A NEW CONCEPT IN STRAPDOWN INERTIAL NAVIGATION

by

JOHN EDWARD BORTZ, SR.

B.S., United States Naval Academy
(1957)

S.M., Massachusetts Institute of Technology
(1965)

SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE
DEGREE OF DOCTOR OF SCIENCE

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June, 1969

Signature of Author _____
Department of Aeronautics
and Astronautics, June, 1969

Certified by _____
Thesis Supervisor

Certified by _____
Thesis Supervisor

Certified by _____
Thesis Supervisor

Certified by _____
Thesis Supervisor

Certified by _____
Thesis Supervisor

Accepted by _____
Chairman, Departmental
Graduate Committee

A NEW CONCEPT IN STRAPDOWN
INERTIAL NAVIGATION

by

John E. Bortz, Sr.

Submitted to the Department of Aeronautics and Astronautics on June 2, 1969 in partial fulfillment of the requirements for the degree of Doctor of Science.

ABSTRACT

The conventional method for updating the coordinate transformation matrix in strapdown inertial navigation systems is to integrate the matrix differential equation

$$\dot{C}^{RB} = C^{RB} [\underline{\omega}_{RB} x]$$

numerically using the incremental outputs from the system gyros. The major problem in this method is the well known phenomenon of noncommutativity of finite rotations. The two conventional ways of combatting errors due to this effect are (a) to update the direction cosine matrix at or near the gyro rebalance frequency using a simple update algorithm or (b) to update after many rebalance cycles using a more sophisticated algorithm. Even the most efficient algorithm places a very heavy burden on the navigation system computer. If the update process is slowed down to ease the computational load, system bandwidth and accuracy are sacrificed.

In the method presented in this thesis, a correction is generated for the noncommutativity phenomenon using analog computing elements. This correction is fed back through the system gyros where it is summed with the torque produced by the angular velocity of the vehicle. Then the gyros integrate and quantize the sum of the angular velocity and the noncommutativity correction torques. As a result, the incremental data from the gyros can be accumulated over time intervals that are from 10 to 1000 times longer than permissible with conventional algorithms and the update accomplished using a simple closed form solution of the $\dot{C} = C[\underline{\omega}x]$ equation. This is accomplished without sacrificing either the accuracy of the update or system bandwidth.

Incidental to this method, a means is developed for extracting an accurate electrical analog of the vehicle angular velocity vector from the pulse-rebalanced gyro triad. This allows the possibility of analog compensation for gyro dynamical errors and a further unburdening of the navigation system computer.

A similar hybrid computational technique is presented for transforming the specific force measurement from the instrument to the navigational coordinate frame.

The experiment and the test results for verifying the hybrid computation method are described, and a detailed error analysis of the hybrid system is presented.

Thesis Supervisor: Wallace E. Vander Velde, Sc.D.
Title: Professor of Aeronautics and Astronautics

Thesis Supervisor: Walter Wrigley, Sc.D.
Title: Professor of Instrumentation and Astronautics

Thesis Supervisor: Walter McKay
Title: Professor of Aeronautics and Astronautics

Thesis Supervisor: James E. Potter, Ph.D.
Title: Associate Professor of Aeronautics and Astronautics

Thesis Supervisor: Arthur H. Lipton, Sc.D.
Title: Chief, Guidance Programs Branch, NASA/ERC

ACKNOWLEDGEMENTS

The author wishes to express his gratitude:

To Professor Wallace E. Vander Velde who, as thesis committee chairman, gave generously of his technical excellance, of his conscientious attention to detail, and, through encouragement and inspiration, of his friendship.

To Professor Walter Wrigley who contributed ideas and direction as a member of the thesis committee and who guided and encouraged the author throughout his doctoral program.

To Professors Walter McKay and James E. Potter who, as thesis committee members, made valuable and greatly appreciated contributions to the thesis.

To Dr. Arthur H. Lipton who, as thesis committee member, contributed both technically and as a friend. Without his support in placing NASA/ERC resources at the author's disposal, the experimentation in this thesis could not have taken place.

To Professor Walter Hollister who, through his constructive criticism and his direct participation, helped clarify and develop the material in Chapter 7.

To Dr. Richard Hayes and Mr. Robert Wedan of NASA/ERC who, through their encouragement, understanding, and sponsorship, made the author's educational program possible.

To Bernard J. Pattern, who through his excellent technical assistance, helped the author throughout the difficult phase of setting up and performing the experiment.

To Robert Wiseman, Edward Spitzer, Robert Stone, Angelo Fanara, and Maurice Lanman for their consultation in setting up the experiment.

To David Downing and Donald Rockwell, who, though engaged in their own thesis research, assisted the author with in-depth discussion during the entire course of this thesis.

To Donna DeCaro who typed the first draft and to Pam Mullin who did such a fine job in typing the final form.

And to his wife, Betty, who proofread the entire thesis and whose intangible contribution has been greater than words can express.

TABLE OF CONTENTS

Chapter No.		Page No.
1	Introduction	
	1.1 A Critique of Strapdown Inertial Navigation	1
	1.2 Coordinate Frames and the Transformation Computation	6
2	The Hybrid Concept	
	2.1 Introduction	13
	2.2 The Hybrid Strapdown System	17
3	The Dynamics of Finite Rotations	
	3.1 A Vector Concept of Rotations	24
	3.2 The Relationship Between $\underline{\phi}_{RB}$ and C^{RB}	26
	3.3 The Theory of Rotation Vectors	32
	3.4 The Goodman Robinson Theorem	39
4	The Measurement of Angular Velocity	
	4.1 Problem Statement	47
	4.2 Filter Analysis	49
	4.3 Filter Design	59
	4.3.1 Gyro Characteristics	59
	4.3.2 Demodulator	59
	4.3.3 Signal Generator Section	63
	4.3.4 Torque Generator Section	66
	4.3.5 Final Section	66
	4.4 Test Results	66

TABLE OF CONTENTS (Cont.)

Chapter No.		Page No.
5	System Mechanization	
	5.1 The Choice of an Equation	69
	5.2 Analog System Configuration	72
	5.3 The Frequency of the Update	76
	5.4 System Error Analysis	77
	5.4.1 Equation Mechanization Error	78
	5.4.2 Analog Computation Error	80
	5.4.3 Gyro Storage Error	89
	5.5 Test Results	91
6	Evaluation of the Hybrid Concept	
	6.1 Preliminary Considerations	96
	6.2 Round Off Error	98
	6.3 Kinematric Response	99
	6.4 Performance Comparison	101
7	The Treatment of Specific Force	
	7.1 Vector Integration in a Rotating Space	111
	7.2 System Mechanization	117
	7.3 Two Sample Problems	120
8	Conclusions and Recommendations	
	8.1 General Discussion	127
	8.2 Gyro Quantization Level	129
	8.3 Computer Round-off Error	130

TABLE OF CONTENTS (Cont.)

Chapter No.			Page No.
8	8.4	Inertial Sensor Design Considerations	130
	8.5	Recommendations	131
		8.5.1 Filter Design	131
		8.5.2 Cross Product Term Generation	132
		8.5.3 The Treatment of Specific Force	133
		8.5.4 Analog Computation Sealing	133
		8.5.5 Analog Inertial Sensor Compensation	133
Appendix			
A	Laning's Theory		134
	A.1	The Angle Sum of Two Vectors	137
	A.2	Algebraic Properties of the Angle Sum Operation	141
	A.3	The Rotation Sum of Two Vectors	145
	A.4	Algebraic Properties of the Rotation Sum Operation	149
	A.5	The Rotation Vector Differential Equation	150
	A.6	The Coordinate Transformation	155
B	ω-Filter Design Details		160
C	Numerical Integration of the $\dot{\phi}$ Equation		165
D	Hybrid Computer Analog Program		185
E	Hybrid Computer Digital Program		198
F	Notation Conventions		220

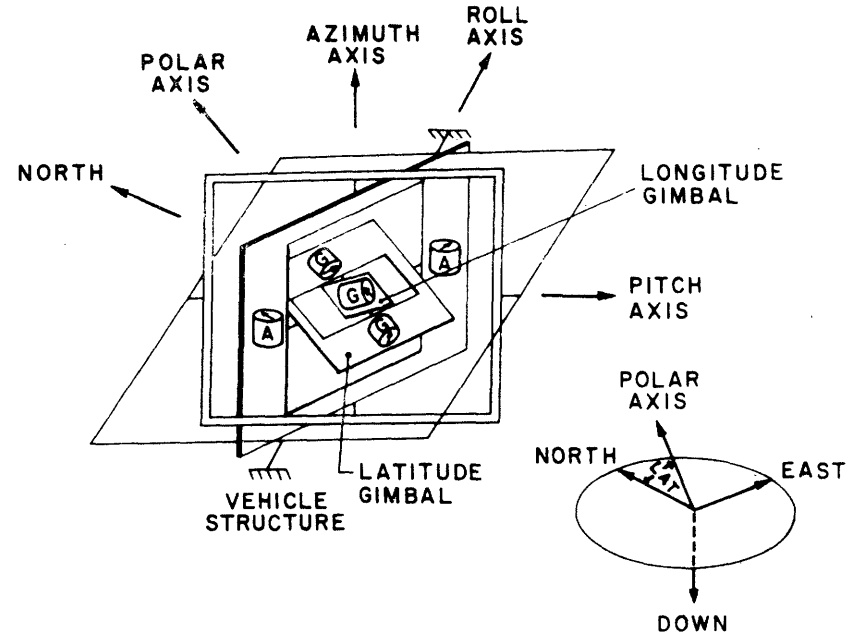
CHAPTER 1

INTRODUCTION

1.1 A Critique of Strapdown Inertial Navigation

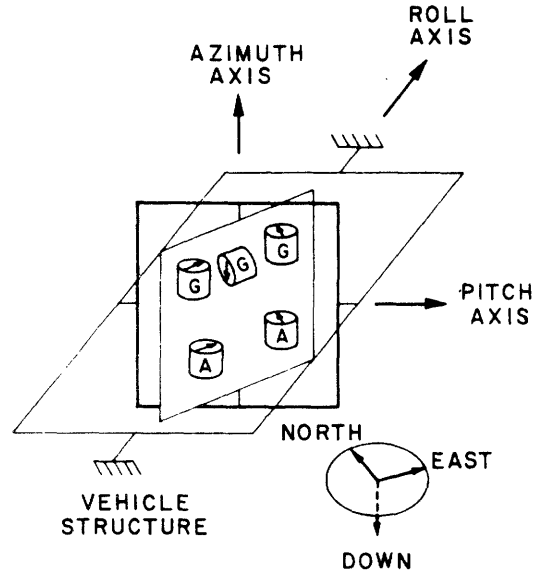
Inertial navigation systems can be classified according to the way they perform the basic operations of inertial navigation (ref. 1). A navigation system is called a geometric system (Figure 1.1) if the orientations of both a reference coordinate frame and a navigational coordinate frame are physically maintained by system gimbals; a semi-geometric or a semi-analytic system if only the orientation of a reference or an intermediate coordinate frame respectively is physically maintained by system gimbals; and an analytic system if the orientation of neither a reference nor an intermediate coordinate frame is physically maintained by system gimbals. An analytic system is commonly termed a strapdown system. (Other names are gimballess or no-gimbal systems.)

Until the present, whenever there is appreciable angular velocity of the vehicle relative to an inertial coordinate frame the best performance has been achieved with a gimballed (geometric, semi-geometric or semi-analytic) system. The performance of a strapdown system is limited primarily by two factors not present in gimballed navigation systems. The



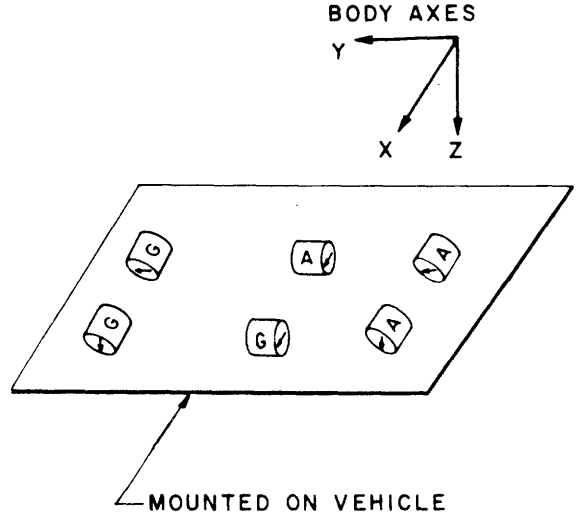
ATTITUDE FROM GIMBAL ANGLES
LATITUDE AND LONGITUDE FROM GIMBAL ANGLES

(a) GEOMETRIC SYSTEM



ATTITUDE FROM GIMBAL ANGLES
COMPUTE LATITUDE AND LONGITUDE

(b) SEMI-GEOMETRIC SYSTEM



COMPUTE ATTITUDE
COMPUTE LATITUDE AND LONGITUDE

(c) ANALYTIC SYSTEM

Figure 1.1.- Inertial system configuration

cause of both limitations is the vehicle-fixed mounting of the inertial sensors in strapdown systems. In gimballed navigation systems, the inertial sensors are mounted on gimbals whose orientations are nominally stationary relative to either the intermediate or the reference coordinate frame. On the other hand, the inertial sensors of a strapdown system partake fully of the vehicle's angular motion. A rebalancing signal must be applied to the gyro to keep its spin axis near its case-fixed reference direction. Furthermore, the measurement of specific force must be transformed from the frame of the measurement into that reference coordinate frame in which the integrations of acceleration are to take place.

The first factor limiting the performance of a strapdown system is that the measurement of input axis angular velocity provided by the gyro must be inferred from the rebalance torque applied to the gyro. As a result, the integral of the input axis angular velocity is known only in terms of the rebalance torque. In gimballed inertial navigation systems, since the gyros do not experience the vehicle's angular velocity, the torque applied to a gyro's torque generator is always small and the calibration of this torque is not of great concern. In a strapdown system, the calibration of the rebalance torque is crucial and is one of its major problems. Also, the error model for a single degree-of-freedom gyro includes error terms which are functions of angular velocity and therefore must be calibrated and compensated.

The second limitation arises from the need to transform the specific force measurement from the Measurement* Frame to the Reference Frame. The coordinate transformation from the Measurement Frame to the Reference Frame must therefore be known. This coordinate transformation is usually computed from the pulses (each pulse representing an increment of integrated angular velocity) obtained from pulse rebalanced gyros. Each pulse represents a finite rotation. However, finite rotations do not commute (ref. 2), since Rotation A followed by Rotation B does not, in general, produce the same result as Rotation B followed by Rotation A. Consequently, coordinate transformations computed from these finite increments include, to some degree, errors resulting from the noncommutativity of finite rotations. The size of these errors depends upon the size of the increment and upon the sophistication of the algorithm used in updating the coordinate transformation.

Thus, the two chief limitations of strapdown navigation systems are the scale factor error and the coordinate transformation computation error. No techniques for reducing the scale factor error are presented in this thesis. It is to the coordinate transformation computation error that attention is addressed. Previously, each new proposal for reducing the coordinate transformation error has been a new algorithm for updating the coordinate transformation using the incremental

---

* A coordinate frame whose name is capitalized refers to a specific frame. When the name is not capitalized the reference is to a class of coordinate frames.

data from the gyros. Of these algorithms, it may be said that they require a great deal of computer capability and capacity. In fact, this heavy computer loading prevented serious strapdown system development until the early 1960's when a new generation of aerospace computers enabled the coordinate transformation computation to become a practical reality.

Strapdown navigation systems have certain advantages over gimballed systems. They are more reliable and more easily maintained than gimballed systems; they are smaller and more flexible in shape; they consume less power since they draw no gimbal torquing power. However, these advantages are irrevelant if an application requires accuracy beyond the capability of strapdown technology. There are inertial navigation system applications for which strapdown systems are considered inadequate, such as the navigation system for manned fighter aircraft, but there are also applications in which strapdown systems are currently being used, such as the back up guidance and navigation system for the Apollo Lunar Module (ref. 3). Further, in at least one application, a strapdown system gives better performance than a gimballed system. When a spacecraft must be stabilized in orientation as, for example, to point a laser beam toward a receiving station on Earth, the entire vehicle becomes the stable member, and the distinction between a gimballed system and a strapdown system vanishes. Vehicle angular motions can be nulled by signals from the gyros in either class of system. If a gimballed system were used, uncertainty in the measurement of gimbal angles would contribute to uncertainty in orientation of the vehicle.

The techniques presented in this thesis will permit further applications for strapdown systems since, when these techniques are used, coordinate transformation computation is not a major limitation in strapdown system technology.

1.2 Coordinate Frames and the Transformation Computation

An explanation is in order concerning the particular choice and role of the various coordinate frames in this thesis. There always exists an inertial coordinate frame (either explicit or implied) in any inertial navigation process since all measurements of specific force and angular velocity are made relative to an inertial coordinate frame. An inertial coordinate frame is a coordinate frame in which a particle in motion in any arbitrary direction, but with no external forces acting upon it, continues in motion with a constant velocity vector. Any accelerometer, whether mounted directly on the vehicle or on a stable platform, measures an inertial quantity whose instantaneous value is the same as it would be if that sensor were fixed in orientation relative to an inertial coordinate frame. If a specific inertial coordinate frame is chosen for an inertial navigation problem, that frame is called the Inertial Frame.

Another coordinate frame that is always involved in the navigation process is a coordinate frame fixed to the vehicle structure. This coordinate frame is called the Body Frame. Other coordinate frames may be introduced as convenient or necessary. For example, in a terrestrial navigation system, it is convenient to introduce a local coordinate frame known as the Navigation Frame. The origin of this coordinate frame

is fixed in the vehicle with one axis along the vertical, another axis orthogonal to the vertical axis and in the plane defined by the vertical and the Earth's rotation axis, and the third axis taken to complete a right handed orthogonal triad. Other coordinate frames are treated in Broxmeyer (ref. 4).

In a strapdown inertial navigation system, there is no gimbal to be maintained in alignment with a reference coordinate frame, so the orientation of a reference frame is maintained in the navigation computer as a mathematical coordinate transformation. The coordinate transformation commonly takes the form of a direction cosine matrix C^{RB} which transforms a vector from its representation in the Body Frame into its equivalent representation in the Reference Frame. The direction cosine matrix is not the only form that the coordinate transformation may take. There are other characterizations of relative orientation such as Euler Angles and Quaternions. For a discussion of these other representations, see references (5) and (6). Except for the singularities in other representations, the direction cosine matrix corresponding to any other representation is unique although the converse is not necessarily true. There may be computational advantages to the use of other representations, but it is not the purpose of this thesis to explore this possibility, but rather to develop a hybrid computational technique that makes use of both incremental gyro output pulses and analog information extracted from the gyros for use in supplementary analog computations. This technique is then compared with representative

digital techniques.

The Body Frame may have arbitrary angular motion relative to the Reference Frame, and the gyros in a strapdown navigation system must measure this motion. The direction cosine matrix is "updated" from these measurements. The conventional method for updating the direction cosine matrix is to integrate numerically the matrix differential equation*

$$\dot{C}^{RB} = C^{RB} [\underline{\omega}^{B}_{RB} x] \tag{1.1}$$

subject to the initial condition

$$C^{RB}(t_o) = C^{RB}_{O} \tag{1.2}$$

where $[\underline{\omega}^{B}_{RB} x]$ is the skew symmetric cross-product matrix formed from the Body Frame components of $\underline{\omega}_{RB}$. It is assumed throughout that the Reference Frame is an inertial coordinate frame although it need only be related to an inertial coordinate frame through another coordinate transformation which is a function of the vehicle's position. In that case, C^{IB} is updated using the gyro measurements and then premultiplied by C^{RI} as evaluated from the vehicle's position coordinates. Then

$$C^{RB} = C^{RI} C^{IB}$$

gives the desired coordinate transformation. Alternatively, pulses representing incremental changes in C^{RI} might be appropriately generated and summed with the gyro pulses representing incremental changes in C^{IB}. This summation of pulses then

*Notation conventions appear in Appendix F.

updates C^{RB} using a suitable digital algorithm.

In order to integrate Eq. (1.1) on a digital computer, the gyros are sampled periodically to get increments of integrated angular velocity from which $\underline{\omega}_{RB}$ can be inferred. As already stated, the fundamental problem in integrating Eq.(1.1) is the noncommutativity of finite rotations. The only case in which the gyro increments, $\Delta\theta_x$, $\Delta\theta_y$, and $\Delta\theta_z$, give a true picture during the interval ΔT is when the direction of $\underline{\omega}_{RB}$ is constant and the $\Delta\theta$'s accurately reflect that direction. For example, suppose $\omega_x = \omega_y = \omega_z$. If at the start of a gyro sampling interval, the x- and the y-gyro floats are close to their thresholds and the z-gyro float is not, the output at the end of the sampling interval might well be a pulse from each of the x- and the y-gyros and no pulse from the z-gyro. Then

$$\underline{l}_{\omega} = \frac{\underline{l}_x\omega_x + \underline{l}_y\omega_y + \underline{l}_z\omega_z}{(\omega_x^2+\omega_y^2+\omega_z^2)^{1/2}}$$

but

$$\underline{l}_{\Delta\theta} = \frac{\underline{l}_x\Delta\theta_x + \underline{l}_y\Delta\theta_y}{(\Delta\theta_x^2+\Delta\theta_y^2)^{1/2}}$$

Obviously

$$\underline{l}_{\omega} \neq \underline{l}_{\Delta\theta}$$

and so the update of the coordinate transformation will be in error. In any case other than

$$\underline{l}_{\omega} = \underline{l}_{\Delta\theta} = \text{constant}$$

over the sampling interval, information is irretrievably lost

in the integration and quantization operations performed by the gyros. It is accurate to say that except where $\underline{\omega}_{RB}$ is constant in orientation or except in the limit as $t \to t_o$, the quantity

$$\underline{\theta}_{RB}(t) = \int_{t_o}^{t} \underline{\omega}_{RB}(\tau)\, d\tau \tag{1.3}$$

is a physically meaningless quantity.

In the hybrid mechanization, a correction $\underline{\sigma}_{RB}(t)$ is generated and added to $\underline{\theta}_{RB}(t)$ so that the resulting vector

$$\underline{\phi}_{RB}(t) = \underline{\theta}_{RB}(t) + \underline{\sigma}_{RB}(t) \tag{1.4}$$

has the physical significance of being the equivalent single axis rotation which takes a coordinate frame from coincidence with the Reference Frame at time t into coincidence with the Body Frame. The existence of this equivalent single axis rotation $\underline{\phi}_{RB}$ is guaranteed by a theorem of Euler which states that the most general displacement of a rigid body with one point fixed is a rotation about an axis through the fixed point (ref. 7). The coordinate transformation $C^{RB}(t)$ can be evaluated as a matrix function of the equivalent rotation $\underline{\phi}_{RB}(t)$.

