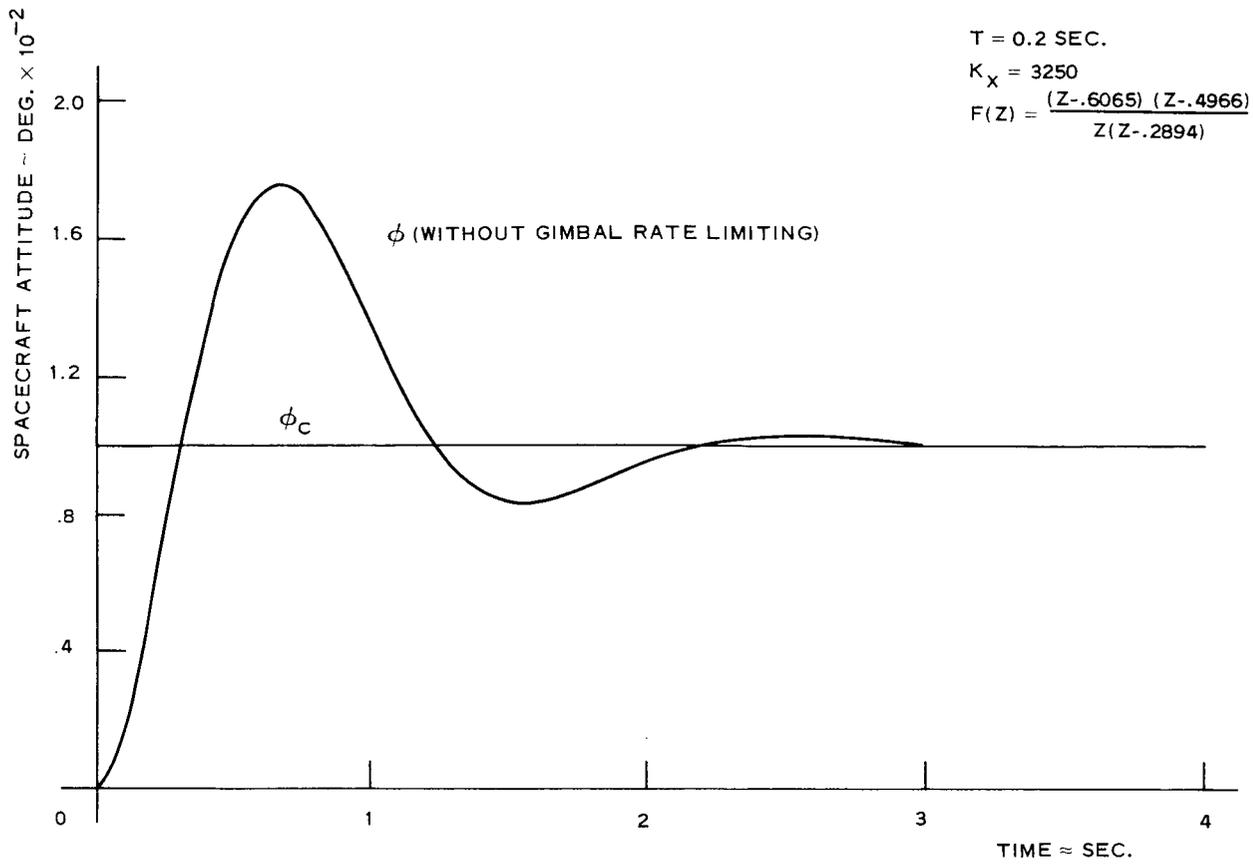
Figure 3-9.   Control Loop Response
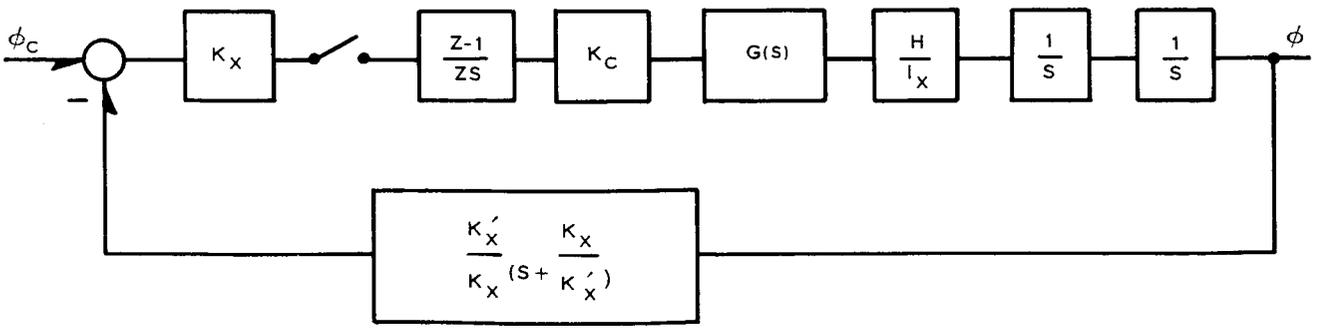
Figure 3-10.   Rate Feedback System
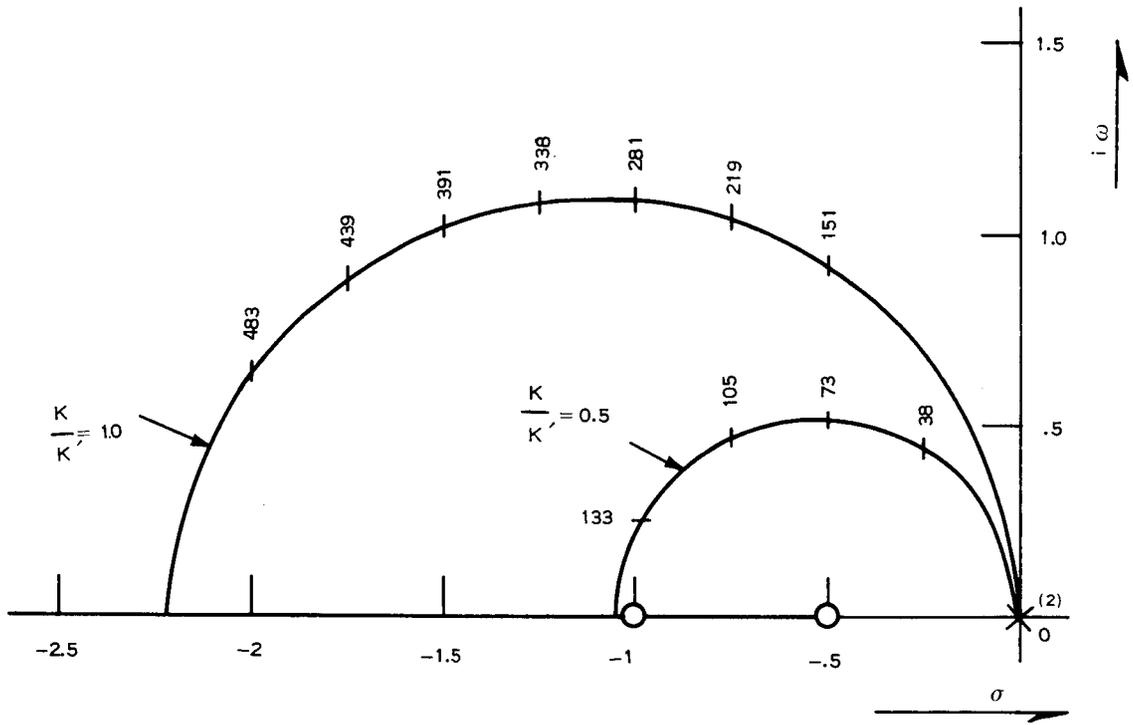
Figure 3-11b. *Rate Feedback System Root Loci for a Set of Lead Zero Locations*
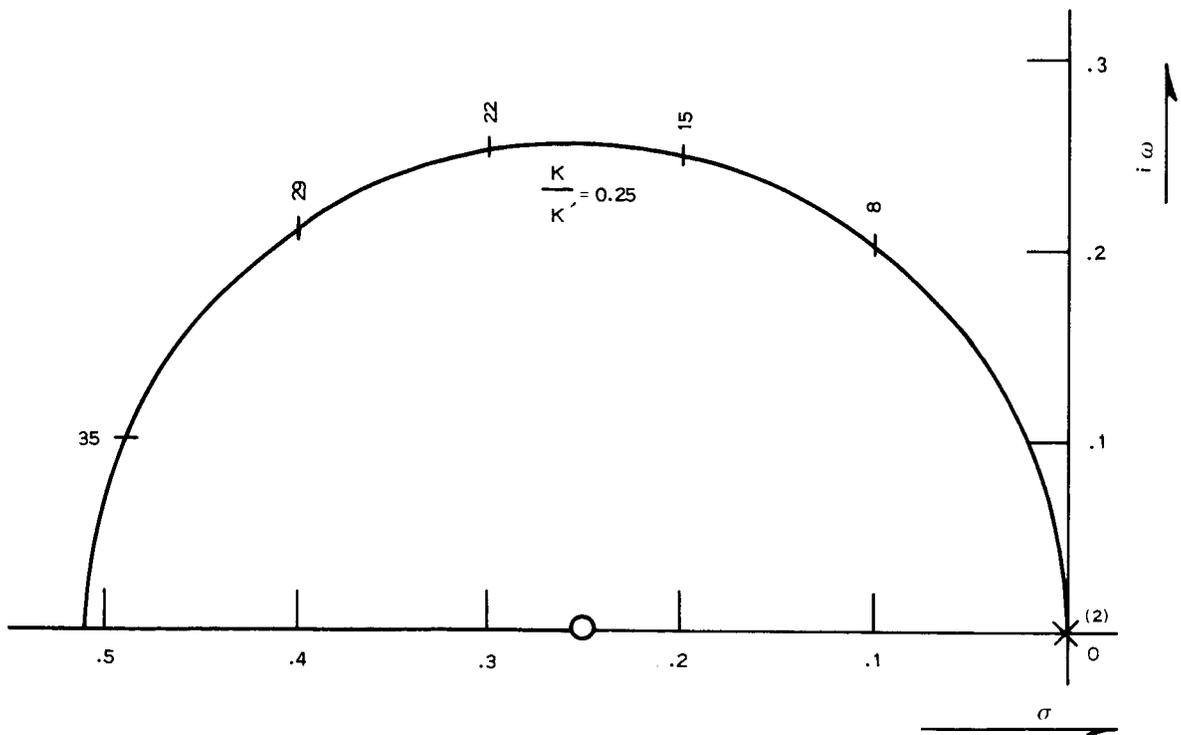
Figure 3-11c. *Rate Feedback System Root Loci for a Set of Lead Zero Locations*

Section 4

CONTROL LAWS

Introduction

How successfully the CMG's control the vehicle attitude depends on the success of computing gimbal rate commands so that the resulting total precession torque vector of the three CMG's closely matches the computed desired torque vector. This is the function of the control law algorithm. To yield the proper gimbal rate commands, the control law has to consider the precession torque vector corresponding to each individual CMG gimbal rate.

In general, the production of a desired torque vector does not require gimbal rates about all six CMG gimbal axes. The control law must make a sensible selection of the gimbal axes to be used for the generation of the desired torque in each computation cycle.

The study has resulted in two different types of control law. The first, called the algebraic control law, selects three gimbal axes based on entirely geometric considerations. It algebraically computes the required gimbal rates about the selected gimbal axes to produce an aggregate CMG precession torque, which exactly matches the computed desired torque. The other control law, called the iterative control law, accomplishes the same end result with increasing exactitude as the allowed number of iterations increases.

The iterative control law has the additional advantage of complete flexibility of selection policy for choosing which gimbals are to be commanded. It makes its selection of each gimbal by finding the axis whose use adds the least "cost" to the total. The iterative control law can be readily tailored to meet specific mission requirements by variations in the form of the cost function.

Because it is fundamental to the control laws developed in the study, a discussion of CMG precession torques, as functions of vehicle rates and gyro gimbal rates, precedes the detailed description of the control laws. Among these is the baseline control law which is one of the control law options in the control computer simulation.

Gyro Precession Torques

In computing gimbal rate commands to generate a desired control torque, relations of CMG precession torques as functions of vehicle rates and CMG gimbal rates are required. In deriving these relations, the magnitude of the spin momentum of each CMG is treated as a constant. However, the control law algorithm uses the values of spin

momentum magnitude as sampled during each computation cycle. The contribution of vehicle rates to CMG spin momenta is insignificant. This approximation is only used in control law computations, and not in the simulation of the equations of motion, which are exact.

The most significant part of the CMG torque acting on the vehicle is that part due to the precession rate of the CMG spin momentum. For the ith CMG, this torque is equal to $(-\underline{\omega}_i \times \underline{H}_i)$, where $\underline{\omega}_i$ is the precession rate of the spinning element of the CMG posessing the spin momentum $\underline{H}_i$. Since $\underline{\omega}_i$ is composed of vehicle rate $\underline{\omega}$ and gimbal rates $\dot{\alpha}_i$, $\dot{\beta}_i$, it is convenient to write the CMG precession torque acting on the vehicle in two parts: that part due to precession at the vehicle rate $\underline{\omega}$, and the part due to precession at the gimbal rates $\dot{\alpha}_i$, $\dot{\beta}_i$.

The components of torque due to precession at body rates is given by:

$$T_{xb} = H_1 (\omega_y \sin\alpha_1 \cos\beta_1 + \omega_z \sin\beta_1) + H_2 (\omega_z \cos\alpha_2 \cos\beta_2 - \omega_y \sin\beta_2)$$
$$- H_3 (\omega_y \cos\alpha_3 + \omega_z \sin\alpha_3) \cos\beta_3 \tag{4-1a}$$

$$T_{yb} = - H_1 (\omega_x \sin\alpha_1 + \omega_z \cos\alpha_1) \cos\beta_1 + H_2 (\omega_z \sin\alpha_2 \cos\beta_2 + \omega_x \sin\beta_2)$$
$$+ H_3 (\omega_x \cos\alpha_3 \cos\beta_3 - \omega_z \sin\beta_3) \tag{4-1b}$$

$$T_{zb} = H_1 (\omega_y \cos\alpha_1 \cos\beta_1 - \omega_x \sin\beta_1) - H_2 (\omega_x \cos\alpha_2 + \omega_y \sin\alpha_2) \cos\beta_2$$
$$+ H_3 (\omega_x \sin\alpha_3 \cos\beta_3 + \omega_y \sin\beta_3) \tag{4-1c}$$

Since these torques are due to body rates ω_x, ω_y, ω_z, they act on the vehicle even when the gimbal rates are zero. When the body rates are insignificant, such as in the attitude hold mode, these torques can be neglected; but during maneuvers, such as in the case of the earth-mapping experiment, their contribution should be subtracted from the total desired control torque $\underline{T}_d$. The remainder of $\underline{T}_d$ is generated by commanding appropriate gimbal rates. In matrix notation, the CMG torques due to precession at the gimbal rates can be expressed as:

$$\underline{T}_c = [B] \underline{\dot{\beta}} \tag{4-2a}$$

where

$$\underline{T}_c = \underline{T}_d - \underline{T}_b$$

and

$$(\underline{\dot{\beta}})^T = (\dot{\alpha}_1, \dot{\alpha}_2, \dot{\alpha}_3, \dot{\beta}_1, \dot{\beta}_2, \dot{\beta}_3) \tag{4-2b}$$

$$[B] = [\underline{a}_1, \ \underline{a}_2, \ \underline{a}_3, \ \underline{b}_1, \ \underline{b}_2, \ \underline{b}_3] \tag{4-2c}$$

with

$$\underline{a}_1 = H_1 \begin{pmatrix} \sin \alpha_1 \cos \beta_1 \\ 0 \\ \cos \alpha_1 \cos \beta_1 \end{pmatrix} \qquad \underline{b}_1 = H_1 \begin{pmatrix} \cos \alpha_1 \sin \beta_1 \\ -\cos \beta_1 \\ -\sin \alpha_1 \sin \beta_1 \end{pmatrix} \tag{4-3a}$$

$$\underline{a}_2 = H_2 \begin{pmatrix} \cos \alpha_2 \cos \beta_2 \\ \sin \alpha_2 \cos \beta_2 \\ 0 \end{pmatrix} \qquad \underline{b}_2 = H_2 \begin{pmatrix} -\sin \alpha_2 \sin \beta_2 \\ \cos \alpha_2 \sin \beta_2 \\ -\cos \beta_2 \end{pmatrix} \tag{4-3b}$$

$$\underline{a}_3 = H_3 \begin{pmatrix} 0 \\ \cos \alpha_3 \cos \beta_3 \\ \sin \alpha_3 \cos \beta_3 \end{pmatrix} \qquad \underline{b}_3 = H_3 \begin{pmatrix} -\cos \beta_3 \\ -\sin \alpha_3 \sin \beta_3 \\ \cos \alpha_3 \sin \beta_3 \end{pmatrix} \tag{4-3c}$$

Substitution of equations 4-2b and 4-2c into equation 4-2a yields Equation (4-4).

$$\underline{T}_c = \dot{\alpha}_1 \underline{a}_1 + \dot{\alpha}_2 \underline{a}_2 + \dot{\alpha}_3 \underline{a}_3 + \dot{\beta}_1 \underline{b}_1 + \dot{\beta}_2 \underline{b}_2 + \dot{\beta}_3 \underline{b}_3 \tag{4-4}$$

It is evident that the torque contribution of the gimbal rate $\dot{\alpha}_1$, for instance, is equal to $\dot{\alpha}_1 \underline{a}_1$, or the contribution of the gimbal rate $\dot{\beta}_2$ is equal to $\dot{\beta}_2 \underline{b}_2$. Therefore, the vectors $\underline{a}_i$ and $\underline{b}_i$ can be considered numerically equal to the precessional torque vectors corresponding to gimbal rates $\dot{\alpha}_i = 1$ rad/sec. and $\dot{\beta}_i = 1$ rad/sec., respectively.

Baseline Control Law

This control law is described in the Baseline attachment to the Statement of Work in the RFP. It is one of the three control laws incorporated in the simulation and is described here for quick reference.

In its sensed feedback version, the CMG gimbal rate commands are computed as:

$$\dot{\alpha}_{1c} = k_1 \sin \alpha_1 + k_3 \cos \alpha_1$$

$$\dot{\alpha}_{2c} = k_2 \sin \alpha_2 + k_1 \cos \alpha_2$$

$$\dot{\alpha}_{3c} = k_3 \sin \alpha_3 + k_2 \cos \alpha_3$$

$$\dot{\beta}_{1c} = k_5 \cos \beta_1$$

$$\dot{\beta}_{2c} = k_6 \cos \beta_2$$

$$\dot{\beta}_{3c} = k_4 \cos \beta_3$$

(4-5)

where

$$k_1 = -K_{e1}\epsilon'_x - K_{r1}(\omega_x - \omega_{xc})$$

$$k_2 = -K_{e2}\epsilon'_y - K_{r2}(\omega_y - \omega_{yc})$$

$$k_3 = -K_{e3}\epsilon'_z - K_{r3}(\omega_z - \omega_{zc})$$

$$k_4 = K_{e4}\epsilon'_x + K_{r4}(\omega_x - \omega_{xc})$$

$$k_5 = K_{e5}\epsilon'_y + K_{r5}(\omega_y - \omega_{yc})$$

$$k_6 = K_{e6}\epsilon'_z + K_{r6}(\omega_z - \omega_{zc})$$

(4-6)

The error terms $\epsilon'_{x, y, z}$ are the indicated attitude errors about respective body axes. It should be noted that when used in the sense of Euler angle errors ϵ'_x, for instance, is defined as

$$\epsilon'_x = \phi' - \phi_c$$

where ϕ' is the indicated and ϕ_c the commanded roll angle. In polarity, this is opposite to the conventional servomechanism definition of error, but is used here to conform to the RFP usage.

The derived rate version of the baseline control law (BCL) is diagrammed in Figure 4-1.

To appreciate the close similarity of the derived and sensed versions of the BCL, consider the signal $-\epsilon''_x$, which is the output of the sampled data compensator $(z - z_0)/z$. By inspection of Figure 4-1 it is possible to write

$$-\epsilon''_x = \frac{z - z_0}{z} (\phi_c - \phi')*$$

(4-7)
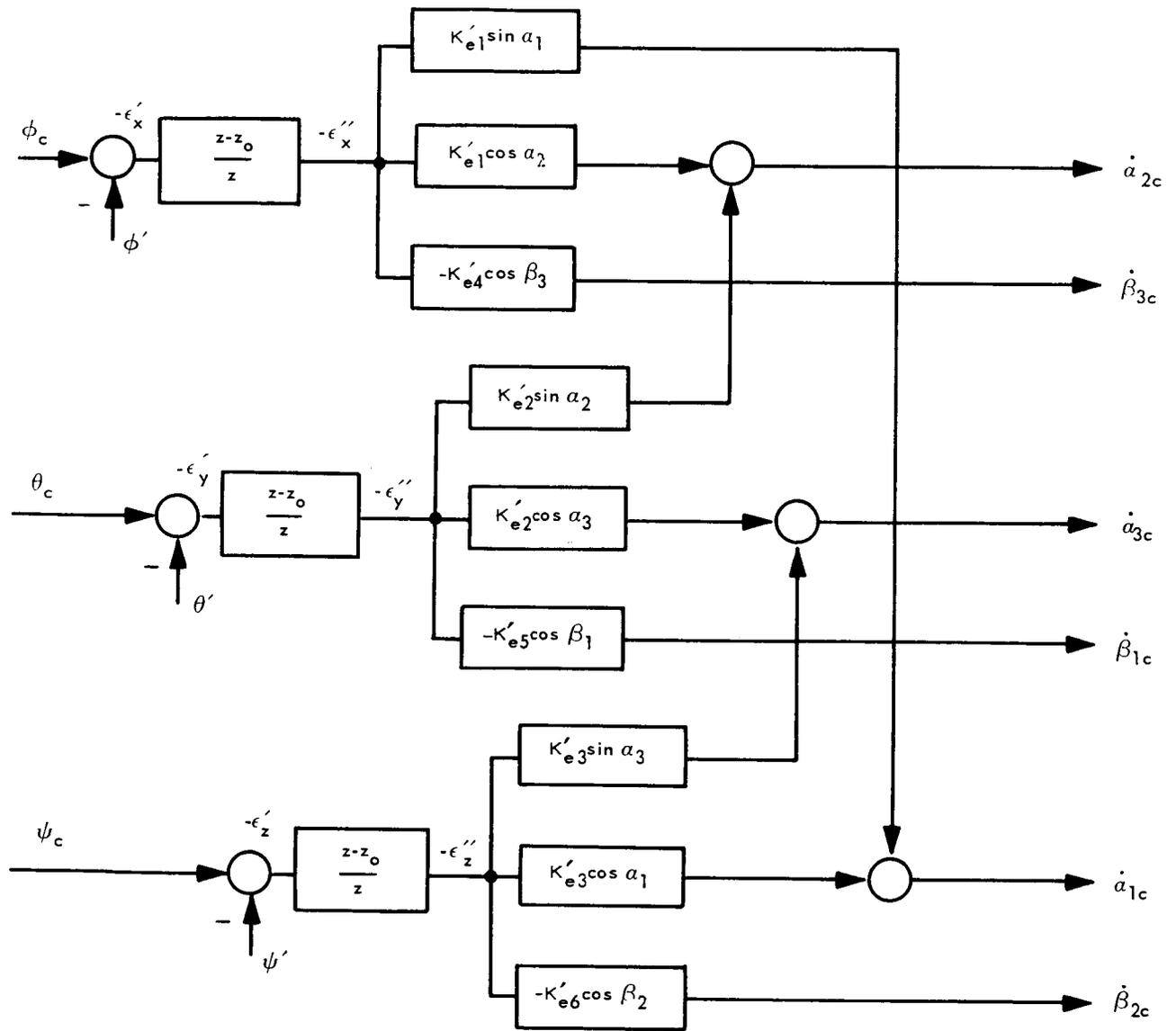
Figure 4-1. Derived Rate Version of the Baseline Control Law

where the asterisk is used to indicate that the error signal has been sampled. Equation (4-7) can be further processed as

$$-\epsilon_x'' = (1 - z_0 z^{-1})(\phi_c - \phi')*$$
$$= (1 + z_0 - z_0 - z_0 z^{-1})(\phi_c - \phi')*$$
$$= [(1 - z_0) + z_0(1 - z^{-1})](\phi_c - \phi')*$$
$$= \left[(1 - z_0) + Tz_0 \frac{(1 - z^{-1})}{T} \right](\phi_c - \phi')* \tag{4-8}$$

Now, note that

$$\frac{1 - z^{-1}}{T}(\phi_c - \phi')* \cong (\omega_{xc} - \omega_x)*$$

Therefore, Equation (4-8) can be rewritten as

$$-\epsilon_x'' = (1 - z_0)(\phi_c - \phi')* + Tz_0(\omega_{xc} - \omega_x)* \tag{4-9}$$

Corresponding to $-\epsilon_x''$ the gimbal rate command $\dot{\alpha}_{1c}$, for instance, is

$$\dot{\alpha}_{1c} = K_{e1}' \sin\alpha_1 \left[(1 - z_0)(\phi_c - \phi')* + Tz_0(\omega_{xc} - \omega_x)* \right]$$

With reference to Equation (4-6) recognition that

$$(1 - z_0) K_{e1}' \approx K_{e1}$$

and

$$K_{e1}'(Tz_0) \approx K_{r1}$$

establishes the fact that the sensed and derived rate version of the control law are functionally identical. There is advantage, however, to the derived rate version in experiments or modes of operation where the vehicle rates are too low to be accurately sensed.

Algebraic Control Law

Let T_d be the desired torque corresponding to an indicated attitude error vector $(\epsilon_x', \epsilon_y', \epsilon_z')$ combined with a rate error vector, if sensed rate feedback is used for loop

stabilization. At gimbal rates equal to zero (meaning the attitude of the CMG's not changing with respect to the vehicle), CMG torques are generated due to precession of the gyros at the vehicle angular rate. This torque vector, $\underline{T}_b$, is given by equations (4-1a), (4-1b) and (4-1c). The torque to be generated by the additional precession rates $\dot{\alpha}_i$, $\dot{\beta}_i$ is what remains of $\underline{T}_d$ after the precession torque due to vehicle rate is subtracted from $\underline{T}_d$. Thus, the task of the control law is to compute appropriate gimbal rate commands to generate the remainder torque

$$\underline{T}_c = \underline{T}_d - \underline{T}_b \qquad (4\text{-}10)$$

According to Equation (4-2a), determination of the values of required gimbal rates to generate the torque vector $\underline{T}_c$ involves the inversion of the matrix [B]. However, [B] is not a square matrix, as it relates six unknown (the six gimbal rates making up the vector $\underline{\beta}$) and three known variables (the three components of $\underline{T}_c$). In other words, the system is redundant or underspecified. To solve Equation (4-2a), i.e., to be able to invert [B], it is necessary to confine the choice of gimbal axes to only three. Since this corresponds to deleting in [B] three of its columns, it reduces [B] to a square matrix. Consequently, Equation (4-2a) can be solved for rates about the selected gimbal axes.

The so-called algebraic control law (ACL) consists of a procedure for selecting three gimbal axes about which the CMG's are commanded to precess, and of an algebraic non-iterative method of solving Equation (4-2a) without formally inverting the matrix [B].

To select the three axes about which rates are to be commanded ACL utilizes in addition to the vectors $\underline{a}_i$ and $\underline{b}_i$ given by equations (4-3a), (4-3b), and (4-3c) the unit angular momentum vectors of the three CMG's

$$\underline{h}_1 = \begin{pmatrix} \cos\alpha_1 \cos\beta_1 \\ \sin\beta_1 \\ -\sin\alpha_1 \cos\beta_1 \end{pmatrix}; \quad \underline{h}_2 = \begin{pmatrix} -\sin\alpha_2 \cos\beta_2 \\ \cos\alpha_2 \cos\beta_2 \\ \sin\beta_2 \end{pmatrix}; \quad \underline{h}_3 = \begin{pmatrix} \sin\beta_3 \\ -\sin\alpha_3 \cos\beta_3 \\ \cos\alpha_3 \cos\beta_3 \end{pmatrix} \qquad (4\text{-}11)$$

and the properties of these vectors as

$$\underline{a}_i \cdot \underline{b}_j = 0 \text{ for } i = j \qquad (4\text{-}12)$$

$$(\underline{a}_i \times \underline{b}_i) \times \underline{h}_i = 0 \qquad (4\text{-}13)$$

the latter of which meaning that $\underline{h}_i$ is normal to both $\underline{a}_i$ and $\underline{b}_i$.

The computational procedure for ACL can be outlined as follows:

1. Given the gimbal angles α_i and β_i, compute the vectors $\underline{a}_i$, $\underline{b}_i$, and $\underline{h}_i$.

2. In order not to chance selecting an entirely new set of gimbal axes at each computation cycle with the consequence of finding it necessary to set already established gimbal rates to zero at each cycle, save a fraction K_S of the currently available gimbal rates. (Note that these rates are essentially the same as those previously commanded by the control computer.) Therefore, compute the torque that still remains to be produced after saving the fractions $K_S\dot\alpha_i$ and $K_S\dot\beta_i$ of the existing gimbal rates. That is, compute

$$\Delta\underline{T} = \underline{T}_c - K_S\left(\sum_{i=1}^{3}\dot\alpha_{ic}\underline{a}_i + \sum_{i=1}^{3}\dot\beta_{ic}\underline{b}_i\right) \tag{4-14}$$

Proceed to find gimbal rates $\Delta\dot\alpha_{ic}$ and $\Delta\dot\beta_{ic}$ to produce $\Delta\underline{T}$. The rate commands $\Delta\dot\alpha_{ic}$ and $\Delta\dot\beta_{ic}$ added to the saved rates $K_S\dot\alpha_{ic}$ and $K_S\dot\beta_{ic}$ constitute the new gimbal rate commands.

3. Compute $(\Delta\underline{T}\cdot\underline{h}_i)$ for i = 1, 2, 3 and find i = m for which the dot product is a minimum. Since $\underline{h}_m$ is normal to the plane formed by the vectors $\underline{a}_m$ and $\underline{b}_m$ (see Equation 4-13), a small component of $\Delta\underline{T}$ along $\underline{h}_m$ means that $\Delta\underline{T}$ has a large component in the plane formed by $\underline{a}_m$ and $\underline{b}_m$. Thus step #3 determines the $\underline{a}_i$, $\underline{b}_i$ plane closes to $\Delta\underline{T}$.

4. Gimbal axes corresponding to α_m and β_m have now been selected. What remains is the selection of a third torque vector $\underline{a}_i$ or $\underline{b}_i$, which with $\underline{a}_m$ and $\underline{b}_m$ already selected form a reasonable triad. The safest practical choice, that is one which will never be bad even though in general not the best, is to choose as the third of the triad an $\underline{a}_i$ or a $\underline{b}_i$ which is closest to being normal to the plane formed by the vectors $\underline{a}_m$ and $\underline{b}_m$. But since $\underline{h}_m$ is normal to $\underline{a}_m$ and $\underline{b}_m$, this choice is tantamount to finding the $\underline{a}_i$ or $\underline{b}_i$ closest to $\underline{h}_m$.

Thus, compute $|\underline{h}_m\cdot\underline{a}_i|$ and $|\underline{h}_m\cdot\underline{b}_i|$ for $i \neq m$ and find i = n for which the dot product is a maximum. Let n = k if the vector selected is an $\underline{a}$, and let n = p if a vector $\underline{b}$ is chosen.

5a. If the choice in step #4 is $\underline{a}_k$ compute the desired gimbal rates $\Delta\dot\alpha_m$, $\Delta\dot\beta_m$ and $\Delta\dot\alpha_k$ as follows:

According to equations (4-12) and (4-13) dotting into the equation

$$\underline{a}_m\Delta\dot\alpha_m + \underline{b}_m\Delta\dot\beta_m + \underline{a}_k\Delta\dot\alpha_k = \Delta\underline{T}$$

in turn the vectors $\underline{h}_m$, $\underline{a}_m$ and $\underline{b}_m$ gives

$$(\underline{a}_k \cdot \underline{h}_m) \, \Delta \dot{\alpha}_k = \Delta \underline{T} \cdot \underline{h}_m$$

$$(\underline{a}_m \cdot \underline{a}_m) \, \Delta \dot{\alpha}_m + (\underline{a}_m \cdot \underline{a}_k) \, \Delta \dot{\alpha}_k = \Delta \underline{T} \cdot \underline{a}_m \qquad (4\text{-}15)$$

$$(\underline{b}_m \cdot \underline{b}_m) \, \Delta \dot{\beta}_m + (\underline{b}_m \cdot \underline{a}_k) \, \Delta \dot{\alpha}_k = \Delta \underline{T} \cdot \underline{b}_m$$

If the first is solved for $\Delta \dot{\alpha}_k$, its substitution into the remaining equations yields the values of $\Delta \dot{\alpha}_m$ and $\Delta \dot{\beta}_m$.