Digital algorithms for numerically integrating Eq. (1.1) try to overcome the noncommutativity of finite rotations. The simpler algorithms achieve their accuracy by updating at high repetition rates using low gyro quantization levels. The more sophisticated algorithms achieve their accuracy by processing the $\Delta\theta$'s obtained over many gyro rebalance intervals as to reconstruct a better history of $\underline{\omega}_{RB}$ than is possible by assuming that $\underline{\omega}_{RB} = \Delta\theta/\Delta T$. In each case, the accuracy of the

update is traded off against the computation burden which the algorithm imposes. These algorithms were so demanding of computer capacity that strapdown navigation systems had to wait a decade longer than gimballed systems to become practical since computer technology during that decade was inadequate for updating the strapdown coordinate transformation.

Only rebalanced gyros are treated in this thesis. In an Electrostatically Suspended Gyro (ESG) for example, the gyro is not torqued, so there are no errors which grow with time because of the computation of the coordinate transformation from incremental data. Instead, the orientation of the angular momentum vector in an ESG relative to the Body Frame is read periodically and the coordinate transformation is evaluated from the readings of at least two ESG's. Such a system has its own set of error mechanisms, but the coordinate transformation computation error is not one of the more significant. Attention will be addressed to systems using one or two degree-of-freedom electrically rebalanced integrating gyros. The context is that of pulse rebalanced gyros, but a set of linearly rebalanced gyros followed by quantizers could be used if of sufficiently high quality.

Systems with redundant inertial measurement capability are not considered here. The problems of redundancy are essentially uncoupled from the problems of transforming specific force and updating the coordinate transformation. Finally, the problem of compensation of inertial sensors for their dynamic errors is not considered. It is true, however, that more effective compensation can be achieved as a by-product of the technique

of deriving analog angular velocity signals from pulse rebalanced gyros.

CHAPTER 2

THE HYBRID CONCEPT

2.1 Introduction

The universal practice of using pure digital computation to update the coordinate transformation has led to an implicit understanding of the strapdown gyro as an inherently digital transducer. For the conventional sampled-data rate extraction and digital processing techniques, it is only required that at a sampling instant, the gyro yield a positive, negative, or zero incremental output. This notion of the gyro and its capabilities is, however, not complete. A better understanding and use of the strapdown gyro can lead to substantial improvements in strapdown system technology. It must be recognized that the strapdown gyro is inherently an analog transducer. It is only the employment of a digital (pulse) rebalance loop that renders the gyro output in incremental form. From this realization, there follows the possibility of extracting useful continuous information from strapdown gyros.

To date, there has been no attempt to use any computational technique except purely digital techniques to maintain the strapdown coordinate transformation. Digital computation methods have advantages, but they have disadvantages as well. The same

can be said for analog computation methods. The purpose of hybrid computation is to combine analog and digital computational methods so as to use the advantages of one method to overcome the disadvantages of the other. This philosophy will be applied to the strapdown coordinate transformation problem.

The main advantage of digital computation is its arbitrarily high precision. Since only one computational operation can be performed at a time, each step in integrating the direction cosine differential equation must be performed serially. Thus there is a definite limit to the repetition rate at which any algorithm can be applied in a specific computer. Since other computations must also be carried out in the strapdown navigation system computer, the bandwidth of the angular velocity environment that can be tracked must be weighed against the computer size and speed requirements of the overall computational load.

The principal advantage of analog computation is that any number of operations can be performed simultaneously. Therefore, the bandwidth of a computational algorithm is the bandwidth of the system as modeled by the analog computer – not the bandwidth that would result from cascading each computing element, as is done in effect on a digital computer. It is practical to obtain a direction cosine computation whose bandwidth exceeds that of the gyros by at least an order of magnitude using analog computation since the analog computer is an electronic device while the gyros are electromechanical in nature. On the other hand, the accuracy and precision of the

analog computation is limited. An analog computing element with an accuracy of 0.01 percent is quite good indeed.

To summarize, the precision of digital computation can be made as high as desired, but the bandwidth of the computation is limited by the overall load. Analog computation bandwidth, relative to gyro bandwidth, is no problem for an analog computer, but the accuracy and precision of the overall computation is limited by the cascaded errors of the individual computing elements to around 0.1 percent (ref. 8). Obviously, analog and digital computers have complementary strengths and weaknesses.

Simulation and computation problems exist that are unsuited for either digital computation or analog computation alone. An aircraft research simulation is an example of such a problem. Also, the strapdown coordinate transformation computation is only marginally suited to all digital computation, and since it has the characteristics listed below, it is natural to explore the possibilities of hybrid computation for this application. Problems suited for hybrid computation have some or all of the following features (ref. 9):

(a) The problem is complex.

(b) The problem must be solved in real time.

(c) Certain parts of the problem require relatively high accuracy.

(d) Certain parts of the problem require relatively high bandwidth.

(e) Both analog and digital information are available.

Analog information may be available naturally as a result of a continuous physical measurement. Digital information results from sampling physical processes. Conversions from one form to the other are possible although not always desirable. For example, the numerical differentiation followed by a digital to analog conversion of the $\Delta\theta$ pulse train from a gyro may be used to obtain an analog representation of angular velocity, but this is inferior to deriving an analog signal directly from the gyro itself, because unacceptable time lags might be introduced in the process of smoothing the results of the numerical differentiation.

(f) The problem is capable of convenient division into parts structured for analog computation and parts for digital computation. The search for prime numbers, say, could not reasonably be structured for analog computation. The real time integration of a complicated system of transcendental differential equations might be difficult to do on a digital computer. This problem is easily structured for hybrid computation while the prime number problem is not.

Effective hybrid computation requires that there be relatively few analog to digital and digital to analog conversion links. The portions of the problem set aside for digital computation and those set aside for analog computation should be relatively uncoupled. Otherwise, problem complexity is aggravated rather than diminished by hybrid computation.

2.2 The Hybrid Strapdown System

The conventional method for maintaining the current value of the coordinate transformation is to numerically integrate the matrix differential equation

$$\dot{C}^{RB}(t) = C^{RB}(t)\ [\underline{\omega}^{B}_{RB}(t)x]$$

subject to the initial condition

$$C^{RB}(t_o) = C^{RB}_o.$$

In the hybrid method, C^{RB} is evaluated as a matrix function of the vector argument $\underline{\phi}_{RB}$ where $\underline{\phi}_{RB}(t)$ is defined as that rotation required to take a coordinate frame from coincidence with the Reference Frame into coincidence with the Body Frame at time t. The differential equation for $\underline{\phi}_{RB}(t)$ is developed in the next chapter. Anticipating this development, one form of the result is

$$\dot{\underline{\phi}}_{RB} = \underline{\omega}_{RB} + \frac{1}{2}\,\underline{\phi}_{RB} \times \underline{\omega}_{RB} + A(\phi_{RB})\underline{\phi}_{RB} \times (\underline{\phi}_{RB}\times\underline{\omega}_{RB}) \tag{2.1}$$

subject to the initial condition

$$\underline{\phi}_{RB}(t_o) = \underline{0} \tag{2.2}$$

where

$$A(\phi_{RB}) = \frac{1}{\phi^2_{RB}}\left(1 - \frac{\phi_{RB}\ \sin\ \phi_{RB}}{2(1-\cos\ \phi_{RB})}\right) \tag{2.3}$$

In Chapter 3, it is also shown that

$$C^{RB} = I + q(\phi_{RB})[\underline{\phi}_{RB}x] + \frac{q^2(\phi_{RB})}{1 + \cos\ \phi_{RB}}\,[\underline{\phi}_{RB}x]^2 \tag{2.4}$$

where

$$q(\phi_{RB}) = \frac{\sin \phi_{RB}}{\phi_{RB}} \tag{2.5}$$

Eq. (2.1) is integrated subject to Eq. (2.3), yielding the current value of $\underline{\phi}_{RB}(t)$. Then $C^{RB}(t)$ is obtained by inserting this value into the right-hand side of Eq. (2.4) which is a matrix function of $\underline{\phi}_{RB}(t)$ only.

A decision is made as to what parts of Eq. (2.1) and Eq. (2.4) are to be solved using analog computing elements and what parts are to be solved on a digital computer. The criteria for the decision are:

(a) The accuracy of the overall coordinate transformation will not be less than if the computation were done entirely on a digital computer using some specified algorithm.

(b) Any part of the computation that can be done using simple analog circuitry will be done that way subject to criterion (a).

The algorithm to which criterion (a) refers will not be specified now, but in Chapter 6, where the hybrid method is compared with purely digital methods, specific choices of algorithms will be made.

In theory, by Shannon's sampling theorem (ref. 10), the non-commutativity error can be overcome to any desired degree using purely digital computation for any input angular motion whose spectrum does not contain frequencies exceeding one-half the gyro sampling frequency. The computational load, however, to approach this theoretical bandwidth limit is entirely out of the question.

The treatment of Eq. (2.4) will be settled first. Assume that a whole number digital representation is available for each component of $\underline{\phi}_{RB}$, and that each component is accurate to, say, five significant figures. Any error in transforming the specific force measurement is as serious as an error in the measurement, thus so if the specific force measurement is accurate to five significant figures as is reasonable in a good quality navigation system, then an analog computer accuracy of three significant figures in the evaluation of $C^{RB}(\underline{\phi}_{RB})$ would not suffice. Hence the evaluation of Eq. (2.4) must be done by a digital computer in order to satisfy criterion (a).

Next a sufficiently accurate whole number representation of $\underline{\phi}_{RB}$ must be obtained. If $\Delta\underline{\phi}_{RB}$ increments were available with the same accuracy as the $\Delta\theta$ increments delivered by the gyro, these increments could be counted in the digital computer to obtain $\underline{\phi}_{RB}$. To obtain $\Delta\underline{\phi}_{RB}$, Eq. (2.1) must be solved in the proper incremental form. Eq. (2.1) may be rewritten as

$$\dot{\underline{\phi}}_{RB} = \underline{\omega}_{RB} + \dot{\underline{\sigma}}_{RB} \tag{2.6}$$

where

$$\dot{\underline{\sigma}}_{RB} = \frac{1}{2}\,\underline{\phi}_{PB} \times \underline{\omega}_{RB} + A(\phi_{RB})\,\underline{\phi}_{RB} \times (\underline{\phi}_{RB} \times \underline{\omega}_{RB}). \tag{2.7}$$

Each gyro in the orthogonal triad is made to integrate and quantize that component of the right hand side of Eq.(2.6) which is parallel to its input axis, thus solving Eq. (2.1) in the desired form.

To see how this is done, consider the performance of a strapdown single degree-of-freedom, electrically rebalanced, integrating gyro (ref. 1).

$$(Is^2+Cs)A = H(\omega_{IA}-\omega_{fb}+\omega_{ext}) + N(\underline{\omega},\underline{f},t) \qquad (2.8)$$

where

I = gyro float inertia about OA (output axis)

C = gyro damping constant about OA

s = time differentiation operator

A = gyro float angle about OA

H = gyro spin angular momentum

ω_{IA} = input axis angular velocity

$H\omega_{fb}$ = a feedback torque applied about OA tending to null A

$N(\underline{\omega},\underline{f},t)$ = a torque function of angular velocity, specific force, and time embodying all the non-ideal performance features of the gyro

$H\omega_{ext}$ = any other non-inertial torque applied about OA

Since $H\omega_{fb}$ is the rebalance torque tending to keep float angle A near null, its time integral will be equal and opposite (except for quantization error in a pulse rebalanced gyro) to the time integral of the sum of all other torques acting on the float.

$$\int_{t_o}^{t_f} H\omega_{fb}\, dt = \int_{t_o}^{t_f} [H(\omega_{IA}+\omega_{ext})+N]\, dt$$

In a pulse rebalanced gyro, the quantized integral of $H\omega_{fb}$ is available as the sum of the rebalance pulses multiplied by the weight of each pulse.

$$\frac{\Delta\theta}{H}(n_+ - n_-) = \int_{t_o}^{t_f} \left(\omega_{IA} + \omega_{ext} + \frac{N}{H}\right) dt + e(q) \qquad (2.9)$$

where $\Delta\theta$ is pulse weight (radians/pulse)

n_+ is the total number of positive pulses on (t_o, t_f)

n_- is the total number of negative pulses on (t_o, t_f)

$e(q)$ is quantization error (information stored in the gyro)

If $H\omega_{ext}$ is taken to be

$$H\omega_{ext} = H\dot{\sigma} - N_{comp}(\underline{\omega}, \underline{f}, t) \qquad (2.10)$$

where σ is a component of $\underline{\sigma}$ defined by Eq. (2.6) N_{comp} is an electrically applied compensation for $N(\underline{\omega}, \underline{f}, t)$ then Eq. (2.8) becomes

$$(Is^2 + Cs)A = H(\omega_{IA} + \dot{\sigma} - \omega_{fb}) - \delta N(\underline{\omega}, \underline{f}, t) \qquad (2.11)$$

where

$$\delta N(\underline{\omega}, \underline{f}, t) = N_{comp}(\underline{\omega}, \underline{f}, t) - N(\underline{\omega}, \underline{f}, t)$$

Neglecting $\delta(\underline{\omega}, \underline{f}, t)$ and using Eq. (2.10) in Eq. (2.9), the result is

$$\int_{t_o}^{t_f} (\omega_{IA} + \dot{\sigma})\, dt = \frac{\Delta\theta}{H}(n_+ - n_-) + e(q) \qquad (2.12)$$

Since $\omega_{IA} + \dot{\sigma}$ may be written as

$$\omega_{IA} + \dot{\sigma} = \underline{l}_{IA} \cdot (\underline{\omega} + \dot{\underline{\sigma}}) \qquad (2.13)$$

Eqs. (2.10) - (2.13) show that by applying an electrical torque proportion to $\underline{l}_{IA} \cdot \dot{\underline{\sigma}}$, the gyro triad can be made to integrate

and quantize Eq. (2.6).

The generation of $\dot{\underline{\sigma}}_{RB}$ can be accomplished using analog circuitry. If $\dot{\underline{\sigma}}_{RB}$ is a term, small in magnitude compared with the maximum value of $|\underline{\omega}_{RB}|$, then the three significant figure accuracy of analog computing elements will suffice. (A system error analysis is found in Chapter 5.)

The hybrid system, which is shown symbolically in Figure 2.1, requires (a) a set of filters which extract as continuous signals the components of $\underline{\omega}_{RB}$ (b) a set of multipliers and summers to take the vector cross products required in Eq. (2.7), (c) a set of integrators to integrate the components of $\dot{\underline{\phi}}_{RB}$ (note that the only purpose of this vector integration is to obtain the vector $\underline{\phi}_{RB}$ for use in generating the cross product terms of $\dot{\underline{\sigma}}_{RB}$), (d) other miscellaneous circuitry which will be explained when introduced (Chapter 5).

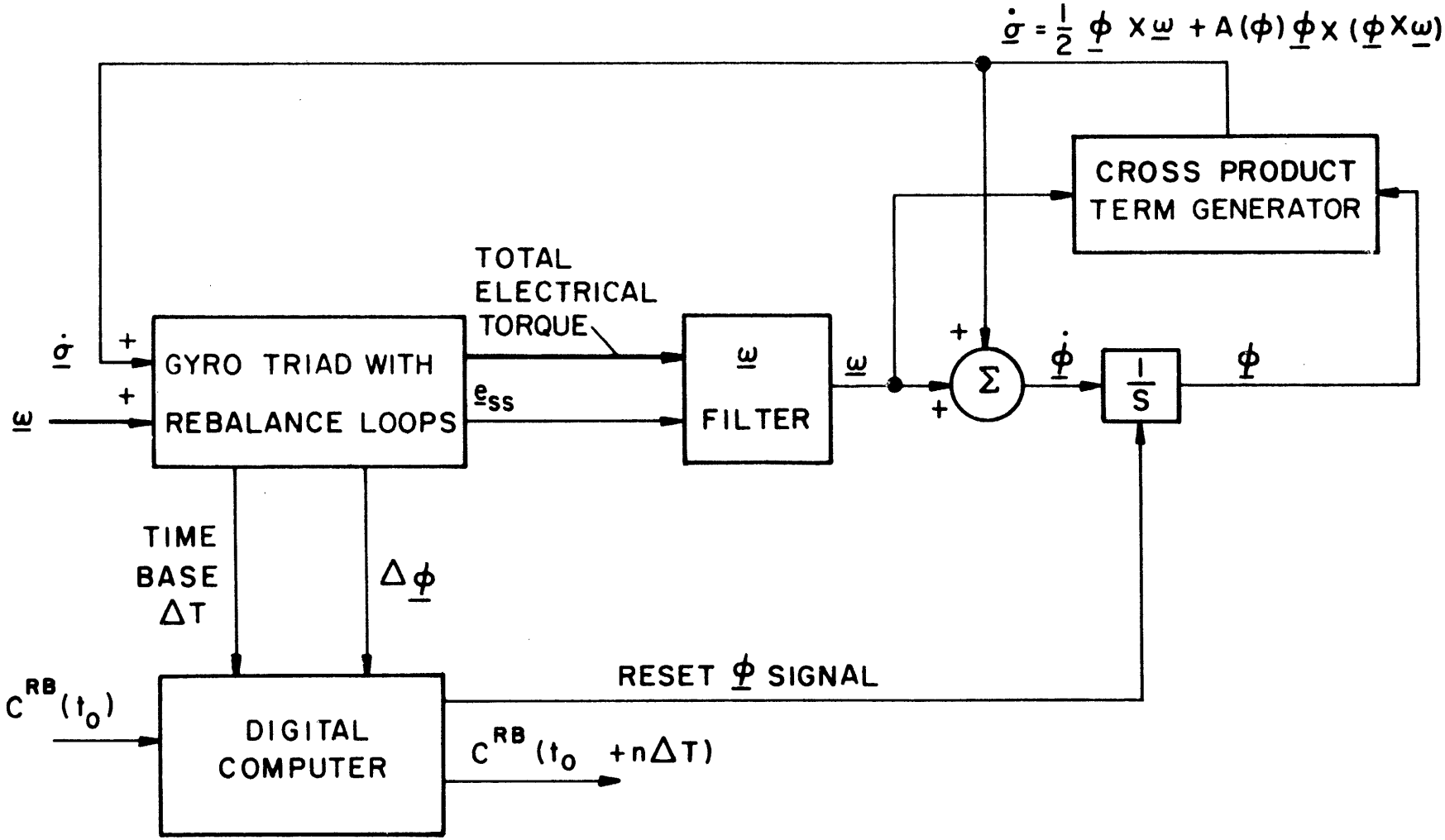


Figure 2.1.- Symbolic system diagram

CHAPTER 3

THE DYNAMICS OF FINITE ROTATIONS

3.1 A Vector Concept of Rotations

In the field of mechanics, it has long been held that finite rotations of one coordinate frame relative to another are not true vectors. In support of this, it is said that finite rotations do not commute; that is, the orientation resulting from taking rotation A and then rotation B is not, in general, the same as the orientation resulting from taking rotation B and then rotation A. Here the matter has rested. With the advent of strapdown navigation systems, which endure the rotational as well as translational aspects of general rigid body motion, new understanding is required of the dynamics of finite rotations. A more sophisticated coordinate transformation algorithm is not the answer to the computation problems that arise. Greater insight into the dynamics of finite rotations is necessary if any significant progress is to be made in reducing the complexity of the coordinate transformation computation.

What is a finite, rigid body rotation? In a physical sense, it is a change in the orientation of one coordinate frame relative to another. By Euler's theorem (ref. 7),

there corresponds to any rotation, an axis of rotation and a magnitude of rotation. Physically speaking, a quantity which has direction and magnitude is a vector. Hence, the following definition is made.

Definition: Let the Reference Frame and the Body Frame be two coordinate frames whose origins coincide. The rotation vector $\underline{\phi}_{RB}$ is defined to be that vector whose direction is parallel to the axis of rotation of the Body Frame with respect to the Reference Frame, and whose magnitude is equal to the angle through which a coordinate frame coincident with the Reference Frame must be rotated about the axis in order to be brought into coincidence with the Body Frame.

Three questions will be answered in this chapter.

(a) Is the rotation vector unique?

(b) What is the relationship between the coordinate transformation C^{RB} and the rotation vector $\underline{\phi}_{RB}$?

(c) What are the laws of addition for rotation vectors and what is the relationship between angular velocity $\underline{\omega}_{RB}$ and the time rate of change $\underline{\dot{\phi}}_{RB}$ of the rotation vector?

The question regarding the uniqueness of the rotation vector can be answered at once on an intuitive basis. (It will be answered rigorously in the next section.) Assume that the Body Frame is initially coincident with the Reference Frame. If the Body Frame is rotated through any integer multiple of 2π radians about any fixed axis, it is again coincident with the Reference Frame. In a more general sense, if the Body Frame is displaced from coincidence with the Reference Frame by some general rotation $\underline{\phi}_{RB}$, then a rotation through $\phi_{RB} + 2n\pi$

radians about $\underline{1}_\phi$ results in the same final displacement

$$\underline{\phi}_{RB} = \underline{\phi}_{RB}\underline{1}_\phi = (\phi_{RB}+2n\pi)\underline{1}_\phi \tag{3.1}$$

Two special cases are worthy of note. The first is the case for which $\phi_{RB} = 2\pi$. Then

$$\underline{\phi}_{RB} = (2\pi+2n\pi)\underline{1}_\phi$$

Since n may be taken to be -1,

$$\underline{\phi}_{RB} = (2\pi-2\pi)\ \underline{1}_\phi = 0\ \underline{1}_\phi = \underline{0}$$

and the axis of rotation is indeterminate. The second special case is the case for which $\phi_{RB} = \pi$. Then

$$\underline{\phi}_{RB} = \pi\underline{1}_\phi$$

Also

$$\underline{\phi}_{RB} = (\pi-2\pi)\underline{1}_\phi = (-\pi)\underline{1}_\phi$$
$$= \pi(-\underline{1}_\phi)$$

By comparing Eqs. (3.2) and (3.3), it is seen that a rotation of π radians about some axis $\underline{1}_\phi$ may be taken in either direction about that axis with the same result.

Thus it is seen that the rotation vector is not unique. To any orientation there corresponds infinitely many rotation vectors (a one fold infinity for $\phi_{RB} \neq 2n\pi$ and a threefold infinity for $\phi_{PB} = 2n\pi$ where n is any integer). The converse is not true, however. In the next section, it will be seen that C^{RB} is a single valued matrix function of $\underline{\phi}_{RB}$.

3.2 The Relationship Between $\underline{\phi}_{RB}$ and C^{RB}

In this section C^{RB} will be derived as a unique matrix function of the rotation vector $\underline{\phi}_{RB}$ where C^{RB} is the coordinate transformation matrix that transforms a vector from a

Body Frame representation into an equivalent Reference Frame representation.

One method for deriving C^{RB} in terms of $\underline{\phi}_{RB}$ is to note that if

$$\underline{\omega}_{RB} = \omega \underline{1}_k \tag{3.4}$$

where $\underline{1}_k$ is a unit vector whose orientation is fixed, then

$$C^{RB}(t) = \exp\left\{(t-t_o)[\omega^B_{RB}x]\right\} \tag{3.5}$$

satisfies the differential equation

$$\dot{C}^{RB} = C^{RB}[\omega^B_{RB}x] \tag{3.6}$$

subject to the initial condition

$$C^{RB}(t_o) = I \tag{3.7}$$

This can be verified by substituting Eq. (3.5) into (3.6). Unfortunately, Eq. (3.6) has a solution in the form of Eq. (3.5) only for the case where $\underline{\omega}_{RB}$ is the zero vector or is constant in direction. This results from the noncommutativity of finite rotations. The right hand side of Eq. (3.5) may be evaluated using the Cayley-Sylvester theorem for matrices (ref. 11). The result is

$$C^{RB} = \frac{1}{\phi^2}\begin{bmatrix} \phi_x^2 + (\phi^2-\phi_x^2)\cos\phi & \phi_x\phi_y(1-\cos\phi) - \phi\phi_z\sin\phi & \phi_x\phi_z(1-\cos\phi) + \phi\phi_y\sin\phi \\ \phi_y\phi_x(1-\cos\phi) + \phi\phi_z\sin\phi & \phi_y^2 + (\phi^2-\phi_y^2)\cos\phi & \phi_y\phi_z(1-\cos\phi) - \phi\phi_x\sin\phi \\ \phi_z\phi_x(1-\cos\phi) - \phi\phi_y\sin\phi & \phi_z\phi_y(1-\cos\phi) + \phi\phi_x\sin\phi & \phi_z^2 + (\phi^2-\phi_z^2)\cos\phi \end{bmatrix} \tag{3.8}$$

where

$$\underline{\phi} = (t-t_o)\underline{\omega}_{RB}$$

$$\phi^2 = \underline{\phi}^T\underline{\phi}$$

Eq. (3.8) can be written more compactly as

$$C^{RB} = \frac{1}{\phi^2}\left\{\underline{\phi}\underline{\phi}^T(1-\cos\phi) + \phi^2 I\cos\phi + \phi^2\frac{\sin\phi}{\phi}[\underline{\phi}\times]\right\} \tag{3.9}$$

Note: When the vector $\underline{\phi}$ is used as the argument of the coordinate transformation C^{PQ}, it is understood (unless otherwise stated) to mean $\underline{\phi}_{PQ}$.

Equation (3.9) can also be derived from purely geometrical arguments. This will afford better insight into the relationship between $\underline{\phi}_{RB}$ and C^{RB} because angular velocity is not introduced, and so there is no restriction that the direction of $\underline{\omega}_{RB}$ be fixed.

Let $\underline{r}^B$ be an arbitrary vector fixed in the Body Frame. Suppose that at $t = t_o$, $C^{RB} = I$. Then

$$\underline{r}_o^B = \underline{r}_o^R$$

Then let the Body Frame rotate with respect to the Reference Frame through the angle ϕ about a Body Frame axis $\underline{u}$. At time t, a new vector is defined by

$$\underline{r}^R(t) = C^{RB}(t)\ \underline{r}^B(t)$$

or since $\underline{r}^B = \underline{r}_o^B$ is fixed in the Body Frame,

$$\underline{r}^R(t) = C^{RB}(t)\ \underline{r}_o^B \tag{3.10}$$

These relationships are shown in Figure 3.1.

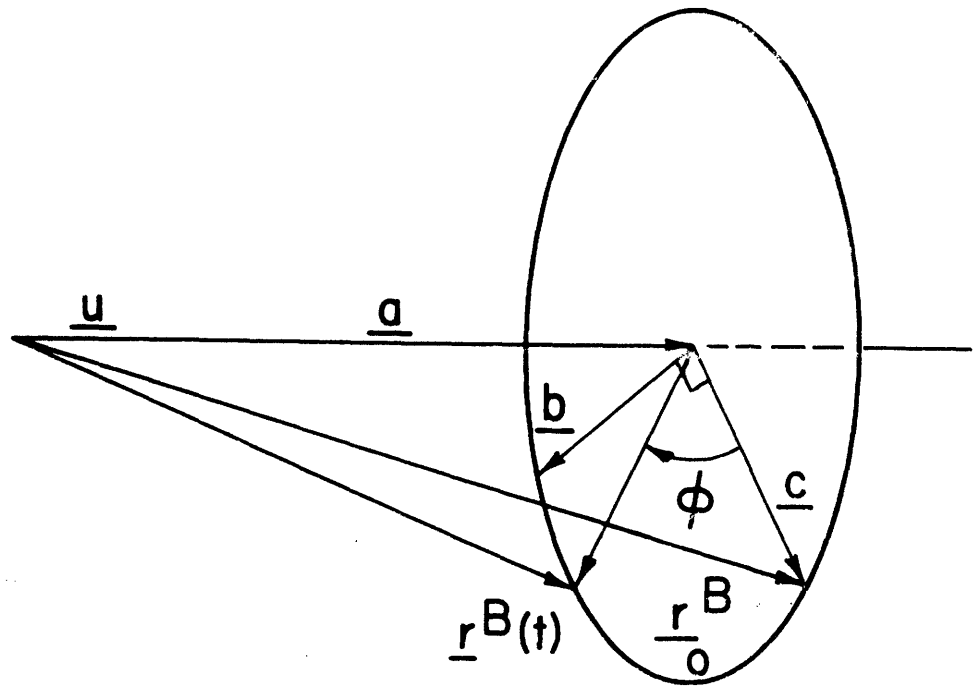


Figure 3.1.- Rotation vector geometry

The relationships used in constructing the figures are

$$\underline{u} \cdot \underline{u} = 1 \qquad \underline{a} = (\underline{r}_o^B \cdot \underline{u})\, \underline{u}$$

$$\underline{u} \cdot \underline{b} = 0 \qquad \underline{b} = \underline{u} \times \underline{r}_o^B$$

$$\underline{u} \cdot \underline{c} = 0 \qquad \underline{c} = \underline{b} \times \underline{u} = \underline{u} \times (\underline{r}_o^B \times \underline{u})$$

$$\underline{b} \cdot \underline{c} = 0$$

In the matrix notation, $\underline{a}$, $\underline{b}$, and $\underline{c}$ become

$$\left.\begin{aligned} \underline{a} &= \underline{u}\underline{u}^T \underline{r}_o^B \\ \underline{b} &= [\underline{u} \times] \underline{r}_o^B \\ \underline{c} &= \underline{r}_o^B - \underline{u}\underline{u}^T \underline{r}_o^B \end{aligned}\right\} \qquad (3.11)$$

From Figure 3.1 it is seen that

$$\underline{r}^R(t) = \underline{a} + \underline{b} \sin \phi + \underline{c} \cos \phi \qquad (3.12)$$

Eqs. (3.10) - (3.12) can be combined to get

$$C^{RB} \underline{r}_o^B = \left\{ \underline{u}\underline{u}^T (1 - \cos \phi) + I \cos \phi + [\underline{u} \times] \sin \phi \right\} \underline{r}_o^B \qquad (3.13)$$

But since $\underline{r}_o^B$ is an arbitrary vector, it follows that

$$C^{RB} = \underline{u}\underline{u}^T (1 - \cos \phi) + I \cos \phi + [\underline{u} \times] \sin \phi \qquad (3.14)$$

Now let

$$\underline{\phi} = \phi \underline{u}$$

Then Eq. (3.1) becomes

$$C^{RB} = \frac{\underline{\phi}\underline{\phi}^T}{\phi^2} (1 - \cos \phi) + I \cos \phi + \frac{\sin \phi}{\phi} [\underline{\phi} \times] \qquad (3.15)$$

which is in agreement with Eq. (3.9).

Other forms of Eq. (3.15) may be readily derived. By analogy with the vector identity

$$\underline{\phi} \times (\underline{\phi} \times \underline{v}) \equiv \underline{\phi}(\underline{\phi} \cdot \underline{v}) - \phi^2 \underline{v}$$

one may write the matrix identity

$$[\underline{\phi} \times][\underline{\phi} \times]\underline{v} \equiv [\underline{\phi} \times]^2 \underline{v} = (\underline{\phi}\underline{\phi}^T - \phi^2 I)\underline{v}$$

and by direct expansion, it can be verified that

$$\frac{\underline{\phi}\underline{\phi}^T}{\phi^2} \equiv \frac{[\underline{\phi} \times]^2}{\phi^2} + I \tag{3.16}$$

Substitution of Eq. (3.16) into Eq. (3.15) gives

$$C^{RB} = I + \frac{\sin \phi}{\phi} [\underline{\phi} \times] + \frac{1 - \cos \phi}{\phi^2} [\underline{\phi} \times]^2 \tag{3.17}$$

or its equivalent

$$C^{RB} = I + \frac{\sin \phi}{\phi} [\underline{\phi} \times] + \frac{1}{1 + \cos \phi} \left(\frac{\sin \phi}{\phi} [\underline{\phi} \times] \right)^2 \tag{3.18}$$

Since $\underline{\phi}_{PB} \times \underline{\phi}_{PB} = \underline{0}$, either Eq. (3.17) or Eq. (3.18) may be used to show that

$$\underline{\phi}^R_{RB} = C^{RB} \underline{\phi}^B_{RB} = I \underline{\phi}^B_{RB} = \underline{\phi}^B_{RB} \tag{3.19}$$

Eq. (3.19) demonstrates that a vector parallel to the axis of rotation is not changed by the rotation.

It follows immediately from Eq. (3.14) that

$$C^{RB}[\phi_{RB} \underline{1}_\phi] = C^{RB}[(\phi_{RB} + 2n\pi) \underline{1}_\phi] \tag{3.20}$$

Eq. (3.20) demonstrates the non-uniqueness of $\underline{\phi}_{RB}$ in describing relative orientation.

Finally, it will be shown that

$$(C^{RB})^T = (C^{RB})^{-1} \tag{3.21}$$

in order to demonstrate that the direction cosine matrix as derived geometrically is indeed an orthogonal matrix. Note that if the rotation $\underline{\phi}_{RB}$ generates the coordinate transformation C^{RB}, then the rotation $-\underline{\phi}_{RB}$, which would return the Body Frame to coincidence with the Reference Frame, must generate the inverse transformation. That is

$$C^{RB}(-\underline{\phi}_{RB}) = [C^{RB}(\underline{\phi}_{RB})]^{-1} \tag{3.22}$$

Direct substitution in Eq. (3.17) shows that

$$C^{RB}(-\underline{\phi}_{RB}) = [C^{RB}(\underline{\phi}_{RB})]^{T} \tag{3.23}$$

Combining Eqs. (3.22) and (3.23) establishes Eq. (3.21) and the orthogonality of C^{RB}. It is natural to define

$$C^{BR}(\underline{\phi}_{RB}) \equiv [C^{RB}(\underline{\phi}_{RB})]^{-1} = C^{RB}(-\underline{\phi}_{RB}) \tag{3.24}$$

3.3 The Theory of Rotation Vectors

It is well known (ref. 7) that rotation vectors do not obey the normal rules for vector addition. If the rotation $\underline{C}$ is desired such that it is equivalent to the combined effect of taking rotation $\underline{A}$ followed by rotation $\underline{B}$, then it is unfortunately not true in general that

$$\underline{C} = \underline{A} + \underline{B}$$

because of the coupling that exists between the rotations $\underline{A}$

and $\underline{B}$. A new operation is defined by Laning* which he calls "rotation sum" and denotes by the symbol "#" such that

$$\underline{C} = \underline{A} \# \underline{B}$$

This is read as "$\underline{C}$ is equal to $\underline{A}$ rotation summed with B".

A simple example serves to indicate the several aspects of the problem. Let the Body Frame and the Reference Frame be two coordinate frames that are coincident at the time $t = 0$. Let the Body Frame experience an angular velocity $\underline{\omega}_{RB}(t)$ with respect to the Reference Frame as follows:

$$\underline{\omega}^{B}_{RB}(t) = \begin{bmatrix} \pi/2 \\ 0 \\ 0 \end{bmatrix} \qquad 0 < t \leq 1$$

and

$$\underline{\omega}^{B}_{RB}(t) = \begin{bmatrix} 0 \\ \pi/2 \\ 0 \end{bmatrix} \qquad 1 < t \leq 2$$

The relative orientations of the Body Frame and the Reference Frame at times $t = 0$, $t = 1$, and $t = 2$ are shown in Figure 3.2. Let $\underline{A} = \underline{\phi}_{RB(1)}$, $\underline{B} = \underline{\phi}_{B(1)B(2)}$, and $\underline{C} = \underline{\phi}_{RB(2)}$. Then from Figure 3.2,

$$\underline{A} = \begin{bmatrix} \pi/2 \\ 0 \\ 0 \end{bmatrix} \qquad \underline{B} = \begin{bmatrix} 0 \\ \pi/2 \\ 0 \end{bmatrix}$$

and

---

*This operation and notation was introduced by Laning (ref.12). The parts of Laning's theory that pertain to this thesis are developed in Appendix A. There the rotation sum operation is derived in terms of elementary vector operations.

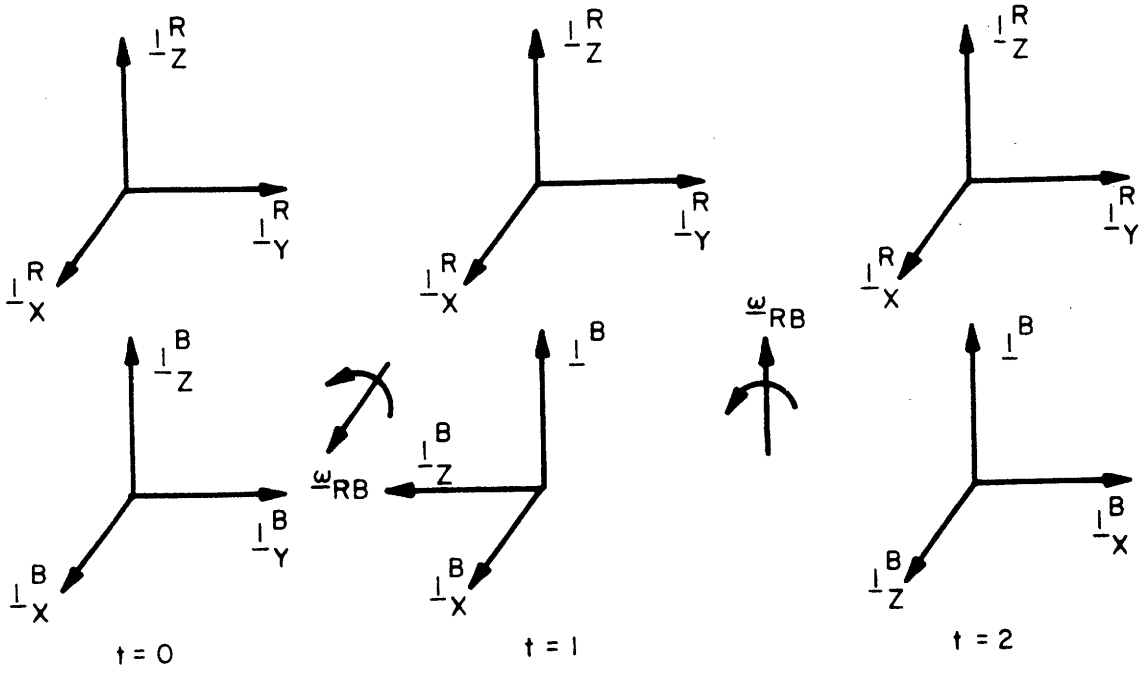


Figure 3.2.- An example of composite rotation

$$\underline{C} = \frac{1}{3}\,(2\pi)\begin{bmatrix} 1/\sqrt{3} \\ 1/\sqrt{3} \\ 1/\sqrt{3} \end{bmatrix}$$

So evidently

$$\underline{C} \neq \underline{A} + \underline{B}$$

Obviously

$$\dot{\underline{\phi}}_{RB}(t) = \underline{\omega}_{RB}(t) \qquad 0 < t \leq 1$$

but just as obviously

$$\dot{\underline{\phi}}_{RB}(t) \neq \underline{\omega}_{RB}(t) \qquad 1 < t \leq 2$$

since

$$\underline{\phi}_{RB}(2) \neq \underline{\phi}_{RB}(1) + \int_1^2 \underline{\omega}_{RB}(t)dt$$

This would appear to contradict the notion that $\underline{\omega}_{RB}(t)$ is the rate of change in the orientation of the Body Frame relative to the Reference Frame. The resolution to this apparent conflict is that the infinitesimal rotation $\underline{\omega}_{RB}(t)dt$ must be "rotation summed" with $\underline{\phi}_{RB}(t)$ in order to get $\underline{\phi}_{RB}(t+dt)$. Mathematically

$$\underline{\phi}_{RB}(t+dt) = \underline{\phi}_{RB}(t) \# \underline{\omega}_{RB}(t)dt$$

A more convenient formulation is to find that infinitesimal rotation $d\underline{\phi}_{RB}$ which when added to $\underline{\phi}_{RB}(t)$ by the normal laws of vector addition yields $\underline{\phi}_{RB}(t+dt)$. That is, an infinitesimal vector $d\underline{\phi}_{RB}$ is sought such that

$$\underline{\phi}_{RB}(t+dt) = \underline{\phi}_{RB}(t) + d\underline{\phi}_{RB} \tag{3.25}$$

It is anticipated that $d\underline{\phi}_{RB}$ will be a function of $\underline{\phi}_{RB}$ and $\underline{\omega}_{RB}dt$. An elegant and rigorous derivation of the differential equation

$$\dot{\underline{\phi}}_{RB} = \underline{f}(\underline{\phi}_{RB}, \underline{\omega}_{RB}) \tag{3.26}$$

due to Laning is presented in Appendix A. Eq. (3.26) will be derived in this section by a more intuitive approach.

The starting point for this derivation is Eq. (3.19)

$$C^{RB}\underline{\phi}_{RB} = \underline{\phi}_{RB}$$

Taking the time derivative of each side of Eq. (3.19) with respect to the Body Frame gives

$$\left(\frac{d}{dt_B} C^{RB}\right)\underline{\phi}_{RB} + C^{RB}\frac{d\underline{\phi}_{RB}}{dt_B} = \frac{d\underline{\phi}_{RB}}{dt_B} \tag{3.27}$$

where the notation d/dt_B indicates that the derivative is taken with respect to the Body Frame. Since Eq. (3.6) is a differential relationship taken with respect to the Body Frame, it can be used in Eq. (3.27) to get, after a slight rearrangement

$$C^{RB}[\underline{\omega}_{RB}\ x]\ \underline{\phi}_{RB} = (I-C^{RB})\frac{d\underline{\phi}_{RB}}{dt_B} \tag{3.28}$$

The notation d/dt_B will be dropped with the understanding that all time derivatives are taken with respect to the Body Frame unless otherwise stated. Premultiplication of each side of Eq. (3.28) by C^{BR} gives

$$[\underline{\omega}_{RB}\ x]\underline{\phi}_{RB} = (C^{BR}-I)\dot{\underline{\phi}}_{RB} \tag{3.29}$$

If the factor $C^{BR}-I$ is expanded by means of Eqs. (3.24) and (3.17) to get

$$C^{BR} - I = \frac{(1-\cos\phi_{RB})}{\phi_{RB}^2}[\phi_{RB}\ x]^2 - \frac{\sin\phi_{RB}}{\phi_{RB}}[\underline{\phi}_{RB}\ x]$$

$$= \left\{\frac{(1-\cos\phi_{RB})}{\phi_{RB}^2}[\underline{\phi}_{RB}\ x] - \frac{\sin\phi_{RB}}{\phi_{RB}}\right\}[\underline{\phi}_{RB}\ x] \tag{3.30}$$

then it can be seen that $C^{RB} - I$ is singular since it can be written as the product of two matrices, one of which, $[\underline{\phi}_{RB}\ x]$, is singular. Therefore, Eq. (3.29) cannot be solved for $\dot{\underline{\phi}}_{RB}$ by premultiplying each side by the inverse of $C^{BR} - I$.

The physical implication is that the factor $C^{BR} - I$ projects $\dot{\underline{\phi}}_{RB}$ onto a two dimensional space. (The crossproducts indicate that this space is orthogonal to $\underline{\phi}_{RB}$.)

As in the case of vectors, it is true that

$$[\underline{\omega}_{RB} \times]\underline{\phi}_{RB} = -[\underline{\phi}_{RB} \times]\underline{\omega}_{RB}$$

Using this identity and Eq. (3.30) in Eq. (3.29), premultiplying each term by $[\underline{\phi}_{RB} \times]/\phi_{RB}^2$, and transferring all non-zero terms to the right hand side gives

$$\underline{0} = -\frac{1}{\phi_{RB}^2}[\underline{\phi}_{RB} \times]^2 \underline{\omega}_{RB} + \frac{(1- \cos \phi_{RB})}{\phi_{RB}^2}[\underline{\phi}_{RB} \times]\dot{\underline{\phi}}_{RB}$$

$$+ \frac{1}{\phi_{RB}^2}\frac{\sin \phi_{RB}}{\phi_{RB}}[\underline{\phi}_{RB} \times]^2 \dot{\underline{\phi}}_{RB} \qquad (3.31)$$

where the second term on the right has been reduced by the identity

$$[\underline{\phi} \times]^3 \underline{V} \equiv -\phi^2[\underline{\phi} \times] \underline{V}$$

Note that when $\underline{\omega}_{RB}$ is constant in direction,

$$\underline{\omega}_{RB} = \dot{\underline{\phi}}_{RB}$$

as was the case in the derivation leading to Eq. (3.9). Moreover, it is true in any case (refer to Appendix A) that

$$\underline{\phi}_{RB} \cdot \underline{\omega}_{PB} = \underline{\phi}_{RB} \cdot \dot{\underline{\phi}}_{RB} \qquad (3.32)$$

that is, $\underline{\omega}_{RB}$ and $\dot{\underline{\phi}}_{PB}$ always have the same component in the direction of $\underline{\phi}_{PB}$. In matrix form Eq. (3.32) is

$$\underline{\phi}_{RB}^{T}\underline{\omega}_{RB} = \underline{\phi}_{RB}^{T}\dot{\underline{\phi}}_{RB}$$

Premultiply each side by $\underline{\phi}_{RB}/\phi_{RB}^2$ and add the resulting equation to Eq. (3.31) to get

$$\frac{\underline{\phi}_{RB}\underline{\phi}_{RB}^{T}}{\phi_{RB}^2} = \frac{1}{\phi_{RB}^2}\left\{-[\underline{\phi}_{RB}\ x]^2 + \underline{\phi}_{RB}\underline{\phi}_{RB}^{T}\right\}\underline{\omega}_{RB}$$

$$+ \frac{(1-\cos\phi_{RB})}{\phi_{RB}^2}[\underline{\phi}_{RB}\ x]\dot{\underline{\phi}}_{RB} + \frac{1}{\phi_{RB}^2}\frac{\sin\phi_{RB}}{\phi_{RB}}[\underline{\phi}_{RB}x]^2\ \dot{\underline{\phi}}_{RB}$$

Finally subtract $[\underline{\phi}_{RB}\ x]^2\dot{\underline{\phi}}_{RB}/\phi_{RB}^2$ from each side and invoke Eq. (3.16) once again. The result in vector notation is

$$\dot{\underline{\phi}} = \underline{\omega} + B(\phi)\underline{\phi}\ x\ \dot{\underline{\phi}} - C(\phi)\underline{\phi}\ x(\underline{\phi}x\dot{\underline{\phi}}) \tag{3.33}$$

where

$$B(\phi) = \frac{1}{\phi^2}(1-\cos\phi) \tag{3.34}$$

$$C(\phi) = \frac{1}{\phi^2}\left(1-\frac{\sin\phi}{\phi}\right) \tag{3.35}$$

Two other forms of the rotation vector differential equation are derived in Appendix A. All forms are mutually equivalent in the sense that any form can be derived from any other form. These equations are listed here for convenience.

$$\dot{\underline{\phi}} = \underline{\omega} + \frac{1}{2}\underline{\phi}\ x\ \underline{\omega} + A(\phi)\underline{\phi}\ x(\underline{\phi}\ x\ \underline{\omega}) \tag{3.36}$$

where

$$A(\phi) = \frac{1}{\phi^2}\left(1 - \frac{\phi\sin\phi}{2(1-\cos\phi)}\right) \tag{3.37}$$

Also

$$\dot{\underline{\phi}} = \underline{\omega} + \underline{\phi} \times \left(\frac{C(\phi)}{B(\phi)} \underline{\omega} + 2A(\phi) \dot{\underline{\phi}} \right) \tag{3.38}$$

To obtain yet another form, when the substitution

$$\underline{\phi} = \underline{1}_{\phi} \; 4 \arctan \frac{\alpha}{4} \tag{3.39}$$

is made in Eq. (3.36), then a lengthy reduction yields

$$\dot{\underline{\alpha}} = \left(1 + \frac{\alpha^2}{16} \right) \underline{\omega} + \frac{1}{2} \underline{\alpha} \times \underline{\omega} + \frac{1}{8} \underline{\alpha} \times (\underline{\alpha} \times \underline{\omega}) \tag{3.40}$$

The coordinate transformation can also be expressed as a function of $\underline{\alpha}$. The result is

$$C^{RB} = I + \frac{1 - \frac{\alpha^2}{16}}{\left(1 + \frac{\alpha^2}{16} \right)^2} [\underline{\alpha} \times] + \frac{1}{2\left(1 + \frac{\alpha^2}{16} \right)^2} [\underline{\alpha} \times]^2 \tag{3.41}$$

The last pair of equations is worthy of note in that no trigonometric functions are involved in either the rotation vector solution $\underline{\alpha}$ or in the coordinate transformation. Eq. (3.39) shows, however, that α is unbounded for $\phi = 2\pi$.