5b. If the choice in step #4 is $\underline{b}_p$ procedure similar to step #5a gives the equations

$$(\underline{b}_p \cdot \underline{h}_m) \, \Delta \dot{\beta}_p = \Delta \underline{T} \cdot \underline{h}_m$$

$$(\underline{a}_m \cdot \underline{a}_m) \, \Delta \dot{\alpha}_m + (\underline{a}_m \cdot \underline{b}_p) \, \Delta \dot{\beta}_p = \Delta \underline{T} \cdot \underline{a}_m \qquad (4\text{-}16)$$

$$(\underline{b}_m \cdot \underline{b}_m) \, \Delta \dot{\beta}_m + (\underline{b}_m \cdot \underline{b}_p) \, \Delta \dot{\beta}_p = \Delta \underline{T} \cdot \underline{b}_m$$

which can be solved for $\Delta \dot{\beta}_p$, $\Delta \dot{\alpha}_m$, and $\Delta \dot{\beta}_m$ as in step #5a.

6. Compute the gimbal rate commands to be issued as:

$$\dot{\alpha}_{ic} = K_s \dot{\alpha}_{ic}$$

for $i \neq m, k$ if the choice in step #4 was $\underline{a}_k$

$$\dot{\alpha}_{kc} = K_s \dot{\alpha}_{kc} + \Delta \dot{\alpha}_k$$

$$\dot{\alpha}_{mc} = K_s \dot{\alpha}_{mc} + \Delta \dot{\alpha}_m$$

and

$$\dot{\beta}_{ic} = K_s \dot{\beta}_{ic}$$

for $i \neq m, p$ if the choice in step #4 was $\underline{b}_p$

$$\dot{\beta}_{mc} = K_s \dot{\beta}_{mc} + \Delta \dot{\beta}_{mc}$$

$$\dot{\beta}_{pc} = K_s \dot{\beta}_{pc} + \Delta \dot{\beta}_{pc}$$

where the rate terms multiplied by K_S are the rate commands which had been issued at the preceding computational cycle.

Part of the preamble for the iterative control law (ICL) is the same as that for the ACL. If the desired control torque is $\underline{T}_d$ and the torque of the three CMGs due to their precession at the vehicle rates is equal to $\underline{T}_b$, the task of the control law is to compute gimbal rate commands $\dot{\alpha}_{ic}$ and $\dot{\beta}_{ic}$ so that the aggregate CMG torque due to precession at the rates $\dot{\alpha}_{ic}$ and $\dot{\beta}_{ic}$ is equal to

$$\underline{T}_c = \underline{T}_d - \underline{T}_b$$

The torque $\underline{T}_c$ is set equal to $\underline{T}_d$ when vehicle rates are insignificant or are too low to be sensed.

The salient feature of the ICL is its iterative selection of incremental gimbal rates to be commanded based on a consideration of the change of the "cost" of a gyro due to the incremental change of the rate of one of its gimbals. The change of the cost of a gyro is equal to the difference of its costs with and without the incremental gimbal rate change.

The cost function used by the ICL to compute the cost of a gyro can be of many forms, depending on the choice of its fundamental objective. The cost function used in the program is of the form

$$f_c = K_\alpha \left| \dot{\alpha}_i \right| + K_\beta \left| \dot{\beta}_i \right| + K_{\alpha\beta} H_i \left| \dot{\alpha}_i \dot{\beta}_i \cos \beta_i \right| .$$

For example, if $K_\alpha = K_\beta = K_{\alpha\beta} = 1$, the objective of a cost function of this type is to make the simultaneous use of both gimbals of the same CMG "expensive". It thus achieves a measure of electrical energy economy, since the total power output of the torque motors when both gimbals are simultaneously used is equal to $(2H_i \dot{\alpha}_i \dot{\beta}_i \cos \beta_i)$, which can be considerably larger than twice the power output of a torque motor corresponding to the use of a single gimbal axis.

For the computational procedure for ICL consider the nth iterative cycle. At the beginning of the nth iteration the computer has available the vectors $\underline{a}_i$ and $\underline{b}_i$, which it has computed at the start of the iterative process from the values of the six CMG gimbal angles. During the first $(n-1)$ steps of the iteration, the computer has made a number of gimbal axis choices for which it has computed rate commands $\dot{\alpha}_{ic}$ and $\dot{\beta}_{ic}$. These rates, if commanded and acquired, would produce a torque $\underline{T}_{(n-1)}$, which leaves still to be produced the remainder torque

$$\underline{T}_{rn} = \underline{T}_c - \underline{T}_{(n-1)}.$$

Corresponding to the rate commands computed during the first $(n-1)$ steps of the iteration, $\dot{\alpha}_{ic}$ and $\dot{\beta}_{ic}$, the computer also has available the individual cost of each gyro up to the beginning of the nth iteration step.

Based on this information, the computer's task during the nth step of iteration is to find the least expensive gimbal axis, about which the incremental rate command $\Delta\dot{\alpha}_{ic}$ (or $\Delta\dot{\beta}_{ic}$) would produce a torque making the magnitude of the remainder $\underline{T}_{r(n+1)}$ less than the magnitude of $\underline{T}_{rn}$.

Even though six gimbal axes are available from which to choose, for clarity, in the accompanying figures the choice between a single outer axis and one inner axis is indicated. The procedure during the nth iteration is as follows:

1. Compute the rates $\Delta\dot{\alpha}_i$ and $\Delta\dot{\beta}_i$ necessary to produce along the vectors $\underline{a}_i$ and $\underline{b}_i$, respectively, the torque vectors $\underline{T}_{\alpha i}$ and $\underline{T}_{\beta i}$, whose components along $\underline{T}_{rn}$ are equal to $\underline{T}_{rn}$. Figure 4-2a shows the relationship between vector $\underline{T}_{\alpha i}$, $\underline{T}_{\beta i}$, and $\underline{T}_{rn}$. The rates $\Delta\dot{\alpha}_i$ and $\Delta\dot{\beta}_i$ computed in this step are called test rates, R_{test}, and their values are used in cost evaluation.

From the definition of R_{test} or from Figure 4-2a, $R_{test} = \Delta\dot{\alpha}_i$ (or $\Delta\dot{\beta}_i$) can be computed as

$$(\underline{a}_i \cdot \Delta\dot{\alpha}_i) \cdot \frac{\underline{T}_{rn}}{\left|\underline{T}_{rn}\right|} = \left|\underline{T}_{rn}\right|$$

(a) R_{test} (b) R_{use}
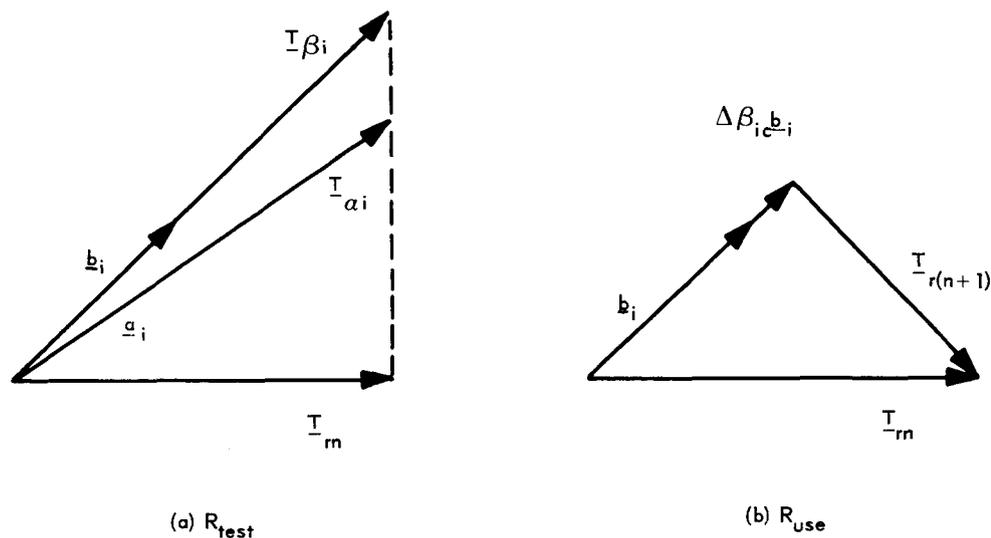
Figure 4-2. Illustration of Method to Compute R_{test} and R_{use}

from which

$$\Delta\dot{\alpha}_i = \frac{\left|\underline{T}_{rn}\right|^2}{(\underline{a}_i \cdot \underline{T}_{rn})} \quad .$$

Similarly,

$$\Delta\dot{\beta}_i = \frac{\left|\underline{T}_{rn}\right|^2}{(\underline{b}_i \cdot \underline{T}_{rn})} \quad .$$

2. Compute the cost accrual corresponding to the use of $\Delta\dot{\alpha}_i$ and $\Delta\dot{\beta}_i$ as

$$(\text{Cost})_{\Delta\dot{\alpha}_i} = f_c(\dot{\alpha}_i + \Delta\dot{\alpha}_i, \dot{\beta}_i) - f_c(\dot{\alpha}_i, \dot{\beta}_i)$$

and

$$(\text{Cost})_{\Delta\dot{\beta}_i} = f_c(\dot{\alpha}_i, \dot{\beta}_i + \Delta\dot{\beta}_i) - f_c(\dot{\alpha}_i, \dot{\beta}_i)$$

3. Select the gimbal axis whose R_{test} results in the smallest cost increment.

4. For illustration, assume that the ith inner gimbal axis is chosen in step No. 3 and refer to Figure 4-2b. In this step an incremental rate command, called R_{use}, is to be computed such that the torque $R_{use}\underline{b}_i$ (or $R_{use}\underline{a}_i$, if $\underline{a}_i$ is chosen in step No. 3) is equal to the component of $\underline{T}_{rn}$ along the vector $\underline{b}_i$ (or $\underline{a}_i$). Thus compute:

$$R_{use}\left|\underline{b}_i\right| = \underline{T}_{rn} \cdot \frac{\underline{b}_i}{\left|\underline{b}_i\right|}$$

for which

$$R_{use} = \Delta\dot{\beta}_{ic} = \frac{(\underline{T}_{rn} \cdot \underline{b}_i)}{\left|\underline{b}_i\right|^2}$$

or in the case of $\underline{a}_i$

$$R_{use} = \Delta\dot{\alpha}_{ic} = \frac{(\underline{T}_{rn} \cdot \underline{a}_i)}{\left|\underline{a}_i\right|^2}$$

4-12

5. Compute the total rate commands up to the nth iteration cycle by adding $\Delta \dot{\beta}_{ic}$ (or $\Delta \dot{\alpha}_{ic}$) to the corresponding rate $\dot{\beta}_{ic}$ (or $\dot{\alpha}_{ic}$) existing at the beginning of the nth iteration cycle.

6. Compute the cost of each axis for these rates.

7. Compute the remaining torque as

$$\underline{T}_{r,\,n+1} = \underline{T}_{rn} - R_{use}\underline{b}_i$$

or as

$$\underline{T}_{r,\,n+1} = \underline{T}_{rn} - R_{use}\underline{a}_i$$

If n is the maximum number of iterations allowed or if

$$\left| \underline{T}_{r,\,n+1} \right| \leq K_{TOLERANCE} \left| \underline{T}_d \right|$$

exit; otherwise, proceed to the next iteration cycle.

CONTROL LAW CHARACTERISTICS

Comparison of Control Laws

Simulation results contained in this section compare system responses for the baseline control law (BCL) and the iterative control law (ICL). Results for the algebraic control law (ACL) are not included as its performance is equivalent to that of the ICL with unlimited iterations and no cost function capability. The cost function used is simply $f_{ci} = |\dot{\alpha}_{ic}| + |\dot{\beta}_{ic}|$; i.e., the gimbal chosen is the one which produces the most control torque for the smallest gimbal rate.

A root locus plot for the spacecraft attitude control system is shown in Figure 5-1, and is plotted as a function of the variable gain (K_c) of the BCL. The nominal operating point is chosen at $s = -2.33 \pm i\,0.50$, for which K_c is equal to its nominal value of two. Since the iterative control law has a unity gain, the error channel gains for use with this control law are adjusted by the factor of two to provide the same system operating point. The two sets of error channel gains are chosen to give the same response for all three spacecraft control channels. For gimbal angles initially at zero, the response is also the same for either control law. The nominal system response to an attitude step command is shown in Figure 5-2. For ease of interpretation, the gimbal rate limiters are not included for the simulation results presented in this section. Also, small (0.01 degree) attitude commands are used so that the system will respond essentially as a linear system.

To compare the two control laws for off-nominal conditions, the case for all gimbal angles initially at 45 degrees is first considered. With the same 0.01 degree step command about the x axis, the system response with the BCL is shown in Figure 5-3. To explain the significant change in system response, consider the moments on the spacecraft as given by the simplified CMG precession torque (Equation 5.1) and the control law (Equation 5.2).

$$
\begin{vmatrix} T_x \\ T_y \\ T_z \end{vmatrix} = H
\begin{vmatrix}
\sin\alpha_1\cos\beta_1 & \cos\alpha_2\cos\beta_2 & 0 & \cos\alpha_1\sin\beta_1 & -\sin\alpha_2\sin\beta_2 & -\cos\beta_3 \\
0 & \sin\alpha_2\cos\beta_2 & \cos\alpha_3\cos\beta_3 & -\cos\beta_1 & \cos\alpha_2\sin\beta_2 & -\sin\alpha_3\sin\beta_3 \\
\cos\alpha_1\cos\beta_1 & 0 & \sin\alpha_3\cos\beta_3 & -\sin\alpha_1\sin\beta_1 & -\cos\beta_2 & \cos\alpha_3\sin\beta_3
\end{vmatrix}
\begin{vmatrix} \dot{\alpha}_1 \\ \dot{\alpha}_2 \\ \dot{\alpha}_3 \\ \dot{\beta}_1 \\ \dot{\beta}_2 \\ \dot{\beta}_3 \end{vmatrix}
\qquad (5.1)
$$
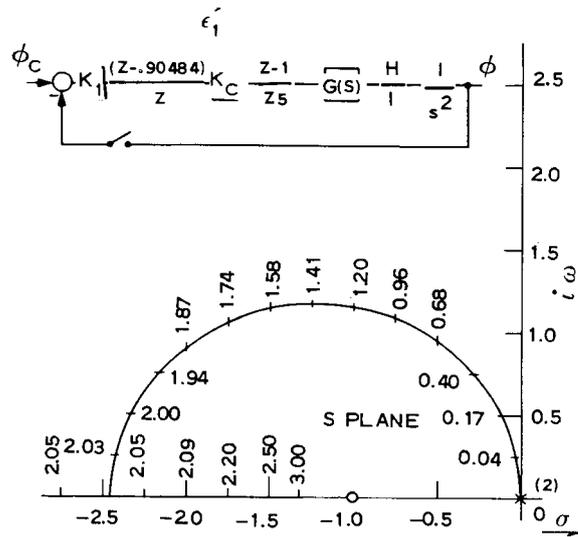
Figure 5-1. System Root Locus

$\alpha_1 = 0 \quad \beta_1 = 0$
$\alpha_2 = 0 \quad \beta_2 = 0$
$\alpha_3 = 0 \quad \beta_3 = 0$
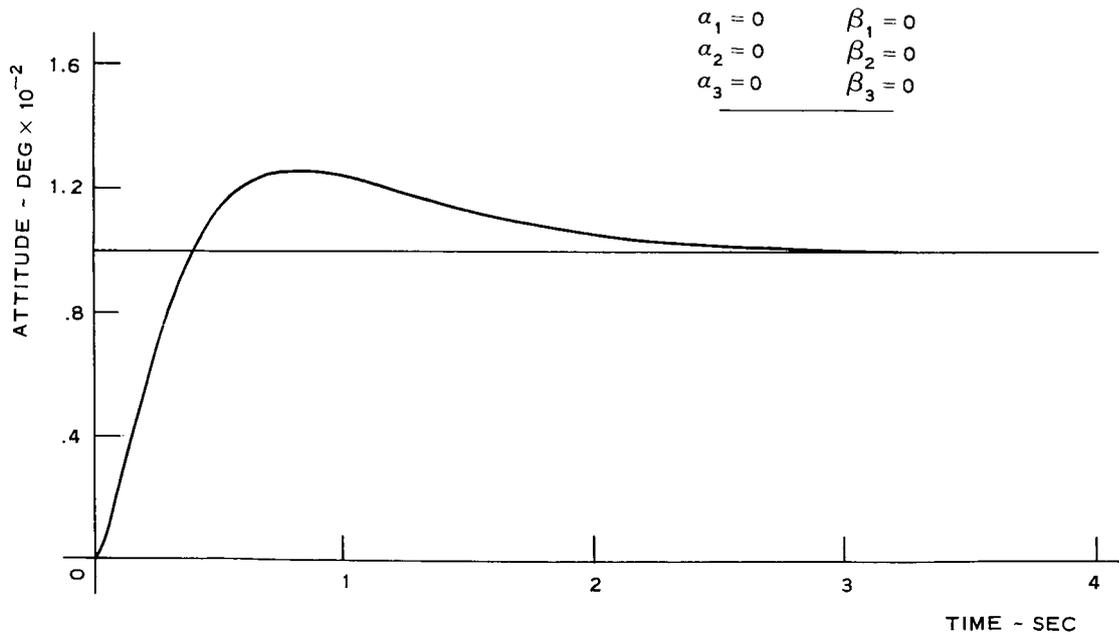
Figure 5-2. Nominal System Response
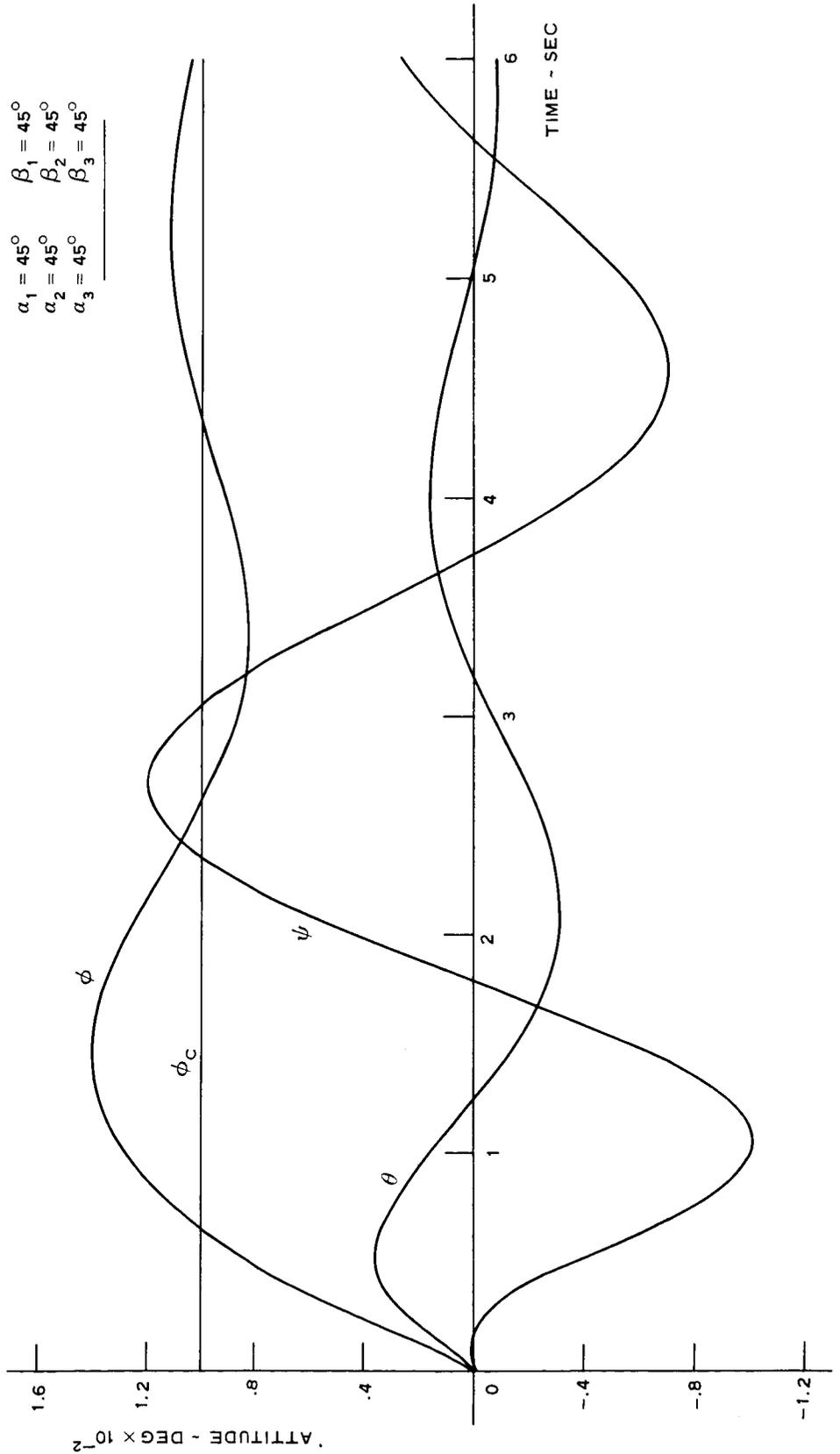
Figure 5-3. System Response with Baseline Control Law

$$\alpha_1 = 45° \qquad \beta_1 = 45°$$
$$\alpha_2 = 45° \qquad \beta_2 = 45°$$
$$\alpha_3 = 45° \qquad \beta_3 = 45°$$

$$\dot{\alpha}_{1c} = \epsilon_1' \sin \alpha_1 + \epsilon_3' \cos \alpha_1$$

$$\dot{\alpha}_{2c} = \epsilon_2' \sin \alpha_2 + \epsilon_1' \cos \alpha_2$$

$$\dot{\alpha}_{3c} = \epsilon_3' \sin \alpha_3 + \epsilon_2' \cos \alpha_3$$

$$\dot{\beta}_{1c} = - \epsilon_2' \cos \beta_1 \tag{5.2}$$

$$\dot{\beta}_{2c} = - \epsilon_3' \cos \beta_2$$

$$\dot{\beta}_{3c} = - \epsilon_1' \cos \beta_3$$

By assuming perfect CMG gimbal loop response, i.e., $\dot{\alpha}_i = \dot{\alpha}_{ic}$ and $\dot{\beta}_i = \dot{\beta}_{ic}$, equations (5.1) and (5.2) may be combined. First consider the moments on the vehicle resulting from an error channel value in the roll control loop. From equations (5.1) and (5.2)

$$T_x = H (\sin^2\alpha_1 \cos\beta_1 + \cos^2\alpha_2 \cos\beta_2 + \cos^2\beta_3)\, \epsilon_1'$$

$$T_y = H (\sin\alpha_2 \cos\alpha_2 \cos\beta_2 + \sin\alpha_3 \sin\beta_3 \cos\beta_3)\, \epsilon_1' \tag{5.3}$$

$$T_z = H (\sin\alpha_1 \cos\alpha_1 \cos\beta_1 - \cos\alpha_3 \sin\beta_3 \cos\beta_3)\, \epsilon_1'$$

where the prime indicates the error channel value at the output of the stabilization difference equation. For the 45 degree gimbal angles, Equation (5.3) becomes

$$T_x = H (1.207)\, \epsilon_1'$$

$$T_y = H (.707)\, \epsilon_1' \tag{5.4}$$

$$T_z = H (0) \qquad \epsilon_1'$$

As can be seen from the root locus (Figure 5-1), the operating point for the commanded roll channel has shifted to $s \simeq -1.0 \pm i\, 1.15$, corresponding to the change in the value of K_c from the nominal value of 2.0 to 1.207. More important is the cross-channel coupling which causes a significant pitching moment even though the pitch command is zero. From Equation (5.4), it would appear that there should be no coupling into the yaw channel, and because of the roll channel response this is true. However, as the pitch channel error builds up, due to roll channel coupling, it in turn couples into the yaw channel. This can be seen by combining equations (5.1) and (5.2) for the moments on the spacecraft due to an error channel value in pitch. The resulting equation is

$$T_x = H (\sin\alpha_2 \cos\alpha_2 \cos\beta_2 - \cos\alpha_1 \sin\beta_1 \cos\beta_1) \; \epsilon_2'$$

$$T_y = H (\sin^2\alpha_2 \cos\beta_2 + \cos^2\alpha_3 \cos\beta_3 + \cos^2\beta_1) \; \epsilon_2' \qquad (5.5)$$

$$T_z = H (\sin\alpha_3 \cos\alpha_3 \cos\beta_3 + \sin\alpha_1 \sin\beta_1 \cos\beta_1) \; \epsilon_2'$$

which for $\alpha_i = \beta_i = 45°$ becomes

$$T_x = H (\quad 0 \quad) \; \epsilon_2'$$

$$T_y = H (1.207) \; \epsilon_2' \qquad (5.6)$$

$$T_z = H (\; .707) \; \epsilon_2'$$

From Equation (5.6), the coupling moment from the pitch channel into the yaw channel is of the same magnitude as the coupling moment from the roll channel into the pitch channel. However, the angular excursion of ψ, in Figure 5-3, is about five times greater than θ. This is because the moment of inertia about the z axis is about five times smaller than that about the y axis. The yaw channel in turn couples back into the roll axis as shown in Equation (5.7). The underdamped response shown in Figure 5-3, which is not predicted by the root locus, is due to the cross-channel coupling via the BCL.

$$T_x = H (\sin\alpha_1 \cos\alpha_1 \cos\beta_1 + \sin\alpha_2 \sin\beta_2 \cos\beta_2) \; \epsilon_3'$$

$$T_y = H (\sin\alpha_3 \cos\alpha_3 \cos\beta_3 - \cos\alpha_2 \sin\beta_2 \cos\beta_2) \; \epsilon_3' \qquad (5.7)$$

$$T_z = H (\cos^2\alpha_1 \cos\beta_1 + \sin^2\alpha_3 \cos\beta_3 + \cos^2\beta_2) \; \epsilon_3'$$

The response shown in Figure 5-4 is for the same initial conditions, but using the ICL with a maximum of six iterations allowed. Since this control law commands a torque vector very nearly equal to the required torque vector, the cross-channel moments are very small. Moreover, since the control law gain is approximately unity, the system operating point, and thus its time response, is essentially invariant.

For an additional comparison, initial gimbal angles which cause the baseline control law to be rather ineffective are selected. With α_1 equal to zero and α_2 equal to ±90°, the baseline control law can command only one gimbal (β_3, modified by $\cos\beta_3$) to generate control torques about the vehicle roll axis. With initial gimbal angles, of
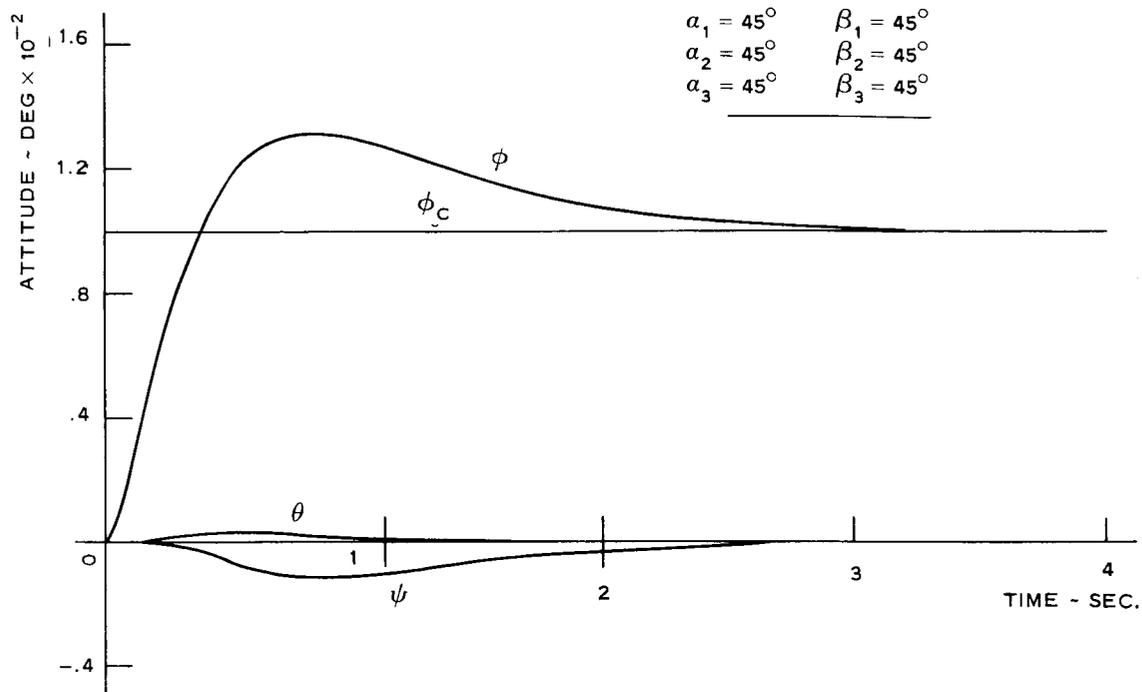
Figure 5-4. System Response with Iterative Control Law (Six Iterations)

$$\alpha_1 = 0° \qquad \beta_1 = 30°$$

$$\alpha_2 = 90° \qquad \beta_2 = 0°$$

$$\alpha_3 = 0° \qquad \beta_3 = 60°$$

equations (5.3), (5.5), and (5.7) reduce to equations (5.8), (5.9), and (5.10).

$$T_x = H (0.25) \; \epsilon_1'$$

$$T_y = H (0) \; \epsilon_1' \qquad\qquad (5.8)$$

$$T_z = H (-0.433) \; \epsilon_1'$$

$$T_x = H (-0.433) \; \epsilon_2'$$

$$T_y = H (2.25) \; \epsilon_2' \qquad\qquad (5.9)$$

$$T_z = H (0) \; \epsilon_2'$$

$$T_x = H(0) \epsilon_3'$$

$$T_y = H(0) \epsilon_3' \qquad\qquad (5.10)$$

$$T_z = H(1.866) \epsilon_3'$$

For this set of gimbal angles, from equation (5.8) and Figure 5-1, the roll channel operating point is at $s = -0.15 \pm i\,0.55$ for the system with the BCL. From the above equations, for a roll channel command, the only cross-channel coupling is from the roll channel into the yaw channel. Thus after the initial transient, (the yaw channel has nearly nominal response for the chosen gimbal settings), the yaw response simply follows the roll channel attitude, and the pitch attitude remains essentially zero. The slow, underdamped system response is shown in Figure 5-5.

The response for the same conditions using the iterative control law with six iterations is shown in Figure 5-6. For this case, the roll channel response is slightly slower than the nominal response of Figure 5-2. This is because the torque-produced vector did not sufficiently converge to the torque-required vector with the six iterations allowed. However, the cross-channel control torques have been eliminated. Figure 5-7 shows the same case with twelve iterations, and considering the severe gimbal positions for this example and the near nominal response produced, 12 iterations may be considered an upper limit for most applications.

While the previous examples have been for admittedly bad initial gimbal setting, it is clear that undesirable cross-channel control torques of significant magnitude will be present when the baseline control law is used, unless the gimbals are in the vicinity of their nominally zero values. For large gimbal excursions, the resulting variable gain can also cause poor system response. Even if the control law gain is adjusted as a function of the gimbal angles to retain good loop response, the undesirable cross-channel control torques will still be present.