3.4 The Goodman Robinson Theorem

The 1950's were a decade in which great advancements were made in the design, manufacture, and testing of gyroscopic instruments. As is often the case, unexpected results were observed in the testing process, and so a theory was developed to explain these results. One such unexpected result was the now famous "coning" phenomenon in which a gyro, when subjected to out of phase sinusoidal oscillations about its spin and output axes, indicates a constant input axis angular velocity when,

in reality, no net change in orientation is occurring about its input axis. In 1957, Goodman and Robinson presented a paper (ref. 13) in which they explained the cause of the coning phenomenon.

The Goodman-Robinson theorem will be presented here because of its historical significance and also because it affords an interesting and independent derivation of Eq. (3.35).

The Goodman-Robinson Theorem

If a rigid body undergoes an arbitrary angular motion with respect to the Reference Frame, but at some time t_f one Body axis, say the i^{th} axis $\underline{l}_i^B$ returns to the orientation it had at t_o, then the net effect of the angular motion was to displace the Body Frame relative to the Reference Frame by a rotation ψ_i taken about $\underline{l}_i^B$ where ψ_i is given by

$$\psi_i = \int_{t_o}^{t_f} \omega_i(t)\, dt + A_i + 2m\pi \qquad (3.42)$$

where n is an integer

A_i is the solid angle described by $\underline{l}_i^B$ and is equal to the surface area traced out by $\underline{l}_i^B$ on the unit sphere

Goodman and Robinson proved the theorem first for the case where the curve traced out on the unit sphere by $\underline{l}_i^B$ is a simple closed curve. Their argument is somewhat hard to visualize geometrically. Broxmeyer (ref. 4) offers a proof that affords considerably more insight. Refer to reference (4) or (12) for a proof.

Goodman and Robinson then extended the theorem to curves which are not closed. They did this simply by postulating a

convenient closure for the curve. The argument is as follows: Suppose the Body Frame and the Reference Frame are coincident at $t = t_o$. An arbitrary angular velocity $\underline{\omega}_{RB}$ on the interval (t_o, t_f) takes the Body Frame from coincidence with the Reference Frame into some orientation at $t = t_f$ that can be uniquely described by the principal value of the rotation vector $\underline{\phi}_{RB}(t_f)$. The curve traced out on the interval (t_o, t_f) can be conceptually closed by conceptually rotating the Body Frame from its relative orientation at $t = t_f$ through the vector rotation $-\underline{\phi}_{RB}$. Then since this conceptually rotated Body Frame again coincides with the Reference Frame, ψ_i in Eq. (3.42) is zero. When this is done,

$$\psi_i = \left[\int_{t_o}^{t_f} \omega_i(t)\, dt + A_i + 2m\pi \right] - \phi_i = 0 \qquad (3.43)$$

If

$$\left| \int_{t_o}^{t_f} \underline{\omega}(t)\, dt \right| < 2\pi \qquad (3.44)$$

then it is true that $m = 0$. Solving Eq. (3.43) for ϕ_i gives

$$\phi_i = \int_{t_o}^{t_f} \omega_i(t)\, dt + A_i \qquad (3.45)$$

whenever Eq. (3.44) holds.

To find the area A_i, Goodman and Robinson used the so-called Green's Theorem in the plane (ref. 14). The result to be employed here is

$$A_i = \frac{1}{2} \oint (xdy - ydx) \tag{3.46}$$

where

A is the plane area within a closed curve C,

$\oint$ is the line integral taken in the positive sense around C,

$f(x,y) = 0$ is the equation of C

The use of Eq. (3.46) will yield an approximate expression for the area within the contour traced out by $\underline{l}_i^B$ as closed by the conceptual rotation $-\underline{\phi}_{RB}$. This is shown as the shaded area in Figure 3.3. In order to adapt Eq. (3.46), a change in coordinate system must be introduced. Let the local area around $\underline{l}_z^B$ (say) be approximated by a plane, and define a ϕ_x, ϕ_y coordinate system, in that plane as shown in Figure 3.4. In terms of this coordinate system, Eq. (3.46) becomes (in parametric form)

$$A_z(t) = \frac{1}{2} \int_{t_o}^{t_f} \left(\phi_x \frac{d\phi_y}{d\tau} - \phi_y \frac{d\phi_x}{d\tau} \right) d\tau \tag{3.47}$$

which when used in Eq. (3.45) gives

$$\phi_z(t) = \int_{t_o}^{t} \left[\omega_z + \frac{1}{2} \left(\phi_x \frac{d\phi_y}{d\tau} - \phi_y \frac{d\phi_x}{d\tau} \right) \right] d\tau \tag{3.48}$$

The equations for ϕ_x and ϕ_y are obtained by cyclic permutation of the indices. Equation (3.49) and the equations for ϕ_y and ϕ_x can be differentiated with respect to time to get the differential equations

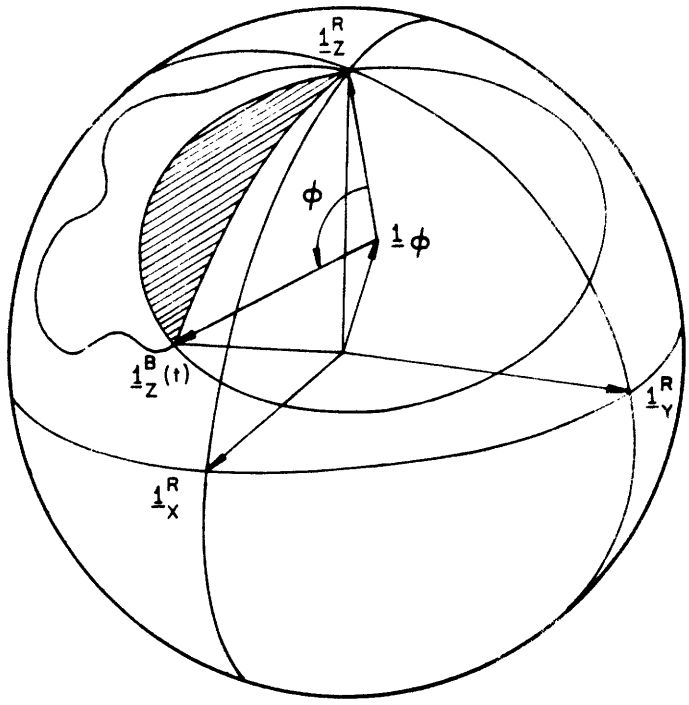


Figure 3.3.- Area traced on unit sphere

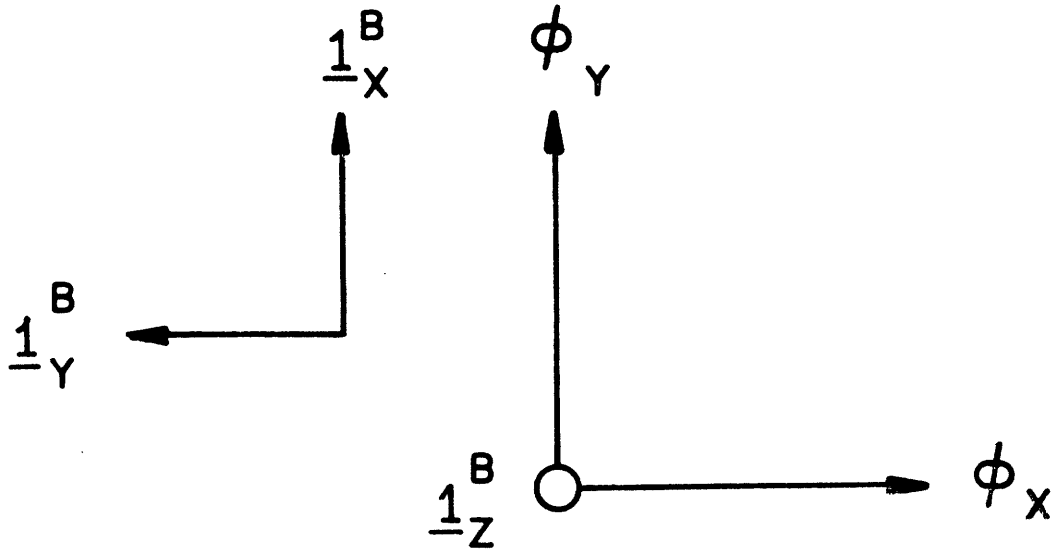


Figure 3.4.- ϕ_x, ϕ_y coordinate system

$$\left.\begin{aligned}\dot{\phi}_x &= \omega_x + \frac{1}{2}\,(\phi_y\dot{\phi}_z - \phi_z\dot{\phi}_y)\\ \dot{\phi}_y &= \omega_y + \frac{1}{2}\,(\phi_z\dot{\phi}_x - \phi_x\dot{\phi}_z)\\ \dot{\phi}_z &= \omega_z + \frac{1}{2}\,(\phi_x\dot{\phi}_y - \phi_y\dot{\phi}_x)\end{aligned}\right\} \qquad (3.49)$$

By defining the vector $\underline{\phi}^T = [\phi_x\ \phi_y\ \phi_z]$, Eq. (3.49) can be written in vector form.

$$\dot{\underline{\phi}} = \underline{\omega} + \frac{1}{2}\,\underline{\phi} \times \dot{\underline{\phi}} \qquad (3.50)$$

This is a reasonably good approximation to Eq. (3.33) which is written here in expanded form.

$$\dot{\underline{\phi}} = \underline{\omega} + \frac{1 - \cos\phi}{\phi^2}\,\underline{\phi} \times \dot{\underline{\phi}} + \frac{1}{\phi^2}\left(1 - \frac{\sin\phi}{\phi}\right)\underline{\phi} \times (\underline{\phi} \times \dot{\underline{\phi}}) \qquad (3.33)$$

By comparing Eq. (3.33) with Eq. (3.50), it is seen that there are two areas of disagreement. They are (a) the coefficient on the $\underline{\phi} \times \dot{\underline{\phi}}$ term and (b) Eq. (3.50) omits the final term in Eq. (3.33). Intuitively, these discrepancies can be explained as follows:

(a) Let A_p be the area of a plane circle of radius ϕ. Then from plane geometry,

$$A_p = \pi\phi^2$$

Let A_S be the spherical area enclosed within the spherical small circle in Figure 3.5 where the central angle from the pole to the small circle is ϕ.

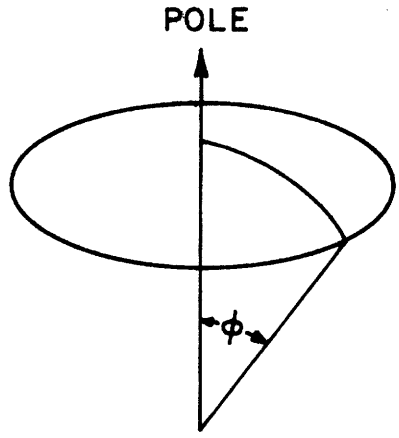


Figure 3.5.- Spherical area enclosed by small cycle

If the sphere is a unit sphere, then from solid geometry,

$$A_S = 2\pi(1 - \cos\phi)$$

The ratio

$$\frac{A_S}{A_p} = \frac{\frac{1}{2}(1 - \cos\phi)}{\phi^2 / 1/2}$$

is exactly the ratio of the coefficient of the $\underline{\phi} x \dot{\underline{\phi}}$ term in Eq. (3.33) to that in Eq. (3.50).

(b) Referring to Figure 3.3, the area given by Eq.(3.47) approximates the sum of the shaded area and the crosshatched area whereas the desired area is only the shaded area. This happens because the closure for this greater area is a great circle which appears as a <u>straight</u> line in Figure 3.4. The actual closure generated by $-\underline{\phi}_{RB}$ is, in general, a small circle which would appear as a <u>curved</u> line in Figure 3.4, but the parametric form of Green's Theorem given by Eq. (3.47)

takes the area enclosed by (a) the curve generated by the parameter t and (b) a straight line from the origin to the location of the point (ϕ_x, ϕ_y) at time t. The area discrepancy (shown as the cross hatched area in Figure 3.3) is equal to

$$\frac{1}{\phi^2}\left(1 - \frac{\sin\phi}{\phi}\right)\underline{\phi} \times (\underline{\phi} \times \dot{\underline{\phi}})$$

CHAPTER 4

THE MEASUREMENT OF ANGULAR VELOCITY

4.1 Problem Statement

In Chapter 2 the analog computation of the $\dot{\underline{\sigma}}_{RB}$ correction term required a triad of continuous signals representing $\underline{\omega}^B_{RB}$. The gyros measure the components of $\underline{\omega}^B_{RB}$, and from a combination of signals, observable in the gyros, the desired signals must be derived.

If a gyro with a linear rebalance loop followed by an integrator and quantizer is used, the generation of the desired analog signals is straightforward. The performance of a single degree-of-freedom integrating gyro* as given by Eq. (2.8) becomes

$$(Is^2+Cs)\ A = H(\omega_{IA}-\omega_{fb}) \tag{4.1}$$

if the non-ideal performance term is neglected and $H\omega_{ext}$ is taken to be zero. $H\omega_{ext} = H\dot{\sigma}$ in a pulse rebalanced gyro, for in that configuration, the summation and integration of $H(\omega_{IA}+\dot{\sigma})$ takes place in the gyro and the quantization operation

*An integrating gyro is so named because of the absence of a mechanical restraint torque proportional to float angle. An integrating gyro may be used in a rate gyro mode by using a linear rebalance loop.

is performed by the rebalance loop. In a gyro with a linear rebalance loop followed by an integrator and quantizer, $\dot{\sigma}$ is neither integrated nor quantized by the gyro and so it is added to ω_{IA} at the input to the integrator-quantizer. In linear rebalance loops, the rebalance torque $H\omega_{fb}$ is proportional to float angle A.

$$H\omega_{fb} = KA$$

With this substitution Eq. (4.1) becomes

$$(Is^2+Cs+K)\ A = H\omega_{IA} \tag{4.2}$$

An electrical signal e_{sg}, proportional to float angle A

$$e_{sg} = K_{sg}A \tag{4.3}$$

is taken to be the output of the gyro. The gyro transfer function is found by using Eq. (4.3) in Eq. (4.2) and taking the Laplace Transform (assuming zero initial conditions)

$$\frac{e_{sg}(s)}{\omega_{IA}(s)} = \frac{K_{sg}H/K}{(I/K)s^2 + (C/K)s + 1} \tag{4.4}$$

In the case of the linear rebalance loop, at low frequencies the signal e_{sg} is proportional to ω_{IA} with a scale factor of $K_{sg}H/K$ volts per radian per second. The high frequency characteristics are modified by the presence of the second order dynamics of the gyro.

For the pulse rebalanced gyro, it is not so obvious how to proceed. One possibility for obtaining a voltage proportional to ω_{IA} is to fit an $(n-1)^{th}$ or lower order polynomial through the last n data points, i.e., the last n $\Delta\theta$'s from

the gyro. This is done in the digital computer. The value of the polynomial (over the interval from the previous sampling instant until the time of the next fitting process) is made available to the analog circuitry by digital to analog converters. With three components of $\underline{\omega}^B_{RB}$ to extract, this computation could not be performed often enough to have significant bandwidth without imposing a rather large load on the computer. The phase lags of low computation rates or the inaccuracies of fast rates have detrimental effects (when $\underline{\omega}_{RB}$ has a broad band power spectral density) on the coordinate transformation (computed by any technique) and on the compensation of the gyro dynamic errors $N(\underline{\omega},\underline{f},t)$. For an analog rate extraction scheme, the pulse rebalancing feature makes it impossible to obtain the desired signals from signal generator observations alone. As will be seen, the torque generator signal supplies the needed additional information for the rate extraction process.

4.2 Filter Analysis

The starting point for this analysis is again Eq. (2.8)

$$(Is^2+Cs)\ A = H\omega_{IA} - H\omega_{fb} + H\dot{\sigma} + N(\underline{\omega},\underline{f},t) \qquad (4.5)$$

where $H\dot{\sigma}$ has been substituted for $H\omega_{ext}$. The block diagram of the gyro modeled by this equation is shown in Figure 4.1. A three level relay is shown in the rebalance loop, but this is not essential; the filter design for a gyro whose rebalance loop utilizes any nonlinear element would be identical to that for the gyro with the three level relay. An intuitive approach

is taken in the design process, and the results are then analyzed to justify the design. Two questions lead to the filter configuration.

(a) Suppose the gyro were operating open loop (no rebalance signal). How should the output of the signal generator be treated in order to obtain the desired signal?

(b) Although the rebalance loop is in fact closed, can the torquing signal be processed and introduced into the filter in such a way as to allow the filter to operate on the signal generator signal of an equivalent open loop gyro?

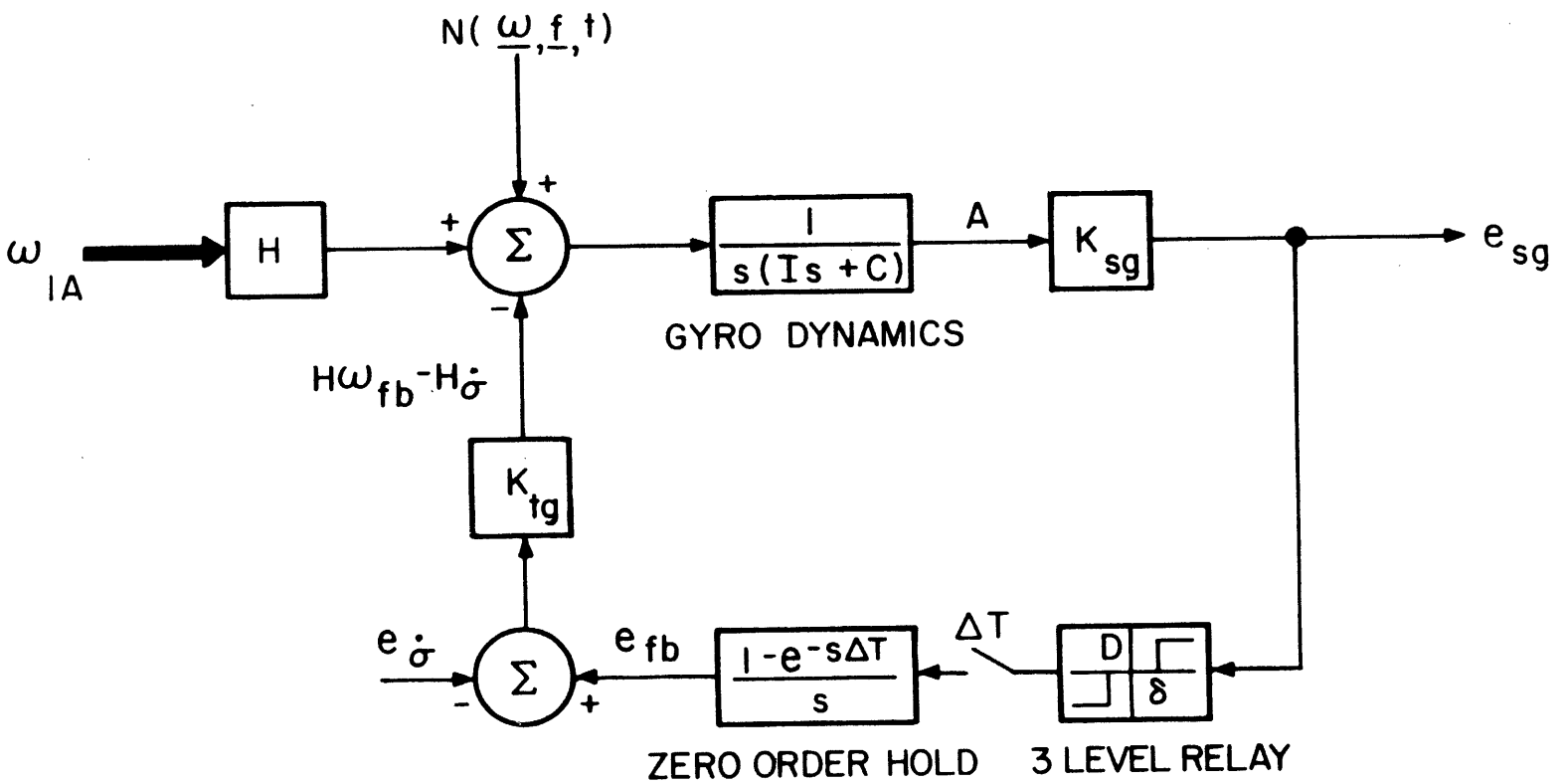
With the gyro operating open loop, the $H\omega_{fb} - H\dot{\sigma}$ term in Eq. (4.5) is zero. Also neglecting the non-ideal performance term, Eq. (4.5) becomes

$$(Is^2+Cs)\ A = H\omega_{IA}$$

Using Eq. (4.3), the gyro transfer function is

$$\frac{e_{sg}(s)}{\omega_{IA}(s)} = \frac{K_{sg}H/C}{s\left(\frac{I}{C}\ s+1\right)} \qquad (4.6)$$