It is also clear from the previous examples that the control laws developed during this study eliminate both of the above problems.
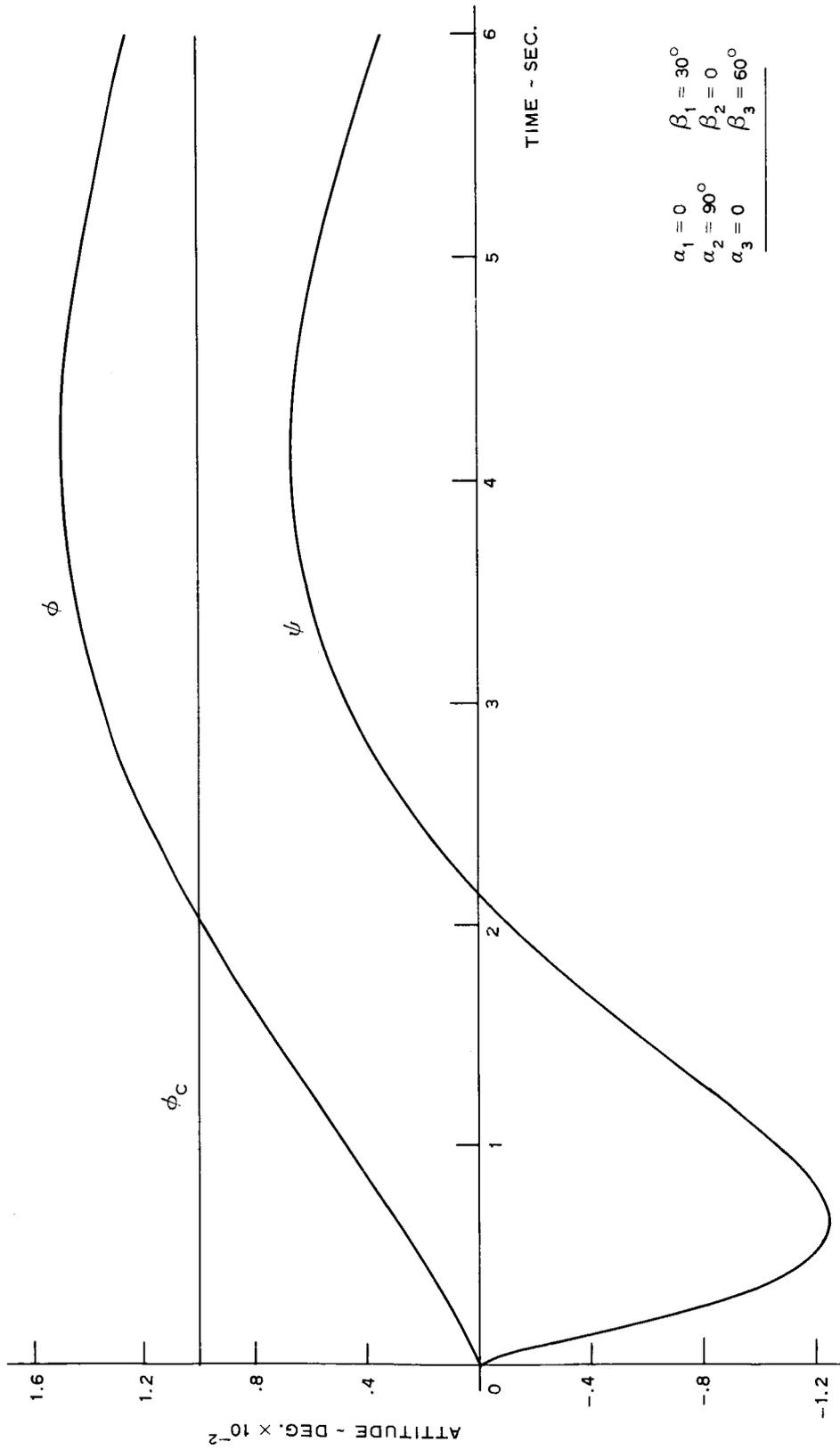
Figure 5-5. System Response with Baseline Control Law
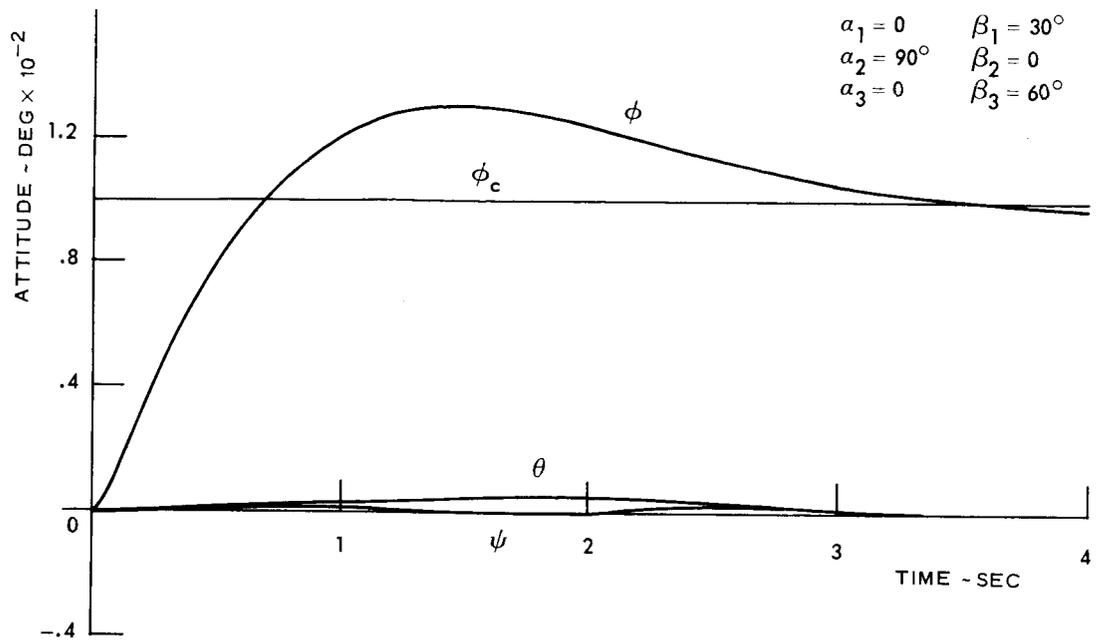
Figure 5-6. System Response with Iterative Control Law (Six Iterations)
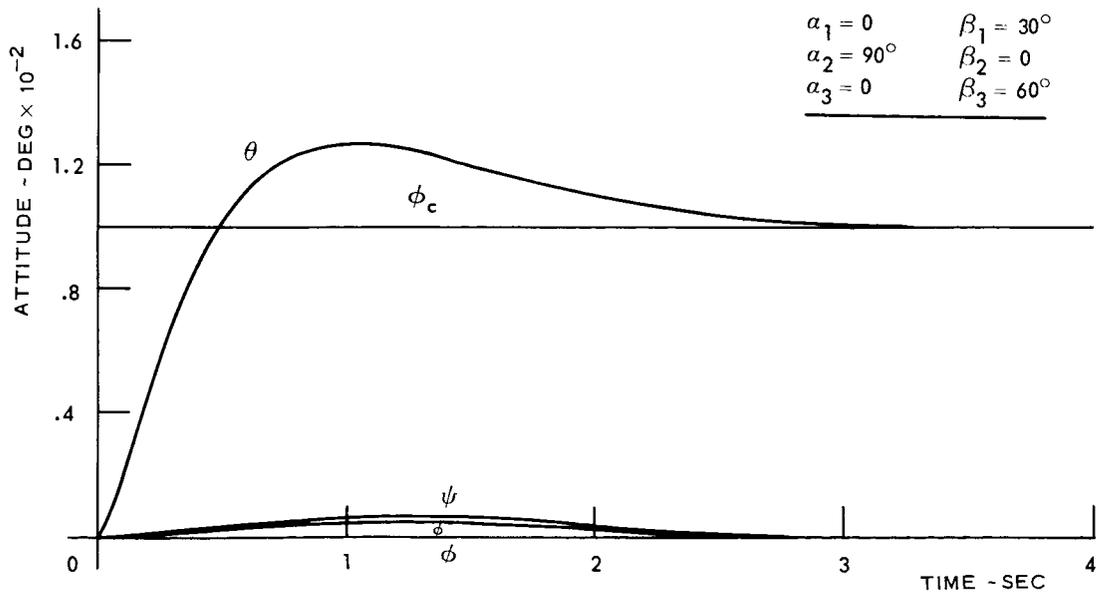
Figure 5-7. System Response with Iterative Control Law (Twelve Iterations)

ACL Versus ICL

The advantages and disadvantages of the two control laws can be tabulated as follows:

	Advantages	Disadvantages
Algebraic Control Law	Represents a moderate load on the control computer. Computes gimbal rates to produce the desired torque vector exactly.	Cannot be adapted to mission requirements. Can be more energy consuming than ICL, since two of the three gimbals it commands always belong to the same CMG.
Iterative Control Law	Can be adapted to mission requirements, and therefore can be made to emphasize energy or accuracy of resultant torques. Can compute gimbal rates to produce torques with any desired accuracy.	Represents a load on the computer considerably higher than ACL.

Consideration of the listed advantages and disadvantages of the two control laws leads to the conclusion that joining the two control laws would combine most of their advantages. Since where one control law is advantageous the other is not, and vice versa, the combination of most of their advantages would imply elimination of most of their disadvantages.

To this end, the following possibility should receive serious consideration. ICL selects two separate gimbal axes during its first two iterations; and regardless of the nature of additional choices in further iterations, most of the desired torque is eventually produced by the two gimbals first selected. Therefore, if ICL is used to select just those two gimbal axes according to a desired cost function, most of its superiority will be exploited. At this point, it is possible to switch to the ACL to select a third a_i or b_i vector to form a suitable triad with the two a_i and/or b_i already selected by ICL. Consequently, the desired rates along the selected gimbal axes can be computed to produce the desired torque vector exactly in a manner similar to that used in ACL.

Section 6

SIMULATION RESULTS

Introduction

A detailed description of the computer simulation program is presented in Volume II of the report. The program basically consists of the following parts:

- Simulation in floating point arithmetic of the environment portion of the system (i. e. , equation of rigid body motion of the system, CMG gimbal servo loops, sensor dynamics and error models).

- Simulation in fixed point arithmetic of the fixed point control computer. The computer word length, sensory data staleness and quantization are adjustable parameters.

- Simulation in floating point arithmetic of the control computer equations to provide the "true" values of the output variables of the fixed point simulation.

A significant number of runs have been made with the simulation corresponding to short intervals of real time. Of necessity, within the scope of the contract, no long computer runs could be made. For example, even though desirable, the long term time history of CMG gimbal angels in experiment #1 could not be studied. As a consequence of this limitation, results that can be reported pertain to what could be ascertained by short runs.

Verification of Root Locus Analysis

The root locus analysis used to select a sampling rate, corresponding values of system gain and z-domain compensation for stable operation and acceptable response, is neccessarily a linear process. The results of this phase of the study, reported in the Stabilization section, were verified by using the simulation to compute the response of the system to step and ramp commands.

It was confirmed that for very small steps the system response obtained by the exact simulation was in essential agreement with what was predicted by the root locus analysis, and the response given by a simplified simulation which did not contain exact CMG gimbal dynamics. For larger step commands, where the CMG gimbal servo non-linearities come into play, the step response from the exact simulation had a slower rise time and lower overshoot.

Once the validity of results based on the linearized model was established, it was not felt necessary to continue to verify every interim trial choice of operating point. However, to decide on final choices of sampling interval T, and of operating point (as defined by choice of T, system gain, and compensation) the simulation was used in the earth mapping experiment to determine how well the attitude and rate accuracy requirements of this experiment were actually satisfied. In this phase of the work, the control computer simulation in floating point arithmetic was used.

Choice of Control Computer Word Length

While programming in fixed point arithmetic the equations to be solved by the simulated control computer, certain operations were singled out as requiring double precision. After the completion of the simulation, preliminary runs pointed out additional computations which required double precision arithmetic.

To specify the control computer word length, the system response was checked for different word lengths and operating points in performance of the earth mapping experiment. This experiment was chosen because, among the four experiments, it is the hardest computationally, and most stringent from an attitude control viewpoint. As the earth mapping experiment takes longer than ten minutes in real time, the word length study was limited to the most critical portions of the experiment. To pinpoint these instances, the simulation with simplified equations of motion and with no CMG gimbal dynamics was used. It was discovered that for tracking a target in the orbital plane, the attitude error about the z axis of the vehicle has the form shown in Figure 6-1. This was obtained for a sampling interval of 0.01 second, K_x = 4742, K_y = 4711, K_z = 942.1 and a compensation zero at z = 0.90484.

As seen from the figure, the maximum attitude errors occur at the instants of peak acceleration and deceleration about the vehicle z axis. Since the attitude rate error is the derivative of the attitude error, the rate error peaks at the instant of maximum rate about the vehicle z axis (i.e., the zero crossing in Figure 6-1 corresponding to the instant the vehicle is directly over the target). Consequently, the conditions of maximum acceleration (where the attitude error is at its peak) and the maximum rate condition (where the rate error is maximal) were singled out for word length studies using the exact simulation.

Runs were made for these two cases with initial conditions of attitude and attitude rates set equal to their precomputed values to minimize transients. This was done to arrive at the quasi-steady state error values in the course of runs corresponding to four seconds in real time.

Word lengths of 17 bits plus sign, 15 bits plus sign, 13 bits plus sign and 11 bits plus sign were used for two sets of runs with the following operating point conditions:
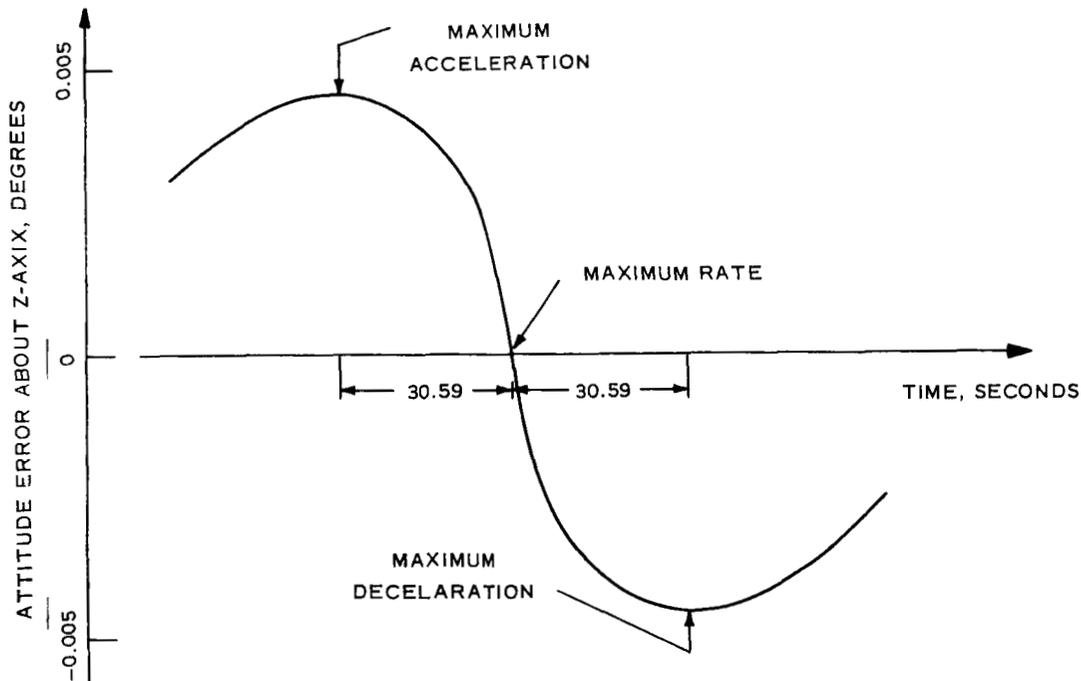
Figure 6-1. Attitude Error about the Vehicle Z Axis in the Earth Mapping Experiment (Simplified simulation result)

1. Sampling interval = 0.1 second
    K_x = 5556, K_y = 5525, K_z = 1105
    Compensation zero at z = 0.8825

2. Sampling interval = 0.15 second
    K_x = 2878, K_y = 2860, K_z = 572
    Compensation zero at z = 0.86071

The results obtained from these runs can be summarized as follows.

The peak attitude error for #1 and #2 operating points are 0.003 degree and 0.005 degree, respectively. The peak rate errors for #1 and #2 operating points are 2×10^{-4} and 4×10^{-4} degrees per second, respectively. Therefore, both operating points satisfy the accuracy specifications of 0.01 degree in attitude and 0.001 degree per second in rate for the earth mapping experiment.

It should be noted that the basic parameter that is varied in this case is the sampling interval from 0.1 to 0.15 second. In each case the zero location of a first order compensation was selected to operate the control loop at about the same natural frequency.

The results obtained for a single operating point but varying the word length indicate, after the initial transient, the same error values (less than one-percent variation for word lengths from 17 bits plus sign to 13 bits plus sign). For word lengths of 11

bits plus sign the error magnitudes increased about 10 percent. Therefore, on the basis of these short runs, it would appear that even 13 bits plus sign is sufficient to satisfy accuracy requirements. This is not unreasonable, since key computations are done double precision. However, for conclusive evidence it is essential to make longer computer runs.

No tradeoff of word length vs sampling rate could be discovered. It is conceivable that longer computer runs may have revealed some tradeoff.

To make these runs, the iterative control law (ICL) was used. Unlike the baseline control law (BCL), ICL generally commands fewer than six gimbals during each sampling interval. And since, during each computation cycle gimbals to be commanded are selected on a "cost" basis (even in runs where the required torque vector does not change direction), the control law is likely to switch gimbal axes as the gimbal angles change. In other words, in a common case, the ICL may command an already established gimbal rate to go to zero and another gimbal axis to essentially take its place. During the runs to establish word length requirement, some irregularity of actually produced torques, whenever a gimbal switching took place, was discovered. A sample of this is shown in Figure 6-2, which is a plot of the rate error about the vehicle z axis with the vehicle directly over the target, at t = 0.

$$\Delta T_s = 0.15 \text{ sec}, \Delta T_f = .075 \text{ sec}$$
$$K_1 = 2878, Z_0 = .86071$$
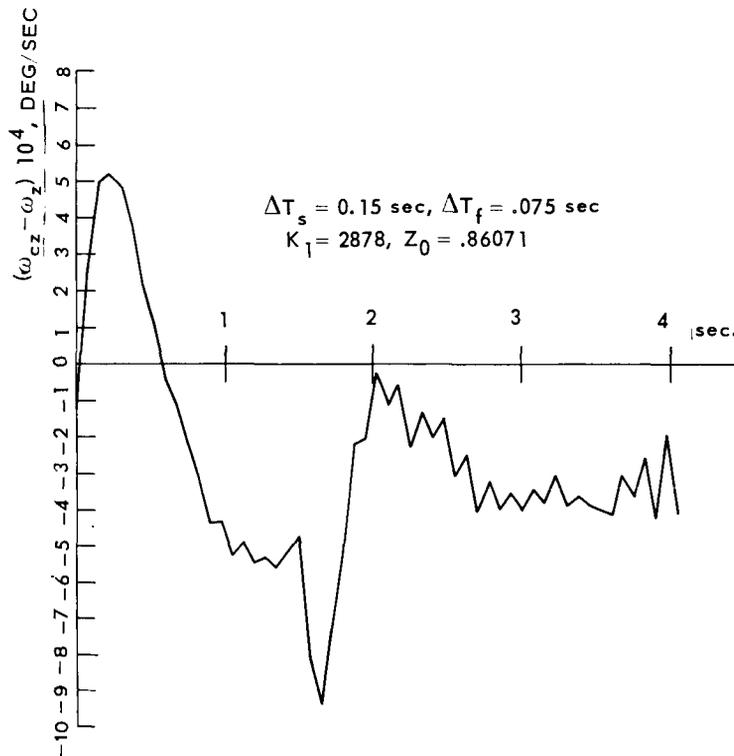
Figure 6-2.   Rate Error About Z—Axis Experiment #1, Vehicle Directly Over Target at t=o.

From the rate error spike, which starts at 1.5 seconds and causes the rate error to change by about 4.5×10^{-4} degrees per second, it is possible to compute the torque disturbance to have an average magnitude of 1.5 foot-pounds for 0.15 second. This is probably because as one gimbal rate is commanded to zero and a second is commanded to take its place, the rate decay of the first does not match the rate rise of the second. Coulomb friction may be the primary contributor to this discrepancy.

The gimbal switching torque disturbance was observed very late in the study when the rate error was plotted for earth-mapping experiment runs. The disturbance is not noticeable in attitude error plots. Consequently, no remedy for it has been implemented. However, it appears that a feature of saving a portion of the past gimbal rates, such as is used in the algebraic control law, would either eliminate or considerably reduce the switching disturbances. In any case, even with the spikes, the rate error specification of 0.001 degree per second is satisfied.

Compensation of Transportation Lags

When the indicated attitude of the vehicle is obtained by updating a direction cosine matrix, any delays in the rate gyro signals directly affect the attitude information. For instance, in the earth-mapping experiment—at a time when the rate about the vehicle z axis is one-degree per second—a delay of 0.1 second causes the indicated attitude to be in error corresponding to a rotation about the z axis by 0.1 degree. Moreover, the introduction of additional phase shift in the control loop affects the system response.

During the simulation studies, to counter the effect of transportation lags in the sensed rate, a simple linear extrapolation of the rate signals was used. Namely, if the signal available at time $t = nT$ is ω_{in}, for $i = x, y,$ or z, and if the signal value at $t = (n-1)T$ was $\omega_{i,n-1}$, the estimated signal

$$(\omega_{in})_{est.} = \omega_{in} + \frac{T_{delay}}{T_{sampling}} (\omega_{in} - \omega_{i,n-1})$$

was used in the direction cosine integration. This was tested in a short run of the earth-mapping experiment showing a tenfold improvement of the attitude error.

Experiment Results

Simulation runs for the earth mapping experiment have indicated that it should be possible to meet its accuracy requirements.

The inertial mode experiment proved unstable due to Coulomb friction and long time constants in the star trackers. Analysis and a computer run without the Coulomb friction have shown the system to be stable. A simulation run with greatly reduced loop gains and a compensation zero location chosen to result in a low natural frequency of the closed loop system also resulted in stable operation. One major source of difficulty is that the star tracker gimbal angle information, rather than the image plane errors, is used for attitude sensing. It is likely that combination of gimbal angels and image plan errors would lead to a stable system.

The horizon spectrometry experiment is operative if lack of star tracker update information can be tolerated.

In the microwave transmission experiment, the accuracy specification can be readily met. In this experiment the attitude error about the y axis is given in a gyro-compassing mode by sensing the component of orbital rate along the z axis. Consequently, as shown in Figure 6-3, the attitude control loop around the y axis is not closed in the conventional manner. A study of the characteristic equation of the attitude control system given in Figure 6-3 indicates that for stable operation, the gain in the forward path of the y-channel, K_z, must be significantly reduced. Simulation shows good system response with K_z equal to unity.
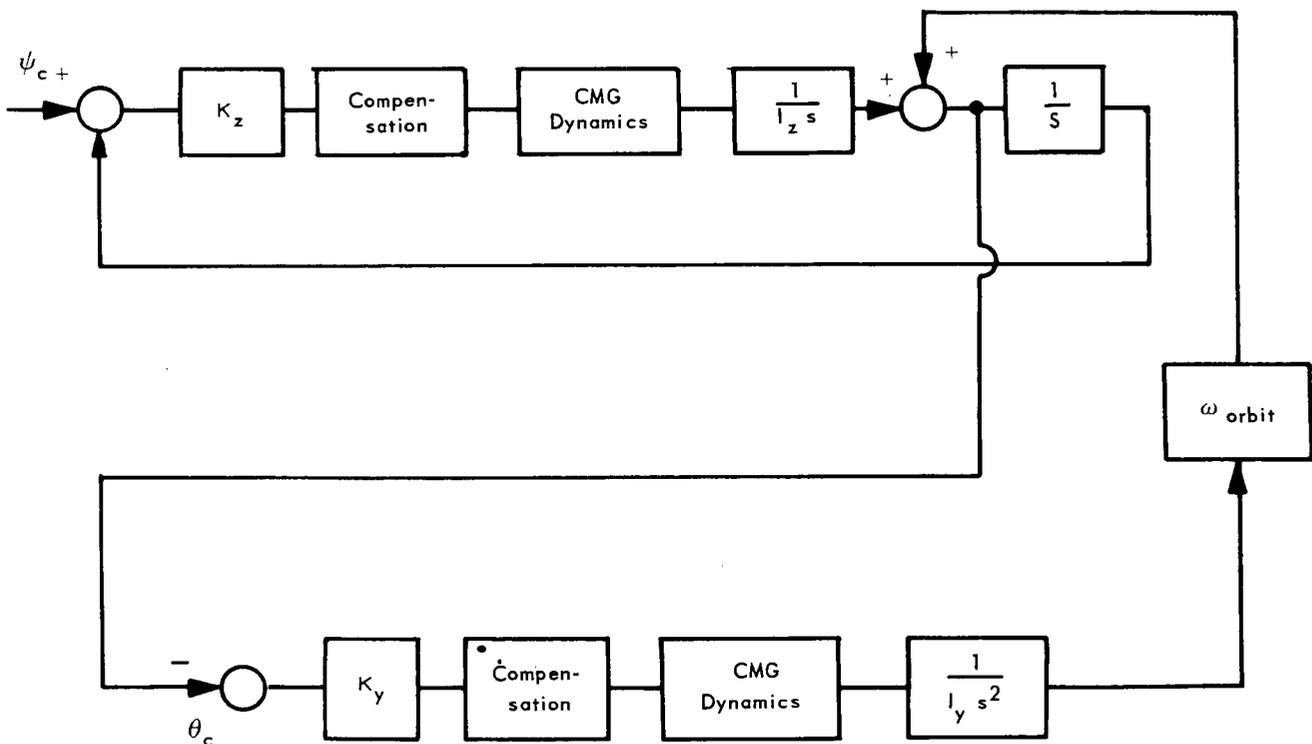
Figure 6-3. *Simplified y and z Axis Attitude Control Loop Showing Coupling Due to Gyrocompassing*

6-6

Section 7

CONTROL AND DISPLAY UNIT

Introduction

Recommendations for the design and layout of the control and display unit are based upon considerations of the type and format of data to be presented, engineering and human factors aspects, and methods of providing control functions such as digital data inserts and/or mode commands. The unit contains the necessary logic to provide a digital interface to the data adapter and computer. The use of integrated circuits and electroluminescent displays produce a control and display unit which minimizes unit size, weight, and power consumption while providing a high unit reliability. The recommended layout for the control and display panel is indicated in Figure 7-1. The design is discussed in detail in Appendix D.

CMG Monitor

The CMG monitor lights are commanded by the control computer as a result of a comparison of each individual quantity against limit values. The ON condition of the lights represent limit excesses in CMG rotor RPM and/or bearing temperatures. All lights can be reset simultaneously by a monitor reset button on the display panel, or individually by program initiation.

Display System

The special purpose display system consists of three rows of electroluminescent lamps, each row representing two address digits and a four digit plus sign data word. Each numeric display device associated with the data word has the capability of displaying a decimal point. In the primary mode of operation, the displays are under control of the display mode switch located on the front panel. The data displayed represents parameters called up by the operator such as body rates, Euler angles, etc. Operation of the display mode switch generates a system interrupt. Upon program recognition and interpretation of the interrupt, the desired data is located, formatted and transferred to the display panel. It should be noted that the control and display unit only initiates the display request and that the continued updating of that display (dynamic parameters) is a function of the CMG control computer program.
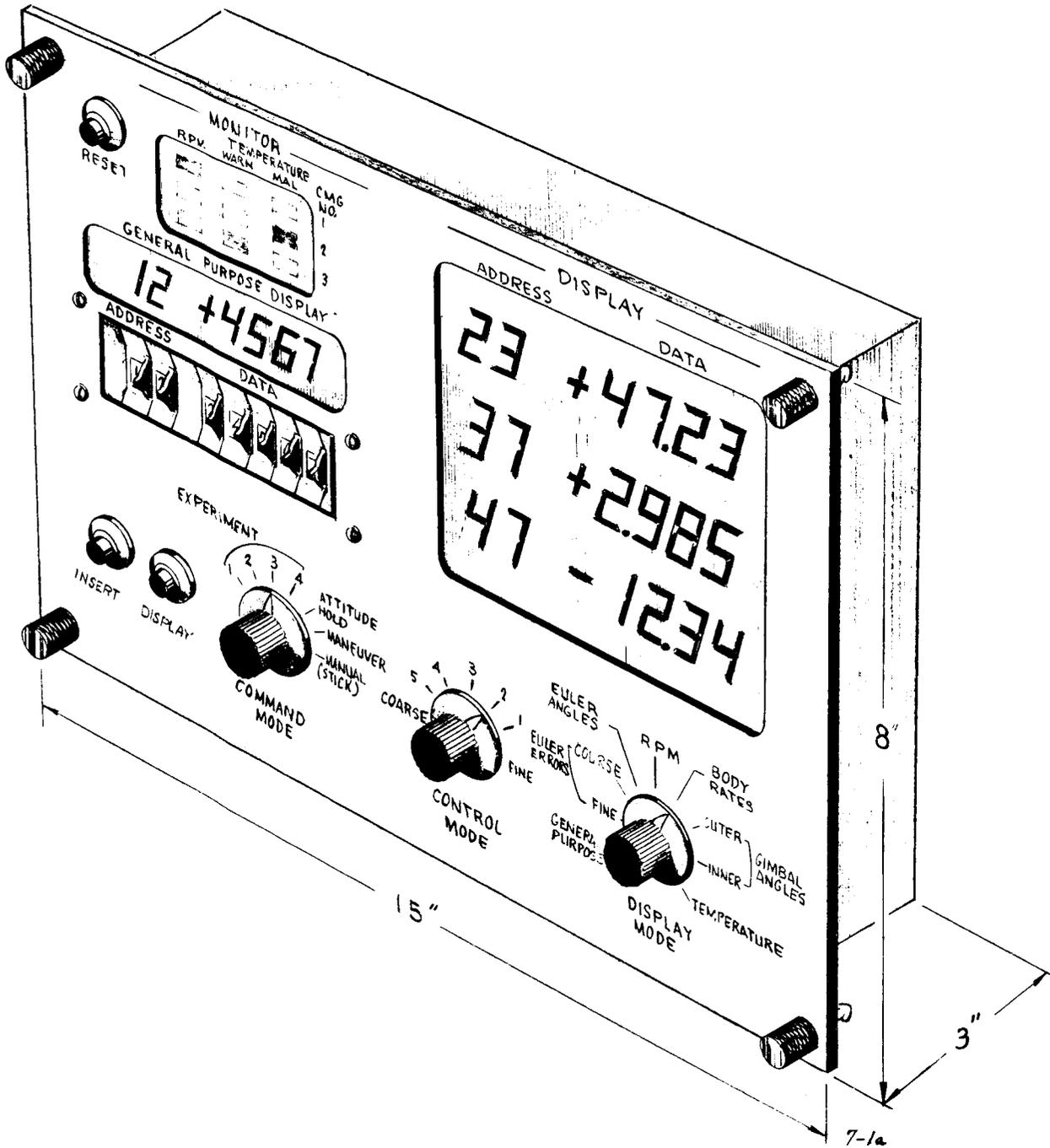
Figure 7-1. Controls and Display Panel

The display system provides a secondary mode of operation in which the display mode switch is used in conjuction with the data insert/display system. When the display mode switch is in the general purpose position, display parameters are called up by selecting an address via the thumbwheel switches and activating the display request.

Decimal Data Insert and Display

The decimal data insert and display consits of seven thumbwheel switches which define address and data, and the display or insert control switches. The system provides the capability to insert/display up to 99 decimal quantities directly into/from the computer. This number can be increased by utilizing extended addressing schemes.

The general purpose display operates in conjunction with the insert and display controls. Functionally, the insert/display unit requires a two-digit address and a display command to call up data from the computer. Both address and data are displayed on one of the four rows under control of the computer program. Selecting a two-digit address and a data word (four digits plus sign) and depressing the insert key will transfer the data to the computer, and (via the computer) display the address and transferred data on the general purpose display. By comparing the displayed address and data against the thumbwheel settings, the display operator can establish that the intended data was properly transmitted and received.

The requested data (or inserted data) will normally be displayed on the general purpose display, except when a special purpose display override command is given (by placing the display mode switch in the general purpose position). This, in conjunction with the correct display address, will cause the data to be displayed on one of the other three display rows. The control computer can be programmed to display up to three quantities per address inserted when in the general purpose display mode. The computer also performs the functions of data formatting, row determination, illegal address checking, etc. , as part of the display system subroutine.