The output e_{sg} is proportional (exclusive of gyro dynamics) to the integral of ω_{IA} as expected, and so the filter must therefore differentiate the signal generator signal. At frequencies above those in the desired range, attenuation at the differentiator output is desirable. This is especially true when the signal generator output is an amplitude modulated signal since a ripple at twice the carrier frequency is the by-product of any demodulation process. This ripple is removed by filtering. The filter function is chosen as



$$H\omega_{\dot{\sigma}} = K_{tg}\, e_{\dot{\sigma}}$$

$$H\omega_{fb} = K_{tg}\, e_{fb}$$

Figure 4.1.- Gyro and rebalance loop

$$\frac{\omega^*(s)}{e_{sg}(s)} = \frac{K_f s}{\left(\frac{s}{\omega_f}\right)^2 + \sqrt{2}\,\frac{s}{\omega_f} + 1} \tag{4.7}$$

where

ω^* is the measured value of ω_{IA}

K_f is the filter gain constant

ω_f is the filter natural frequency

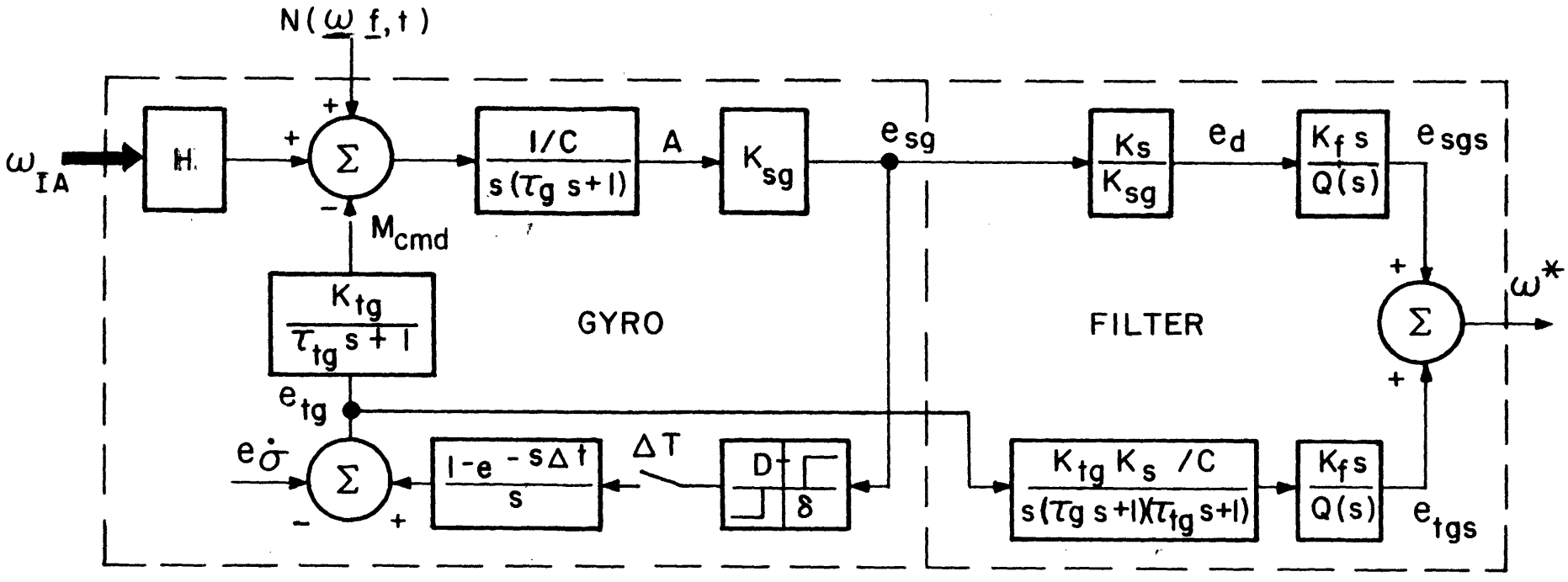
Notice that the denominator is an underdamped second order term. The particular choice of damping constant, $\sqrt{2}/2$, gives the filter a maximally flat response (ref. 14). This filter function is called the signal generator section of the ω-Filter.

With the ω-Filter given by Eq. (4.7), the overall transfer of the angular velocity measurement is

$$\frac{\omega^*(s)}{\omega_{IA}(s)} = \frac{HK_fK_{sg}/C}{(\tau_g s+1)(s^2/\omega_f^2+\sqrt{2}S/\omega_f+1} \tag{4.8}$$

where τ_g is the gyro time constant I/C.

The gyro is, of course, not operated open loop in a strapdown system. But with the torque generator signal passed through the exact electrical analog of the gyro and entered into the filter of Eq. (4.7) with the opposite sense as the torque generator signal (which has passed through the real gyro), then as far as the ω-Filter is concerned, the gyro is operating open loop. Such a filter concept is shown in Figure 4.2(a). In this filter section, the torque generator



FILTER DERIVATION

Figure 4.2(a).- Filter deviation

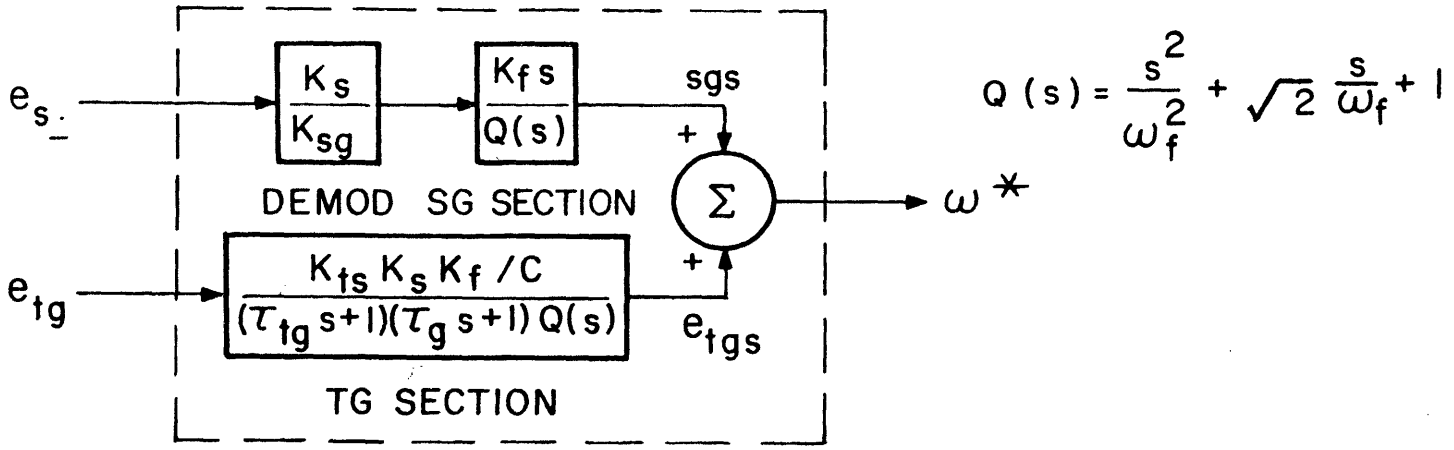


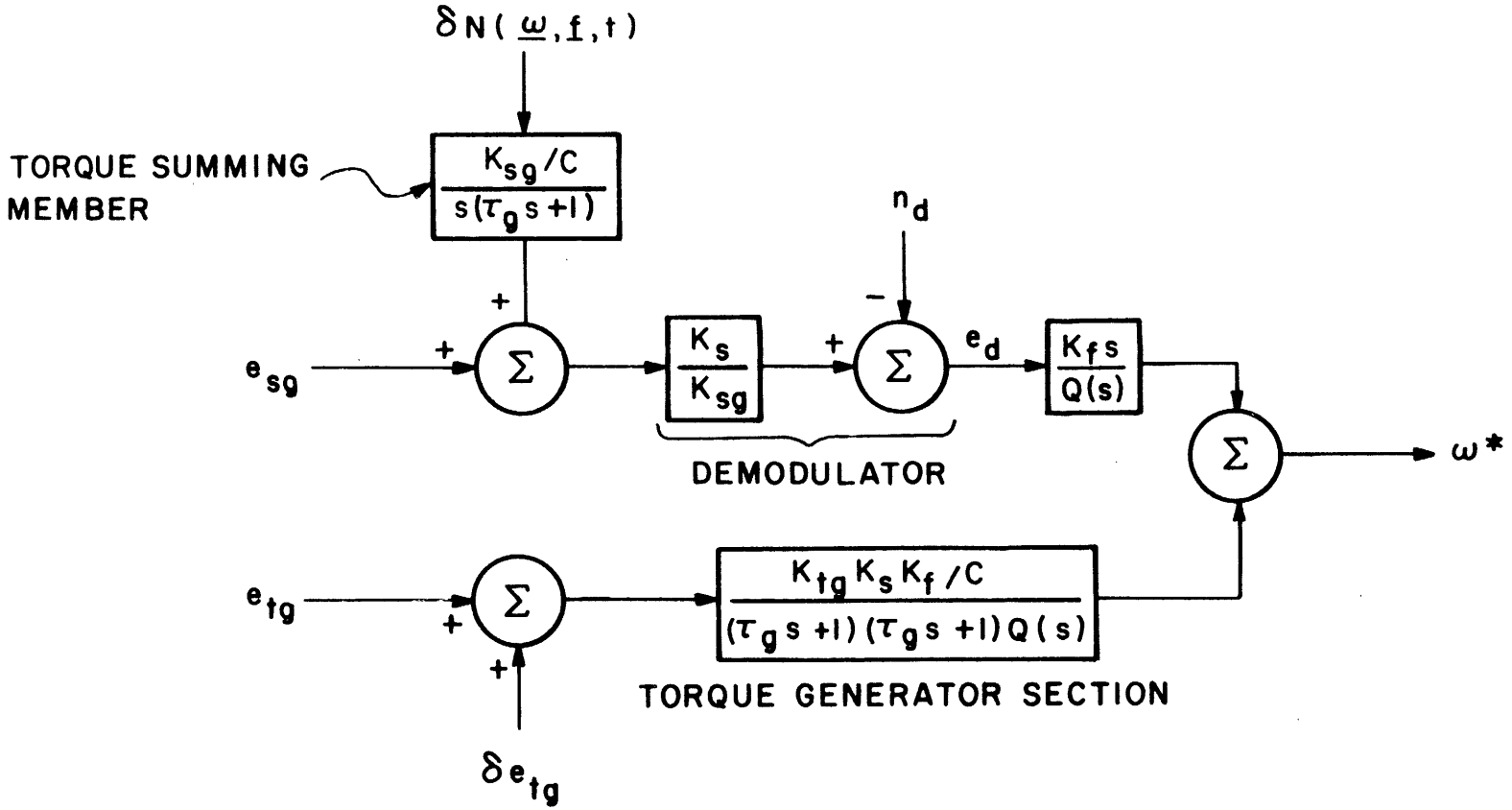
Figure 4.2(b).- Actual ω-Filter

section, the differentiation cancels* the integration performed in the gyro. The result is shown in Figure 4.2(b).

With the ω-Filter implemented in this way, Eq. (4.8) is still valid as the basic transference of the measurement process. There are three main sources of error in this system. They are (a) the uncompensated portion of the non-ideal performance term $\delta N(\underline{\omega},\underline{f},t)$, (b) an error δe_{tg} due to imperfect cancellation of the torque generator signal, and (c) an error n_d which arises in the demodulation process. The way these error sources effect the system is shown in Figure 4.3.

The non-ideal performance term $N(\underline{\omega},\underline{f},t)$ is, in reality, a sum of terms. This sum includes a constant term, a series of terms which are functions of angular velocity $\underline{\omega}$, and a series of terms which are functions of specific force $\underline{f}$. Each term is characterized by a coefficient which may or may not be capable of unique determination in a calibration process. These coefficients may change with time and there is an uncertainty in the determination of each coefficient. It is the practice, in mechanizing a high quality inertial navigation system, to compensate for the non-ideal performance as much as possible. Assume this has been done through application of a compensation torque $N_c(\underline{\omega},\underline{f},t)$ to the gyro float. The ability

*It is acceptable to cancel an integration with a differentiation, but the attempt to cancel a differentiation with an integration results in the situation known as non-observability because the bias component on the input to the differentiator is not known (observable) and hence cannot be restored after the integration.



$$Q(s) = -\frac{s^2}{\omega_f} + \sqrt{2}\,\frac{s}{\omega_f} + 1$$

Figure 4.3-. Errors in $\underline{\omega}$ measurement

to apply this compensation implies the ability to measure angular velocity and specific force and to compute the correction. When this is done, the error δN in applying the compensation is given by

$$\delta N(\underline{\omega},\underline{f},t) = N_c(\underline{\omega},\underline{f},t) - N(\underline{\omega},\underline{f},t) \qquad (4.9)$$

In Figure 4.3 this error is shown as an error torque acting on the torque summing member. The statistics of this error torque are rather complicated and no attempt will be made to discuss them here. Since δN is a function of angular velocity and specific force, it is expected that this noise will be correlated with the input.

The error due to imperfect cancellation of the electrically applied torques is the result of inexact voltage transmission from the torquer input to the filter torque generator section. (The difference between the real gyro dynamics and the simulated gyro path dynamics might be considered at this point but this additional level of attention is not justified since the $\dot{\underline{\sigma}}$ correction signal to be generated from filter output is itself small in magnitude compared to $|\underline{\omega}_{RB}|$. This error is shown entering the torque generator section of the filter in Figure 4.3.

Observe that δe_{tg} can assume (neglecting dynamics) one of three levels depending on whether a positive, negative, or zero pulse of rebalance torque is being generated. Again, this error is correlated with the input. In fact, the statistical properties of δe_{tg} are identical with those of the rebalance torque itself. These statistics will not be developed here.

The final error source, the demodulation error voltage n_d, is given by the expression [see Eq. (4.15)]

$$n_d = K_s(\cos 4\pi ft)A \tag{4.10}$$

where f is the modulation carrier frequency. This error is a consequence of the demodulation process. It also is a function of float angle A. Since its power is concentrated at the frequency 2f, it is easy to attenuate this error by filtering. Since n_d is a function of float angle A, it is correlated with the input.

Filtering which attempts to discriminate against the imperfect compensation error $\delta N(\underline{\omega},\underline{f},t)$ and the imperfect torque cancellation error δe_{tg} on the basis of different power spectra of noise and input signal will not be too effective since these errors are themselves functions of the input signal. The best way to suppress these noises is to suppress their sources. The demodulator noise n_d on the other hand is not a result of gyro and rebalance loop imperfections, but arises as a consequence of the demodulation technique. Since most of its power (except for sideband power generated by the variations in float angle A) is concentrated at twice the modulation carrier frequency f, this noise can be conveniently attenuated by filtering. The filter function for the signal generator is

$$F_{sg}(s) = K_f s/Q(s) \tag{4.11}$$

and is chosen to have the desired bandpass for the angular velocity ω_{IA} and the desired attenuation to the demodulator noise n_d.

4.3 Filter Design

The filter is conveniently divided into 4 sections, a demodulator, a signal generator section, a torque generator section, and a final section. The relationship of these sections is shown in block diagram form in Figure 4.4. The gyro characteristics and the design of each of the filter sections will be discussed now. A transfer function will be developed for each filter section. The actual circuit for each filter section is found in Appendix B. Section 4.3 may be omitted by those whose interest is in ω-Filter performance and not its detailed design. Performance and test results are presented in Section 4.4.

4.3.1 Gyro Characteristics

The mechanical and dynamic characteristics of the Honeywell GG 334A gyro necessary for the design of the ω-Filter are shown in Table 4.1

4.3.2 Demodulator

The signal generator excitation voltage is used in the demodulator and is given by

$$e_p = 5 \sin 2\pi ft \text{ volts} \tag{4.12}$$

The signal generator secondary signal is given by

$$e_s = K_{sg} A \sin 2\pi ft \text{ volts} \tag{4.13}$$

In order to obtain a voltage proportional to the float angle, this secondary signal must be demodulated. The demodulator consists of a multiplier whose transference is given by

TABLE 4.1

GG 334A CHARACTERISTICS

Name	Parameter	Value	Units
H	Angular Momentum	2×10^{5}	gm-cm^2/sec
C	Output Axis Damping	5×10^{5}	dyne-cm-sec
I	Output Axis Inertia	250	gm-cm^2
K_{sg}	Signal Generator Sensitivity	25-30	volts(peak)/radian
A_t	Threshold Float Angle	6×10^{-5}	radians
ΔT	Sampling Interval	1/3600	seconds
$\Delta\theta$	Quantization Level	2^{-14}	radians
τ_{tg}	Torquer Time Constant	5×10^{-5}	seconds
f	Signal Generator Carrier Frequency	28.8×10^{3}	hertz
D^*	Relay Output Voltage	5	volts
K^*_{tg}	Torque Generator Sensitivity	8×10^{4}	dyne-cm/volt
e_p^{**}	Signal Generator Excitation	5	volts (peak)
$\omega_{IA,MAX}$	Maximum Input Angular Velocity	2	radians/second

---

*These are parameters which are given for the simulated gyro path only. In the real gyro, the output of the three level relay is a current. A 5 volt signal is generated for readout purposes only.

**e_p actually has a peak value of 7.35 volts but it is attenuated to 5 volts peak for use in the demodulator.

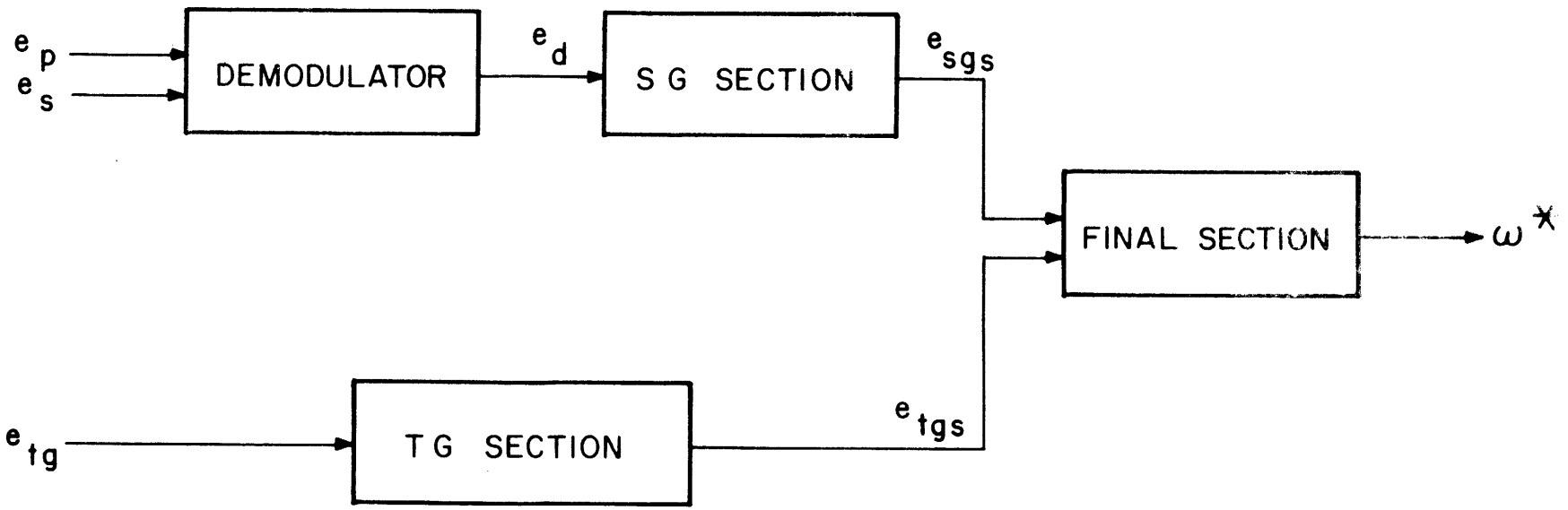


Figure 4.4.- Filter section schematic

$$z = xy/K_m$$

where z is the output voltage, x and y are input voltages, and K_m is equal to the maximum permissible voltage level of either input. For the demodulation process, it is necessary to amplify e_s and then multiply it by e_p. If the amplified e_s is $K_a e_s$, then the demodulator output is

$$e_d = K_a e_p e_s / K_m \tag{4.14}$$

The multiplier chosen for this filter (see Appendix B) has a maximum permissible input of 5 volts. Therefore K_m = 5 volts and using Eqs. (4.12) - (4.14),

$$\begin{aligned} e_d &= K_a 5 K_{tg} A \sin^2 2\pi ft/5 \\ &= K_{sg} K_a A \tfrac{1}{2} (1 - \cos 4\pi ft) \\ &= K_s (1 - \cos 4\pi ft) A \end{aligned} \tag{4.15}$$

where

$$K_s = \tfrac{1}{2} K_{sg} K_a \tag{4.16}$$

K_s may be regarded as the equivalent signal generator sensitivity for the signal flow path through the ω-Filter. Since K_a is dimensionless, the dimensions of K_s are the same as those of K_{sg}, viz., volts per radian. Substituting Eq. (4.10) into Eq. (4.15) gives

$$e_d = K_s A - n_d \tag{4.17}$$

Suppose it is desired when $A = A_T$, that $K_s A_t$ = 1.8 volts. From Eq. (4.15), the maximum value of $e_d = 2K_s A$. Thus at

threshold, $e_{d,\ max} = 3.6$ volts. The difference between this and the 5 volt maximum allows for the possibility of exceeding the threshold without saturating the multiplier. Then

$$\begin{aligned} K_s &= 1.8/A_T \\ &= 1.8/6\text{x}10^{-5} \\ &= 3\text{x}10^4 \text{ volts/radian} \end{aligned} \tag{4.18}$$

From Eqs. (4.16) and (4.18) and Table 4.1, the required value for preamplifier gain is

$$\begin{aligned} K_a &= 2\ K_s/K_{sg} \\ &= 2\text{x}3\text{x}10^4/30 \\ &= 2000 \end{aligned} \tag{4.19}$$

4.3.3 Signal Generator Section

This section implements the transfer function

$$F_{sg}(s) = \frac{K_f \omega_f \frac{s}{\omega_f}}{\frac{s^2}{\omega_f^2} + \sqrt{2}\,\frac{s}{\omega_f} + 1} \tag{4.20}$$

where K_f is chosen so that an overall scale factor which relates $\omega_{IA',\ MAX}$ to ω^*_{MAX} has some desired value and ω_f is chosen so as to achieve the desired bandpass on one hand and adequate filtering of n_d on the other.