Control and Command Mode Switches

The command mode switch allows the astronaut, via the control panel, to select the various modes of operation, i. e. , experiment, attitude hold, maneuver and manual. In the manual mode of operation, the astronaut uses the three row special-purpose display in conjunction with the display mode switch, to assist him in the control function.

The control mode switch permits the astronaut to choose a different set of control loop gains and/or compensation coefficients stored within the computer. Each of the switch positions corresponds to a predesigned control loop operating point, varying from course to fine (most accurate) in five increments. Other sets may be used by utilizing the insert data switches to define intermediate control levels.

Characteristics

The characteristics of the recommended control and display unit are:

Weight   = 6. 8 pounds

Volume   = 360 cubic inches

Power   = 7. 1 watts (average)

Reliability = 0. 97 (based on 50% utilization for a 30-day mission).

Section 8

CONTROL COMPUTER SIZING AND RECOMMENDATIONS

Control Computer Sizing Data

A basic sizing operation on the math flow necessary to implement the functions in Figure 8-1 has been performed. Volume II of this report includes the detailed math flow which conforms to Figure 8-1.

To obtain generally applicable data the following items have been considered:

a. Breakdown by function — Facilitates calculation of computational load if functions are combined, altered, or eliminated.

b. Breakdown by instruction type

   Add (load register, add, store)

   D. P. Add (load double word, D. P. add, store double word)

   Multiply (multiply, divide)

   D. P. Multiply (D. P. multiply, D. P. divide)

c. Breakdown by minor loop so that machine with and without indexing can be sized. This includes the add type instructions for incrementing and testing index registers and a count of the number of passes through the loop.

d. I/O operations have not been included since all I/O manipulations and timings are unique to each hardware configuration.

The computer speed and memory requirements for these functions can be determined from the preceding tabulation. The cycle execution time is obtained by choosing the longest path through the math flow and using the corresponding instruction count for those functions in conjunction with the execution times for the chosen machine. The memory load requirements are obtained from the data locations and instruction count for the control computer program listed on the last two pages.
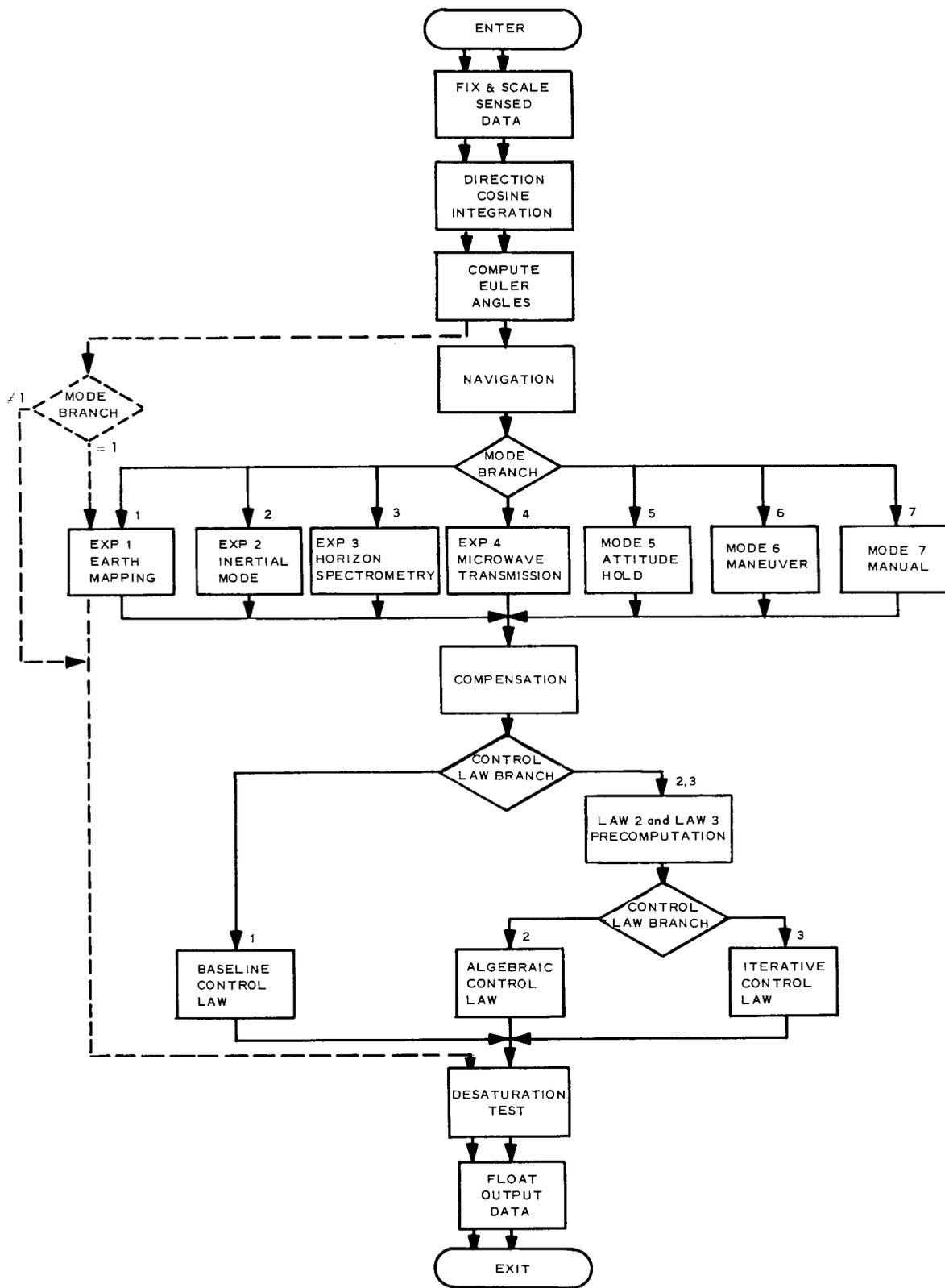
Figure 8-1. Simulated Control Computer Math Flow

The following is a tabulation of the sizing data.

Direction Cosine Integration

Adds	D.P. Adds	Multiplies	D.P. Multiplies	Adds for Indexing	Passes Through Loop
6	--	2	--	2	3
15	--	6	--	2	3
5	--	--	--	2	3
4	1	1	--	4	6
5	1	1	--	4	6
1	1	--	--	2	3
5	9	--	6	--	1

Instructions Executed

140	24	36	6	72

Euler Angles and Trig. Functions

Adds	D.P. Adds	Multiplies	D.P. Multiplies	Adds for Indexing	Passes Through Loop	
72	--	7	--	--	2	arctan
46	--	8	--	--	1	
3	--	1	--	2	3	arcsin
58	--	7	--	2	8	sin
58	--	7	--	2	12	sin
58	--	7	--	--	2	sin
6	--	--	--	--	1	

Instructions Executed

1481	--	179	--	46

Position and Velocity Computations

Adds	D.P. Adds	Multiplies	D.P. Multiplies	Adds for Indexing	Passes Through Loop
--	17	--	9	--	1
--	2	--	2	2	10
--	21	--	--	--	1
--	2	--	2	2	9
--	20	--	--	--	1
--	6	--	4	2	3
21	--	--	--	--	1

Instructions Executed

21	114	--	59	44

Experiment #1

Adds	D. P. Adds	Multiplies	D. P. Multiplies	Adds for Indexing	Passes Through Loop	
7	26	--	10	--	1	
1	7	--	2	2	3	
23	--	4	--	--	1	SQRT
3		1	--	2	3	
16	2	3	--	--	1	
58	--	7	--	--	2	
5	1	1	--	2	3	
	2	--	1	2	9	
20	--	6	--	--	1	
2	--	1		2	3	
5	--	--	--	--	1	
3	2	1	--	2	3	
6	1	1	--	--	1	
2	2	1	--	2	3	
3	12	1	6	--	1	
2	2	--	--	2	3	
5	--	--	--	--	1	
2	12	--	9	--	1	

Instructions Executed

Adds	D. P. Adds	Multiplies	D. P. Multiplies	Adds for Indexing
257	113	44	40	60

Experiment #2

Adds	D. P. Adds	Multiplies	D. P. Multiplies	Adds for Indexing	Passes Through Loop
16	11	8	3	--	1
2	--	--	--	2	3

Instructions Executed

Adds	D. P. Adds	Multiplies	D. P. Multiplies	Adds for Indexing
22	11	8	3	6

Experiment #3

Adds	D.P.Adds	Multiplies	D.P. Multiplies	Adds for Indexing	Passes Through Loop	
1	1	--	--	2	3	
7	6	5	--	--	1	
23	--	4	--	--	2	SQRT
3	--	1	--	2	6	
21	--	11	--	--	1	
3	--	--	--	--	1	
6	--	2	--	2	2	
24	12	9	--	--	1	
22	8	7	--	--	1	
4	12	4	--	2	3	
1	1	--	--	2	6	
--	6	--	6	--	1	
1	1	--	--	2	3	
16	18	4	6	--	1	
58	--	7	--	--	2	sin

Instructions Executed

Adds	D.P.Adds	Multiplies	D.P. Multiplies	Adds for Indexing
309	98	80	12	46

Experiment #4

Adds	D.P.Adds	Multiplies	D.P. Multiplies	Adds for Indexing	Passes Through Loop
9	3	--	--	--	1

Instructions Executed

Adds	D.P.Adds	Multiplies	D.P. Multiplies	Adds for Indexing
9	3	--	--	--

Mode 5

Adds	D.P.Adds	Multiplies	D.P. Multiplies	Adds for Indexing	Passes Through Loop
2	--	--	--	2	3
3	--	--	--	--	1
4	--	--	--	2	3
3	--	--	--	2	3
5	8	5	--	--	1

Instructions Executed

Adds	D.P.Adds	Multiplies	D.P. Multiplies	Adds for Indexing
35	8	5	--	18

Mode 6

Adds	D. P. Adds	Multiplies	D. P. Multiplies	Adds for Indexing	Passes Through Loop
3	--	--	--	--	1
3	--	--	--	2	3
7	--	--	--	--	1
12	--	--	--	2	2
14	--	3	--	--	1
3	1	1	--	2	3
1	7	--	--	--	1
3	6	6	--	--	1
1	1	--	--	2	3
4	--	--	--	2	3
13	--	5	--	--	1
3	--	--	--	2	3
5	8	5	--	--	1

Instructions Executed

112	27	22	--	34	

Mode 7

Adds	D. P. Adds	Multiplies	D. P. Multiplies	Adds for Indexing	Passes Through Loop
3	--	--	--	--	1
1	1	--	--	2	3
2	--	--	--	2	3
3	6	6	--	--	1
1	1	--	--	2	3
4	--	--	--	2	3
13	--	5	--	--	1
3	--	--	--	2	3
5	8	5	--	--	1

Instructions Executed

57	18	16	--	30	

Compensation

Adds	D.P. Adds	Multiplies	D.P. Multiplies	Adds for Indexing	Passes Through Loop
--	11	--	--	2	3
10	--	--	--	--	1
1	5	1	1	--	3
3	--	--	--	--	1
8	--	2	2	2	3
11	--	--	--	2	6

Instructions Executed

Adds	D.P. Adds	Multiplies	D.P. Multiplies	Adds for Indexing
106	48	9	9	24

Baseline Control Law

Adds	D.P. Adds	Multiplies	D.P. Multiplies	Adds for Indexing	Passes Through Loop
27	--	9	--	--	1

Instructions Executed

Adds	D.P. Adds	Multiplies	D.P. Multiplies	Adds for Indexing
27	--	9	--	--

Precomputation for Control Laws 2 and 3

Adds	D.P. Adds	Multiplies	D.P. Multiplies	Adds for Indexing	Passes Through Loop
4	--	1	--	2	3
26	--	3	--	--	1
31	--	2	--	2	3
12	--	--	--	--	1
54	--	36	--	--	1
3	--	--	--	2	3

Instructions Executed

Adds	D.P. Adds	Multiplies	D.P. Multiplies	Adds for Indexing
206	--	48	--	18

Algebraic Control Law

Adds	D. P. Adds	Multiplies	D. P. Multiplies	Adds for Indexing	Passes Through Loop
3	--	--	--	--	1
6	--	1	--	2	3
12	--	4	--	--	1
5	--	1	--	2	3
6	--	4	--	2	9
10	--	1	--	--	1
3	--	1	--	2	3
6	--	2	--	2	3
3	--	1	--	2	9
7	--	--	--	--	1
17	--	--	--	2	2
8	--	2	--	2	3
4	--	2	--	2	9
13	--	--	--	--	1
29	--	--	--	2	2
15	--	1	--	--	1
12	--	4	--	2	3
6	--	6	--	--	1
6	--	1	--	--	1
6	--	--	--	2	3

Instructions Executed

419	--	109	--	104

Iterative Control Law

Adds	D.P.Adds	Multiplies	D.P.Multiplies	Adds for Indexing	Passes Through Loop
4	--	--	--	--	1
10	--	2	--	2	3
4	--	2	--	2	9
8	--	--	--	--	1
3	--	1	--	2	3
7	--	2	--	2	3
31	--	5	--	--	3
4	--	--	--	2	3
6	--	2	--	2	9
9	--	--	--	--	1
32	--	2	--	2	3
31	--	5	--	--	6
5	--	1	--	--	1
3	--	1	--	2	3
6	--	1	--	--	1
31	--	5	--	--	1
3	--	1	--	2	3
3	--	1	--	--	1
4	--	--	--	--	1

Instructions Executed

$205+420\cdot ITER$	$48+68\cdot ITER$	--	$36+42\cdot ITER$	

Desaturation

Adds	D.P.Adds	Multiplies	D.P.Multiplies	Adds for Indexing	Passes Through Loop
22	--	--	--	2	3
11	--	--	--	--	1
27	10	--	--	2	3

Instructions Executed

158	30	--	--	12

Instruction Count for the Control Computer Program (with indexing capability)

	Instructions Using Single Precision Data	Instructions Using Double Precision Data
Direction Cosine Integration	67	18
Euler Angles and Trig. Functions	6	0
Position and Velocity Computations	27	85
Experiment #1	173	98
Experiment #2	28	14
Experiment #3	158	77
Experiment #4	9	3
Mode 5	28	8
Mode 6	89	15
Mode 7	10	1
Compensation	54	21
Baseline Control Law	36	0
Preliminaries to Control Laws	178	0
Algebraic Control Law	262	0
Iterative Control Law	157	0
Desaturation	64	10
Sine and Cosine Subroutine	65	0
Arc sine Subroutine	27	0
Arc tangent Subroutine	79	0
Square Root Subroutine	33	0
Cost Subroutine	36	0
TOTALS	1586	350

Data Locations Required for the Control Computer Program

	Single Precision* Locations	Double Precision* Locations
Input	37	--
Constants	16	--
Direction Cosine Integration	40	9
Euler Angles and Trig. Functions	25	--
Position and Velocity	3	61
Experiment #1	43	30
Experiment #2	14	--
Experiment #3	28	2
Experiment #4	--	--
Modes 5, 6, and 7	18	5
Compensation	72	3
Baseline Control Law	6	--
Preliminaries to Control Laws	37	--
Algebraic Control Law	37	--
Iterative Control Law	25	--
Desaturation	15	3
Subroutines	23	--
TOTALS	439	113

* Each entry represents the number of new data locations needed for that section of the program.

Computer Recommendations

The following paragraphs consider the applicability of two IBM 4 Pi space computer models for application. Characteristics of the two models are discussed in Appendix E.

Model TC. —The approximate memory load for the tabulated functions is 7500 bytes. With an estimate of an additional 2500 byte locations for scaling, CMG monitor, control and display, I/O and self-test and diagnostics, the total memory load estimate for the Model TC is 10 K bytes. Thus, for the presently defined system, the 16 K byte memory version of the Model TC appears sufficient from this standpoint.

For determining the execution time for the TC, the longest path through the math flow includes Experiment #1 of the command mode choices. Without control law computations, the resulting cycle time from the tabulated functions is approximately 78 milliseconds. For iterative control law, the execution time is dependent upon both the number of iterations and the cost function chosen. With six iterations, the execution times with the simplest cost function ($C_i = |\dot{\alpha}| + K |\dot{\beta}_i|$) and the most complex cost function considered are approximately 73 and 103 milliseconds, respectively. The total cycle time, including scaling time, is then approximately 165 milliseconds with the simpler cost function and 200 with the more complex one. For the baseline and algebraic control laws, the execution times are approximately 18.5 and 39.5 milliseconds respectively.

To allow for the peripheral computations (those not required for the control loops) a cycle time of 250 to 300 milliseconds is considered necessary. For the specified accuracy requirements, sampling intervals of this order are regarded as approaching the upper allowable limit. Based on this and the limited amount of spare memory, the model TC is recommended only if the presently specified system definition does not significantly increase in either size or accuracy requirements.

One should note that some saving in execution time can be achieved by more efficient programming techniques. In particular, in an experiment mode the Euler angles are computed for display purposes only, and their computation could be spread over several sampling intervals. Also a linear extrapolation technique may prove useful for the position and velocity calculation, thus allowing a lower integration rate. However, these reductions will probably be more than offset by undefined functions such as telemetry.

The increased capability of the model CP-2 leaves no doubt as to its capability for this application, and it also allows for considerable growth capacity. For the worst case timing discussed for the model TC, the model CP-2 execution time is only 46 milliseconds, a decrease by nearly a factor of four. Therefore, if a potentially significant increase in either memory or accuracy requirements is anticipated, the model CP-2 is recommended.

Section 9

RECOMMENDATIONS FOR FURTHER STUDIES

In the preceding section it is stated that 0.25 to 0.3 seconds is regarded as approaching the lower allowable limit for the sampling interval if the present accuracy requirements are still to be achieved. The question is whether or not the stability compensation can be designed to provide a fast enough system response at these lower sampling intervals while maintaining sufficient gain margin. If use of a control computer in the 4 Pi Model TC class is considered, further stability analysis and system simulation should be performed to determine the feasibility of these lower sampling rates.

It will be necessary to occasionally desaturate the CMGs. To optimize this is a problem of considerable magnitude, which requires answers as to what criterion to use to decide when to desaturate, and, once the decision is made, what procedure to use to find the best desaturation policy. It appears likely that a feasible scheme will share some of the ingredients of the algebraic and interative control laws, such as vectors $\underline{a}_i$, $\underline{b}_i$ and unit CMG angular momentum vectors $\underline{h}_i$. Since it is felt that there is still need for additional work in the area of control laws, it would be very logical to couple this in a single study effort with an investigation of the desaturation problem.

Appendix A

EXPERIMENT EQUATIONS

General

A number of corrections, modifications, and additions were made to the experiment equations supplied in the RFP. They are documented here so that Appendix A can be used as a supplement to the SPACECRAFT SENSOR DYNAMICS section of the RFP to form the source material on which the corresponding portion of the simulation is based.

So that they are not lost in the detail, the following points are mentioned first:

1. For programming uniformity, the definition of the I-frame for the Horizon Spectrometry Experiment (Experiment #3) in the RFP is adapted for all four experiments. According to this definition, the I-frame is geocentric with its Y_I axis along the earth's rate vector and the X_I and Z_I axes in the equatorial plane forming a right handed triad.

   This choice does not affect the first two experiments, since in those experiments the choice of I-frame is immaterial as long as it is known how it is defined. However, in the Microwave Transmission Experiment (Experiment #4) some of the equations had to be modified to comply with the choice of I-frame definition different from that specified for Experiment #4 in the RFP. The pertinent changes of Experiment #4 equations are listed under Modifications.

2. It is important to note that in the RFP the rotations A_c (telescope azimuth angle command in Experiment #1) and e_i (star tracker elevation gimbal angles in Experiment #3) are left handed. The angles e_i in Experiment #2 are right handed.

3. In the experiments section of the RFP, the attitude error is given as a small angle error vector $\underline{\epsilon}$, which when primed corresponds to indicated errors. The components of $\bar{\epsilon}$ are ϵ_x, ϵ_y, and ϵ_z, which are the small rotations necessary to rotate the B_c-frame (commanded vehicle attitude) into the B-frame (actual vehicle attitude) as shown by the following equation.

$$
\begin{pmatrix} \underline{i} \\ \underline{j} \\ \underline{k} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & \epsilon_x \\ 0 & -\epsilon_x & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & -\epsilon_y \\ 0 & 1 & 0 \\ \epsilon_y & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & \epsilon_z & 0 \\ -\epsilon_z & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \underline{i}_c \\ \underline{j}_c \\ \underline{k}_c \end{pmatrix}
$$

When these rotations are considered singly they indicate that:

ϵ_i = (Actual rotation about the i th vehicle axis)
    − (Desired rotation about the i th vehicle axis).

This is opposite in polarity to the error signal required in the attitude control loops. Therefore, it is necessary to recognize that the polarity of the components of the small angle error vector as supplied by the experiment equations has to be changed in order to achieve negative feedback.

Corrections

The correct version of the equations found in error in the SPACECRAFT SENSOR DYNAMICS section of the RFP are listed bearing their original numbering.

In the Inertial Mode experiment, the following equation should read:

$$\epsilon_z' = \frac{1}{\Delta'} (s_{2zI} U_{1x} - s_{1zI} U_{2x})$$ equation (3) of the set 2.1

$$\Delta_2 = \cos E_1 \cos E_2 \sin (A_1 - A_2)$$

$$\Delta_3 = \sin A_1 \sin E_1 \cos E_1$$ 4th and 5th lines of the set 2.2

$$\Delta y_1 = (a_1 - a_1') \cos e_1$$

$$\Delta y_2 = (a_2 - a_2') \cos e_2$$ equation (10-a) and (10-b) of the set 2.5

(This is in order to have negative feedback in the star tracker azimuth servo simulation. Otherwise, if Δy is defined as in the RFP, it is necessary to use $(-\Delta y)$ as the error quantity in the azimuth servo loop.)

In the Horizon Spectrometry experiment, the following equations should read:

$$\sin \phi = \frac{u}{\sqrt{u^2 + w^2}}$$ equations (7) and (8) of the set 2.1.3

$$\cos \phi = \frac{w}{\sqrt{u^2 + w^2}}$$ (They appear correctly in the Appendix A of the Horizon Spectrometry experiment.)

$$\Delta = \cos A_1 (\cos E_1)^2 \cos E_2 \sin (A_2 - A_1)$$

$$\Delta' = \cos E_1 \cos E_2 \sin (A_2 - A_1)$$ first two equations of the set 2.1.14

$$\left.\begin{array}{l} \Delta x_1 = (a_1 - a_1') \cos e_1 \\[8pt] \Delta z_1 = (e_1 - e_1') \\[8pt] \Delta x_2 = (a_2 - a_2') \cos e_2 \\[8pt] \Delta z_2 = (e_2 - e_2') \end{array}\right\} \quad \text{equations (36), (37), (38), (39) of the set 2.3.2}$$

$$\alpha_{XT} = -\int_0^t (\Omega_X \cos a' + \Omega_Y \sin a') \, dt' \qquad \text{equation (40) of the set 2.3.3}$$

Modifications

Earth mapping experiment. —The direction cosine subroutines differ from corresponding equations in the RFP as follows:

In the $M_{B_C I}$ matrix the elements of the third column stay constant at their initial values. That is,

$$M_{13} = M_{130}$$

$$M_{23} = M_{230}$$

$$M_{33} = M_{330}$$

The updating subroutine for $M_{B_C I}$ solves for the first column elements by integrating according to equations (8-1), (8-3), (8-5) of the set 2.1. The elements of the second column are obtained by the cross product of the third and first column elements. This is done to insure better orthogonality of the updated matrix.

In the case of the M_{BI} matrix, the elements of the first two rows are updated using the first six equations of the set 2.2. The elements of the third row are obtained by the cross product of the first row elements into the second row elements.

Inertial mode experiment. —The method indicated in the RFP to account for the base loading of the star tracker telescopes is by consideration of the vehicle small angle motion as resolved along the outer and inner gimbal axes of the star trackers. In the star tracker block diagram these components are differentiated to obtain the rate of angular motion of the star trackers about their respective gimbal axes.

In the simulation, the same end has been accomplished by resolving the vehicle rate vector $\underline{\omega}$ along the telescope gimbal axes so that the need for differentiation is eliminated. With this modification in equations 2.6, the components of spacecraft perturbation rates along gimbal axes become:

$\underline{\omega}_x$ along the outer gimbal axis of both trackers

and

$$\underline{\omega} \cdot y_1' = \omega_y \cos a_1' + \omega_z \sin a_1'$$
$$\underline{\omega} \cdot y_2' = \omega_y \cos a_2 + \omega_z \sin a_2$$

equations (11-a) and (11-b) of the set 2.6

along the inner gimbal axes for both trackers. Note that in Equation (11-b) the bore-sighting value of the azimuth angle is used since in this experiment the actual tracker azimuth a_2' is not required.

Microwave transmission experiment. —The adoption of the I-frame definition in the Horizon Spectrometry experiment for all four experiments necessitates the modification of some of the equations in the Microwave transmission experiment. They are modified to read as follows:

$$\epsilon_x = m_{12} \cos \phi - m_{32} \sin \phi$$
$$\epsilon_y = -m_{13} \sin \phi \sin \lambda + m_{23} \cos \lambda - m_{33} \cos \phi \sin \lambda$$
$$\epsilon_z = m_{11} \sin \phi \cos \lambda + m_{21} \sin \lambda + m_{31} \cos \phi \cos \lambda$$

equations (7), (8), and (9) of the set 2.2

$$\lambda_R = \sin^{-1} \left(\frac{y}{r} \right)$$
$$\cos \phi = \frac{z}{\sqrt{x^2 + z^2}}$$
$$\sin \phi = \frac{x}{\sqrt{x^2 + z^2}}$$

equations (12), (13), and (14) of the set 2.2

$$y(0) = 0, \ z(0) = r(0)$$

$$\begin{pmatrix} m_{110} & m_{120} & m_{130} \\ m_{210} & m_{220} & m_{230} \\ m_{310} & m_{320} & m_{330} \end{pmatrix} = \begin{pmatrix} -1/\sqrt{2} & 0 & 1/\sqrt{2} \\ 1/\sqrt{2} & 0 & 1/\sqrt{2} \\ 0 & 1 & 0 \end{pmatrix}$$

suggested initial condition, equation set of 2.3

Additions to the Earth Mapping Experiment

In the RFP, the assumption is made that the slant range to the target, $\underline{s}$, and the vector slant range rate with respect to the earth in inertial coordinates, $(\underline{V})_I$, are known. That is, they are for instance provided by an on-board guidance computer. If, on the other hand, they are to be computed by the control computer, it has to carry out Kepler orbit extrapolation of data transmitted to the vehicle from tracking stations.

In the digital simulation of the environment of the system and of the fixed point control computer, the vehicle velocity and position in the I-frame is computed by two different methods. To provide correct values against which to compare the results of the fixed point limited word length control computer, in the environment portion of the simulation the Kepler's equation is solved iteratively by floating point arithmetic. The simulation of the control computer uses a Taylor Series expansion of the coordinates and velocity of a body in Keplerian motion (the so called f and g series method[*]) to update the vehicle position and velocity from information assumed telemetered to the vehicle at the start of the experiment.

Navigation by f and g series. —Given the initial position and velocity vectors at time t_0, the position of the vehicle at time t is computed by using the Taylor series expansion

$$\underline{u} = f \, \underline{u}_0 + g \, \dot{\underline{u}}_0$$

where

$$\underline{u} = \frac{\underline{\rho}}{\rho_{Earth}} = \text{Normalized position vector at time t}$$

$$\underline{u}_0 = \frac{\underline{\rho}_0}{\rho_{Earth}} = \text{Normalized position vector at } t = t_0$$

$$\dot{\underline{u}}_0 = \frac{\dot{\underline{\rho}}_0}{V_0} \doteq \text{Normalized velocity vector at } t = t_0$$

$$V_0 = \text{Circular velocity at earth's radius}$$

---

[*]Sconzo, P., LeSchak, A. R., Tobey, R., "Symbolic Computation of f and g Series by Computer," IBM Technical Report 00.1262-1, June 1965.

and

$$f = \sum_{n=0}^{\infty} f_n \frac{\tau^n}{n!}$$

$$g = \sum_{n=0}^{\infty} g_n \frac{\tau^n}{n!}$$

for

$$\tau = \left(\frac{t - t_0}{T_0} \right) 2\pi$$

with T_0 = Period of circular orbit at earth's radius. Expressions for f_n and g_n are given in the cited reference.

To update the vehicle velocity the Taylor series expansion

$$\underline{\dot{u}} = f' \underline{u}_0 + g' \underline{\dot{u}}_0$$

is used, where

$\underline{\dot{u}}$ = Normalized velocity vector at time t

$$f' = \sum_{n=1}^{\infty} f_n \frac{\tau^{(n-1)}}{(n-1)!}$$

$$g' = \sum_{n=1}^{\infty} f_n \frac{\tau^{(n-1)}}{(n-1)!}$$

Number of terms required in the f and g series. —To compute desired vehicle angular rate and attitude, the method used in the RFP is to compute the desired rates from

$$\omega_{cz} = \Omega_{EIz} + \Omega_{EIy} \tan A_c + \frac{V_{cx}}{s \cos A_c}$$

(A-1)

$$\omega_{cx} = \omega_{cy} = 0$$

and determine the desired vehicle attitude by updating the properly initialized matrix M_{B_cI} by using the rates in (A-1). The matrix M_{B_cI} relates the commanded vehicle coordinates to the I-frame.

Alternatively, it is possible to use (A-1) to compute desired rate, but determine the desired vehicle attitude from the components of the slant range vector $\underline{s}$. Since the slant range vector is given by

$$\underline{s} = \underline{\rho} - \underline{R} \qquad\qquad\qquad (A-2)$$

determination of desired attitude by using this method places an accuracy requirement on the computation of the vehicle position vector ρ. For a given update interval this accuracy requirement should be equivalent to the accuracy requirement on ω_{cz} when ω_{cz} is essentially integrated through the same updating interval to compute M_{B_cI} matrix. Therefore, the accuracy requirement on ρ in (A1) to meet the attitude accuracy specification of experiment #1 is used as one requirement by which to determine the number of terms needed in the f and g series.

To consider a worst case, when the vehicle is directly overhead the target, the vehicle position vector ρ and the commanded vehicle y-axis should be colinear. Therefore, if an attitude accuracy requirement ten times smaller than 0.01° (specified in the RFP) is adapted, it follows that

$$\frac{\left| \rho_{x_c} \right|}{200 \text{ N.M.}} \leq \frac{0.001°}{57.3}$$

or

$$\left| \rho_{x_c} \right| \leq \frac{0.2}{57.3} \text{ N.M.}$$

But in the f and g series method the position vector ρ is normalized by dividing it by the earth's radius, ρ_ϵ. Consequently, the f and g series should contain enough terms so that

$$\frac{\left| \rho_{x_c} \right|}{\rho_\epsilon} \leq \frac{0.2}{57.3 \times 3440} = 1.015 \times 10^{-6} \qquad\qquad (A-3)$$

This is one requirement. In addition, it is necessary to establish requirements corresponding to the ω_{cz} computation according to the Equation (A-1). Again for the vehicle directly over the target case, Equation (A-1) reduces to

$$\omega_{cz} = \Omega_{EIz} + \frac{V_{cx}}{s}$$
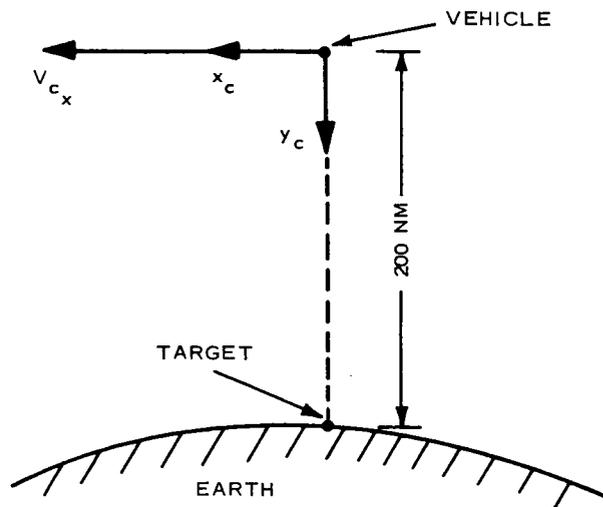
Figure A-1. Orientation of Commanded Vehicle Coordinates When the Vehicle is Directly Over the Target

from which

$$\Delta \omega_{cz} = \frac{\Delta V_{cx}}{s} - \frac{V_{cx}}{s^2} \Delta s \tag{A-4}$$

The accuracy specified in the RFP for rate is 0.001°/sec. Adoption of a specification limiting the contributions of each error source in (A-4) ten times smaller than 0.001°/sec yields the requirements

$$\frac{\left|\Delta V_{cx}\right|}{s} \leq \frac{10^{-4}}{57.3} \tag{A-5}$$

and

$$\frac{\left|V_{cx}\right|}{s^2} \left|\Delta s\right| \leq \frac{10^{-4}}{57.3} \tag{A-6}$$

For the purposes of this section it is safe to assume that the velocity of the vehicle with respect to the earth is the same as its inertial velocity, which is updated by the use of the f and g series. Given that in this method $\underline{V}$ is normalized by dividing V_O (orbital velocity at earth's radius) into it, from Equation (A5) derives the requirement

$$\frac{\left|\Delta V_{cx}\right|}{V_o} \leq \frac{10^{-4}s}{57.3V_o} = \frac{10^{-4} \times 200 \times 6080}{57.3 \times 25900}$$

(A-7)

$$\frac{\left|\Delta V_{cx}\right|}{V_o} \leq 0.82 \times 10^{-4}$$

This is the second condition the f and g series update has to satisfy.

Finally, from Equation (A-6) the third requirement is obtained. It can be readily shown that the component of error in ρ_- contributing to Δs is ρ_{yc} (Figure A-1). Thus setting $\left|\Delta s\right| = \left|\Delta \rho_{yc}\right|$ in equation (A6), and normalizing $\Delta \rho_{yc}$ yields the requirement

$$\frac{\left|\Delta \rho_{yc}\right|}{\rho_\epsilon} \leq \frac{10^{-4}}{57.3} \frac{s^2}{\left|V_{cx}\right| \rho_\epsilon}$$

where for 200 N.M. circular orbit the substitution of

$$\left|V_{cx}\right| = 4.15 \text{ NM/sec}$$

yields

$$\frac{\left|\Delta \rho_{yc}\right|}{\rho_\epsilon} \leq \frac{10^{-4} (200)^2}{57.3 (4.15) (3440)}$$

$$\frac{\left|\Delta \rho_{yc}\right|}{\rho_\epsilon} \leq 0.49 \times 10^{-5}$$

(A-8)

This added to the already derived requirements

$$\frac{\left|\rho_{xc}\right|}{\rho_\epsilon} \leq 1.015 \times 10^{-6}$$

(A-3)

$$\frac{\left|\Delta V_{cx}\right|}{V_o} \leq 0.82 \times 10^{-4}$$

(A-7)

completes the accuracy specifications on the f and g series computations.

To determine the number of terms of the f and g series necessary to meet its accuracy requirements, if updating a circular orbit at earth's radius is considered, it can be shown that for 10 terms the inaccuracies corresponding to the Equations A3, A7, and A8 at the end of an updating period of quarter of an orbit are:

$$\frac{\left|\rho_{xc}\right|}{\rho_\epsilon} = 1.01 \times 10^{-6} \qquad\qquad \text{(see Equation A-3)}$$

$$\frac{\left|\Delta\rho_{yc}\right|}{\rho_\epsilon} = 0.35 \times 10^{-6} \qquad\qquad \text{(see Equation A-8)}$$

$$\frac{\left|\Delta V_{cx}\right|}{V_o} = 0.35 \times 10^{-5} \qquad\qquad \text{(see Equation A-7)}$$

Since these satisfy all three requirements, the f and g series computations programmed as part of the control computer simulation contain 10 terms.

Computation of $\underline{V}$. —The RFP defines $\underline{V}$ as the slant range vector rate of change with respect to the earth; i.e.,

$$\underline{\dot{s}}_E = \underline{V} \qquad\qquad\qquad \text{(A-9)}$$

therefore, from

$$\underline{s} = \underline{\rho} - \underline{R}$$

where

$\underline{\rho}$ = Position vector of the vehicle

$\underline{R}$ = Position vector of the target

it follows that

$$\underline{V} = \underline{\dot{s}}_E = \underline{\dot{\rho}}_E - \underline{\dot{R}}_E \qquad\qquad \text{(A-10)}$$

But since $\underline{\dot{R}}_E$, the rate of change of the target position with respect to the earth, is iqual to zero Equation (A-10) reduces to

$$\underline{V} = \underline{\dot{s}}_E = \underline{\dot{\rho}}_E \qquad\qquad \text{(A-11)}$$

On the other hand, from

$$\frac{d\rho}{dt}\Big|_I = \dot{\rho}_E + \Omega_{EI} \times \rho$$

$\dot{\rho}_E$ is equal to

$$\dot{\rho}_E = \frac{d\rho}{dt}\Big|_I - \Omega_{EI} \times \rho \tag{A-12}$$

Finally substitution of Equation (A-12) into (A-11) yields

$$V = \dot{s}_E = \frac{d\rho}{dt}\Big|_I - \Omega_{EI} \times \rho \tag{A-13}$$

where

$\frac{d\rho}{dt}\Big|_I$ = Vehicle velocity vector with respect to inertial space supplied by f and g series updating

Ω_{EI} = Earth's rate vector with respect to inertial space

<u>Computation of $\dot{s}$.</u> —Since $\dot{s}$ is the rate of change of the magnitude of the slant range, it can be obtained by differentiating

$$|s| = (s \cdot s)^{1/2}$$

with respect to the earth. That is,

$$\frac{d}{dt}|s| = \dot{s} = \frac{1}{2}\frac{s \cdot \dot{s}_E}{(s \cdot s)^{1/2}} + \frac{1}{2}\frac{\dot{s}_E \cdot s}{(s \cdot s)^{1/2}}$$

$$\dot{s} = \dot{s}_E \cdot \left(\frac{s}{|s|}\right) \tag{A-14}$$

where $\dot{s}_E$ is given by Equation (A-13). Note that $\dot{s}$, the rate of change of the magnitude of the slant range, according to Equation (A-14) is the component of the rate of change of the slant range vector along the unit slant range vector.

Appendix B

CMG GIMBAL LOOPS

The block diagram of the CMG rigid body gimbal loop dynamics, supplied by LRC, is shown in Figure B-1. It includes the dominant cross-gimbal coupling terms ($H\cos\beta$). The numerical values of the parameters in the diagram are:

$$K_g = 276 \qquad\qquad b = 55$$

$$K_{g2} = 1.165 \qquad\qquad c = 150$$

$$K_f = 1.53 \qquad\qquad G = 67.6$$

$$K_m = 53.7 \qquad\qquad I_\alpha = 29.1$$

$$\sigma_m = 328 \qquad\qquad I_\beta = 23.5$$

$$a = 0.15 \qquad\qquad H = 1000$$

Linear transfer functions representing the gimbal loop dynamics are required for the design of the spacecraft sampled-data attitude control system. These are obtained as follows.

A linearized version of Figure B-1 is shown in Figure B-1a which is then reduced to that of Figure B-1b where:

$$d_3 = \sigma_m + a + c$$

$$d_2 = K_m K_f G^2 C_\alpha + \sigma_m (a+c) + ac$$

$$d_1 = \sigma_m ac + K_m G^2 C_\alpha [K_f(a+c) + K_g K_{g2}]$$

$$d_0 = K_m G^2 C_\alpha (K_f ac + K_g K_{g2} b)$$

$$d_3' = \sigma_m + a + c$$

$$d_2' = K_m K_f G^2 C_\beta + \sigma_m (a+c) + ac$$

$$d_1' = \sigma_m ac + K_m G^2 C_\beta [K_f(a+c) + K_g K_{g2}]$$

$$d_0' = K_m G^2 C_\beta (K_f ac + K_g K_{g2} b)$$
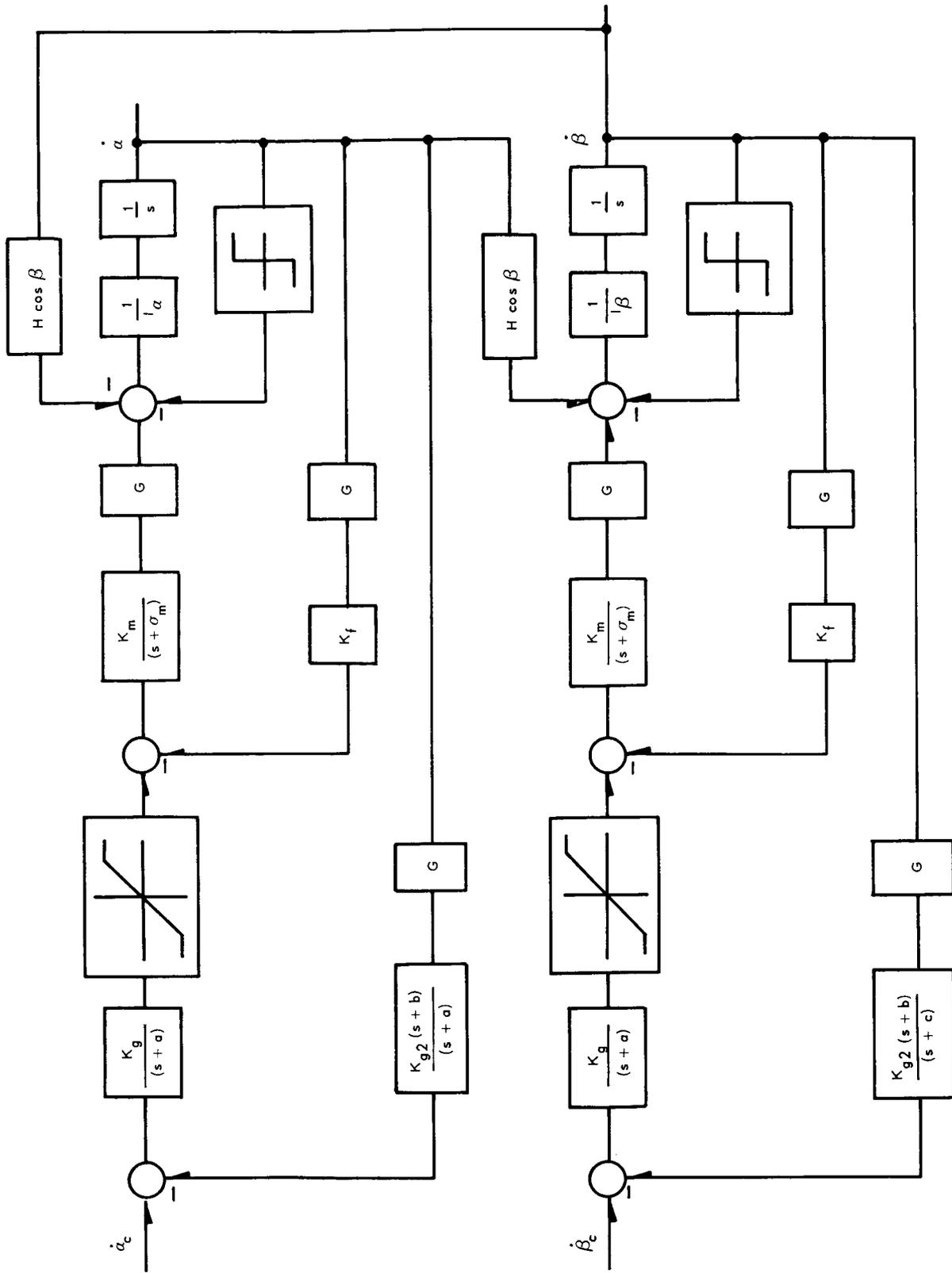
Figure B-1. CMG Gimbal Loops
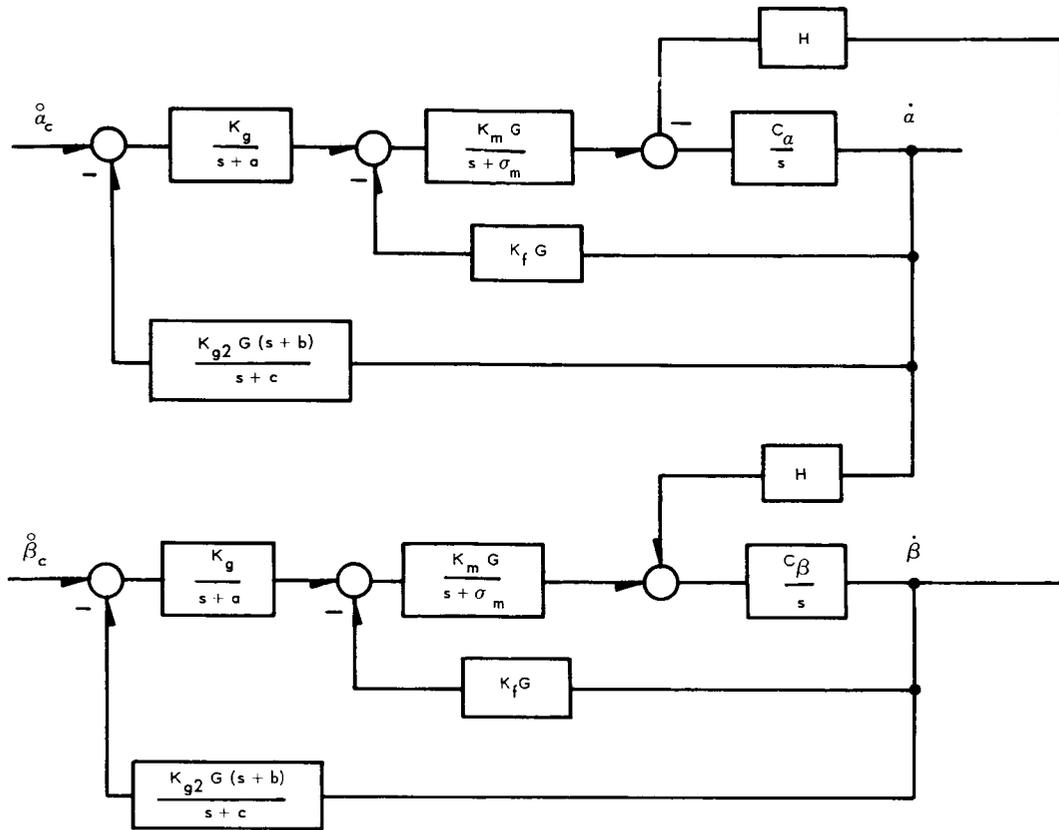
Figure B-1a
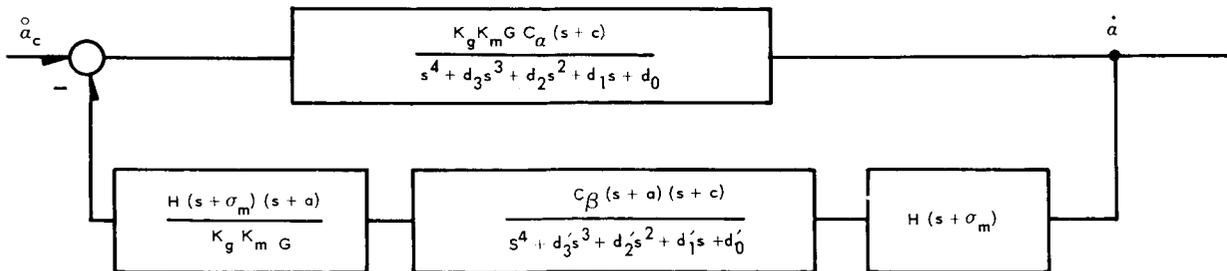
Figure B-1b

From Figure B-1b the closed loop transfer function for the outer gimbal loop is:

$$\frac{\dot{\alpha}}{\dot{\alpha}_c} = \frac{K_g K_m G C_\alpha (s+c)(s^4 + d_3' s^3 + d_2' s^2 + d_1' s + d_0')}{(s^4 + d_3 s^3 + d_2 s^2 + d_1 s + d_0)(s^4 + d_3' s^3 + d_2' s^2 + d_1' s + d_0') + H^2 C_\alpha C_\beta (s + \sigma_m)^2 (s+a)^2 (s+c)^2}$$

and similarly for the inner gimbal loop:

$$\frac{\dot{\beta}}{\dot{\beta}_c} = \frac{K_g K_m G C_\beta (s+c)(s^4 + d_3 s^3 + d_2 s^2 + d_1 s + d_0)}{(s^4 + d_3 s^3 + d_2 s^2 + d_1 s + d_0)(s^4 + d_3' s^3 + d_2' s^2 + d_1' s + d_0') + H^2 C_\alpha C_\beta (s + \sigma_m)^2 (s+a)^2 (s+c)^2}$$

The approximate roots of the fourth-order polynomials are $(s + 52.8)$ $(s + 327)$ $(s^2 + 98s + 10720)$ for the polynomial with the primed coefficients and $(s + 62.3)$ $(s + 327)$ $(s^2 + 88.4s + 7324)$ for the polynomial with the unprimed coefficients. The closed loop poles, which are the same for both gimbal loops, are obtained from the root locus plot shown in Figure B-2. The resulting transfer functions for the gimbal control loops are:

$$\frac{\dot{\alpha}}{\dot{\alpha}_c} = \frac{K_\alpha (s + 150)(s + 52.8)(s^2 + 98s + 10720)}{(s^2 + 80s + 14820)(s^2 + 74s + 2458)(s^2 + 148s + 7157)(s + 327.5)}$$

$$\frac{\dot{\beta}}{\dot{\beta}_c} = \frac{K_\beta (s + 150)(s + 62.3)(s^2 + 88.4s + 7324)}{(s^2 + 80s + 14820)(s^2 + 74s + 2458)(s^2 + 148s + 7157)(s + 327.5)}$$

Because the poles were obtained from the root locus in an approximate fashion, the values of K_α and K_β are computed to give the correct steady-state response. The numerical values are 34,760 and 43,110 for K_α and K_β respectively.
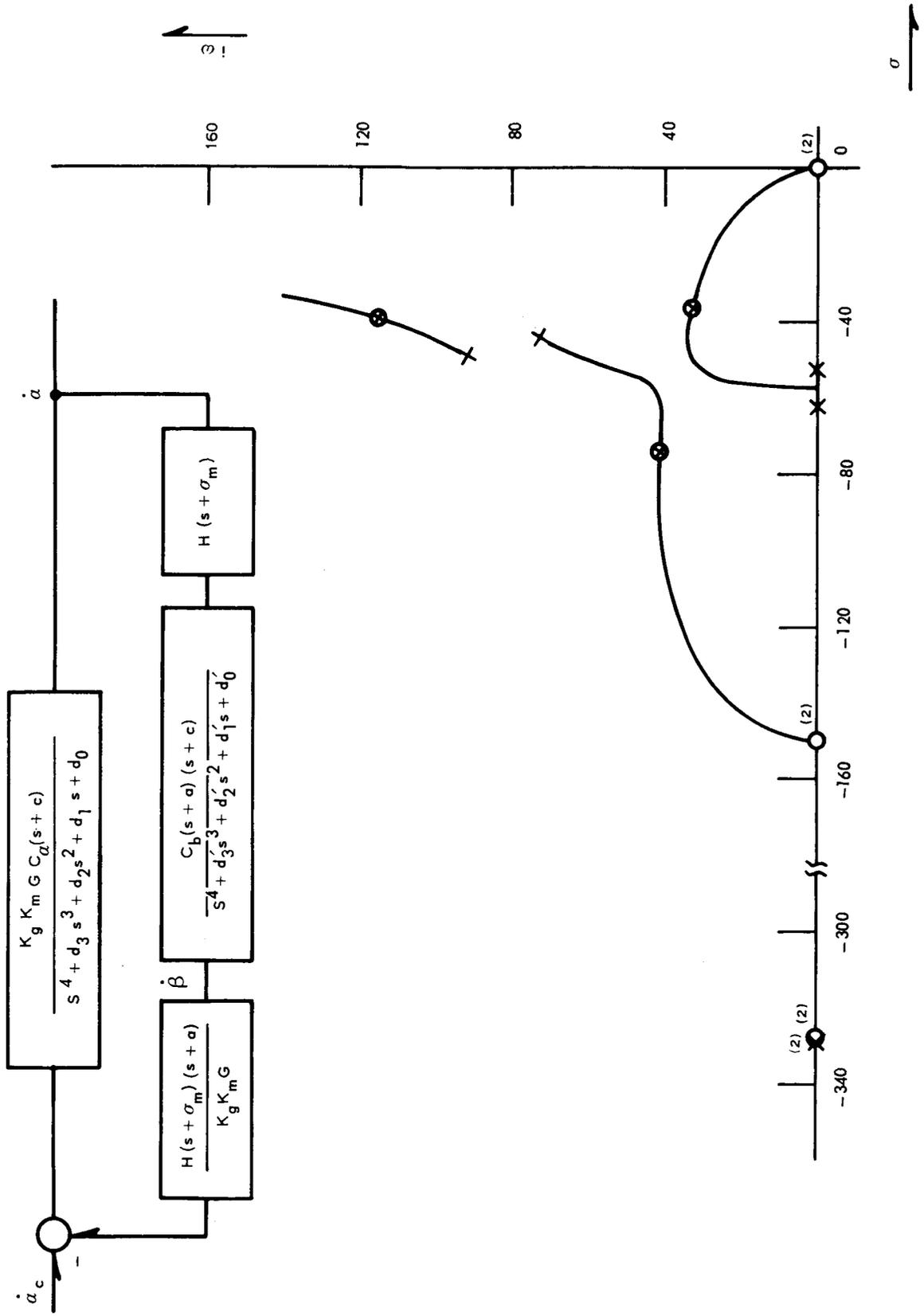
Figure B-2. CMG Gimbal Loop Root Locus

Appendix C

EQUATIONS OF MOTION

Introduction

To write the equations of rotational motion of a vehicle with three CMG's plus a single moving mass within the vehicle it is necessary to use the generalized angular momentum equation*

$$\underline{M} = \frac{d\underline{H}}{dt} + \underline{S} \times \underline{a} \tag{C-1}$$

where

$\underline{M}$ = External torque about the C.M. of the main vehicle (i.e., without the moving mass)

$\underline{H}$ = Angular momentum of the entire system about the C.M. of the main vehicle

$\underline{S} = \displaystyle\sum_{i=1}^{n} m_i \underline{q}_i$ = First moment of mass of the total system about the C.M. of the main vehicle

$\phantom{\underline{S}} = m_M \underline{q}_M$ for the case of a single moving mass m_M which at time t is at the position $\underline{q}_M$ with respect to the C.M. of the main vehicle.

$\underline{a}$ = Inertial acceleration of the C.M. of the main vehicle.

Note that for the case of a single moving mass m_M the angular momentum vector $\underline{H}$ in Equation (C-1) is given by

$$\underline{H} = \underline{H}_{VEHICLE} + \underline{H}_{GYROS} + m_M \underline{q}_M \times \frac{d\underline{q}_M}{dt}$$

from which

$$\frac{d\underline{H}}{dt} = \frac{d}{dt}(\underline{H}_V + \underline{H}_G) + m_M \underline{q}_M \times \frac{d^2 \underline{q}_M}{dt^2} \tag{C-2}$$

---

*Grubin, C., "Dynamics of a Vehicle Containing Moving Parts", Transactions of the ASME, September 1962.

It can be shown that, the inertial acceleration of the C.M. of the main vehicle of mass m_S can be expressed as

$$\underline{a} = \frac{\underline{F}}{m_M + M_S} - \frac{m_M}{m_M + m_S} \frac{d^2}{dt^2} \underline{q}_M$$

where if $\underline{F}$, the resultant of external forces, is neglected as suggested in the RFP

$$\underline{a} = - \frac{m_M}{m_M + m_S} \frac{d^2}{dt^2} \underline{q}_M \qquad (C-3)$$

Substitution of Equations (C-2) and (C-3) into (C-1) yields

$$\underline{M} = \frac{d}{dt} (\underline{H}_V + \underline{H}_G) + Q\underline{q}_M \times \left(\frac{d^2}{dt^2} \underline{q}_M \right) \qquad (C-4)$$

where

$\underline{H}_V$ = Angular momentum of the main vehicle plus the contribution to the angular momentum of each CMG taken as a point mass about the C.M. of the main vehicle.

$\underline{H}_G$ = Total angular momentum of the three CMG's, each taken about its own C.M.

$Q = \dfrac{m_M m_S}{m_M + m_S}$

$\underline{q}$ = $x\underline{I} + y\underline{J} + z\underline{K}$ = The position at time t of the moving mass m_M with respect to the C.M. of the main vehicle.

Equation (C-4) is used to write the equations of rotational motion of the system studied. Note that, Equation (C-4) is an approximation, in the sense that if $\underline{F}$ is not neglected the left hand side of Equation (C-4) would be composed of

$$\left(\underline{M} + \frac{m_M}{m_M + m_S} \underline{F} \times \underline{q}_M \right)$$

which is the total external torque about the instantaneous C.M. of the entire system.

Since the total system possesses nine degrees of freedom (three for the vehicle, two for each CMG, and none for the moving mass because its motion is assumed specified as a function of time), nine scalar simultaneous differential equations of motion are needed to describe its dynamics.

The first three of these equations are obtained by the use of Equation (C-4) where the inertia tensor of the vehicle about the C.M. of the vehicle without the moving mass is

$$\underline{\underline{I}}_V (0) = \begin{pmatrix} I_{xxs} & - I_{xys} & - I_{xzs} \\ - I_{xys} & I_{yys} & - I_{yzs} \\ - I_{xzs} & - I_{yzs} & I_{zzs} \end{pmatrix} \tag{C-5}$$

When the contribution of the moving mass is accounted for in the manner indicated by Equation (C-4), the following inertia terms (and their derivatives) of the entire system about the instantaneous C.M. of the entire system appear in the equations of motion:

$$I_{xx} = I_{xxs} + Q \, (y^2 + z^2)$$

$$I_{yy} = I_{yys} + Q \, (x^2 + z^2)$$

$$I_{zz} = I_{zzs} + Q \, (x^2 + y^2)$$

$$\tag{C-6}$$

$$I_{xy} = I_{xys} + Q \, (xy)$$

$$I_{xz} = I_{xzs} + Q \, (x\,z)$$

$$I_{yz} = I_{yzs} + Q \, (y\,z)$$

The CMG inertia tensors about their respective C.M. (point 0_i) are given as follows:

$$\underline{\underline{I}}_{a1}(^0_1) = \begin{pmatrix} A_a & 0 & 0 \\ 0 & B_a & 0 \\ 0 & 0 & C_a \end{pmatrix}; \quad \underline{\underline{I}}_{b1}(^0_1) = \begin{pmatrix} A_b & 0 & 0 \\ 0 & B_b & 0 \\ 0 & 0 & C_b \end{pmatrix}; \quad \underline{\underline{I}}_{g1}(^0_1) = \begin{pmatrix} A_g & 0 & 0 \\ 0 & B_g & 0 \\ 0 & 0 & B_g \end{pmatrix}$$

$$\underline{\underline{I}}_{a2}(^0_2) = \begin{pmatrix} C_a & 0 & 0 \\ 0 & A_a & 0 \\ 0 & 0 & B_a \end{pmatrix}; \quad \underline{\underline{I}}_{b2}(^0_2) = \begin{pmatrix} C_b & 0 & 0 \\ 0 & A_b & 0 \\ 0 & 0 & B_b \end{pmatrix}; \quad \underline{\underline{I}}_{g2}(^0_2) = \begin{pmatrix} B_g & 0 & 0 \\ 0 & A_g & 0 \\ 0 & 0 & B_g \end{pmatrix} \text{(C-7)}$$

$$\underline{\underline{I}}_{a3}(^0_3) = \begin{pmatrix} B_a & 0 & 0 \\ 0 & C_a & 0 \\ 0 & 0 & A_a \end{pmatrix}; \quad \underline{\underline{I}}_{b3}(^0_3) = \begin{pmatrix} B_b & 0 & 0 \\ 0 & C_b & 0 \\ 0 & 0 & A_b \end{pmatrix}; \quad \underline{\underline{I}}_{g3}(^0_3) = \begin{pmatrix} B_g & 0 & 0 \\ 0 & B_g & 0 \\ 0 & 0 & A_g \end{pmatrix}$$

where

 subscript a refers to the outer gimbal

 subscript b refers to the inner gimbal

 subscript g refers to the gyro rotor.

Note that in Equation (C-7) the inner gimbal inertias pertain to the inner gimbal ring only, while the outer gimbal inertias include in addition to the outer gimbal ring inertia also the inertias of the inner gimbal torquer and that of the inner gimbal sensor.

 CMG deployment with respect to the vehicle axes x, y, z and the CMG coordinate frames, with respect to which the CMG inertia tensors are defined, are shown in Figure (C-1).

 To see how the torquer rotor inertias about the axis of rotation and the gear ratios come into the picture consider the total angular momentum of the CMG #1 assembly (rotor, inner gimbal, outer gimbal) given as follows:

$$\underline{(H)}_{CMG\#1} = \underline{\underline{I}}_{a1}(^0_1) \cdot \underline{\omega}'_1 + \underline{\underline{I}}_{b1}(^0_1) \cdot (\underline{\omega} + \dot{\alpha}_1 \underline{J} + \dot{\beta}_1 \underline{k}_1)$$

$$+ \underline{\underline{I}}_{g1}(^0_1) \cdot (\underline{\omega} + \dot{\alpha}_1 \underline{J} + \dot{\beta}_1 \underline{k}_1 + \Omega_1 \underline{i}_1) + (\omega_y + G_\alpha \dot{\alpha}_1) J_{m\alpha} \underline{J} \quad \text{(C-8)}$$

$$+ (\omega_x \sin\alpha_1 + \omega_z \cos\alpha_1 + G_\beta \dot{\beta}_1) J_{m\beta} \underline{k}_1$$
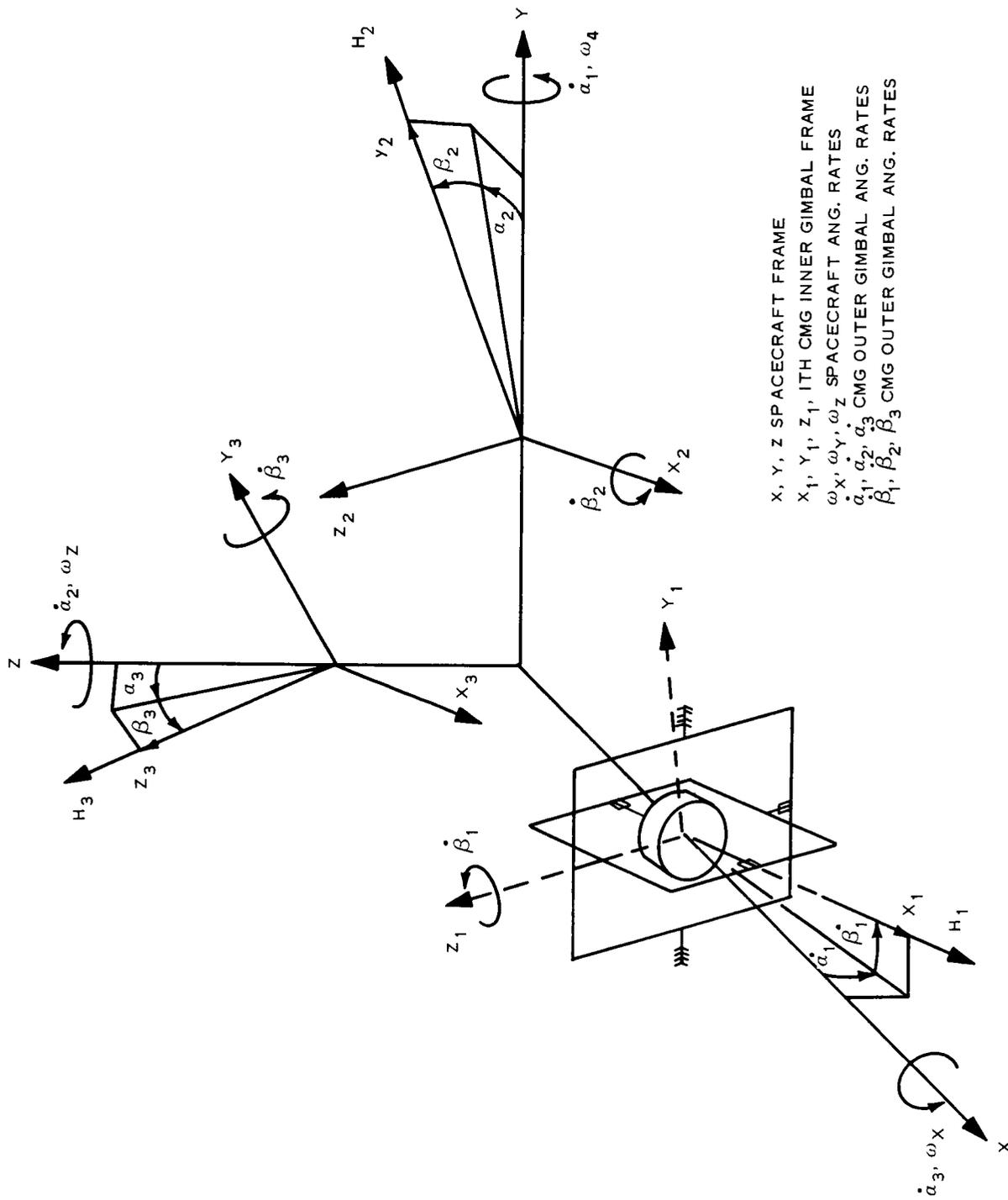
x, y, z   SPACECRAFT FRAME

$x_1, y_1, z_1,$ ITH CMG INNER GIMBAL FRAME

$\omega_x, \omega_y, \omega_z$ SPACECRAFT ANG. RATES

$\dot{\alpha}_1, \dot{\alpha}_2, \dot{\alpha}_3$ CMG OUTER GIMBAL ANG. RATES

$\dot{\beta}_1, \dot{\beta}_2, \dot{\beta}_3$ CMG OUTER GIMBAL ANG. RATES

Figure C-1.   Reference Frames

where

$$(\omega_y + G_\alpha \dot{\alpha}_1) \, J_{m\alpha} = \text{Angular momentum of the outer gimbal torquer rotor about the outer gimbal axis}$$

$$(\omega_x \sin\alpha_1 + \omega_z \cos\alpha_1 + G_\beta \dot{\beta}_1) \, J_{m\beta} = \text{Angular momentum of the inner gimbal torquer rotor about the inner gimbal axis}$$

for

$J_{m\alpha}$ = Outer gimbal torquer rotor inertia about the outer gimbal axis

$J_{m\beta}$ = Inner gimbal torquer rotor inertia about the inner gimbal axis

G_α = Gear ratio for outer gimbal drive

G_β = Gear ratio for inner gimbal drive

Of the nine scalar equations of motion, the last six are obtained as follows:

For an outer gimbal axis equation of motion (for instance, for CMG #1), the following relation is used:

$$\frac{d}{dt} (\underline{H}_{\alpha 1}) \cdot \underline{J} = M_{T\alpha 1} \tag{C-9}$$

where

$M_{T\alpha 1}$ = Total torque interaction between the vehicle and CMG #1 assembly about α_1 - axis.