The actual chain of elements in the transference path from ω_{IA} to ω^* is shown in Figure 4.5(a). The equivalent transference is shown in Figure 4.5(b). 10 volts was chosen for the maximum value of ω^*. Then since $\omega_{IA,\ MAX} = 2$ radians/second, it is necessary that

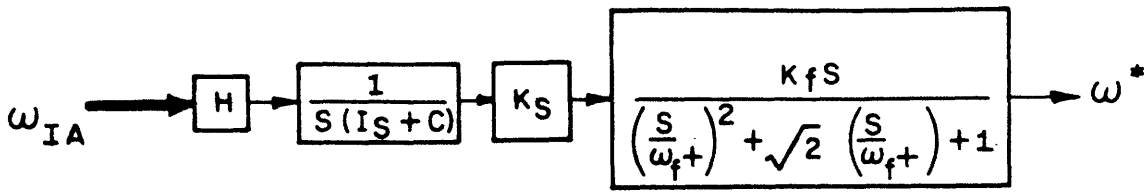


Figure 4.5(a).- Actual arrangement of elements

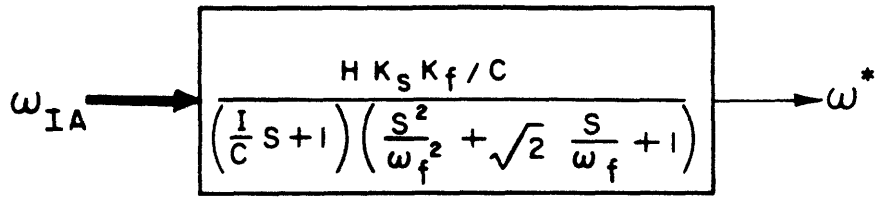


Figure 4.5(b).- Equivalent transference

$$H\,K_s K_f/C = \omega^*/\omega_{IA,\,MAX}$$

$$= 5 \text{ volt-sec/radian}$$

Therefore

$$K_f = 5C/HK_s$$

$$= \frac{5x5x10^5}{2x10^5x3x10^4} \text{ sec}$$

$$= 4.17x10^{-4} \text{ sec} \qquad (4.21)$$

The gyro float time constant τ_g is I/C, which from Table 4.1 is

$$\tau_g = 5x10^{-4} \text{ sec} \qquad (4.22)$$

or equivalently, the gyro break frequency ω_g is

$$\omega_g = 1/\tau_g$$

$$\omega_g = 2\text{x}10^3 \text{ rad/sec} \tag{4.23}$$

The noise n_d has a characteristic frequency ω_n given by

$$\omega_n = 4\pi f$$

$$= 4\pi\text{x}2.88\text{x}10^4$$

$$= 3.62\text{x}10^5 \text{ rad/sec}$$

Therefore, choose

$$\omega_f = 10^4 \text{ rad/sec} \tag{4.24}$$

This is sufficiently above the gyro float break frequency so that the input signal bandpass is still primarily limited by the float lag. It is also sufficiently below the demodulator noise frequency so that the demodulator noise is heavily attenuated by the filter.

Note from Figures 4.2a and 4.3 that ω_{IA} is attenuated by a second order characteristic in the filter since the float integration and filter differentiation cancel. In contrast with this n_d experiences only a first order attenuation since in its path there is no integration to cancel the filter differentiation.

By substituting Eqs. (4.21) and (4.24) into Eq. (4.20), the signal generator section transfer function becomes

$$F_{sg}(s) = \frac{4.17\left(\frac{s}{10^4}\right)}{\left(\frac{s}{10^4}\right)^2 + \sqrt{2}\left(\frac{s}{10^4}\right) + 1} \tag{4.25}$$

4.3.4 Torque Generator Section

This section must simply duplicate the path that must be taken by the torquing signal. From Figure 4.2b, it can be seen that the transfer function of the torque generator section is

$$F_{tg}(s) = \frac{K_{tg}K_sK_f/C}{\left(\tau_{tg}s+1\right)\left(\tau_g s+1\right)\left(\frac{s^2}{\omega_f^2}+\sqrt{2}\,\frac{s}{\omega_f}+1\right)} \qquad (4.26)$$

Using values from Table (4.1) and Eqs. (4.18), (4.21), and (4.24), this becomes

$$F_{tg}(s) = \frac{2}{\left(5\,\frac{s}{10^5}+1\right)\left(5\,\frac{s}{10^4}+1\right)\left(\frac{s^2}{10^8}+\sqrt{2}\,\frac{s}{10^4}+1\right)} \qquad (4.27)$$

4.3.5 Final Section

The final section is responsible for combining the outputs of the signal generator and the torque generator sections. It is merely a summing amplifier.

4.4 Test Results

The ω-Filter was built (Appendix B) and tested. The results were generally as expected although a perfect cancellation of the torque generator signal was not achieved. At the filter output, there were small residual pulses of opposite sign occurring before and after the generation of a torque pulse in the gyro. These residual pulses had a one volt magnitude. In order to suppress these pulses, a lag was introduced at the output stage having a 10^{-2} second time

constant. The residual pulse were attenuated by a factor of 10, but the first break frequency was reduced to 100 rad/sec. The filter gain and phase vs. frequency plot is shown in Figure 4.6.

The filter was adjusted to have zero output when the float angle A was constant and to have a 1.000 volt (1/10 of full scale voltage) when the gyro pulse rate was 3600 pulses in 10 seconds (1/10 of maximum pulse rate). Thus the filter was adjusted for zero error at two points. The output voltage error at any intermediate point was less than 1 percent of full scale voltage. A suggested design goal is an output voltage error of less than 1 percent of the nominal output voltage at any intermediate point.

Conditions adversely affecting the filter performance were:

(a) The gyros were not temperature controlled. This rendered an exact electrical analog of the gyros difficult to achieve.

(b) The signal generator signal used by the filter was taken from the output of the rebalance loop preamplifier. The gain of this preamplifier tends to have a different value above the threshold for generating a torque pulse than below. It is recommended that the signal generator signal itself be used rather than the rebalance loop preamplifier output.

(c) The multiplier used in the demodulator has a nominal bandwidth of 25 KHz. The multiplier output signal is a 57.6 KHz signal. A better multiplier is recommended.

(d) The gyro wheels were not on. The result of this factor is difficult to assess.

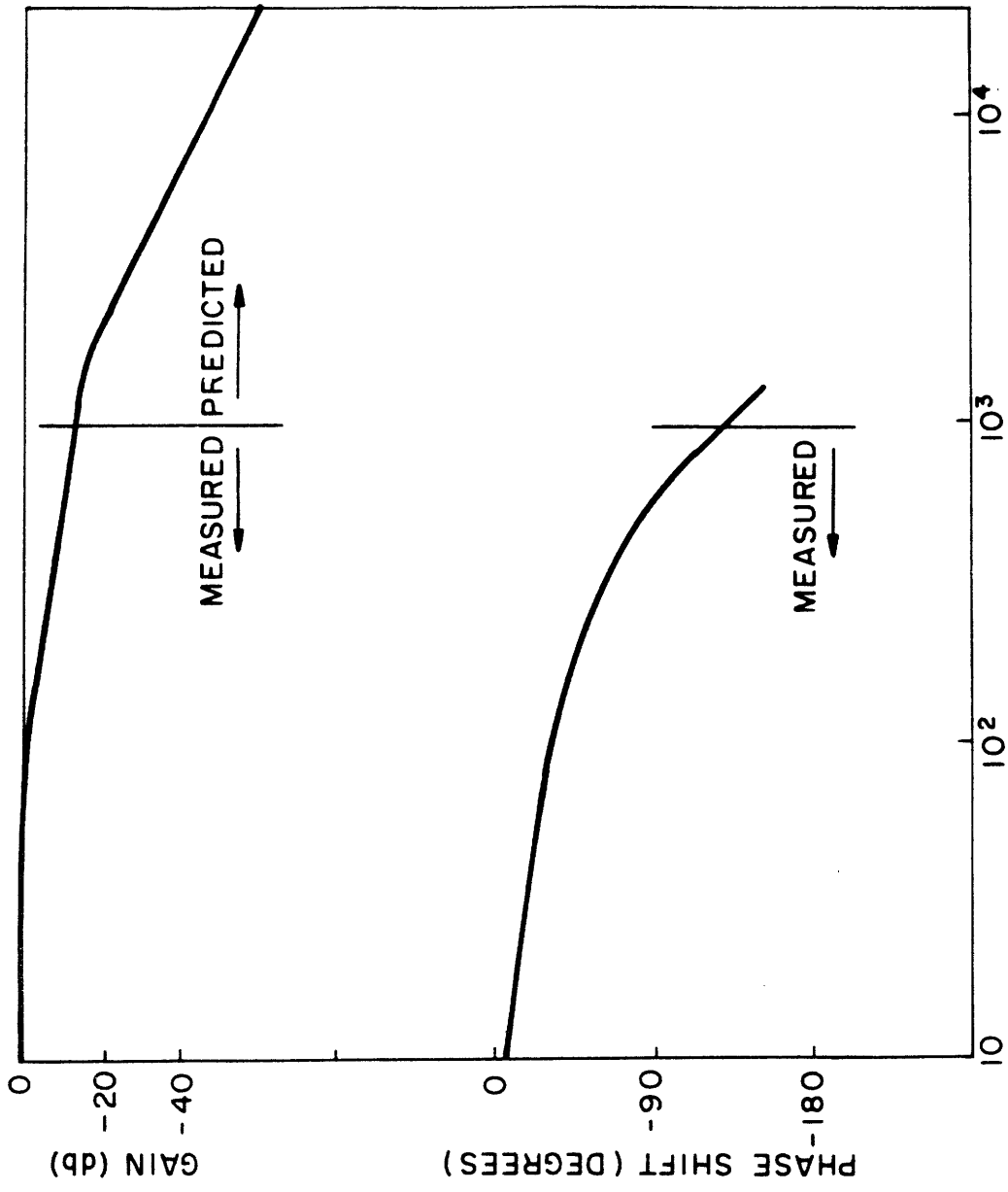


Figure 4.6.- ω filter gain, phase vs ω

CHAPTER 5

SYSTEM MECHANIZATION

5.1 The Choice of an Equation

The system to be mechanized is that system which integrates $\underline{\dot{\phi}}_{RB}$ and evaluates the coordinate transformation C^{RB} given by Eq. (3.17). In Section 3.3, four forms of the rotation vector differential equation were given as Eqs. (3.36), (3.33), (3.38), and (3.40). They are rewritten here for convenience.

$$\underline{\dot{\phi}} = \underline{\omega} + \frac{1}{2}\,\underline{\phi} \times \underline{\omega} + A(\phi)\,\underline{\phi} \times (\underline{\phi} \times \underline{\omega}) \tag{5.1}$$

$$\underline{\dot{\phi}} = \underline{\omega} + B(\phi)\underline{\phi} \times \underline{\dot{\phi}} - C(\phi)\,\underline{\phi} \times (\underline{\phi} \times \underline{\dot{\phi}}) \tag{5.2}$$

$$\underline{\dot{\phi}} = \underline{\omega} + \frac{C(\phi)}{B(\phi)}\,\underline{\phi} \times \underline{\omega} + 2A(\phi)\underline{\phi} \times \underline{\dot{\phi}} \tag{5.3}$$

and

$$\underline{\dot{\alpha}} = \left(1 + \frac{\alpha^2}{16}\right)\underline{\omega} + \frac{1}{2}\,\underline{\alpha} \times \underline{\omega} + \frac{1}{8}\,\underline{\alpha} \times (\underline{\alpha} \times \underline{\omega}) \tag{5.4}$$

where

$$\underline{\alpha} = \underline{1}_{\phi}\, 4 \tan \frac{\phi}{4} \tag{5.5}$$

When expanded as infinite series, A, B, and C are [from Eqs. (3.37), (3.34) and (3.35)]

$$A = \frac{1}{12} + \frac{\phi^2}{720} + \frac{\phi^4}{30,240} + \frac{\phi^6}{1,209,600} + \cdots \tag{5.6}$$

$$B = \frac{1}{2} - \frac{\phi^2}{24} + \frac{\phi^4}{720} - \frac{\phi^6}{40,320} + \cdots \tag{5.7}$$

$$C = \frac{1}{6} - \frac{\phi^2}{120} + \frac{\phi^4}{5040} - \frac{\phi^6}{362,880} + \cdots \tag{5.8}$$

A desirable property of the equation to be used is that when $\underline{l}_\omega$ is constant, $\dot{\underline{\phi}} = \underline{\omega}$. In other words, it is desirable to be able to cast the system equation in the form

$$\dot{\underline{\phi}} = \underline{\omega} + \dot{\underline{\sigma}} \tag{5.9}$$

In this form, $\dot{\underline{\sigma}}$ can be regarded as the correction rate for the noncommutativity phenomenon, and in the absence of a noncommutativity effect ($\underline{l}_\omega$ constant), $\underline{\phi}$ is just the same as it would be if the $\dot{\underline{\sigma}}$ loop in Figure 2.1 were opened. This loop could actually be opened when the direction of $\underline{\omega}$ is essentially constant over the update interval. Equation (5.5) introduces a scaling applying to Eq. (5.4). As a result, this form of the rotation vector differential equation is subject to additional sources of analog computation error over the other three forms and will not be considered further.

If only the first term of the series expansions for A, B, and C is used in Eqs. (5.1), (5.2) and (5.3), then the approximate relations

$$\dot{\underline{\phi}} = \underline{\omega} + \frac{1}{2}\,\underline{\phi} \times \underline{\omega} + \frac{1}{12}\,\underline{\phi} \times (\underline{\phi} \times \underline{\omega}) \tag{5.10}$$

$$\dot{\underline{\phi}} = \underline{\omega} + \frac{1}{2}\,\underline{\phi} \times \dot{\underline{\phi}} - \frac{1}{6}\,\underline{\phi} \times (\underline{\phi} \times \dot{\underline{\phi}}) \tag{5.11}$$

$$\dot{\underline{\phi}} = \underline{\omega} + \underline{\phi} \times \left(\frac{1}{3}\,\underline{\omega} + \frac{1}{6}\,\dot{\underline{\phi}} \right) \tag{5.12}$$

are obtained. The most significant neglected term in Eq.(5.10) is

$$\frac{\phi^4}{720}\,\underline{1}_\phi \times(\underline{1}_\phi \times \underline{\omega})$$

where $\phi = |\underline{\phi}|$. In Eq. (5.11), the most significant neglected term is

$$-\frac{\phi^3}{24}\,\underline{1}_\phi \times \dot{\underline{\phi}}$$

and in Eq. (5.12) it is

$$\frac{\phi^3}{90}\,\underline{1}_\phi \times \underline{\omega} + \frac{\phi^3}{720}\,\underline{1}_\phi \times \dot{\underline{\phi}}$$

Obviously, for any value of $|\underline{\phi}|$, Eq. (5.10) gives a superior result to that of Eq. (5.11). Hence, Eq. (5.11) is excused from further consideration. While the approximation errors in Eq. (5.12) are two orders of magnitude larger than those in Eq. (5.10), Eq. (5.12) enjoys a distinct computational advantage over Eq. (5.10). In Eq. (5.10), two cross products are required, while in Eq. (5.12) only one is required. Recognizing the fact that analog computational elements do contribute errors, the impact of the errors generated by the additional cross product requirement of Eq. (5.10) would have to be assessed before a choice could be made between Eqs. (5.10) and (5.12) strictly on the basis of relative accuracy.

Relative accuracy, however, need not be the basis for the choice. The choice of a particular equation need only be substantiated on the basis of the absolute accuracy of its solution. The scale factor uncertainty of the gyros used in

the experiment (Honeywell GG 334A gyros) has an rms value of about 200 ppm. Therefore an error of 20 ppm can be tolerated in the equation without significant effect on the solution. This justifies the choice of Eq. (5.12) subject to a restriction on ϕ_{MAX} that will hold the error in $\underline{\dot{\phi}}$ to less than 20 ppm. Using Eq. (5.9), Eq. (5.12) becomes

$$\underline{\dot{\phi}} = \underline{\omega} + \underline{\phi} x \left(\frac{1}{2} \underline{\omega} + \frac{1}{6} \underline{\dot{\sigma}} \right) \tag{5.13}$$

The accuracy and the stability of Eq. (5.3) for large ϕ ($\phi < \pi$ radians) and of Eq. (5.13) for small ϕ have been verified by a highly accurate digital computer integration. This verification is given in Appendix C.

5.2 Analog System Configuration

The analog system mechanizing Eq. (5.13) is shown symbolically in Figure 5.1. The considerations of voltage scaling are also indicated. Since each signal in an analog computer is a voltage, each real variable being modelled is a product of its corresponding analog voltage and a scale factor. In particular,

$$\underline{\omega} = k_{\omega} \underline{e}_{\omega} \tag{5.14}$$

$$\underline{\dot{\phi}} = k_{\dot{\phi}} \underline{e}_{\dot{\phi}} \tag{5.15}$$

$$\underline{\dot{\sigma}} = k_{\dot{\sigma}} \underline{e}_{\dot{\sigma}} \tag{5.16}$$

$$\underline{\phi} = k_{\phi} \underline{e}_{\phi} \tag{5.17}$$

One scale factor, k_{ω}, can be chosen at once. Since the maximum voltage on the analog computer is 100 volts and since the maximum angular velocity for the gyros is 2 radians per

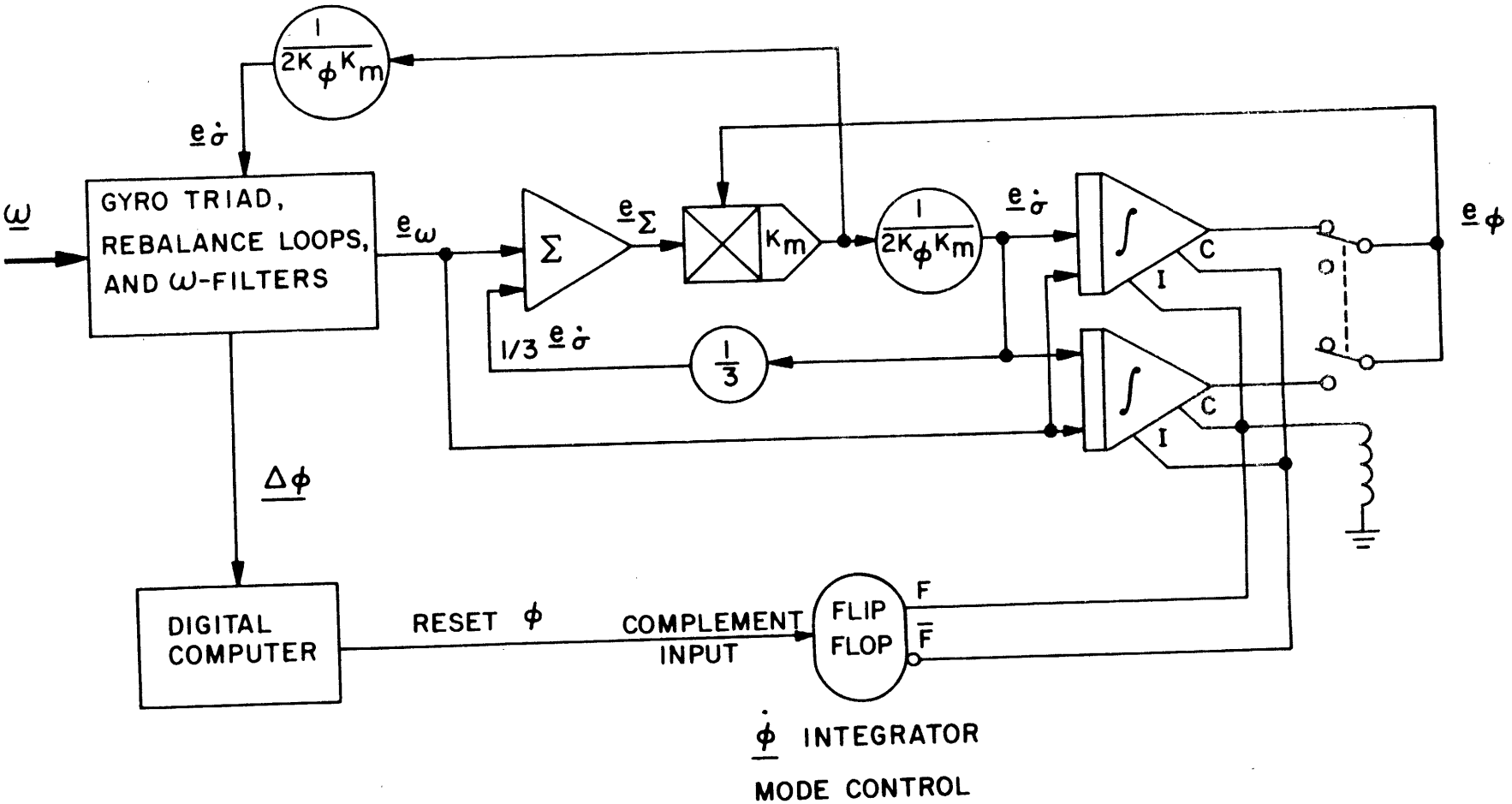


Figure 5.1.- Analog system mechanization

second (Table 4.1),

$$k_\omega = \frac{|\underline{\omega}_{MAX}|}{|\underline{e}_{\omega,MAX}|} = \frac{2 \text{ rad/sec}}{100 \text{ volts}}$$

$$k_\omega = 2\text{x}10^{-2} \text{ rad/sec/volt} \tag{5.18}$$

$\underline{\dot{\phi}}$ does not appear as an electrical signal in Figure 5.1, so the choice of $k_{\dot{\phi}}$ is immaterial. $\underline{e}_\omega$ and $\underline{e}_{\dot{\sigma}}$ are summed together to obtain $\underline{e}_{\dot{\phi}}$ and so $k_{\dot{\sigma}}$ is chosen to be the same as k_ω.

$$k_{\dot{\sigma}} = 2\text{x}10^{-2} \text{ rad/sec/volt} \tag{5.19}$$

Before choosing k_ϕ, an expression will be derived for the first neglected term in Eq. (5.13). Since

$$\underline{\dot{\phi}} = \underline{\omega} + \underline{\dot{\sigma}}$$

it is seen from Eq. (5.13) that

$$\underline{\dot{\sigma}} = \frac{1}{2}\,\underline{\phi}\text{x}\underline{\omega} + \frac{1}{6}\,\underline{\phi}\text{x}\underline{\dot{\sigma}}$$

$$= \frac{1}{2}\,\underline{\phi}\text{x}\underline{\omega} + \frac{1}{6}\,\underline{\phi}\text{ x}\left(\frac{1}{2}\,\underline{\phi}\text{x}\underline{\omega} + \frac{1}{6}\,\underline{\phi}\text{x}\underline{\dot{\sigma}}\right)$$

$$= \frac{1}{2}\,\underline{\phi}\text{x}\underline{\omega} + \frac{1}{12}\,\underline{\phi}\text{ x}(\underline{\phi}\text{x}\underline{\omega}) + \frac{1}{36}\,\underline{\phi}\text{ x}(\underline{\phi}\text{x}\underline{\dot{\sigma}})$$

Note from Eq. (5.13) that

$$\underline{\phi} \cdot \underline{\dot{\sigma}} = 0$$

so the last term of the preceeding equation becomes

$$\frac{1}{36}\,\underline{\phi}\text{ x}(\underline{\phi}\text{x}\underline{\dot{\sigma}}) = -\frac{\phi^2}{36}\,\underline{\dot{\sigma}}$$

and therefore

$$\dot{\underline{\sigma}} = \frac{1}{2}\,\underline{\phi}x\underline{\omega} + \frac{1}{12}\,\underline{\phi}\times(\underline{\phi}x\underline{\omega}) - \frac{\phi^2}{36}\,\dot{\underline{\sigma}}$$

$$\left(1 + \frac{\phi^2}{36}\right)\dot{\underline{\sigma}} = \frac{1}{2}\,\underline{\phi}x\underline{\omega} + \frac{1}{12}\,\underline{\phi}\times(\underline{\phi}x\underline{\omega})$$

$$\dot{\underline{\sigma}} = \left(1 + \frac{\phi^2}{36}\right)^{-1}\left[\frac{1}{2}\,\underline{\phi}x\underline{\omega} + \frac{1}{12}\,\underline{\phi}\times(\underline{\phi}x\underline{\omega})\right]$$

$$\dot{\underline{\sigma}} \approx \left(1 - \frac{\phi^2}{36}\right)\left[\frac{1}{2}\,\underline{\phi}x\underline{\omega} + \frac{1}{12}\,\underline{\phi}\times(\underline{\phi}x\underline{\omega})\right]$$

$$\dot{\underline{\sigma}} \approx \frac{1}{2}\,\underline{\phi}x\underline{\omega} + \frac{1}{12}\,\underline{\phi}\times(\underline{\phi}x\underline{\omega}) - \frac{\phi^2}{72}\,\underline{\phi}x\underline{\omega} - \frac{\phi^2}{432}\,\underline{\phi}\times(\underline{\phi}x\underline{\omega})$$

By comparing this result with Eq. (5.1) and using Eq. (5.6) and (5.9) it can be seen that the first neglected term in Eq. (5.13) is

$$\text{FNT} = \frac{\phi^3}{72}\,\underline{1}_{\phi}x\underline{\omega} \tag{5.20}$$

Eq. (5.20) shows that the maximum error magnitude in Eq. (5.13) is of the order of $\phi^3/72$ times that component of $\underline{\omega}$ which is perpendicular to $\underline{\phi}$. If ϕ_{MAX} is taken to be 0.1 radian, then the maximum error is 14 ppm times the component of $\underline{\omega}$ perpendicular to $\underline{\phi}$. Since ϕ will range anywhere from 0 to ϕ_{MAX} and since the average of $\underline{1}_{\phi}x\underline{\omega}$ will be somewhat less than ω_{MAX}, the choice of

$$\phi_{MAX} = 0.1 \text{ radian} \tag{5.21}$$

results in a probable error not more than $5\times10^{-6}|\underline{\omega}|$ and a guaranteed error of not more than $14\times10^{-6}|\underline{\omega}|$. With ϕ_{MAX} as given by Eq. (5.21), k_{ϕ} is then

$$k_{\phi} = \frac{0.1 \text{ rad}}{100 \text{ volts}}$$

$$= 10^{-3} \text{ rad/volt} \tag{5.22}$$

5.3 The Frequency of the Update

The hybrid computational technique obviates the need for high frequency updating for the purpose of maintaining an accurate direction cosine matrix. Accuracy is maintained by generating an analog correction signal $\underline{\dot{\sigma}}$ which accounts for the noncommutativity phenomenon encountered in finite rotations. This allows the accumulation of the $\Delta\underline{\phi}$ pulses from the gyros over arbitrarily long intervals of time, subject, of course, to the restriction that an update must occur when ϕ exceeds ϕ_{MAX}.