$\underline{H}_{\alpha 1}$ = Angular momentum of the entire CMG #1 assembly.

The total torque interaction between the vehicle and a torqued load is partly through the base of the torquer and partly throught the base of the gear box. It is possible to show that the total torque interaction can be expressed as

$$M_{T\alpha 1} = M_{\alpha 1} + (1 - G_\alpha) (G_\alpha \ddot{\alpha}_1 + \dot{\omega}_y) J_{m\alpha} \tag{C-10}$$

where

$M_{\alpha 1}$ = G_α times the torque produced by the outer gimbal torquer.

Equation (C-10) is substituted into Equation (C-9) so that the resulting equation of motion is a function of $M_{\alpha 1}$, which is the torque that appears in the outer gimbal servo loop at the output of the gear box.

For an inner gimbal axis equation of motion (for instance, for CMG #1) the following equation is used:

$$\frac{d}{dt}(\underline{H}_{\beta 1}) = M_{T\beta 1} \qquad (C-11)$$

where

$M_{T\beta 1}$ = Total torque interaction between inner and outer gimbals of CMG #1 about the β_1-axis.

$\underline{H}_{\beta 1}$ = Angular momentum of CMG #1 inner gimbal assembly.

In this case, it can be shown that

$$M_{T\beta 1} = M_{\beta 1} + (1 - G_\beta)(G_\beta \ddot{\beta}_1 + \dot{\omega}_x \sin\alpha_1 + \dot{\omega}_z \cos\alpha_1 + \omega_x \dot{\alpha}_1 \cos\alpha_1 - \omega_z \dot{\alpha}_1 \sin\alpha_1) J_{m\beta}$$

$$(C-12)$$

where

$M_{\beta 1}$ = G times the torque produced by the inner gimbal torque.

Equation (C-12) is substituted into Equation (C-11) to make it a function of $M_{\beta 1}$.

List of Equations of Motion

The nine scalar equations of motion can be arranged in the form

$$
\begin{pmatrix}
a_{11} & a_{12} & a_{13} & a_{14} & a_{15} & a_{16} & a_{17} & 0 & a_{19} \\
a_{12} & a_{22} & a_{23} & 0 & a_{25} & a_{26} & a_{27} & a_{28} & a_{29} \\
a_{13} & a_{23} & a_{33} & a_{34} & a_{35} & 0 & a_{37} & a_{38} & a_{39} \\
a_{14} & 0 & a_{34} & a_{44} & 0 & 0 & 0 & 0 & 0 \\
a_{15} & a_{25} & a_{35} & 0 & a_{55} & 0 & 0 & 0 & 0 \\
a_{16} & a_{26} & 0 & 0 & 0 & a_{66} & 0 & 0 & 0 \\
a_{17} & a_{27} & a_{37} & 0 & 0 & 0 & a_{77} & 0 & 0 \\
0 & a_{28} & a_{38} & 0 & 0 & 0 & 0 & a_{88} & 0 \\
a_{19} & a_{29} & a_{39} & 0 & 0 & 0 & 0 & 0 & a_{99}
\end{pmatrix}
\begin{pmatrix}
\dot{\omega}_x \\
\dot{\omega}_y \\
\dot{\omega}_z \\
\ddot{\beta}_1 \\
\ddot{\alpha}_1 \\
\ddot{\beta}_2 \\
\ddot{\alpha}_2 \\
\ddot{\beta}_3 \\
\ddot{\alpha}_3
\end{pmatrix}
=
\begin{pmatrix}
b_1 \\
b_2 \\
b_3 \\
b_4 \\
b_5 \\
b_6 \\
b_7 \\
b_8 \\
b_9
\end{pmatrix}
\qquad (C-13)
$$

where, for example, the first scalar equation is

$$a_{11}\dot{\omega}_x + a_{12}\dot{\omega}_y + a_{13}\dot{\omega}_z + a_{14}\ddot{\beta}_1 + a_{15}\ddot{\alpha}_1 + a_{16}\ddot{\beta}_2 + a_{17}\ddot{\alpha}_2 + a_{19}\ddot{\alpha}_3 = b_1$$

Note that the coefficient matrix in Equation (C-13) can be looked upon as a generalized inertia tensor, and as the choice of the subscripts of its elements indicate it is a symmetric matrix.

The content of the elements a_{ij} and b_i in Equation (C-13) are given by the equation sets (C-15) and (C-16), where to achieve some measure of conciseness the identifications listed in the Equation set (C-14) are used.

Components of outer gimbal ring inertial rate along the outer gimbal axes of the ith CMG

Components of inner gimbal ring inertial rate along the inner gimbal axes of the ith CMG

$$\omega'_{x1} = \omega_x \cos\alpha_1 - \omega_z \sin\alpha_1 \qquad \omega_{x1} = \omega'_{x1}\cos\beta_1 + \omega'_{y1}\sin\beta_1$$

$$\omega'_{y1} = \omega_y + \dot{\alpha}_1 \qquad \omega_{y1} = -\omega'_{x1}\sin\beta_1 + \omega'_{y1}\cos\beta_1$$

$$\omega'_{z1} = \omega_x \sin\alpha_1 + \omega_z \cos\alpha_1 \qquad \omega_{z1} = \omega'_{z1} + \dot{\beta}_1$$

$$\omega'_{x2} = \omega_x \cos\alpha_2 + \omega_y \sin\alpha_2 \qquad \omega_{x2} = \omega'_{x2} + \dot{\beta}_2$$

$$\omega'_{y2} = -\omega_x \sin\alpha_2 + \omega_y \cos\alpha_2 \qquad \omega_{y2} = \omega'_{y2}\cos\beta_2 + \omega'_{z2}\sin\beta_2$$

$$\omega'_{z2} = \omega_z + \dot{\alpha}_2 \qquad \omega_{z2} = -\omega'_{y2}\sin\beta_2 + \omega'_{z2}\cos\beta_2$$

$$\omega'_{x3} = \omega_x + \dot{\alpha}_3 \qquad \omega_{x3} = \omega'_{x3}\cos\beta_3 - \omega'_{z3}\sin\beta_3$$

$$\omega'_{y3} = \omega_y \cos\alpha_3 + \omega_z \sin\alpha_3 \qquad \omega_{y3} = \omega'_{y3} + \dot{\beta}_3$$

$$\omega'_{z3} = -\omega_y \sin\alpha_3 + \omega_z \cos\alpha_3 \qquad \omega_{z3} = \omega'_{x3}\sin\beta_3 + \omega'_{z3}\cos\beta_3$$

(C-14)

where

$$\omega_x,\ \omega_y,\ \omega_z = \text{Rates about vehicle axes}$$

$$a_{11} = I_{xx} + [A_a + (A_b + A_g) \cos^2\beta_1 + (B_b + B_g) \sin^2\beta_1] \cos^2\alpha_1$$

$$+ (C_a + C_b + B_g + J_{m\beta}) (\sin^2\alpha_1 + \cos^2\alpha_2)$$

$$+ [A_a + (A_b + A_g) \cos^2\beta_2 + (B_b + B_g) \sin^2\beta_2] \sin^2\alpha_2$$

$$+ J_{m\alpha} + B_a + (A_b + A_g) \sin^2\beta_3 + (B_b + B_g) \cos^2\beta_3$$

$$a_{12} = -I_{xy} + (A_b + A_g - B_b - B_g) \cos\alpha_1 \sin\beta_1 \cos\beta_1$$

$$+ [C_a - A_a + C_b + B_g + J_{m\beta} - (A_b + A_g) \cos^2\beta_2 - (B_b + B_g) \sin^2\beta_2] \sin\alpha_2 \cos\alpha_2$$

$$- (A_b + A_g - B_b - B_g) \sin\alpha_3 \sin\beta_3 \cos\beta_3$$

$$a_{13} = -I_{xz} + [C_a - A_a + C_b + B_g + J_{m\beta} - (A_b + A_g) \cos^2\beta_1 - (B_b + B_g) \sin^2\beta_1] \sin\alpha_1 \cos\alpha_1$$

$$- (A_b + A_g - B_b - B_g) (\sin\alpha_2 \sin\beta_2 \cos\beta_2 - \cos\alpha_3 \sin\beta_3 \cos\beta_3)$$

$$a_{14} = (C_b + B_g + J_{m\beta}G_\beta) \sin\alpha_1$$

$$a_{15} = (A_b + A_g - B_b - B_g) \cos\alpha_1 \sin\beta_1 \cos\beta_1$$

$$a_{16} = (C_b + B_g + J_{m\beta}G_\beta) \cos\alpha_2$$

$$a_{17} = -(A_b + A_g - B_b - B_g) \sin\alpha_2 \sin\beta_2 \cos\beta_2$$

$$a_{19} = B_a + J_{m\alpha}G_\alpha + (A_b + A_g) \sin^2\beta_3 + (B_b + B_g) \cos^2\beta_3$$

$$a_{22} = I_{yy} + B_a + J_{m\alpha} + (A_b + A_g) \sin^2\beta_1 + (B_b + B_g) \cos^2\beta_1$$

$$+ (C_a + C_b + B_g + J_{m\beta}) (\sin^2\alpha_2 + \cos^2\alpha_3)$$

$$+ [A_a + (A_b + A_g) \cos^2\beta_2 + (B_b + B_g) \sin^2\beta_2] \cos^2\alpha_2$$

$$+ [A_a + (A_b + A_g) \cos^2\beta_3 + (B_b + B_g) \sin^2\beta_3] \sin^2\alpha_3$$

$$a_{23} = -I_{yz} - (A_b + A_g - B_b - B_g)(\sin\alpha_1 \sin\beta_1 \cos\beta_1 - \cos\alpha_2 \sin\beta_2 \cos\beta_2)$$

$$+ [C_a - A_a + C_b + B_g + J_{m\beta} - (A_b + A_g)\cos^2\beta_3 - (B_b + B_g)\sin^2\beta_3]\sin\alpha_3 \cos\alpha_3$$

$$a_{25} = B_a + J_{m\alpha}G_\alpha + (A_b + A_g)\sin^2\beta_1 + (B_b + B_g)\cos^2\beta_1$$

$$a_{26} = (C_b + B_g + J_{m\beta}G_\beta)\sin\alpha_2$$

$$a_{27} = (A_b + A_g - B_b - B_g)\cos\alpha_2 \sin\beta_2 \cos\beta_2$$

$$a_{28} = (C_b + B_g + J_{m\beta}G_\beta)\cos\alpha_3$$

$$a_{29} = -(A_b + A_g - B_b - B_g)\sin\alpha_3 \sin\beta_3 \cos\beta_3$$

$$a_{33} = I_{zz} + B_a + J_{m\alpha} + (C_a + C_b + B_g + J_{m\beta})(\cos^2\alpha_1 + \sin^2\alpha_3)$$

$$+ (A_b + A_g)\sin^2\beta_2 + (B_b + B_g)\cos^2\beta_2$$

$$+ [A_a + (A_b + A_g)\cos^2\beta_1 + (B_b + B_g)\sin^2\beta_1]\sin^2\alpha_1$$

$$+ [A_a + (A_b + A_g)\cos^2\beta_3 + (B_b + B_g)\sin^2\beta_3]\cos^2\alpha_3$$

$$a_{34} = (C_b + B_g + J_{m\beta}G_\beta)\cos\alpha_1$$

$$a_{35} = -(A_b + A_g - B_b - B_g)\sin\alpha_1 \sin\beta_1 \cos\beta_1$$

$$a_{37} = B_a + J_{m\alpha}G_\alpha + (A_b + A_g)\sin^2\beta_2 + (B_b + B_g)\cos^2\beta_2$$

$$a_{38} = (C_b + B_g + J_{m\beta}G_\beta)\sin\alpha_3$$

$$a_{39} = (A_b + A_g - B_b - B_g)\cos\alpha_3 \sin\beta_3 \cos\beta_3$$

$$a_{44} = a_{66} = a_{88} = C_b + B_g + J_{m\beta}G_\beta^2$$

$$a_{55} = B_a + J_{m\alpha}G_\alpha^2 + (A_b + A_g)\sin^2\beta_1 + (B_b + B_g)\cos^2\beta_1$$

Equations (C-15) - Concluded

$$a_{77} = B_a + J_{m\alpha}G_\alpha^2 + (A_b + A_g)\sin^2\beta_2 + (B_b + B_g)\cos^2\beta_2$$

$$a_{99} = B_a + J_{m\alpha}G_\alpha^2 + (A_b + A_g)\sin^2\beta_3 + (B_b + B_g)\cos^2\beta_3$$

$$M_x - \dot{H}_1 \cos\alpha_1 \cos\beta_1 + \dot{H}_2 \sin\alpha_2 \cos\beta_2 - \dot{H}_3 \sin\beta_3$$

$$-\left\{ -(\omega_y + G\dot\alpha_1) \omega'_{z1} J_{m\alpha} \cos\alpha_1 + (\omega'_{z1} + G\dot\beta_1) \omega'_{y1} J_{m\beta} \cos\alpha_1 + [\dot\alpha_1 J_{m\beta} + (\omega_y + G\dot\alpha_1) J_{m\alpha}] \omega'_{x1} \sin\alpha_1 + (\omega_z + G\dot\alpha_2) \omega'_{x2} J_{m\alpha} \sin\alpha_2 \right.$$

$$\left. + [\dot\alpha_2 J_{m\beta} + (\omega_z + G\dot\alpha_2) J_{m\alpha}] \omega'_{y2} \cos\alpha_2 - (\omega'_{x2} + G\dot\beta_2) \omega'_{z2} J_{m\beta} \sin\alpha_2 - (\omega'_{y3} + G\dot\beta_3) \omega'_{z3} J_{m\beta} \right\}$$

$$+ \dot{I}_{xy}\omega_y - \dot{I}_{xx}\omega_x + \dot{I}_{xz}\omega_z - Q[\omega_y (\dot{x}\dot{y} - \dot{y}\dot{x}) + \omega_z (\dot{x}\dot{z} - \dot{z}\dot{x}) + (\dot{y}\ddot{z} - \ddot{y}\dot{z})]$$

$$- [-A_a \cos\alpha_1 \omega'_{z1} \dot\alpha_1 + C_a \sin\alpha_1 \omega'_{x1} \dot\alpha_1 - (A_b + A_g) \cos\alpha_1 \cos^2\beta_1 \omega'_{z1} \dot\alpha_1 + (A_b + A_g) \cos\alpha_1 \cos\beta_1 \omega_{y1} \dot\beta_1$$

$$- (B_b + B_g) \cos\alpha_1 \sin^2\beta_1 \omega'_{z1} \dot\alpha_1 + \cos\alpha_1 \sin\beta_1 (B_b + B_g) \omega_{x1} \dot\beta_1 + (C_b + B_g) \sin\alpha_1 \omega'_{x1} \dot\alpha_1$$

$$+ C_a \cos\alpha_2 \omega'_{y2} \dot\alpha_2 + A_a \sin\alpha_2 \omega'_{x2} \dot\alpha_2 + (C_b + B_g) \cos\alpha_2 \omega'_{y2} \dot\alpha_2 + (A_b + A_g) \sin\alpha_2 \cos^2\beta_2 \omega'_{x2} \dot\alpha_2$$

$$- (A_b + A_g) \sin\alpha_2 \cos\beta_2 \omega_{z2} \dot\beta_2 + (B_b + B_g) \sin\alpha_2 \sin^2\beta_2 \omega'_{x2} \dot\alpha_2 - (B_b + B_g) \sin\alpha_2 \sin\beta_2 \omega_{y2} \dot\beta_2$$

$$+ (B_b + B_g) \sin\beta_3 \cos\beta_3 \omega'_{y3} \dot\alpha_3 - (B_b + B_g) \cos\beta_3 \omega_{z3} \dot\beta_3 - (A_b + A_g) \sin\beta_3 \cos\beta_3 \omega'_{y3} \dot\alpha_3$$

$$+ (A_b + A_g) \sin\beta_3 \omega_{x3} \dot\beta_3]$$

$$-\left\{ (\omega_z^2 - \omega_y^2) I_{yz} + \omega_y \omega_z (I_{zz} - I_{yy}) + \omega_x (\omega_z I_{xy} - \omega_y I_{xz}) + \omega'_{y1} \omega'_{z1} (C_a - B_a) \cos\alpha_1 + \omega'_{x1} \omega'_{y1} (B_a - A_a) \sin\alpha_1 \right.$$

$$+ \omega_{y1} \omega_{z1} (C_b - B_b) \cos\alpha_1 \cos\beta_1 - [\omega_{x1} \omega_{z1} (A_b + A_g - C_b - B_g) + \Omega_1 \omega_{z1} A_g] \cos\alpha_1 \sin\beta_1$$

$$+ [\omega_{x1} \omega_{y1} (B_b + B_g - A_b - A_g) - \Omega_1 \omega_{y1} A_g] \sin\alpha_1 + \omega'_{y2} \omega'_{z2} (B_a - A_a) \cos\alpha_2$$

$$- \omega'_{x2} \omega'_{z2} (C_a - B_a) \sin\alpha_2 + [\omega_{y2} \omega_{z2} (B_b + B_g - A_b - A_g) - \Omega_2 \omega_{z2} A_g] \cos\alpha_2$$

$$- \omega_{x2} \omega_{z2} (C_b - B_b) \sin\alpha_2 \cos\beta_2 + [\omega_{x2} \omega_{y2} (A_b + A_g - C_b - B_g) + \Omega_2 \omega_{x2} A_g] \sin\alpha_2 \sin\beta_2 + \omega'_{y3} \omega'_{z3} (A_a - C_a)$$

$$\left. + [\omega_{y3} \omega_{z3} (A_b + A_g - C_b - B_g) + \Omega_3 \omega_{y3} A_g] \cos\beta_3 + \omega_{x3} \omega_{y3} (C_b - B_b) \sin\beta_3 \right\}$$

$$b_1 =$$

C-12

$$M_y - \dot{H}_1 \sin\beta_1 - \dot{H}_2 \cos\alpha_2 \cos\beta_2 - \dot{H}_3 \sin\alpha_3 \cos\beta_3$$

$$-\left\{ (\omega'_{z1} + G_\beta \dot{\beta}_1) \omega'_{x1} J_{m\beta} + [\dot{\alpha}_2 J_{m\beta} + (\omega_z + G_\alpha \dot{\alpha}_2) J_{m\alpha}] \omega'_{y2} \sin\alpha_2 + (\omega'_{x2} + G_\beta \dot{\beta}_2) \omega_{z2} J_{m\beta} \cos\alpha_2 - (\omega_z + G_\alpha \dot{\alpha}_2) \omega'_{x2} J_{m\alpha} \cos\alpha_2 \right.$$

$$\left. + (\omega_x + G_\alpha \dot{\alpha}_3) \omega'_{y3} J_{m\alpha} \sin\alpha_3 - (\omega'_{y3} + G_\beta \dot{\beta}_3) \omega'_{x3} J_{m\beta} \sin\alpha_3 + [\dot{\alpha}_3 J_{m\beta} + (\omega_x + G_\alpha \dot{\alpha}_3) J_{m\alpha}] \omega'_{z3} \cos\alpha_3 \right\}$$

$$- I_{yy} \omega_y + I_{yz} \omega_z + I_{xy} \omega_x - Q[\omega_x (yx - x\ddot{y}) + \omega_z (y\dot{z} - z\dot{y}) + (z\ddot{x} - x\ddot{z})]$$

$$- \left[-(A_b + A_g) \sin\beta_1 \cos\beta_1 \omega'_{z1} \dot{\alpha}_1 + (A_b + A_g) \sin\beta_1 \omega_{y1} \dot{\beta}_1 + (B_b + B_g) \sin\beta_1 \cos\beta_1 \dot{\omega}'_{z1} \dot{\alpha}_1 \right.$$

$$- (B_b + B_g) \cos\beta_1 \dot{\omega}'_{x1} \dot{\beta}_1 + C_a \sin\alpha_2 \omega'_{y2} \dot{\alpha}_2 - A_a \cos\alpha_2 \omega'_{x2} \dot{\alpha}_2 + (C_b + B_g) \sin\alpha_2 \omega'_{y2} \dot{\alpha}_2$$

$$- (A_b + A_g) \cos\alpha_2 \cos^2\beta_2 \omega'_{x2} \dot{\alpha}_2 + (A_b + A_g) \cos\alpha_2 \cos\beta_2 \omega_{z2} \dot{\beta}_2 - (B_b + B_g) \cos\alpha_2 \sin^2\beta_2 \omega'_{x2} \dot{\alpha}_2$$

$$+ (B_b + B_g) \cos\alpha_2 \sin\beta_2 \omega_{y2} \dot{\beta}_2 + C_a \cos\alpha_3 \omega'_{z3} \dot{\alpha}_3 + A_a \sin\alpha_3 \omega'_{y3} \dot{\alpha}_3$$

$$+ (B_b + B_g) \sin\alpha_3 \sin^2\beta_3 \omega'_{y3} \dot{\alpha}_3 - (B_b + B_g) \sin\alpha_3 \sin\beta_3 \omega_{z3} \dot{\beta}_3 + (C_b + B_g) \cos\alpha_3 \omega'_{z3} \dot{\alpha}_3$$

$$+ (A_b + A_g) \sin\alpha_3 \cos^2\beta_3 \omega'_{y3} \dot{\alpha}_3 - (A_b + A_g) \sin\alpha_3 \cos\alpha_3 \omega_{x3} \dot{\beta}_3 \right]$$

$$-\left\{ \omega_x \omega_z (I_{xx} - I_{zz}) + (\omega_x^2 - \omega_z^2) I_{xz} + \omega_y (\omega_x I_{yz} - \omega_z I_{xy}) + \omega'_{x1} \omega'_{z1} (A_a - C_a) \right.$$

$$+ \omega_{y1} \omega_{z1} (C_b - B_b) \sin\beta_1 + [\omega_{x1} \omega_{z1} (A_b + A_g - C_b - B_g) + \Omega_1 \omega_{z1} A_g] \cos\beta_1$$

$$+ \omega'_{y2} \omega'_{z2} (B_a - A_a) \sin\alpha_2 + \omega'_{x2} \omega'_{z2} (C_a - B_a) \cos\alpha_2 + [\omega_{y2} \omega_{z2} (B_b + B_g - A_b - A_g) - \Omega_2 \omega_{z2} A_g] \sin\alpha_2$$

$$+ \omega_{x2} \omega_{z2} (C_b - B_b) \cos\alpha_2 \cos\beta_2 - [\omega_{x2} \omega_{y2} (A_b + A_g - C_b - B_g) + \Omega_2 \omega_{x2} A_g] \cos\alpha_2 \sin\beta_2$$

$$+ \omega'_{x3} \omega'_{z3} (B_a - A_a) \cos\alpha_3 - \omega'_{x3} \omega'_{y3} (C_a - B_a) \sin\alpha_3 + [\omega_{y3} \omega_{z3} (A_b + A_g - C_b - B_g) + \Omega_3 \omega_{y3} A_g] (\sin\alpha_3 \sin\beta_3)$$

$$\left. + [\omega_{x3} \omega_{z3} (B_b + B_g - A_b - A_g) - \Omega_3 \omega_{x3} A_g] \cos\alpha_3 - \omega_{x3} \omega_{y3} (C_b - B_b) \sin\alpha_3 \cos\beta_3 \right\}$$

$$b_2 =$$

Equations (C-16) - Continued

$$
b_3 = \left\{
\begin{aligned}
& M_z + \dot{H}_1 \sin\alpha_1 \cos\beta_1 - \dot{H}_2 \sin\beta_2 - \dot{H}_3 \cos\alpha_3 \cos\beta_3 \\[4pt]
& - \left\{ (\omega_y + G_\alpha \dot{\alpha}_1)\, \omega'_{z1}\, J_{m\alpha} \sin\alpha_1 - (\omega'_{z1} + G_\beta \dot{\beta}_1)\, \omega'_{y1}\, J_{m\beta} \sin\alpha_1 + [\dot{\alpha}_1 J_{m\beta} + (\omega_y + G_\alpha \dot{\alpha}_1) J_{m\alpha}]\, \omega'_{x1} \cos\alpha_1 - (\omega'_{x2} + G_\beta \dot{\beta}_2)\, \omega'_{y2}\, J_{m\beta} \right. \\[4pt]
& \left. - (\omega_x + G_\alpha \dot{\alpha}_3)\, \omega'_{y3} J_{m\alpha} \cos\alpha_3 + (\omega'_{y3} + G_\beta \dot{\beta}_3)\, \omega'_{x3} J_{m\beta} \cos\alpha_3 + [\alpha_3 J_{m\beta} + (\omega_x + G_\alpha \dot{\alpha}_3) J_{m\alpha}]\, \omega'_{z3} \sin\alpha_3 \right\} \\[4pt]
& + \dot{I}_{xz}\, \omega_x + \dot{I}_{yz}\, \omega_y - \dot{I}_{zz}\, \omega_z - Q\left[\omega_x (z\ddot{x} - x\ddot{z}) + \omega_y (z\ddot{y} - y\ddot{z}) + (x\ddot{y} - y\ddot{x}) \right] \\[4pt]
& - \left[A_a \sin\alpha_1\, \omega'_{z1} \dot{\alpha}_1 + C_a \cos\alpha\, \omega'_{x1} \dot{\alpha}_1 + (A_b + A_g) \sin\alpha\cos^2\beta_1\, \omega'_{z1} \dot{\alpha}_1 - (A_b + A_g) \sin\alpha \cos\beta_1\, \omega_{y1} \dot{\beta}_1 \right. \\[4pt]
& + (B_b + B_g) \sin\alpha_1 \sin^2\beta_1\, \omega'_{z1} \dot{\alpha}_1 - (B_b + B_g) \sin\beta_1 \sin\alpha_1\, \omega_{x1} \dot{\beta}_1 + (C_b + B_g) \cos\alpha_1\, \omega'_{x1} \dot{\alpha}_1 - (A_b + A_g) \sin\beta_2 \cos\beta_2\, \omega'_{x2} \dot{\alpha}_2 \\[4pt]
& + (A_b + A_g) \sin\beta_2\, \omega_{z2} \dot{\beta}_2 + (B_b + B_g) \sin\beta_2 \cos\beta_2\, \omega'_{x2} \dot{\alpha}_2 - (B_b + B_g) \cos\beta_2\, \omega_{y2} \dot{\beta}_2 + C_a \sin\alpha_3\, \omega_{z3} \dot{\alpha}_3 \\[4pt]
& - A_a \cos\alpha_3\, \omega'_{y3} \dot{\alpha}_3 - (B_b + B_g) \cos\alpha_3 \sin^2\beta_3\, \omega'_{y3} \dot{\alpha}_3 + (B_b + B_g) \cos\alpha_3 \sin\beta_3\, \omega_{z3} \dot{\beta}_3 \\[4pt]
& \left. + (C_b + B_g) \sin\alpha_3\, \omega'_{z3} \dot{\alpha}_3 - (A_b + A_g) \cos\alpha_3 \cos^2\beta_3\, \omega'_{y3} \dot{\alpha}_3 + (A_b + A_g) \cos\alpha_3 \cos\beta_3\, \omega_{x3} \dot{\beta}_3 \right] \\[4pt]
& - \left\{ \omega_x \omega_y (I_{yy} - I_{xx}) + (\omega_y^2 - \omega_x^2) I_{xy} + \omega_z (\omega_y I_{xz} - \omega_x I_{yz}) - \omega'_{y1} \omega'_{z1} (C_a - B_a) \sin\alpha_1 \right. \\[4pt]
& + \omega'_{x1} \omega'_{y1} (B_a - A_a) \cos\alpha_1 - \omega_{y1} \omega_{z1} (C_b - B_b) \sin\alpha_1 \cos\beta_1 + [\omega_{x1} \omega_{z1} (A_b + A_g - C_b - B_g) + \Omega_1 \omega_{z1} A_g] \sin\alpha_1 \sin\beta_1 \\[4pt]
& + [\omega_{x1} \omega_{y1} (B_b + B_g - A_b - A_g) - \Omega_1 \omega_{y1} A_g] \cos\alpha_1 + \omega'_{x2} \omega'_{y2} (A - C_a) + \omega_{x2} \omega_{z2} (C_b - B_b) \sin\beta_2 \\[4pt]
& + [\omega_{x1} \omega_{y1} (B_b + B_g - A_b - A_g) - \Omega_1 \omega_{y1} A_g] \cos\alpha_1 + \omega'_{x2} \omega'_{y2} (A - C_a) + \omega_{x2} \omega_{z2} (C_b - B_b) \sin\beta_2 \\[4pt]
& + [\omega_{x2} \omega_{y2} (A_b + A_g - C_b - B_g) + \Omega_2 \omega_{x2} A_g] \cos\beta_2 + \omega'_{x3} \omega'_{z3} (B_a - A_a) \sin\alpha_3 \\[4pt]
& + \omega'_{y3} \omega'_{z3} (C_a - B_a) \cos\alpha_3 - [\omega_{y3} \omega_{z3} (A_b + A_g - C_b - B_g) + \Omega_3 \omega_{y3} A_g] \cos\alpha_3 \sin\beta_3 \\[4pt]
& \left. + [\omega_{x3} \omega_{z3} (B_b + B_g - A_b - A_g) - \Omega_3 \omega_{x3} A_g] \sin\alpha_3 + \omega_{x3} \omega_{y3} (C_b - B_b) \cos\alpha_3 \cos\beta_3 \right\}
\end{aligned}
\right.
$$

$$b_4 = M_{\beta 1} - (C_b + B_g + J_{m\beta}G_\beta)\,\omega'_{x1}\dot{\alpha}_1 + \omega_{y1}\,\Omega_1\,A_g + (A_b + A_g - B_b - B_g)\,\omega_{x1}\,\omega_{y1}$$

$$b_5 = \left\{ \begin{aligned} & M_{\alpha 1} + (\omega'_{z1} + G_\beta\dot{\beta}_1)\,\omega'_{x1}J_{m\beta} - \dot{H}_1\sin\beta_1 \\[6pt] & + (A_b + A_g)\,[\sin\beta_1\cos\beta_1\,\omega'_{z1}\dot{\alpha}_1 + \sin^2\beta_1\omega'_{x1}\dot{\beta}_1 - \sin\beta_1\cos\beta_1\,(\omega_y + \dot{\alpha}_1)\,\dot{\beta}_1] \\[6pt] & + (B_b + B_g)\cos\beta_1\,[-\sin\beta_1\omega'_{z1}\dot{\alpha}_1 + \cos\beta_1\omega'_{x1}\dot{\beta}_1 + \sin\beta_1\,(\omega_y + \dot{\alpha}_1)\,\dot{\beta}_1] \\[6pt] & - \omega_{z1}\,\Omega_1 A_g\cos\beta_1 + (C_a - A_a)\,\omega'_{x1}\,\omega'_{z1} \\[6pt] & + (B_b - C_b)\,\omega_{y1}\,\omega_{z1}\,\sin\beta_1 - (A_b + A_g - C_b - B_g)\cos\beta_1\,\omega_{x1}\,\omega_{z1} \end{aligned} \right.$$

$$b_6 = M_{\beta 2} - (C_b + B_g + J_{m\beta}G_\beta)\,\omega'_{y2}\dot{\alpha}_2 + \omega_{z2}\,\Omega_2 A_g + (A_b + A_g - B_b - B_g)\,\omega_{y2}\,\omega_{z2}$$

$$b_7 = \left\{ \begin{aligned} & M_{\alpha 2} + (\omega'_{x2} + G_\beta\dot{\beta}_2)\,\omega'_{y2}J_{m\beta} - \dot{H}_2\sin\beta_2 \\[6pt] & - (A_b + A_g)\sin\beta_2\,[-\cos\beta_2\omega'_{x2}\dot{\alpha}_2 - \sin\beta_2\omega'_{y2}\dot{\beta}_2 + (\omega_z + \dot{\alpha}_2)\,\dot{\beta}_2\cos\beta_2] \\[6pt] & - (B_b + B_g)\cos\beta_2\,[\sin\beta_2\omega'_{x2}\dot{\alpha}_2 - \cos\beta_2\omega'_{y2}\dot{\beta}_2 - (\omega_z + \dot{\alpha}_2)\,\dot{\beta}_2\sin\beta_2] \\[6pt] & - \omega_{x2}\,\Omega_2 A_g\cos\beta_2 - (A_a - C_a)\,\omega'_{x2}\,\omega'_{y2} \\[6pt] & + (B_b - C_b)\,\omega_{x2}\,\omega_{z2}\,\sin\beta_2 - (A_b + A_g - C_b - B_g)\,\omega_{x2}\,\omega_{y2}\cos\beta_2 \end{aligned} \right.$$

$$b_8 = M_{\beta 3} - (C_b + B_g + J_{m\beta}G_\beta)\,\omega'_{z3}\dot{\alpha}_3 + \omega_{x3}\,\Omega_3\,A_g + (A_b + A_g - B_b - B_g)\,\omega_{x3}\,\omega_{z3}$$

Equations (C-16) - Concluded

$$
b_9 = \begin{cases}
M_{\alpha 3} + (\omega'_{y3} + G_\beta \dot{\beta}_2)\, \omega'_{z3} J_{m\beta} - \dot{H}_3 \sin\beta_3 \\[2ex]
- (B_b + B_g) \cos\beta_3 \left[\sin\beta_3\, \omega'_{y3} \dot{\alpha}_3 - \cos\beta_3\, \omega'_{z3} \dot{\beta}_3 - (\omega_x + \dot{\alpha}_3)\, \dot{\beta}_3 \sin\beta_3 \right] \\[2ex]
+ (A_b + A_g) \sin\beta_3 \left[\cos\beta_3\, \omega'_{y3} \dot{\alpha}_3 + \sin\beta_3\, \omega'_{z3} \dot{\beta}_3 - (\omega_x + \dot{\alpha}_3)\, \dot{\beta}_3 \cos\beta_3 \right] \\[2ex]
- \omega_{y3} \Omega_3 A_g \cos\beta_3 - (A_a - C_a)\, \omega'_{y3} \omega'_{z3} \\[2ex]
+ (B_b - C_b)\, \omega_{x3} \omega_{y3} \sin\beta_3 - (A_b + A_g - C_b - B_g)\, \omega_{y3} \omega_{z3} \cos\beta_3
\end{cases}
$$

Appendix D

CONTROL AND DISPLAY UNIT

Introduction

The performance capabilities of the control and display unit have been summarized in an earlier section. The following describes the methods of implementing the requirements and specific equipment recommendations.

Interface with Data Adapter/Computer

A block diagram of the system is shown in Figure D-1, which indicates the necessary interface. By interfacing the control and display unit with a data adapter, characteristics of the CMG control computer (I/O speed and number of channels, signal levels, word length, serial or parallel operation, number and type of control lines, etc.) can remain undefined until the actual computer has been chosen.
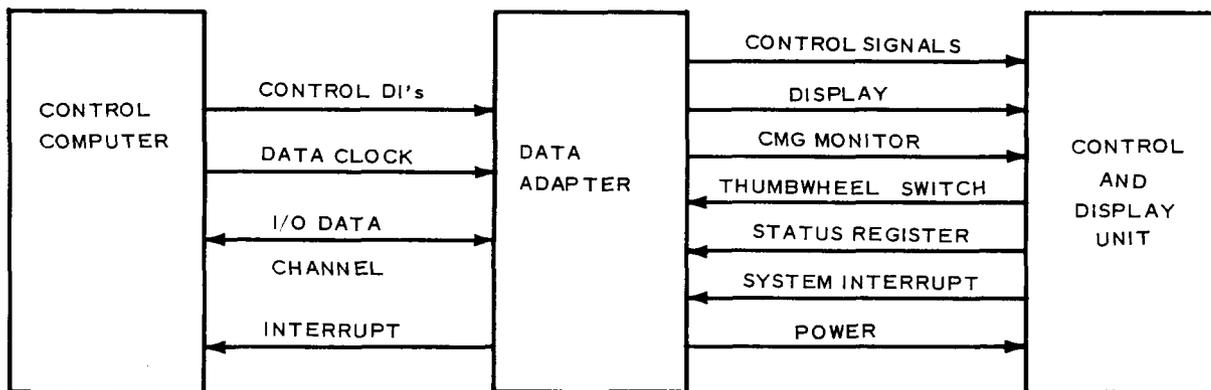
Figure D-1. System Block Diagram

Computer I/O Characteristics

The interface between the computer and the data adapter however, shall consist of the following minimum requirements.

1. Interrupt Line—A single interrupt line is required to allow the control and display unit to request certain computer subroutines. Upon interrupt the computer branches into a search routine which determines the source of the interrupt.

2. Data Channel—A single channel is required which has the capability for two way data transfer. The assumed word length is 16 bits, either serial or parallel.

3. Data Clock—The computer shall provide the necessary gating to accomplish transfer of information through the I/O data channel. The clock is also required so that the control and display unit can generate the proper control signals.

4. Control Lines—Four control lines are required to indicate the content of the information on the I/O buss (data or address), and to indicate the direction of data flow.

Increased capability within the computer, such as several I/O data channels, ability to accept and/or generate a number of discretes, more than one interrupt line of the same level, or more than a single priority interrupt system will reduce the required control logic in the control and display unit.

Data Flow

A block diagram of the control and display unit data flow is shown in Figure D-2. The unit contains the necessary logic to perform the display and control functions previously specified. Logic control is accomplished by utilizing the computer input control lines and internal data clock.

On the basis that only a single two way I/O channel is available for data transfer between the control and display unit and the computer, the data adapter must contain a register which acts as a temporary buffer to collect information from the various control and display unit registers for subsequent transmission to the computer. It also acts as the common temporary buffer for all output data transmissions from the computer. The computer counterpart of this register (the accumulator) is the common register which sends data to, or collects data from the data adapter. The selection of sender or receiver of information within the control and display unit is the function of the address decoder. When informed by the computer that the data is to be interpreted as address information, it decodes the information from the I/O data channel. Direction of data flow is controlled by a read and a write line. Generally, two I/O operations are required per data transfer; the first sets up the address and the second accomplishes the transfer.
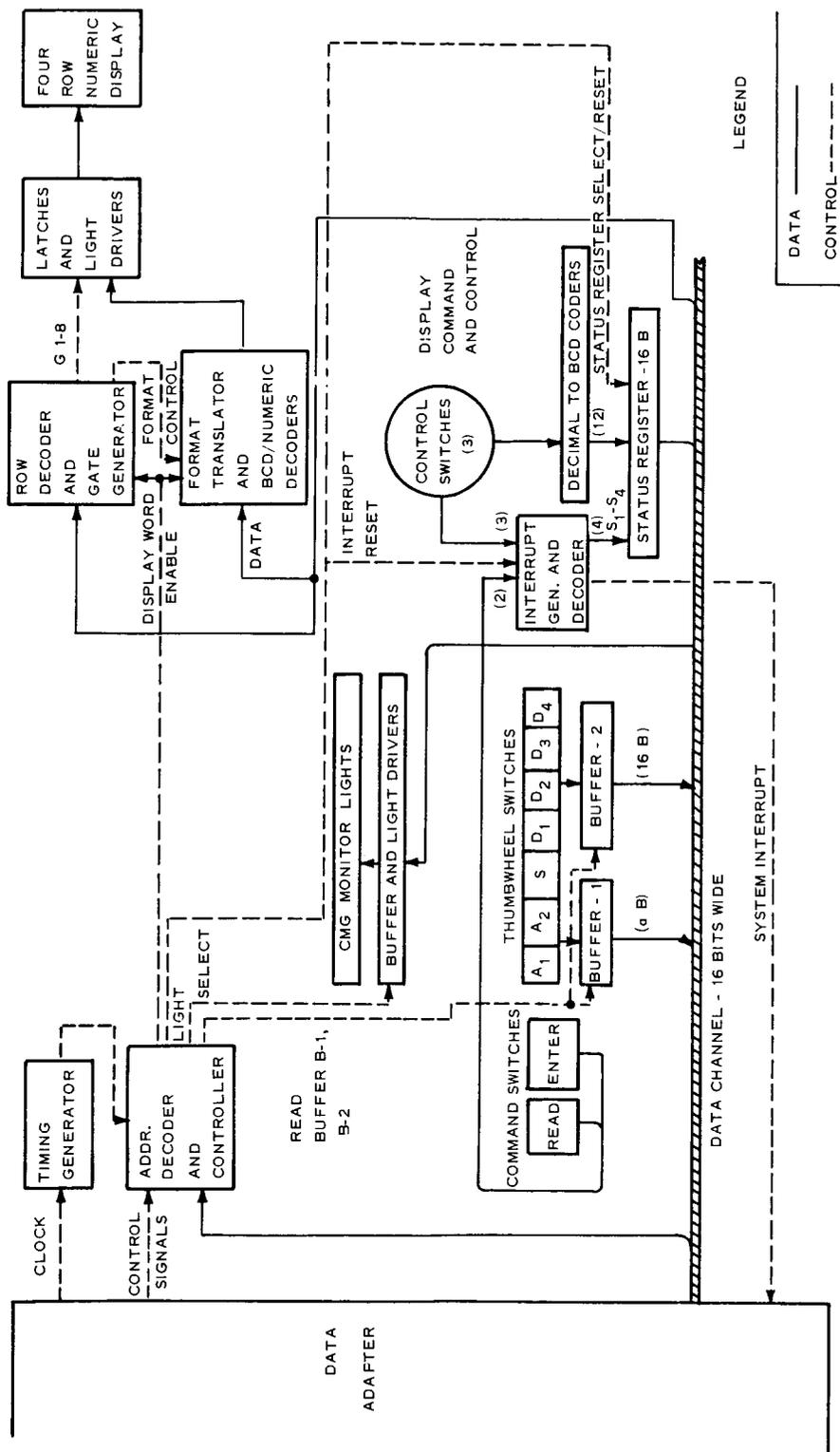
Figure D-2. Control and Display Unit Block Diagram (Single Data Channel Configuration)

The common I/O buffer register in the data adapter would logically be located in the control and display unit if it were the only equipment interfacing with the computer. Since the data adapter will also interface with a number of other peripheral equipments, the buffer register will conceivably be used for functions other than control and display.

Interrupt Line

The interrupt line provides program efficiency in that the computer does not have to interrogate the various mode switches, digital data insert switches, etc., on a repetitive basis to determine if the status has changed since the last time they were sampled. In the system as presently defined, there are four sources of external system interrupts; the digital data insert and display switches and the three mode switches. An insert or display command, or a change in a mode switch position will cause the interrupt generator and decoder to initiate a system interrupt to the computer. Upon recognition of the interrupt, the computer issues an I/O instruction to read the status register. A program subroutine decodes the first four bits of the status register to determine the source of the interrupt. For an interrupt associated with one of the control switches, the program makes use of a portion of the data already transferred into the computer from the status register. If the interrupt is associated with the thumbwheel switches, the appropriate read data buffer commands are issued. The status register and interrupt latch are reset by a read command from the computer. The interrupt decoder also determines which buffer registers are to be loaded. The data format for the status register, which is loaded regardless of the interrupt source, is:

A	B	C	D
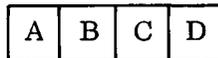
A – Coded interrupt identity
B – Display mode position – BCD
C – Control mode position – BCD
D – Command mode position – BCD

CMG Monitor

Control of the light turn on or turn off is accomplished by an I/O write operation which defines the address and bits in the correct register position. A one in the assigned bit position will turn the associated light on, and a zero will turn it off. It should be noted that status information for all lights is supplied as a result of a single I/O data transfer. Care must be taken in the program that the change in status of a single quantity does not affect other status bits. Also, since a manual reset capability exists, status information must be periodically transferred to the display by the computer program.

The electroluminescent (EL) display devices are solid state with high reliability and present efficiencies in weight, volume, and power over other types of display devices considered (see Display Device Trade-Off section). The devices form numerics by illuminating combination of straight line segments as shown in Figure D-3. Each individual segment to be illuminated requires AC excitation, and is controlled by a binary to numeric switching matrix or translator which decodes the data for display. Since the translator inputs go to a set of latches, which provide a memory capability, the input data is required only during the time the read gate is on.
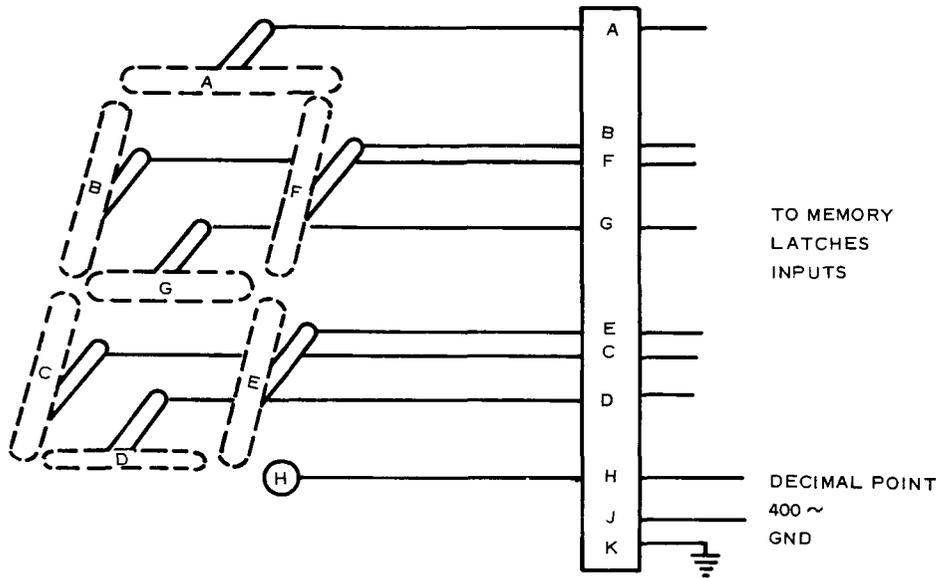
Figure D-3.   Electroluminescent Numeric Display

Figure D-4 shows a typical method of driving the EL display devices. A row of displays is serviced by two separate sets of latches, operating on separate gating signals, which control the SCR display drivers. In the figure, A_1 represents Address 1, A_2 Address 2, D_1 Data position 1, etc. Each numeric requires a four digit BCD input to define the decimal number, and D_1 through D_4 must also be capable of displaying a decimal point which requires a fifth bit. Obviously the assumed 16-bit word length will require two I/O operations for each display row. The format for the data transfer includes numeric, decimal point, and row information as shown in Figure D-5. The row decoder and gate generator, in conjunction with the control signal decoder, provide one of eight gate signals to the display memory latches.
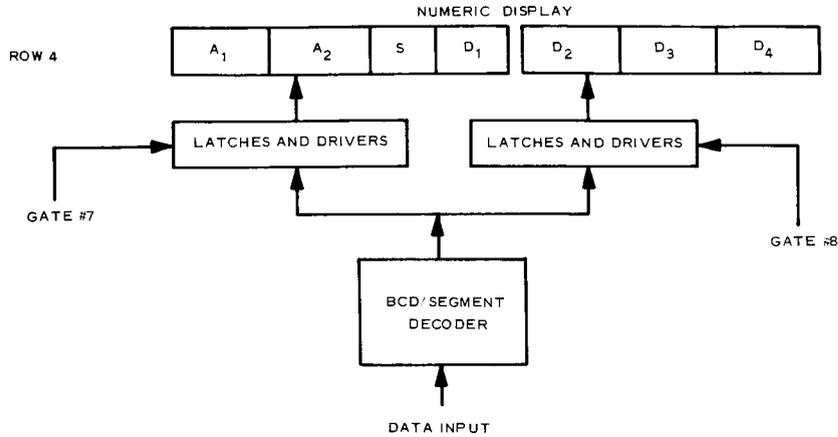
Figure D-4.   Display Decoder/Memory/Driver

As seen in Figure D-4, gate seven allows latch group seven to read the data decoder for a row four, word one operation. The row four, word two decode will generate gate eight which updates the remainder of the display.
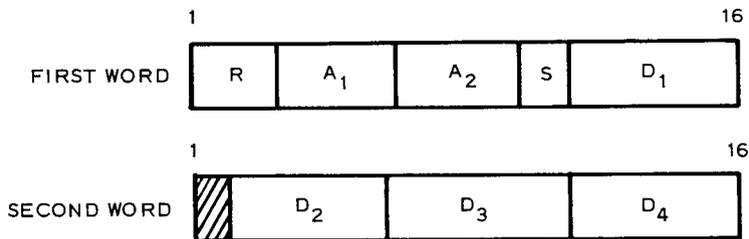
R   - Row Identifier (1-4)
A_1 - Address Digit 1 - 8-4-2-1 BCD
A_2 - Address Digit 2 - 8-4-2-1 BCD
S   - Sign
D_1 - Data Digit 1 - 16-8-4-2-1 BCD
        where 16-bit initiates decimal point
D_2 - D_4 - Same as D_1

Figure D-5.   Display Data Format

The logic functions, data formatting and I/O operations described pertain to both the general purpose single row (defined as row four) display and the three row special purpose display. The control computer contains the necessary subroutines for conversion, coding and outputting the required display information.

D-6

Decimal Data Insert and Display

In the case of a data insert, the operator depresses the insert button after setting up the address and data on the thumbwheel switches. The insert command loads the status register (see Figure D-3) and generates a system interrupt. After processing the interrupt, the control computer reads buffers B-1 and B-2 which contain data in BCD. The computer performs the indicated operation and transfers the data back to the general purpose display to allow verification. The numeric to BCD decoder is an integral part of the thumbwheel switches.

The display operation is similar to the insert except that a single I/O operation is sufficient to transfer the address data into the computer.

Data format for the buffers associated with the thumbwheel switches is as follows:

$$\text{Data Buffer \#1} \quad \overset{1}{\underset{}{\boxed{\text{////} \mid A_1 \mid A_2 \mid S}}} \overset{16}{}$$

$$\text{Data Buffer \#2} \quad \overset{1}{\underset{}{\boxed{D_1 \mid D_2 \mid D_3 \mid D_4}}} \overset{16}{}$$
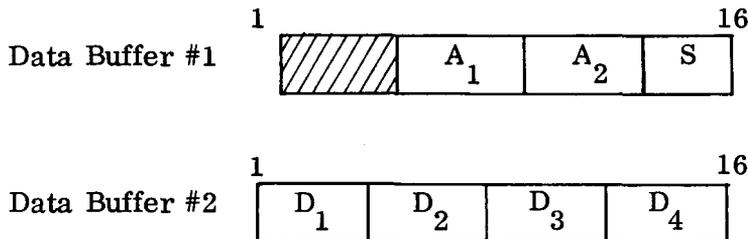
where
A_1 = thumbwheel switch address position 1 – BCD
A_2 = thumbwheel switch address position 2 – BCD, etc.

Data buffer one is transferred to the computer as a result of a display request. Address data $A_1 A_2$ are used to find the correct display data, and the sign data is discarded. Both data buffers are transferred to the computer on an insert command.

Control and Command Mode Switches

The control and command mode switches issue an interrupt to the computer each time the position of the switch position is changed. The computer upon recognition and interpretation transfers the mode switch information through the I/O data channel to the computer. Description of the data flow and data formats are described in a previous section titled Interrupt Line.

Light Emitting Diodes

Potentially, light emitting diodes have numerous advantages over other display devices which presently exist. However, because of the newness of the solution grown GaP diodes, and the experimental nature of the mountings, data as to their performance outside the laboratory environment is insufficient. Continuing development in material technology for high yield, low cost diode processing, and improved packaging and mounting concepts is required before they can be qualified for space applications.

Display Device Trade-Off

Various display devices were considered for use in the CMG control and display panel. Their implications with respect to the panel's weight, power, volume, reliability, and applicability in presenting data in the required time frame and format were studied. Within each group of display devices considered there is a range of design techniques used to obtain the same end result. Because of this the characteristics deviate widely in such areas as weight, power, access time, etc.

The study attempts to use representative characteristics of the display devices (and their drivers and decoders) for each class of display considered. The trade is limited to those which are known to meet the environmental requirements for space applications. The display device must be able to display numeric and sign information in a character height of approximately one-half inch. Alphanumeric display capability is not required. It is desirable that the device be able to display decimal point independently of the numeric. The decoders considered are those which require low level logic signal input so that they are compatible with the control logic within the control and display unit. It is also desirable that they have a latching or memory capability.

The four alternatives considered for display are electromechanical, incandescent light bar, cold-cathode glow discharge and electroluminescent. Table D-1 shows a comparison of these devices. The numbers are based upon a 28-character numeric display which includes the necessary drivers and decoders. The panel area specified is the smallest possible without considering mounting arrangements. The volume reflects the decoders as they are presently packaged by the manufacturer. It is felt that significant reductions in volume could be realized by repackaging certain of the decoders (EL and Cold Cathode are the best candidates).

Table D-2 indicates the comparison of the four display devices considered on the basis of operator recognition under dynamic and static conditions, and operator interpretation of each display in its most common failure modes.

TABLE D-1. ENGINEERING FACTORS COMPARISON MATRIX

	Electro-mechanical	Incandescent Light Bar	Cold Cathode Glow-Discharge	Electro Luminescent
Panel area (sq. in.)	38	23.5	32.5	24.5
Character height (in.)	0.28	0.5	0.60	0.5
Volume (cu. ft.)	0.081	0.028	0.143	0.118
Weight (pounds)	11.8	2.33	3.1	3.1
Power during update (watts)	28.0*	10.5	20.8	4.1
Power, steady-state (watts)	0.0	10.5	20.8	4.1
Failure rate (failure in 10^6 hours)	1530	3930	1400	6.1
*Average power — assumes one row updated at a time.				

The comparison matrix indicates that the incandescent light bar readout requires the least panel area, volume and weight of the readouts considered. The electro-luminescent display comes in second in these three categories. The incandescent readout however, requires 2.5 times the power of the EL devices and has a failure rate 230 times higher. The human factors matrix indicates that the EL display is more desirable than any of the other three readouts considered.

From the above comparisons, the electroluminescent display devices are recommended.

Data Insert Considerations

Two methods of inserting data were considered and compared to the method suggested in the original proposal. The original method was to provide three sets of thumbwheel switches (four digits plus sign) and have each set capable of inserting information associated with a single parameter. The two considered are keyboard, and one set of thumbwheel switches (four digits plus sign and a two digit address). Both of these, when compared with the originally proposed method, provide increased flexibility and reduced control and display panel volume.

TABLE D-2. DISPLAY DEVICES — HUMAN FACTORS CONSIDERATIONS

Comparison Factor	Electromechanical	Incandescent Light Bar	Cold Cathode	Electro Luminescent
Failure Mode and Affect on Usage	Device stops operation completely. Operator may not be aware of failure since a digit is still displayed. (4.0)	Segment failure which results in false display. Some false displays will be recognizable as such. (3.0)	Cathode fails resulting in no illuminated display digit. Failure obvious to operator. (2.0)	No catastrophic type failures. Device degrades in light output as a function of power and usage. (1.0)
Recognition- Static	Excellent (1.0)	Good (2.5)	Good to medium pre-biasing of the off lights creates a background light within the device itself. (4.0)	Good if not used in high ambient light. (2.5)
Recognition- Dynamic	Poor due to amount of time required to select and display digits. (4.0)	Good (1.5)	Poor for dynamic displays. (3.0)	Good if not used in high ambient light. (1.5)
Overall Rating	9.0	7.0	9.0	5.0

Either the keyboard or the seven-digit thumbwheel arrangement appear to be acceptable for the CMG control and display panel, and both are preferrable to the original data insert concept. The seven-digit thumbwheel is recommended for the following reasons:

1. Less hardware in terms of switches, buttons, etc.

2. Direct verification of address and/or data prior to use of that information.

3. No extra program or hardware involvement in determining position of address and data digits, number of digits presented, etc.

The mechanics of using the thumbwheel switches are not as desirable as the keyboard, but in a low data rate system the thumbwheel switches are acceptable.

Electrical Design

Texas Instruments Series 2400 monolithic circuits and GE C5D series silicon controlled rectifiers are considered for the logic and switching functions. The Texas Instrument Series 2400 is the standard 5400 line supplied to IBM specification for the 4 Pi militarized aerospace computer family. The basic circuit is a Transistor-Transistor Logic (TTL) positive NAND gate.

A summary of the electrical characteristics of Series 2400 TTL logic circuits is:

Power Supply	5V±0.25V
Operating Case Temperature	–55° C to +125° C
Fan Out	14 (Max.)
Logic "0" Level	0V to 0.44V
Logic "1" Level	2.40V to 4.5V
Input Switching Threshold	1.0V to 1.8V
Average Power per NAND Gate	11.4 MW
Average Power per Trigger	41 MW
Minimum Noise Margins	400 MV per Level
Typical Propagation Delay	15 NS in Computer Environment

Applications of Series 2400 circuits have shown that an average density of three circuits per flatpack is typically obtained. The Series 2400 integrated circuits are completely qualified for military applications.

The silicon controlled rectifiers used in conjunction with the EL displays provide the necessary low power switching for the segment matrix which forms the various numerics. The silicon controlled rectifiers, in the present configuration are packaged as discrete components in TO-5 cans.

Total logic and switching requirements are implemented with 227 flatpacks and 200 discrete components. Power consumption is estimated at 7.07 watts nominal and 8.93 watts peak. The power for both logic and displays is provided by an external source.

Mechanical Design

The general packaging concept for the CMG unit is shown in Figure 7-1. The physical characteristics are:

Maximum dimensions	8" x 15" x 3"
Weight	4.4 pounds
Volume	360 cubic inches

The major elements of the assembly are three electroluminescent display devices, an array of thumbwheel switch modules, two pluggable logic subassemblies herein referred to as "pages", and three rotary and three pushbutton switches. The EL assemblies provide two numeric displays (one row and three rows) and a nine light monitor panel. Mechanical support for these elements is provided by an aluminum front panel. All sub-units mounted to the panel are enclosed by an aluminum cover to effect a dust seal. The cover is slightly smaller than the panel to create a peripheral overhang suitable for securing the assembly to a mounting frame.

The logic pages are formed by assembling T^2L flatpacks and discrete components to multilayer interconnection boards which are bonded to an aluminum support plate. The pages are supported on stand-offs and are electrically interconnected to the EL readouts by flat printed wiring (RFI shielded) and to the other components by discrete wiring.

The I/O connector is mounted on the back of the dust cover and interconnected with discrete wiring cable of sufficient length to allow cover disassembly for access to all major components and test points.

Reliability

Reliability analysis of the proposed CMG control and display unit resulted in an MTBF estimate of 13,400 hours. Reliability of the unit for a 30-day mission where the unit is turned off for half the mission is 0.97. (An off time failure rate of one-tenth of the on time failure rate was used for this calculation.)

The failure rates and component parts count used in making the estimates are shown in Table D-3. The model used for caluclating the MTBF is based on the constant component part failure rate concept. It is assumed that component parts are failure independent and that any part failure constitutes equipment failure.

TABLE D-3. CMG CONTROL AND DISPLAY UNIT, FAILURE RATE ESTIMATES

Components	Failure Rate λ (Failures/ Million Hours)	Quantity η	$\eta\lambda$
SN 2458	0.0228	24	0.5472
2459	0.0185	4	0.0740
2460	0.0119	1	0.0119
2461	0.0087	64	0.5568
2473	0.0540	117	6.3180
Silicon Control Rectifiers	0.1	200	20.0000
EL Display Signals	0.03	213	6.3900
Switches			
Rotary	3.2	3	9.6000
Thumbwheel	4.0	7	28.0000
Pushbutton	0.34	3	1.0200
Connections Solder	0.0001	3948	0.3948
MIB Segments	0.000017	1974	0.0336
MIB Holes	0.000428	3948	1.6897
Connector	0.101	2	0.2020
$\Sigma\eta\lambda$ (Failures/Million Hours)			74.8380

Failure rates are based upon an average component part hotspot operating temperature of 65° C, 1968 production schedule, continuous operations, a benign ground/orbital environment and procurement of high reliability parts for production hardware.

Appendix E

SYSTEM 4 Pi COMPUTER CHARACTERISTICS

Model TC

Summary. —The Model TC is a general purpose, stored program digital computer. It is designed for the low cost, medium range of computational performance. The hardware requirements are minimized by utilizing a byte (8 bits) data transfer format and efficient utilization of hardware registers to perform multifunctions. The computer is subdivided into a memory, processor, and I/O section. All signals entering or leaving the computer are digital in nature. In the configuration presently being fabricated, the power is supplied from an external source.

The 2.5 microsecond, 8192 byte core memory has a parallel 8 bit transfer path to the processor section and operates within a basic 3 usec machine cycle. The computer has the capability to address 64K bytes of main store.

The processor is organized as a bus system design with an 8 bit fully parallel data flow. The processor contains the facilities for addressing main storage, for fetching or storing information, for arithmetic and logical processing of data, instruction sequencing, timing, and for initiating the communication between storage and external equipment.

Working registers are located both in hardware and memory. A total of 54 instructions are implemented. Processing rates on the order of 48,500 operations/ second are attained by optimizing the machine organization, monolithic logic circuitry, and the high speed core memory.

A summary of Model TC characteristics is contained in Table E-1.

Data and instruction formats. —The TC computer transmits information between main storage and the processor in an 8 bit unit of information called a byte, the basic building block of all formats.

The basic arithmetic operand is the 16 bit fixed point binary word (sign + 15 bits) and is formed by two successive 8 bit memory words. There is no parity check of the memory byte. A four byte operand is used for double precision as well as products and dividends. Two byte operands must be located in main storage on even byte

TABLE E-1. SYSTEM 4 PI/MODEL TC CHARACTERISTICS

Type	General Purpose, digital
Organization	Byte parallel, fixed point, fractional, two's complement notation for negative numbers
Instructions	54
Instruction Word	1, 2, or 3 bytes
Data Word (Single Precision)	2 bytes (S + 15 bits)
Data Word (Double Precision)	4 bytes (S + 31 bits)
Computer Cycle Time	3 usec
Memory Cycle	2.5 usec

	Short Format	Long Format
Add Time	15 usec	18 usec (12 usec if BR-0)
Multiply Time	51 usec	54 usec (48 usec if BR-0)
Operations/Sec	48,500 typical	

Physical

Size	9.75" x 17.12" x 4" (.37 cu. ft.)
Weight	17.3 lbs.
Memory	8192 bytes (8 bits) expandable to 16K bytes
Circuitry	T^2L integrated circuits
Power	60 watts
Environment	MIL-E-5400 Class 2
Reliability	7,500 hours MTBF Predicted

boundaries. Four byte operands must be positioned so that the two least significant bits of the binary operand address are zero. Figure E-1 shows the basic word formats used in the TC Computer.