Since the direction cosine matrix is not an end in itself, but exists (a) in order to transform vectors from the Body Frame into the Reference Frame and (b) to describe the attitude of the vehicle; it should be these considerations which specify the frequency at which it must be updated.

It is important to obtain a good transformation of the accelerometer measurement from the Body to the Reference Frame. As in the case of the conventional direction cosine matrix update process, this transformation can be done by a simple application of the direction cosine matrix to each $\Delta\underline{V}$ (increment of integrated specific force) by accumulating the $\Delta\underline{V}$'s over a longer time interval and then applying a more sophisticated transformation algorithm. In the former approach, a sufficiently accurate direction cosine matrix must be available as often as the accelerometers are sampled. In the latter approach, the overall computational burden of the transformation process is reduced somewhat by using a more sophisticated

algorithm at a slower rate. Typical transformation rates range from 100 to 3000 transformations per second (ref. 15).

The navigation equations are integrated less frequently, with typical rates on the order of 1 integration step every 10 seconds (ref. 16). This means that from the point of view of navigation requirements, the transformation of velocity increments can take place at a slower rate than that given in the previous paragraph. Even the implementation of a guidance law (such as the nulling of the velocity-to-be-gained vector) does not require a high transformation rate since only very low vehicle angular rates (except perhaps for vehicle angular vibrations) would occur in such a guidance process.

The foregoing paragraph provides the rationale for the hybrid velocity transformation method presented in Chapter 7. There it is seen that a direction cosine matrix update frequency of

$$f_u = 10 \text{ updates/second} \tag{5.23}$$

is quite adequate for most velocity transformation purposes.

The important point is that in the hybrid method, it is the use of the coordinate transformation matrix that governs the frequency of the update, and not as in conventional methods where the accuracy of the updating process governs the frequency of the update. This is because in the hybrid method, the same accuracy is available at lower update frequencies.

5.4 System Error Analysis

There are three error sources in the hybrid computational process in addition to the gyro errors. These are:

(a) Equation mechanization error

(b) Analog computation error

(c) Gyro storage error

The coupling between the error sources is weak. Therefore, these sources are regarded as uncoupled. Each error source will now be examined in detail.

5.4.1 Equation Mechanization Error

This error was derived in Section 5.1 and was approximated by Eq. (5.20) which may be rewritten as

$$\delta\dot{\underline{\phi}}_e = \dot{\underline{\phi}}_a - \dot{\underline{\phi}} \cong \frac{\phi^3}{72}\,\underline{1}_\phi \times \underline{\omega} \tag{5.24}$$

where

$\delta\dot{\underline{\phi}}_e$ is the equation mechanization error

$\dot{\underline{\phi}}_a$ is the approximation to $\dot{\underline{\phi}}$ as given by Eq. (5.13)

An approximation to the rms angular velocity drift due to equation mechanization error is found as follows:

$$\left(\delta\dot{\phi}_e\right)^2 = \delta\dot{\underline{\phi}}_e \cdot \delta\dot{\underline{\phi}}_e = \frac{\phi^6}{72\cdot 72}\,\underline{1}_\phi \times \underline{\omega} \cdot \underline{1}_\phi \times \underline{\omega}$$

$$= \frac{\phi^6}{72\cdot 72}\,[\omega^2 - (\underline{1}_\phi \cdot \underline{\omega})^2]$$

Now suppose that ϕ grows linearily with time; i.e., that

$$\phi = kt$$

until

$$kt = 0.1 \text{ radian}$$

(at which time $\underline{\phi}$ is reset to zero). Suppose further that

$$\underline{1}_{\phi} \cdot \underline{\omega} = \frac{\sqrt{2}}{2}\,\omega$$

This is equivalent to an assumption that the component of $\underline{\omega}$ parallel to $\underline{\phi}$ is the same as that perpendicular to $\underline{\phi}$. Also suppose that $|\underline{\omega}|$ is a constant. Then

$$\delta\dot{\phi}^2(t) = \frac{k^6 t^6 \omega^2}{(72)^2 x 2}$$

This is averaged over one cycle from t = 0 to t = 0.1/k to get the mean square value of $\delta\phi$.

$$\overline{\delta\dot{\phi}^2} = \frac{1}{0.1/k} \int_0^{0.1/k} \frac{k^6 t^6 \omega^2}{72 \cdot 72 \cdot 2}\, dt$$

$$= \frac{10 k^7 \omega^2}{72 \cdot 72 \cdot 2} \left[\frac{t^7}{7}\right]_0^{0.1/k} = \frac{\omega^2}{(72)^2 x 2 x 7 x 10^6}$$

$$= (3.71 x 10^{-6})^2\, \omega^2$$

or

$$\delta\dot{\phi}_{e,rms} = 3.71 x 10^{-6}\, \omega \text{ rad/sec} \qquad (5.25)$$

Let $\delta\dot{\phi}_{SF}$ be the error in $\dot{\phi}$ caused by gyro scale factor uncertainty. In even the best strapdown gyro available today,

$$\delta\dot{\phi}_{SF} = 10^{-5}\, \omega$$

The value of $\delta\dot{\phi}_{e,rms}$ given by Eq. (5.25) compares favorably with this.

5.4.2 Analog Computation Error

In order to analyze the effects of the analog computing elements, it is necessary to assume an error model for each analog computing element. These models are:

Integrator

(a) Initial condition error $\delta\underline{\phi}_o$

(b) Input voltage offset $\underline{\alpha}$

(c) Dynamics $D_I(s)$

(d) Transfer function

$$\underline{v}_{out}(s) = \frac{D_I(s)}{s}\,[\underline{v}_{in}(s)+\underline{\alpha}(s)] + \delta\underline{\phi}_o$$

Summing Amplifier

(a) Input voltage offset $\underline{\beta}$

(b) Dynamics $D_\Sigma(s)$

(c) Transfer function

$$\underline{v}_{out}(s) = D_\Sigma(s)\,[\underline{v}_1(s)+\underline{v}_2(s)+\underline{\beta}]$$

Multiplier

(a) Multiplication error ε

(b) Output noise $\underline{\eta}$

(c) Dynamics $D_M(s)$

(d) Transfer function

$$\underline{v}_{out}(s) = D_M(s)\,[(1+\varepsilon)\underline{v}_1\underline{v}_2(s)] + \underline{\eta}$$

Notes

(a) It is not strictly proper to use Laplace transform notion for the multiplier error model since a multiplier is

a nonlinear computing element. It will be seen, however, that this error analysis is equivalent to a perturbation analysis and the nominal product may be cancelled by subtracting it from the unperturbed product. This will become clearer as the linearization proceeds. The result, after the nominal product is cancelled, is linear and the Laplace transformation notation is then appropriate.

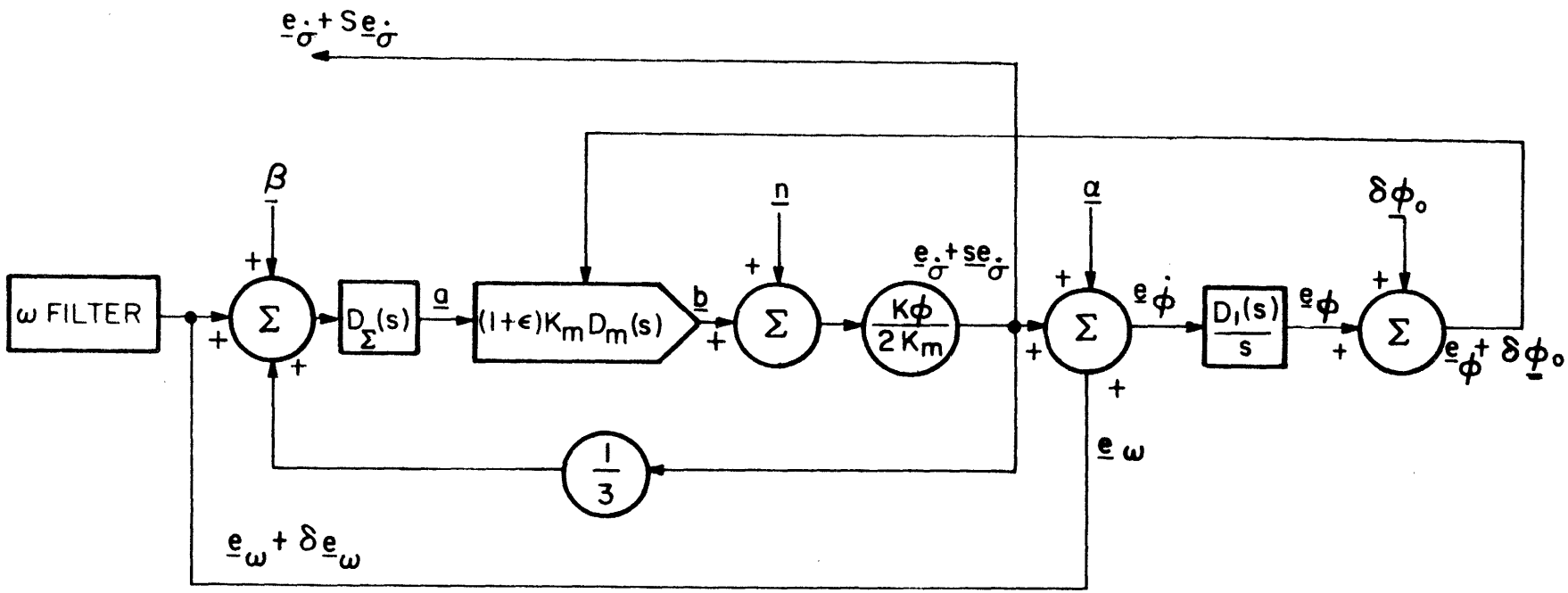
(b) No error model is assigned to the coefficient potentiometers. Instead, the summing amplifier noise $\underline{\beta}$ will be made large enough to account for this omission.

(c) Only one summing amplifier is shown in Figure 5.1 although the actual system mechanization (see Appendix D) contains several. Here again, $\underline{\beta}$ will be taken large enough to account for the noise contributed by the neglected summing amplifiers.

The analog computation system (except for the ω-Filter) is shown in Figure 5.2 with its error sources. Notice that the complementary integrators shown in Figure 5.1 have not been included in Figure 5.2. Since only one integrator is in the circuit at any one time, only one has been shown in Figure 5.2.

The object of this error analysis will be to evaluate the relationship between the error sources, symbolically denoted $\underline{d}(s)$, and the analog computational drift rate $\delta\dot{\underline{\sigma}}$. Mathematically

$$\delta\dot{\underline{\sigma}}(s) = F(s)\underline{d}(s) \tag{5.26}$$



$K_\phi = 10^{-3}$ RAD/VOLT

$K_m = 10^{-2}$ VOLTS^{-1}

Figure 5.2- Error sources in analog computation

where

$$\underline{d}^T = [\underline{\alpha}^T \underline{\beta}^T e \underline{\eta}^T \delta \underline{e}_\omega^T \delta \underline{\phi}_o^T] \tag{5.27}$$

From Figure 5.2

$$\underline{e}_{\dot{\sigma}} + \delta \underline{e}_{\dot{\sigma}} = \frac{1}{2} \frac{k_\phi}{k_m} (\underline{b}+\underline{\eta}) \tag{5.28}$$

$$\underline{b} = (\underline{e}_\phi + \delta \underline{\phi}_o) \times \underline{a}(1+\varepsilon) \; K_M D_M(s) \tag{5.29}$$

$$\underline{a} = D_\Sigma(s) \left[\underline{e}_\omega + \delta \underline{e}_\omega + \underline{\beta} + \frac{1}{3} (\underline{e}_{\dot{\sigma}} + \delta \underline{e}_{\dot{\sigma}}) \right] \tag{5.30}$$

$$\underline{e}_\phi + \delta \underline{\phi}_o = \frac{D_I(s)}{s} (\underline{e}_\omega + \delta \underline{e}_\omega + \underline{e}_{\dot{\sigma}} + \delta \underline{e}_{\dot{\sigma}} + \underline{\alpha}) + \delta \underline{\phi}_o \tag{5.31}$$

Combining Eqs. (5.28) - (5.31) and neglecting 2nd and higher order terms gives

$$\underline{e}_{\dot{\sigma}} + \delta \underline{e}_{\dot{\sigma}} = \frac{k_\phi}{2} \frac{D_1(s)}{s} \left(\frac{2}{3} \underline{e}_{\dot{\sigma}} x \underline{e}_\omega \right)(1+\varepsilon)$$

$$+ \frac{k_\phi}{2} \frac{D_1(s)}{s} \left[\underline{e}_\omega x \left(\underline{\beta} - \frac{2}{3} \delta \underline{e}_{\dot{\sigma}} - \underline{\alpha} \right) + \underline{e}_{\dot{\sigma}} x \left(\underline{\beta} + \frac{2}{3} \delta \underline{e}_\omega - \frac{1}{3} \underline{\alpha} \right) \right]$$

$$- \frac{k_\phi}{2} D_2(s) \left(\underline{e}_\omega + \frac{1}{3} \underline{e}_{\dot{\sigma}} \right) \times \delta \underline{\phi}_o + \frac{1}{2} \frac{k_\phi}{k_m} \underline{\eta}$$

where

$$D_1(s) = D_\Sigma(s) \; D_M(s) \; D_I(s)$$

$$D_2(s) = D_\Sigma(s) \; D_M(s)$$

Now subtract out the nominal value of $\underline{e}_{\dot{\sigma}}$ (set all error sources to zero to get nominal) and collect terms containing the factor $\delta \underline{e}_{\dot{\sigma}}$ on the left hand side. This gives

$$\delta \underline{e}_{\dot{\sigma}} + \frac{k_\phi}{3} \frac{D_1(s)}{s} \underline{e}_\omega \times \delta \underline{e}_{\dot{\sigma}}$$

$$= \frac{k_\phi}{3} D_1(s) \underline{e}_{\dot{\sigma}} \times \underline{e}_\omega \frac{\varepsilon}{s}$$

$$+ \frac{k_\phi}{2} D_1(s) \left[\underline{e}_\omega \times \frac{\underline{\beta} - \underline{\alpha}}{s} + \underline{e}_{\dot{\sigma}} \times \frac{\underline{\beta} + \frac{2}{3} \delta \underline{e}_\omega - \frac{1}{3} \underline{\alpha}}{s} \right]$$

$$- \frac{k_\phi}{2} D_2(s) \left(\underline{e}_\omega + \frac{1}{3} \underline{e}_{\dot{\sigma}} \right) \times \delta \underline{\phi}_o + \frac{1}{2} \frac{k_\phi}{k_m} \underline{\eta} \qquad (5.32)$$

The integration indicated by the 1/s factor refers always to the error sources. Thus the error $\delta \underline{e}_{\dot{\sigma}}$ tends to grow with time for non-zero error sources. In order to further simplify Eq. (5.32) assume that

$$\delta \underline{e}_{\dot{\sigma}} >> \frac{k_\phi}{3} D_1(s) \underline{e}_\omega \times \frac{\delta \underline{e}_{\dot{\sigma}}}{s}$$

Since $k_\phi = 10^{-3}$ rad/volt, this is a good approximation for most run times. Note that if $|\underline{e}_\omega|$ is large, the run time will be small and vice versa. The effect of the analog circuit dynamics can also be ignored since the response of all analog computing elements used in this experiment is flat to 20 KHz whereas the gyro response rolls off at 2 KHz. Further, since $|\underline{\phi}| \leq .1$, Eqs. (5.9) and (5.10) reveal that

$$|\dot{\underline{\sigma}}| << |\underline{\omega}|$$

and consequently $|\underline{e}_{\dot{\sigma}}| << |\underline{e}_\omega|$. In view of these assumptions Eq. (5.32) can be simplified to

$$\delta \underline{e}_{\dot{\sigma}} = \frac{k_\phi}{3} \underline{e}_{\dot{\sigma}} \times \underline{e}_\omega \frac{\varepsilon}{s} + \frac{k_\phi}{2} \underline{e}_\omega \times \frac{\underline{\beta}-\underline{\alpha}}{s}$$

$$+ \frac{k_\phi}{3} \underline{e}_{\dot{\sigma}} \times \frac{\delta \underline{e}_\omega}{s} - \frac{k_\phi}{2} \underline{e}_\omega \times \delta \underline{\phi}_o$$

$$+ \frac{1}{2} \frac{k_\phi}{k_m} \underline{\eta} \qquad (5.33)$$

The auto correlation function for $\delta \underline{e}_{\dot{\sigma}}$ is given by (ref. 17)

$$\phi_{\delta e_{\dot{\sigma}} \delta e_{\dot{\sigma}}}(\tau) = E[\delta \underline{e}_{\dot{\sigma}}(t) \cdot \delta \underline{e}_{\dot{\sigma}}(t+\tau)] \qquad (5.34)$$

where E is the expectation operator. In order to evaluate the auto correlation function it is necessary to transform Eq.(5.33) to the time domain. This operation gives

$$\delta \underline{e}_{\dot{\sigma}}(t) = \frac{k_\phi}{3} \underline{e}_{\dot{\sigma}} \times \underline{e}_\omega \int_o^t h(t-x)\ \varepsilon(x)\ dx$$

$$+ \frac{k_\phi}{2} \underline{e}_\omega \times \int_o^t h(t-x)[\underline{\beta}(x)-\underline{\alpha}(x)]\ dx$$

$$+ \frac{k_\phi}{3} \underline{e}_{\dot{\sigma}} \times \int_o^t h(t-x)\ \delta \underline{e}_\omega(x) dx$$

$$- \frac{k_\phi}{2} \underline{e}_\omega \times \delta \underline{\phi}_o(t) + \frac{1}{2} \frac{k_\phi}{k_m} \underline{\eta}(t) \qquad (5.35)$$

where h(t) is the unit step function u(t), the impulse response of an integrator. Now use Eq. (5.35) in Eq. (5.34) and reduce using the following assumptions:

(a) The noise sources are unbiased and uncorrelated

(b) The three components of $\underline{\alpha}$, $\underline{\beta}$, $\delta \underline{e}_\omega$, $\delta \underline{\phi}_o$, and $\underline{\eta}$ are equal ($\alpha_x = \alpha_y = \alpha_z$, etc.).

(c) Assume

$$E[\underline{e}_\omega \cdot \underline{\alpha}] = 0$$

$$E[\underline{e}_\omega \cdot \underline{\beta}] = 0$$

$$E[\underline{e}_\sigma \cdot \delta\underline{e}_\omega] = 0$$

$$E[\underline{e}_\omega \cdot \delta\underline{\phi}_o] = 0$$

(d) All error sources are Gaussian white noise processes such that

$$\phi_{\varepsilon\varepsilon}(t) = Q_\varepsilon \delta(t)$$

$$\phi_{\alpha\alpha}(t) = 3Q_\alpha \delta(t)$$

$$\phi_{\beta\beta}(t) = 3Q_\beta \delta(t)$$

$$\phi_{\delta e_\omega \delta e_\omega}(t) = 3Q_{\delta e_\omega}(t)$$

$$\phi_{\delta\phi_o \delta\phi_o}(t) = Q_{\delta\phi_o}\delta(t)$$

$$\phi_{\eta\eta}(t) = Q_\eta \delta(t)$$

where $\delta(t)$ is the unit impulse and Q_i is the variance parameter of the i^{th} process. The factor 3 arises because of assumption (b).