Two instruction word formats are used in the computer to provide ease of programming and to achieve good bit efficiency. These instruction formats are compatible with the 8 bit memory organization, being 8, 16, or 24 bits in length. All instructions of 16 or 24 bits in length cause a reference to main store. Some of the 8 bit instructions (e. g., SKIP ON CARRY, ZERO Q) make no reference to main store.

The short instruction format is utilized for the most common instructions, thereby minimizing the program memory size while the longer (16 bit) instructions provide greater flexibility and larger operand addressing. The 24 bit short format-immediate provides 16 bits of immediate data along with an 8 bit operation code and displacement field.

The long format instruction consists of a flag bit to designate the long format, a five bit instruction code, a two bit base register field, and an 8 bit displacement field. This instruction requires two memory bytes. Operand addressing is to a displacement of 256 syllables with respect to one of three base registers. The displacement is a direct address to the first 256 syllables of memory when the base register field contains zero for most instructions.

In each format, the first bit is a flag bit denoting a short or long instruction format. The short format also contains a three bit instruction code, and a four bit field which is used as either a displacement for operand addressing or to specify the number of shifts. Short format operand addressing is to a displacement of 16 syllables with respect to soft base Register 1 (operand fetching only).

Addressing. —Byte locations in storage are consecutively numbered starting with zero; each number is considered the address of the corresponding byte. A group of bytes in storage is addressed by the left most byte of the group. The addressing capability permits a maximum of 65, 546 bytes, using a 16 bit binary address. The TC computer memory addressing hardware permits the addressing of 16, 384 bytes, although 8,192 bytes are presently implemented. This set of main storage addresses includes some locations reserved for special purposes.

For addressing purposes, operands can be grouped in two classes: explicitly addressed operands in main storage and immediate operands placed as part of the instruction stream in main storage.

To permit the ready relocation of program segments and to provide for the flexible specifications of input, output, and working areas, all instructions referring to main
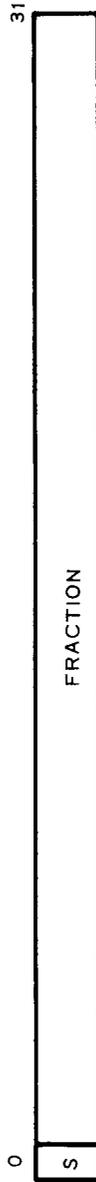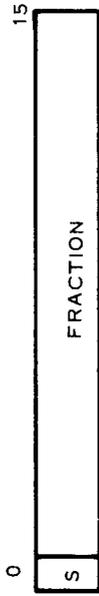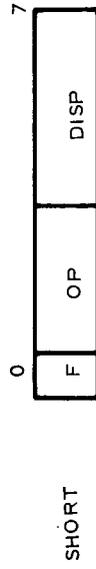
Figure E-1. Word Formats

storage have been given the capacity of employing a full address. The address used to refer to main storage is generated as follows:

Base Address (B) is a 16 bit number contained in a soft general register specified by the program in the B field of the instruction. The base address can be used as a means of static relocation of programs and data. The base address provides for addressing the entire main storage. A total of three base registers are located in main store and contain base addresses used in effective address generation. A special base register designated linkage register (L) is also used but only for branch type instructions.

Displacement (D) is a number contained in the instruction formats. It is included in every address computation. The displacement provides for relative addressing up to 256 syllables for long format instructions and 16 syllables for short format instructions.

In forming the address, the base address is treated as unsigned 16 bit positive binary integers. The displacement is similarly treated as a positive binary integer. The two are added as 16 bit binary numbers ignoring overflow after the displacement syllable address is converted into an effective byte address. Since every address includes a base, the sum is always 16 bits long although only 14 bits are implemented in the computer. The resulting address is the effective address of the left most operand byte.

Processor Section. —The data flow in the processor section is via one of three data buses: the X Bus, the Y Bus, and the Main Bus. The X bus gates one of eight registers to the left side of the adder. The Y bus gates one of three registers to the right side of the adder. Output data from the adder is distributed to any one of seven registers via the main bus. Data inputs and outputs to and from the I/O are gated onto the main bus.

The 8 bit adder features the use of binary full adders packaged one bit per flatpack, thereby improving performance and cost packaging effectiveness. The adder executes all arithmetic and effective address update functions. In addition to the add function, the adder also has the capability to perform AND, OR, and exclusive OR functions.

The hardware registers of the processor section are also packaged very efficiently with two register bits contained in a flatpack without the need for external gating hardware.

The following hardware registers are included in the design:

1. Accumulator (A)—The 16 bit accumulator is the principal arithmetic register.

2. Multiplier-Quotient Register (Q)—The Q register is a 16 bit low order extension of the accumulator.

3. Memory Buffer Register (MBR)—The MBR is an 8 bit register that receives the output from the memory.

4. Displacement Register (D)—The D register is an 8 bit register that stores the address displacement. The TC computer possesses a displacement field of up to 256 syllables for long format instructions and up to 16 syllables for short format instructions. It is also used as a temporary storage register on various arithmetic operations.

5. Instruction Register (I)—The instruction register is a 14 bit register that holds the address of the next sequential instruction.

6. Operation Register (O)—The operation register is an 8 bit register used to store the instruction code.

7. Memory Address Register (MAR)—The Memory Address Register is a 14 bit register used to store and select the proper memory address. The two least significant bits are configured as a counter.

8. Shift and Multiply Counter (S/M)—The S/M counter is used to count iteration during shift, multiply, and divide instruction. The five bit counter allows for a maximum of 32 shifts in either direction.

9. Status Register (SR)—The status register contains the wait state bit, the carry indicator, and the two interrupt masks.

The interruption system used in the TC computer permits a change in state of the computer in response to conditions external to the system. Two interrupts have been designed:

1. Interrupt #0—Major priority for such interrupts as power off interrupt.

2. Interrupt #1—Non-critical priority interrupts that can be interrogated by program control. An example of this is a real time clock interrupt.

An interrupt consists of storing the updated Instruction Counter (IC) and executing an automatic indirect branch. The IC is saved in a location permanently assigned to the interrupt.

Storage section. —The TC computer utilizes an 8,192 word by 8 bit destructive readout core memory. The memory has a 2.5 microsecond read-write cycle time. Each cycle consists of a read followed by a write operation. The computer initiates a

start memory every three microseconds. The access time of the memory is approximately .9 usec. A coincident current (3-D) selection scheme is utilized. The memory array requires four memory mats packaging two bits per mat. Operation is possible over the ambient temperature range of -55° C to +100° C.

The memory uses external data buffer (memory buffer) and memory address lines (14) sent from the processor section. The system is organized such that the "X" half select driver is clocked first since it drives the long dimension. The "Y" current thus has time to reach full amplitude at a slower rate, thereby reducing driver supply voltage requirements.

Timing is generated internally to the memory section. Address decoding is carried out by logic circuits internal to the memory section.

The monolithic circuits used in the memory are designed to "state-of-the-art" groundrules obtained from integrated circuit manufacturers. The high voltage, high current address driver and inhibit driver consist of monolithic pre-driver transformers coupled to discrete chip output stages packaged in flatpacks. The sense amplifier is a monolithic circuit with additional resistors added for threshold setting.

The storage element is an IBM 0.0137" by 0.021" by 0.0045" (nominally 13/21) lithium nickel ferrite toroidal core that is designed to operate over a wide temperature range.

Input/Output section. —The Input/Output section of the TC computer performs the following functions:

1. Data Formatting—The I/O section provides for a byte to serial data conversion to allow the processor to interface with other systems. The 8 bit main data bus is converted to a 1 mc serial readout of one syllable burst duration.

2. Real-Time Clock Generation—A real-time clock is located in the I/O section and is used to generate timed interrupts. The contents of the counter are periodically interrogated by the processor. Two low frequency clocks are also generated by the real time clock.

3. Device Address Generation—The external device addresses are decoded in the I/O section.

4. Interrupt Control—The buffer and control logic associated with interrupts is a part of the I/O section.

Computer Packaging. —The TC computer is packaged in a ruggedized aluminum structure that contains three major subassemblies: Processor, Memory, and Input/

Output and Wiring Harness. The structure has dimensions 9.75" x 17.12" x 4.0" and occupies .37 cubic feet. The heat generated from the subassemblies is removed by forced air cooling with the aid of a Rotron Propimax II exhaust fan.

Electrical interfacing is accomplished with four input/output connectors (118 pins/connector) located on the front panel. AC and DC power is applied through separate connectors. Lightweight magnesium covers at the top and back of the unit facilitates removal of the subassemblies, provides radio frequency attenuation, and prevents entrance of foreign materials. The subassemblies are mounted at the bottom of the structure.

The computer is designed to meet the requirements of the MIL-E-5400 Class 2 specification. The three subassemblies are all of the same basic construction differing only in the type of electronic components that are mounted.

Each subassembly (page) normally consists of two rectangular shaped large SLT boards, a thermal mounting plate, the I/O connectors, and the circuit components.

The processor page contains all the logic circuitry for the computer as defined in previous sections. It consists of two SLT-type printed circuit boards with transistor-transistor logic (TTL) integrated circuit flatpacks soldered on. These PC boards are sandwiched about a metal supporting structure which contains three 98 pin Saturn type connectors. An insulator separates each board from the supporting structure. Feed-throughs are provided to allow electrical connection between the two boards of the page. The SLT board contains two levels of signal wires and two internal ground-voltage planes.

The I/O page is identical in design to the CPU page except that due to less electronics only one SLT board and two Saturn connectors are provided.

The memory subassembly contains an 8,192 x 8 bit memory array and the necessary drive and sense circuitry packaged on two SLT boards with one Saturn connector for I/O communication.

The memory array consists of the following subdivisions:

1. 4 memory planes
2. 2 diode printed circuit boards
3. 3 resistor termination boards
4. 1 X line jumper board

The array subassembly is disconnectable from the memory page, thereby allowing flexibility in production and test.

<u>Software support.</u> —The support programming system provided with the Model TC will assist the problem programmer in the production of operational software for the computer by providing:

1. Programming aids that will enable a program to be expressed in a language which is readily understood

2. Facilities for storing, modifying, and accessing precoded programs

3. A program checkout system to ensure operational program readiness

4. Effective documentation manuals.

The assembler program and language will allow a programmer to make efficient use of all the features of the proposed computer while relieving him of the details of machine coding. The assembler will translate symbolic instructions into machine language instructions, assign storage locations, and perform auxiliary functions necessary to produce an executable machine language program.

Service programs will assist the programmer by providing routines for performing frequently used functions, including system tape generation and source program edit facilities. The service programs consist of a source program editor and a set of utility programs. The source program editor is designed to operate with the assembler and is, in fact, called by the assembler. The function of the editor is to update and alter existing source programs at the source statement level. The utility programs consist of the following:

1. System job control program
2. System tape maintenance programs
3. Preparation program.

The system job control program provides for standard transition of control from one support program to another; for example, from assembler to dynamic program checkout system. This control is accomplished through user-supplied control cards specifying the processing programs desired.

The dynamic program checkout system (DPCS) will provide dynamic program analysis of the computer at the instruction word level of operation.

The DPCS will execute any given set of computer instructions as they appear on the symbolic output listing of the assembler. DPCS control will be exercised by user-provided programs written in a high level computer language (e. g. FORTRAN IV). These programs will call simulator subroutines which will manage the operation of the simulator. To aid in program debugging and checkout, the DPCS will provide a

number of outputs concerning data flow, internal operation, and results of computations. These outputs will be generated by various trace and dump routines within the simulator.

The DPCS will provide an accurate simulation of the functional operation of the computer and associated input/output devices. Every mode of computer operation subject to program control will be exercised with DPCS, just as on the actual equipment. The DPCS will operate with the same arithmetic constraints as the actual computer; likewise, the DPCS will have the same program restrictions as the actual computer. The DPCS will aid in providing a near real-like situation to the simulated object program, and will greatly assist the problem programmers in debugging the operational programs.

A self-test routine to detect and isolate Model TC computer malfunctions is designed to be co-resident with the operational program. At periodic intervals the operational program transfers control to the self-test routine to begin a functional go/ no-go indication. The routine is intended to exercise main working registers, data flow and controls, and instruction control logic. Failure of any active computer circuitry, as well as use of invalid data, will cause an error indication. The storage requirements of the self-test routine for Model TC are variable, depending on the degree of testing desired, and the interference allowable with the operational program.

Instruction list. —A complete list of Model TC instructions and execution times is contained in Table E-2.

Model CP-2

Summary. —The Model CP-2 is a general purpose, stored program, digital computer. It is designed for real-time control applications such as navigation, guidance, or weapons delivery.

Major subsections of the computer are storage, processor, and I/O section. All signals entering or leaving the computer are digital in nature. In the configuration presently being fabricated, a power supply is included within the processor structure.

The 2.5 microsecond, 8,448 word cord storage has a parallel 36 bit transfer path to the processor section. Two parity bits and two storage protect bits insure accurate transfer and control of data located in storage. The storage section may be expanded up to 33,792, 36 bit words.

The processor is organized as a bus system design with a 16 bit fully parallel data flow. The processor contains the facilities for addressing main storage, for fetching or storing information, for arithmetic and logical processing of data, instruc-

TABLE E-2. INSTRUCTION LIST

OPERATION	FORMAT	EXECUTION TIME
ADD	Short	15
ADD	Long	18*
ADD	Immediate	9
ADD DOUBLE	Long	24*
AND	Immediate	9
AND	Long	18*
ARITHMETIC COMPLEMENT	Short	3
BRANCH	Immediate	9
BRANCH	Long	12
BRANCH BACKWARD ON MINUS	Long	6
BRANCH BACKWARD ON PLUS	Long	6
BRANCH BACKWARD UNCONDITIONAL	Long	6
BRANCH BACKWARD ON ZERO	Long	6
BRANCH FORWARD ON MINUS	Long	6
BRANCH FORWARD ON PLUS	Long	6
BRANCH FORWARD UNCONDITIONAL	Long	6
BRANCH FORWARD ON ZERO	Long	6
BRANCH INDIRECT	Long	12
BRANCH INDIRECT & STORE IC	Long	18
COMPARE	Short	9
COMPARE	Long	18*
COMPARE DOUBLE	Long	24*
DIVIDE	Long	54*
EXCHANGE A & Q	Short	3
EXCLUSIVE OR	Long	18*
INPUT/OUTPUT CONTROL	Long	24*
LOAD A	Short	15
LOAD A	Long	18*
LOAD A	Immediate	9

OPERATION	FORMAT	EXECUTION TIME
LOAD BASE	Long	18
LOAD BASE 1	Immediate	15
LOAD BASE 2	Immediate	15
LOAD BASE 3	Immediate	15
LOAD DOUBLE	Long	24*
LOAD STATUS WORD	Long	12
MODIFY BASE	Long	18
MULTIPLY	Short	51
MULTIPLY	Long	54*
NO OPERATION	Short	3
OR	Long	18
SHIFT LEFT DOUBLE	Short	6-18
SHIFT RIGHT ARITHMETIC DOUBLE	Short	6-18
SKIP ON CARRY	Short	3
STORE A	Short	15
STORE A	Long	18*
STORE BASE	Long	18
STORE DOUBLE	Long	24*
STORE IC	Long	18
STORE STATUS WORD	Long	12
SUBTRACT	Short	15
SUBTRACT	Long	18*
SUBTRACT DOUBLE	Long	24*
TALLY	Long	12-24
ZERO Q	Short	3

*Execution Time is 6 usec Less if B=0.

tion sequencing, timing, and for initiating the communication between storage and external equipment.

Working registers are located in high speed hardware. A total of 35 instructions are implemented presently. Processing rates on the order of 125,000 operations/ second are attained by optimizing the machine organization, monolithic logic circuitry, and the high speed core storage.

A summary of Model CP-2 characteristics is contained in Table E-3.

Data and instruction formats. —The standard or single precision data word is 32-bits long (Figure E-2).

```
0                                                          31
┌──┬────────────────────────────────────────────────────┐
│S │                    FRACTION                         │
└──┴────────────────────────────────────────────────────┘
```
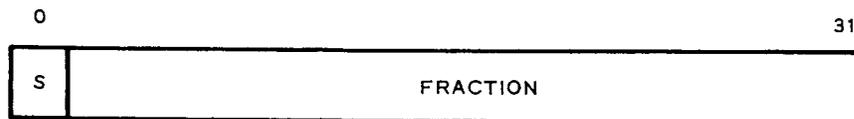
Figure E-2.   Data Format

Positive numbers are always in true binary form. Negative numbers are in two's complement form. The sign bit, position 0, is always 0 for positive numbers and 1 for negative numbers.

A half-precision data word of 16 bits is provided to increase processing speeds in areas where full accuracy is not required. Addressing is to the half-word level, but 32-bit data words must have an even address.

Two basic instruction word formats are used (Figure E-3).

```
0                           15
┌──────┬───┬───┬───────────┐
│  OP  │ F │ T │   DISP    │
└──────┴───┴───┴───────────┘
```

```
0                             16          31
┌──────┬───┬───┬───┬───┬──────────┬─────────┐
│  OP  │ F │ T │ IA│ INT│  COND   │  ADDR   │
└──────┴───┴───┴───┴───┴──────────┴─────────┘
```
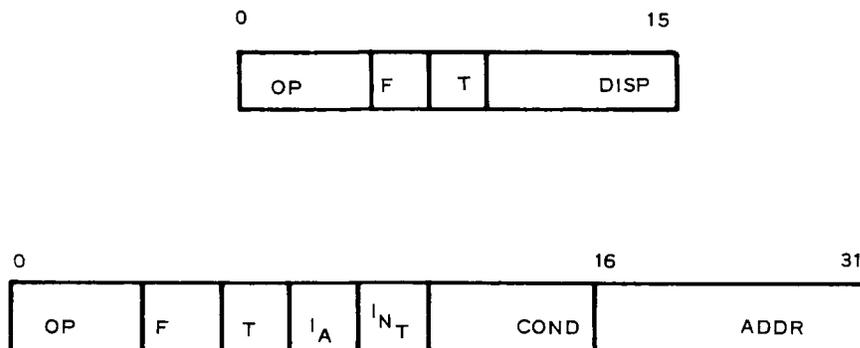
Figure E-3.   Instruction Word Formats

E-13

TABLE E-3. SYSTEM/4 PI MODEL CP-2 CHARACTERISTICS

Type	General purpose, digital
Organization	Parallel, fixed point, fractional, 2's complement notation
Instructions	35
Instruction Word	16 or 32 bits
Data Word	16 or 32 bits
Computer Cycle Time	417 nanoseconds
Storage Cycle	2.5 microseconds

Typical Execution Times	Short Format	Long Format
Add	3.75 usec	5 usec
Multiply	18.13 usec	19.38 usec

Operations/Second	125,000 typical

Physical

Size	10" x 7.6" x 19.6 (.86 cu. ft.)
Weight	47 lbs.
Storage	8,448 – 36 bit words expandable to 33,792 words
Circuitry	TTL integrated circuits
Power	240 watts
Environment	MIL-E-5400, Class 2
Operational MTBF	4,000 hours

In a typical program, long format instructions are used infrequently, allowing a larger program to be contained in storage. The bits within the instruction words are used in the following manner:

OP   These 5 bits define which operation is to be performed by the computer.

F   This format bit controls the instruction format. A "zero" indicates a normal instruction, and a "one" indicates a long instruction.

T   These two Index Tag bits address the Index Register or Instruction Register (I) used in the address modification.

DISP   These bits are called the displacement and, with normal format instructions only, are usually added to the Instruction Register or the index register specified by T. The modified address is defined as the effective address (EA). If T is 00, the displacement is added to the Instruction Register. The displacement is in 2's complement form if negative.

IA   Except for the Modify Index and Skip instruction with T=00, the Indirect Addressing bit is used in the long instruction format. If "zero", addressing will be direct. If a "one", addressing will be indirect.

INT   This bit specifies that the Branch or Skip on Condition (BSC) instruction is to be interpreted as a "Branch Out" when used in an interrupt routine.

COND   These six bits specify the condition of indicators that are interrogated on a BSC or Branch and Store Instruction Counter (BSI) instruction.

ADDRESS   These 16 bits usually specify a core storage address in a long instruction. The address can be modified by the contents of an index register, or used as an indirect address if the IA bit is on.

Addressing. —All core storage locations are directly addressable under program control. The effective address may be determined by examining the tag, flag, and indirect address bits shown in Figure E-3.

The recognition in the instruction of an indirect address control bit will cause the address portion to be treated as an indirect address. The address after indexing, if specified, gives the location of the effective address. An additional storage cycle is not required if index register one is used. One level of indirect addressing is provided.

The effective address for operands if formed as shown in Table E-4.

TABLE E-4. EFFECTIVE ADDRESS GENERATION

Index Reg.	Relative Address	Direct Address	Indirect Address
0	Program Counter + DISP*	ADDRESS	C (ADDRESS)**
1	XR1 + DISP	ADDRESS + XR1	C (ADDRESS + XR1)
2	XR2 + DISP	ADDRESS + XR2	C (ADDRESS + XR2)
3	XR3 + DISP	ADDRESS + XR3	C (ADDRESS + XR3)
*DISP may be positive or negative (2's complement) **C means "contents of" ADDRESS field in instruction			

Processor section. —The processor is organized as a single address, fixed point, two's complement, fractional arithmetic system. It consists of the basic timing, register organization, arithmetic logic unit, I/O interface, and interrupt interface.

The data flow contains eight 16-bit storage registers, two 16-bit buffer registers, one 16-bit storage address register, one 32-bit storage data register, one 16-bit operation code register, one 17-bit adder, a bus system, and associated gates and controls. The data flow of the central processor is 16 bits wide. Data flow is organized around a two-clock system with a cycle of approximately 417 nanoseconds. Timing is generated from a frequency source internal to the CPU.

Functions of the hardware registers are as follows:

A1 and A2—These two 16-bit registers comprise the accumulator, the principal arithmetic register. The quotient is also contained in the accumulator following a division operation.

B1 and B2—These two 16 bit auxiliary registers are used to hold the multiplicand or divisor.

IXI—A 16-bit register used in calculating an effective address.

Program Counter - This 16-bit register is used to determine the location of the next instruction to be fetched from storage.

Storage Address Register - A 16-bit register used to determine the next storage location fetched. May be loaded from either the Program Counter or with the calculated effective operand address.

Storage Data Register - A 32-bit register used to hold the word read from the storage array.

Operation Register - A 16-bit register which holds the current instruction to be executed.

Next Instruction Register - A 16-bit register containing the next instruction to be executed, or part of the current instruction for long formats.

Buffer Register - A 16-bit register which contains the output of the adder. Its contents are distributed over the bus system to other hardware registers.

Storage section. —The MS is a three dimensional, coincident current, 8,448 word, 36-bit ferrite core array. Thirty-two bits contain data, two bits are allocated to parity, and two bits are used for storage protection. Access time for this system is approximately 0.9 usec, and it can operate continuously at a 2.5 usec cycle. Core storage addresses begin at 0000 and end at 8,447.

The nonalterable storage capability has been implemented by storage protection tests in every instruction which modifies storage. Each 36-bit storage word contains two storage protection bits, one for each half-word. These bits differentiate between protected and scratch pad locations, and are assigned when core storage is loaded from either the AGE or from the I/O Unit to prevent inadvertent modification of the data by the program. Any attempt to store data in a protected location will be prevented by an Invalid Address interrupt in computer control.

DRO cores with storage protection and power sequencing effectively provide program protection equivalent to most NDRO storage elements. DRO also allows complete flexibility in assigning storage between program and scratch pad, thus simplifying programming and allowing maximum use of the total storage capacity.

The storage element used is an IBM 13/21 (ID/OD in mils) lithium nickel ferrite. It is designed to operate over a temperature range of -55° C to +105° C.

The plane used in the core array is a militarized version of a commercial IBM plane. The plane is approximately 6" x 6" x 0.156" and contains 16,896 cores.

Welded connections are used throughout the memory array. This plane is machine-wired on IBM high volume production equipment. Foam pads are placed between the coated planes to meet the vibration requirements.

Input/Output section. —The input-output section of the computer provides high-speed data transfer on two independent two-way, 17-bit parallel channels. The number of channels can be expanded to three without redesign or engineering modification to the computer. Input-output data transfer on two of the channels is accomplished under external control. Data transfer on the third channel is accomplished under program control. This growth capacity consists of one external controlled output channel and one program controlled input channel.

In addition, the input-output section provides capacity for discrete inputs and outputs. Storage cycle requests for externally controlled channels have priority over all other normal storage cycle requests and are honored at the completion of the existing storage cycle.

Each externally-controlled input channel has a block of 2048, 17-bit words of storage reserved for input data. Direct input to storage will occur under control of the external device, but only when the automatic input/output mode has been enabled by a program instruction, which specifies the channel to be enabled. Each externally controlled input channel is capable of data transmission at a word rate of 60 KC interleaved with information transmission on the other externally controlled channels.

Externally-controlled output channels function in much the same manner, referencing a 2048 word block of storage and transferring data out at a 60KC word rate.

The processor may handle up to 40 discrete input lines, with eight of these operating as interrupt lines.

The interrupts are tested upon the completion of each instruction.

Interrupts are divided into two levels. The highest level consists of one interrupt, and the other level contains the remaining seven. The recognition of an interrupt in the lowest level causes all other interrupts in that level to be inhibited until the interrupt processing program is completed. The high-level interrupt is not inhibited while a low-level interrupt is being processed. Occurrence of the high priority interrupt causes a program branch. Upon completion of the high priority interrupt program, the interrupted program is resumed. The interrupted program could have been the main program or any one of seven lower priority interrupt programs.

Provisions are also included in the I/O Section for handling 16 discrete outputs.

Computer packaging. —The Model CP-2 is packaged using pluggable construction. The pluggable electronic subassembly, called a page, consists of two multilayer printed circuit boards bonded to a metal frame. An insulator separates each board from the frame. Two 98-pin connectors, identical to the ones being used for the Saturn Guidance Computer subassembly, are fastened along the lower edge of the frame. The connector was developed for this application and has demonstrated its performance and reliability in extensive tests and field service. It has been qualified for use in man-rated space vehicles. Feed-through connections and test points are provided along the upper edge of the frame. The page is fastened to the supporting structure at the two mounting flanges along an axis through the page center of gravity. Additional support is provided by the connectors. The page will be cooled by thermal conduction through the frame and mounting flange thermal interface. Keyed guide pins prevent installation in the wrong locations.

The multilayer printed circuit boards are made of several layers of etched, copper-clad, epoxy-glass laminates which are bonded together under heat and pressure. These boards have been in production since 1962 and have been successfully used in many systems produced by IBM including the Titan II, Titan III, Gemini, Saturn I, and Saturn V Guidance Computers.

The integrated circuit flatpacks are soldered to etched patterns on the surface of the multilayer printed circuit boards. Each page contains two multilayer boards with up to 78 flatpacks per board. Discrete components are soldered in plated holes or on the board surface, depending on terminal configuration. A conformal coating is applied for component support and environmental protection.

The storage array is mounted in a housing which serves also to mount pluggable electronic subassemblies and acts as a thermal path. Pluggable electronic sub-assemblies similar to those in the computer are used to mount the storage circuits. A distribution multilayer printed circuit board is used to interconnect storage circuits and the array. The main storage assembly is removable as a module. Electrical connections to the assembly are made through pluggable input/output connectors.

The power supply assembly consists of three pluggable regulator modules and a transformer - RFI filter assembly. High-power components are fastened directly to the base plates for maximum heat transfer. A conformal coating is applied to the printed circuit board assemblies for environmental protection.

Software support. —A sophisticated package of computer programs will be provided for operation on an IBM System/360. In addition, assemblers will be delivered which will operate on System/360 and the Model CP-2. This support package will assist the programmer in producing operational programs for the Model CP-2 by providing:

1. Programming aids that will enable a program to be expressed in a language which is readily understood

2. Facilities for storing, modifying, and accessing precoded programs

3. A program checkout system to ensure central computer operational program readiness.

The software support programs delivered with the Model CP-2 are the following:

1. The assemblers will assemble programs executable on the Model CP-2. These programs will translate symbolic instructions into machine language instruction, assign storage locations, and perform auxiliary functions which allow the programmer to make efficient use of all of the features of the Model CP-2 while relieving him of the details of machine language coding. This program provides for:

   a. The assembly of relocatable object programs
   b. Linkage between programs assembled at different times
   c. Various listings under programmer control
   d. A significant amount of error detection
   e. An assemble-and-run capability when operating with the dynamic program checkout system described below
   f. Automatic selection of long and short format instructions
   g. Macro processing.

2. Service Programs which assist the programmer by performing frequently used functions such as:

   a. Combining and relocating programs assembled at separate times
   b. Handling the transition from assembly to execute as described above
   c. Maintaining the library tape.

3. A dynamic program checkout system (simulator) which will provide for dynamic program analysis of the Model CP-2 at the instruction word level of operation. This program will execute any instructions and aid in program debugging by providing:

   a. Simulator control through a user's FORTRAN IV program
   b. User program access to simulated Model CP-2 object program computed data
   c. Object program correction
   d. Program debugging options such as dumps, snaps, and traces
   e. Input/output and interrupt initiation and response.

TABLE E-5. INSTRUCTION LIST

OPERATION	TYPICAL EXECUTION TIMES	
	Short Format	Long Format
ADD	3.75	5.0
ADD HALFWORD	3.75	5.0
AND	3.75	5.0
BRANCH ON CONDITION	----	4.58
BRANCH OUT ON CONDITION	----	5.42
BRANCH AND STORE INSTRUCTION COUNTER	----	7.71
COMPARE	4.79	6.04
COMPARE HALFWORD	3.96	5.21
DIRECT INPUT OUTPUT CONTROL	2.50	3.75
DIVIDE	47.29	48.54
EXCLUSIVE OR	3.75	5.0
INSERT STORAGE PROTECT BIT	----	5.42
LOAD A	3.75	5.0
LOAD A HALFWORD	3.75	5.0
LOAD INDEX	2.08	3.33
LOAD Q	3.75	5.0
MODIFY INDEX	2.08	3.33
MODIFY STORAGE HALFWORD	----	5.83
MULTIPLY	18.13	19.38
OR	3.75	5.0
SHIFT LEFT	Varies	----
SHIFT LEFT AND COUNT	Varies	----
SHIFT LEFT AND COUNT DOUBLE	Varies	----
SHIFT LEFT DOUBLE	Varies	----
SHIFT RIGHT ARITHMETIC	Varies	----
SHIFT RIGHT ARITHMETIC DOUBLE	Varies	----
SHIFT RIGHT LOGICAL	Varies	----
SHIFT RIGHT ROTATE DOUBLE	Varies	----
SKIP ON CONDITION	2.71	----
STORE A	4.58	5.83
STORE A HALFWORD	5.42	6.67
STORE INDEX	5.42	6.67
STORE Q	4.58	5.83
SUBTRACT	3.75	5.0
SUBTRACT HALFWORD	3.75	5.0

The self-test program is designed to be co-resident with the operational program. At periodic intervals, the operational program transfers control to the self-test routine to begin a functional go/no-go computer check. The self-test routine, when entered, cycles until interrupted by the operational program. Test results are progressive — results may depend upon prior test action, but can be interrupted at any point. There is no requirement for system restoration upon entering or exiting the test routine. The operational program also contains checks for reasonableness of data. The self-test routine covers the processor registers, data flow and controls, and the I/O interface circuitry to the extent that it will not interfere with the operational programs.

Tests performed by the self-test routine include patterns to check all possible adder combinations, carry paths, and carry look-ahead paths. Similarly, patterns and processor actions are used to check the function of the internal registers, shifter and associated data paths, and instruction control logic.

Instruction. —A complete list of Model CP-2 instructions and execution times is contained in Table E-5.