Using these assumptions, a lengthy reduction gives

$$\phi_{\delta e_{\dot\sigma} \delta e_{\dot\sigma}}(t) = \sum_{i=1}^{5} \phi_i(t) \tag{5.36}$$

where

$$\phi_1(t) = \frac{k_\phi^4}{36} e_\omega^4 e_\phi^2 Q_\varepsilon k_t t \tag{5.37}$$

$$\phi_2(t) = \frac{3}{4} k_\phi^2 e_\omega^2 (Q_\alpha + Q_\beta)\ k_t t \tag{5.38}$$

$$\phi_3(t) = \frac{1}{4} k_\phi^4 e_\omega^2 e_\phi^2 Q_{\delta e_\omega} k_t t \tag{5.39}$$

$$\phi_4(t) = \frac{3}{4} k_\phi^2 e_\omega^2 Q_{\delta\phi_o} \tag{5.40}$$

$$\phi_5(t) = \frac{3}{4} \frac{k_\phi^2}{k_m^2} Q_\eta \tag{5.41}$$

$$k_t = 1 \text{ second} \tag{5.42}$$

It is clear from Eqs. (5.37) - (5.41) that $\phi_{\delta e_{\dot\sigma} \delta e_{\dot\sigma}}(t)$ is not a stationary time function even if each noise source is a stationary random process. This is because of the integration of some of the noise signals. The rms value of a Gaussian white noise process passed through an integration grows as the square root of time. (See Appendix C for a plot of the results of integrating an input angular velocity corrupted by Gaussian white noise.)

These results can be put in a more practical form. Let

$$\delta\dot\sigma_{rms} = k_{\dot\sigma} \delta e_{\dot\sigma, rms}$$

$$= k_{\dot\sigma} \left[\phi_{\delta e_{\dot\sigma} \delta e_{\dot\sigma}}(t) \right]^{1/2}$$

$$n_{i,rms} = Q_i^{1/2}$$

where $n_{i,rms}$ is the rms value of the i^{th} noise source. Then considering only one noise source at a time and using

$$\omega = k_\omega e_\omega$$

$$e_\phi = (k_\omega / k_\phi) e_\omega t$$

the following results are obtained

$$\delta\dot{\sigma}_{rms}(\varepsilon) = \frac{k_\phi}{k_\omega} \frac{\omega^3}{6} \varepsilon_{rms}(k_t t^3)^{1/2} \tag{5.43}$$

$$\delta\dot{\sigma}_{rms}(\alpha) = \frac{\sqrt{3}}{2} k_\phi \omega \alpha_{rms}(k_t t)^{1/2} \tag{5.44}$$

$$\delta\dot{\sigma}_{rms}(\beta) = \frac{\sqrt{3}}{2} k_\phi \omega \beta_{rms}(k_t t)^{1/2} \tag{5.45}$$

$$\delta\dot{\sigma}_{rms}(\delta e_\omega) = \frac{1}{2} k_\phi \omega^2 \delta e_{\omega,rms}(k_t t^3)^{1/2} \tag{5.46}$$

$$\delta\dot{\sigma}_{rms}(\delta\phi_o) = \frac{3}{2} k_\phi \omega \delta\phi_{o,rms} \tag{5.47}$$

$$\delta\dot{\sigma}_{rms}(\eta) = \frac{3}{2} \frac{k_\phi}{k_m} k_\omega \eta_{rms} \tag{5.48}$$

Typical values are given in Table 5.1 for the noise sources considered in this section. Table 5.2 was constructed using the value t = 0.5 sec and the values for k_ω and k_ϕ given in Eqs. (5.18) and (5.22).

TABLE 5.1

TYPICAL NOISE SOURCE VALUES

Source	ε	α	β	δe_ω	$\delta\phi_o$	η
RMS Value	0.01	2.5×10^{-3}	2.5×10^{-3}	1	5×10^{-3}	7.5×10^{-3}

RMS value is in volts except for ε which is dimensionless.

TABLE 5.2

RMS DRIFT IN ANALOG COMPUTATION

Argument	$\delta\dot{\sigma}_{rms}(A)$		
A	$\omega=0$	$\omega=1$	$\omega=2$
ε	0	3×10^{-5}	2.4×10^{-4}
α	0	1.5×10^{-6}	3.0×10^{-6}
β	0	1.5×10^{-6}	3.0×10^{-6}
δe_{ω}	0	3.5×10^{-4}	1.4×10^{-3}
$\delta\phi_{o}$	0	4.5×10^{-6}	9.0×10^{-6}
η	1.4×10^{-5}	1.4×10^{-5}	1.4×10^{-5}

Units of $\delta\dot{\sigma}_{rms}$ are radians/second

From Table 5.2 it is clear that the multiplier noise sources ε and $\underline{\eta}$ are the most serious sources. This is because all other noise sources must pass through a multiplication by ϕ and as a result are scaled by k_{ϕ}.

5.4.3 Gyro Storage Error

At each evaluation of C^{RB}, there is an error in C^{RB} due to the quantization of $\underline{\phi}$. This error doesn't become permanent until $\underline{\phi}$ is reset to zero. The digital computer computes

$$C^{NB}(t) = C^{NR}C^{RB}(t) \tag{5.47}$$

where C^{NR} is the initial condition matrix which premultiplies C^{RB}. This initial condition matrix relates the Reference Frame to some specific Navigation Frame and is given by

$$C^{NR} = C^{NB}(t_r) \tag{5.48}$$

where t_r is the time at which $\underline{\phi}_{RB}$ was last reset to zero. Thus the quantization error in $\underline{\phi}$ doesn't become incorporated into C^{NR} until $\underline{\phi}_{RB}$ is reset to zero.

Furthermore, the only recoverable part of the quantization error is the error in $\underline{\dot{\sigma}}$. Since

$$\Delta\underline{\phi} = \int_{t}^{t+\Delta T} (\underline{\omega}(\tau)+\dot{\sigma}(\tau))\ d\tau$$

it can be seen that the only part of $\Delta\underline{\phi}$ that is a function of $\underline{\phi}$ is

$$\Delta\underline{\dot{\sigma}} = \int_{t}^{t+\Delta T} \dot{\sigma}(\tau)d\tau = \int_{t}^{t+\Delta T} \underline{\phi}\ x\left(\frac{1}{3}\ \omega + \frac{1}{6}\ \underline{\dot{\phi}}\right)d\tau$$

The rms magnitude of the quantization error e(q)

$$e_{rms}(q) = \left(\frac{\Delta\phi^2}{2}\right)^{1/2} \left(\frac{\Delta\theta^2+\Delta\sigma^2}{2}\right)^{1/2}$$

This is a result of the assumption that the quantization error is uniformly distributed over the interval $(0,\Delta\phi)$. The part of the quantization error due to the quantization of $\underline{\dot{\sigma}}$ is approximately

$$e_{rms}(q_{\Delta\sigma}) = \frac{\Delta\sigma}{\Delta\theta} \left(\frac{\Delta\phi^2}{2}\right)^{1/2}$$

This can be approximated by

$$e_{rms}(q_{\Delta\sigma}) = \frac{\dot{\sigma}}{\omega} \left(\frac{\Delta\phi^2}{2}\right)^{1/2} \qquad (5.49)$$

To evaluate Eq. (5.49) as a function of quantization level $\Delta\phi$, an expression for the relative magnitudes of $\underline{\dot{\sigma}}$ and $\underline{\omega}$ must be obtained. From Eqs. (5.9) and (5.13) it is seen that

$$\underline{\dot{\sigma}} \cong \frac{1}{2}\ \underline{\phi}x\underline{\omega}$$

and so as an approximate relationship

$$\dot{\sigma}^2 = \frac{\phi^2\omega^2}{4} - \frac{(\underline{\phi} \cdot \underline{\omega})^2}{2}$$

As an extreme case $\underline{\phi} \cdot \underline{\omega} = 0$. Then

$$\dot{\sigma}^2 = \frac{\phi^2\omega^2}{4}$$

and

$$\frac{\sigma}{\omega} = \frac{\phi}{2}$$

so Eq. (5.49) may be written

$$e_{rms}(q_{\Delta\sigma}) = \frac{\phi}{2}\left(\frac{\Delta\phi^2}{2}\right)^{1/2}$$

The quantization error is incorporated in the initial condition matrix when $\underline{\phi}$ is reset to zero. In the experimental system, $\phi = .1$ when reset, so as a final result

$$e_{rms}(q_{\Delta\sigma}) = \frac{0.05}{\sqrt{2}} \Delta\phi \qquad (5.50)$$

5.5 Test Results

An experiment was conducted to verify the theoretical results of Chapter 3. The experimental system consisted of:

(a) An inertial sensing unit consisting of three Honeywell DDG 334A gyros.

(b) A set of ω-Filters of the design described in Appendix B.

(c) An analog computer patched as described in Appendix D to generate $\underline{\dot{\sigma}}$; to transmit the $\Delta\phi$ pulses and the basic clock frequency (f_c = 3600 pulses/sec. This was scaled in the analog

computer logic section by a factor of 36) to the digital computer; and to synchronize the $\underline{\phi}$ resetting process in both the digital and analog computers.

(d) A digital computer to accumulate the $\Delta\phi$ pulses and to periodically evaluate $C^{NB}(t)$.

The results of the actual experiment were compared with the results of the numerical integration of the $\dot{\underline{\phi}}$ equation given in Appendix C. Two situations were tested. They were

1. An initial value of $\underline{\phi}$ on one axis and an angular velocity about another axis.

2. Out of phase sinusoids about mutually orthogonal axis.

The test results are shown in Table 5.3 and Table 5.4. In these tables, the error in the noncommutativity correction generation is indicated at the right hand side of the page by ϵ_{nc}. In Table 5.3, one error component in each run is labelled "ω" at the right hand side. This error is due to the inexact application of the input along that axis and is not an error in the hybrid computation since the computation responds to the input. In all cases the noncommutativity correction error was less than a one pulse error ($\Delta\phi = 5.555 \times 10^{-4}$). In a pulse rebalanced gyro, a one pulse error is to be anticipated because of gyro storage and quantization. The storage error is a function of the threshold of the relay in the rebalance loop and the quantization error is a function of the pulse weight.

TABLE 5.3

TEST RESULTS CONSTANT ω

$\phi(0)$ radians	ω rad/sec	Run Time seconds	$\underline{\phi}(t_f)$ Experimental	Theoretical	Error $\underline{\phi}_{ex} - \underline{\phi}_{th}$	
0.1	0.0		0.100000	0.099917	0.000083	
0.0	-0.2	1/2	-0.099444	-0.099917	0.000473	ω
0.0	0.0		-0.005000	-0.005000	0.000000	←nc
0.1	0.0		0.100000	0.099917	0.000083	
0.0	-0.1	1	-0.099444	-0.099917	0.000473	ω
0.0	0.0		-0.005000	-0.005000	0.000000	←nc
0.1	0.0		0.100000	0.099917	0.000083	
0.0	0.0	10	-0.005555	-0.005000	-0.000555	←nc
0.0	0.01		0.100555	0.099917	0.000638	ω
0.0	0.1		0.100555	0.099917	0.000638	ω
0.1	0.0	1	0.100000	0.099917	0.000083	←nc
0.0	0.0		-0.005000	-0.005000	0.000000	
0.0	0.01		0.099444	0.099917	-0.000473	ω
0.0	0.0	10	0.005000	0.005000	0.000000	←nc
0.1	0.0		0.100000	0.099917	0.000083	

$\Delta\phi = 0.00055555$

TABLE 5.4

TEST RESULTS-CONING MOTION

Coning Amplitude	Coning Ang. Freq.	Run Time	Drift Angle ψ		Error
		seconds	Experimental	Theoretical	$\psi_{exp}-\psi_{th}$
θ	ω	t		$1/2\theta^2\omega t$	
.02	10	1	0.002222	0.002000	0.000222 ←nc
.02	10	1	0.001666	0.002000	-0.000333 ←nc
.02	10	10	0.024444	0.020000	0.004444 ←nc
.002	100	10	0.001666	0.002000	-0.000333 ←nc

$\Delta\phi = 0.00055555$

Note: Drift Angle ψ refers to the drift in orientation (about an axis which experiences no angular velocity) induced by the coning process. The theoretical value is the predicted value. The experimental value is the coning drift correction generated by the system.

Test Note

The test was performed with the gyro wheels off. Electrical torques were applied to the torque summing member to simulate the torques due to physical inputs. A test table was not available to generate an accurate coning motion profile. The sine wave oscillator used to generate the out-of-phase sinusoids to simulate coning motion was not entirely free of bias, so the mean value of $\underline{\phi}$ generated by the ϕ integrators tended to increase rather rapidly, thus preventing low frequency coning experiments or long time duration experiments. Also, there were no accelerometers on the inertial sensing unit. The gyros were operated at their normal operating temperature, but were afforded only coarse temperature control. Without the wheel and accelerometer heat sources, the inertial measurement unit block temperature distribution was too abnormal to allow fine temperature control by the gyro temperature controllers acting alone.

CHAPTER 6

EVALUATION OF THE HYBRID CONCEPT

6.1 Preliminary Considerations

In this chapter, the hybrid computation will be compared with conventional schemes. It is necessary to define those performance features to be compared, since when any comparisons are made, it is important to correctly choose the basis upon which those comparisons are made.

It sometimes happens that error analyses treat those errors that admit readily of mathematical formulation and the physical significance, the sources, or the meaning of an error is often not adequately understood. An example of this practice occurs in the evaluation of strapdown coordinate transformation computation algorithms. The example is the evaluation of the extent to which the computed direction cosine matrix loses its orthogonality property. That is, the direction cosine matrix should be an orthogonal matrix. Accordingly, it should have a unity determinant, and the rows (columns) should be mutually orthogonal. It is easy to analyze mathematically, the growth of the determinant and the loss of perpendicularity of the row (column) vectors, and such an analysis is commonly performed. A better way, however, would

be to impose orthogonality as a constraint and to periodically perform a computation that reorthogonalizes the direction cosine matrix. Then (as makes good physical sense) the overall drift of the computed Navigation Frame relative to the true Navigation Frame is the only criterion for algorithm performance.

There are two ways in which a computed direction cosine matrix can become nonorthogonal. Let

$$C^{NB} = C^{NR}C^{RB} \tag{6.1}$$

where the N-Frame is the Navigation Frame in which the navigation problem is to be solved, the R-Frame is the Reference Frame which is taken to be the initial coordinate frame from which the change in orientation of the Body Frame is reckoned over the current updating cycle, the B-Frame is the Body Frame. The nonorthogonality modes are:

(1) If C^{NR} and C^{RB} are orthogonal matrices to within the limits of computer precision, then C^{NB} tends to become nonorthogonal in the round-off process that occurs when C^{NR} and C^{RB} are multiplied together.

(2) The algorithm that generates C^{RB} does not generate an orthogonal matrix.

Mode (1)

In the hybrid algorithm, a new Reference Frame is established when only $\underline{\phi}_{RB}$ is reset to zero. Therefore the average frequency at which nonorthogonality mode (1) generates an error that becomes permanently incorporated in C^{NR} is

$$f_{hyb}(1) = \omega_{avg}/\phi_{max}$$

where ω_{avg} is the average angular velocity magnitude.

For the scaling used in the thesis experiment, ϕ_{max} = 0.1 rad and ω_{avg} = 1 rad/sec and so

$$f_{hyb}(1) \leqq 20/\text{sec}$$

In purely digital algorithms, a new Reference Frame is established at each update cycle. Therefore

$$f_{dig}(1) = f_{update}$$

Mode (2)

The hybrid digital algorithm generates an orthogonal matrix. In the hybrid system, Eq. (3.18) is mechanized; its orthogonality is established by Eqs. (3.23) and (3.24). Therefore nonorthogonality mode (2) does not occur in the hybrid evaluation of C^{RB}. In the case of the digital algorithm, nonorthogonality mode (2) is a direct function of the specific algorithm used. Thus

$$f_{hyb}(2) = 0$$

$$f_{dig}(2) = f_{update}$$

6.2 Round-Off Error

Round-off error in the direction cosine matrix computation occurs because of the finite word length in the digital computer. In addition to causing nonorthogonality, round-off error also causes drift in the orientation of the computed Navigation Frame with respect to the actual Navigation Frame. Round-off error is not discussed further here except to note

that as for other arithmetic (as opposed to kinematic) errors in the direction cosine matrix, the round-off error in the hybrid algorithm accumulates at the rate at which $\underline{\phi}_{RB}$ is reset to zero. In purely digital algorithms, the round-off error accumulates at the rate at which the updating process occurs. For a good discussion of round-off error, see Reference 17.

6.3 Kinematic Response

From Eq. (5.9),

$$\dot{\underline{\phi}}_{RB} = \underline{\omega}_{RB} + \dot{\underline{\sigma}}_{RB}$$

it is seen that the angular velocity experienced by the strap-down inertial measurement unit gives rise to either a zero or a non-zero noncommutativity rate $\dot{\underline{\sigma}}_{RB}$. When $\dot{\underline{\sigma}}_{RB} = \underline{0}$, there is no kinematic coupling of the angular velocity component along one axis into an orientation rate about another axis. In this case, the direction of $\underline{\omega}_{RB}$ is fixed. If the direction of $\underline{\omega}_{RB}$ changes with time, then $\dot{\underline{\sigma}}_{RB} \neq \underline{0}$. For example, if

$$\underline{\omega}_{RB} = \begin{bmatrix} -\omega_c \sin\phi \sin\omega_c t \\ \omega_c \sin\phi \cos\omega_c t \\ (1-\cos\phi)\,\omega_c \end{bmatrix} \tag{6.2}$$

where ω_c is called the coning frequency, then for

$$\underline{\phi}(t_o) = \begin{bmatrix} \phi \\ 0 \\ 0 \end{bmatrix}$$

it can be shown by direct substitution into Eq. (5.2)

$$\dot{\underline{\phi}}_{RB} = \underline{\omega}_{RB} + \frac{1}{\phi_{RB}^2}(1-\cos\phi_{RB})\,\underline{\phi}_{RB}\times\dot{\underline{\phi}}_{RB} + \frac{1}{\phi_{RB}^2}\left(1-\frac{\sin\phi_{RB}}{\phi_{RB}}\right)\underline{\phi}_{RB}\times(\underline{\phi}_{RB}\times\dot{\underline{\phi}}_{RB}) \tag{5.2}$$

that

$$\underline{\phi}_{RB}(t) = \begin{bmatrix} \phi\cos\omega_c t \\ \phi\sin\omega_c t \\ 0 \end{bmatrix} \tag{6.3}$$

is a solution for the orientation resulting from the $\underline{\omega}_{RB}$ given by Eq. (6.2). This is the classical coning motion. It is readily seen from Eqs. (6.1) - (6.3) that

$$\dot{\sigma}_z = -(1-\cos\phi)\omega_c \tag{6.4}$$

since $\phi_z = 0$. The case where $\underline{\phi}\cdot\underline{\omega} = 0$ (as in the classical coning situation) results in a maximum kinematic coupling of angular velocity with orientation and $|\dot{\underline{\sigma}}_{RB}|$ is a maximum for a given $|\underline{\omega}_{RB}|$. Thus, coning motion provides an excellent set of circumstances for testing the accuracy of direction cosine algorithms since there is a known closed form solution for the coning motion direction cosine matrix. C^{RB} is evaluated by means of Eq. (3.18) using $\underline{\phi}_{RB}$ as given by Eq. (6.3).

In summary, an algorithm may be completely evaluated by two test cases: (a) no kinematic coupling; $|\dot{\underline{\sigma}}_{RB}(\underline{\omega}_{RB})| = 0$ (the direction of $\underline{\omega}_{RB}$ is fixed) and (b) maximum kinematic coupling; $|\dot{\underline{\sigma}}_{RB}(\underline{\omega}_{RB})|$ is a maximum for a given $|\underline{\omega}_{RB}|$ (classical coning motion).

6.4 Performance Comparison

Four algorithms have been selected for performance comparison. They are:

1. First order Taylor series

$$C^{NB} = C^{NR}(I+[\underline{\theta}x]) \tag{6.5}$$

2. Second order Taylor series

$$C^{NB} = C^{NR}(I+[\underline{\theta}x] + \frac{1}{2}[\underline{\theta}x]^2) \tag{6.6}$$

3. Second order Runge-Kutta

$$C^{NB} = C^{NR}(I+[\underline{\theta}x] + 5[\underline{\theta}_1 x][\underline{\theta}_2 x] - \frac{3}{2}[\underline{\theta}x]^2 - \frac{3}{2}[\underline{\theta}_2 x]^2) \tag{6.7}$$

4. Hybrid

$$C^{NB} = C^{NR}(I+P[\underline{\phi}x] + Q[\underline{\phi}x]^2) \tag{6.8}$$

where

$\underline{\theta}$ and $\underline{\phi}$ are the sum of the $\Delta\underline{\theta}$'s and $\Delta\underline{\phi}$'s respectively over the update interval T.

$\underline{\theta}_1$ and $\underline{\theta}_2$ are the sum of $\Delta\underline{\theta}$'s over the first and second half respectively of the update interval T

$$P = 1 - \frac{\phi^2}{6} + \frac{\phi^4}{120} - \frac{\phi^6}{5040} \cong \frac{\sin\phi}{\phi}$$

$$Q = P^2\left(2 - \frac{\phi^2}{2} + \frac{\phi^4}{24} - \frac{\phi^6}{720}\right)^{-1} \cong \frac{\sin^2\phi}{\phi^2(1-\cos\phi)}$$

The relative computer loading factor (RLF) per update for each algorithm is shown in column 2 of Table 6.1. The rectangular integration algorithm (First order Taylor series)

TABLE 6.1

ALGORITHM COMPLEXITY AND PERFORMANCE

1	2	3	4	5	6
Algorithm	Relative Load Factor	Drift Rate $\underline{\omega}=0$	Drift Rate $\underline{1}_{\omega}$ = const.	Drift Rate Coning $f_u > 2f_c$	Drift Rate Coning $f_u < 2f_c$
1st Order Taylor Series	1	0	$\frac{1}{2}\,\omega^2/f_u$	$\frac{1}{8}\,\frac{\phi^2\omega_c^2}{f_u}$	$\frac{1}{2}\,\phi^2\omega_c$
2nd Order Taylor Series	1.5	0	$\frac{1}{4}\,\omega^2/f_u$	$\frac{1}{16}\,\frac{\phi^2\omega_c^2}{f_u}$	$\frac{1}{2}\,\phi^2\omega_c$
2nd Order Runge-Kutta	2.3	0	$\frac{1}{10}\,\omega^3/f_u^2$	$\frac{1}{40}\,\frac{\phi^2\omega_c^3}{f_u^2}$	$\frac{1}{2}\,\phi^2\omega_c$
Hybrid	1.9	1.4×10^{-5}	1.4×10^{-5} $+3.5\times10^{-7}\omega^3$	$1.4\times10^{-5}+\frac{\tau_g^2\omega_c^2}{1+\tau_g^2\omega_c^2}\,\frac{1}{2}\,\phi^2\omega_c$	

f_u = update frequency

f_c = coning frequency

$\underline{\omega}$ = angular velocity

ϕ = coning amplitude

$\omega_c = 2\pi f_c$ (rad/sec)

τ_g = gyro float time constant

Drift Rates units are radian/second

is the simplest of all the direction cosine algorithms and so it has been assigned an RLF per update of unity.

The drift rate for $\underline{\omega}_{RB} = \underline{0}$ is given in Column 3 of Table 6.1. In each of the all digital algorithms, Eqs. (6.5)-(6.7), all terms but the matrix I are zero for $\underline{\omega}_{RB} = \underline{0}$ (since this results in $\theta_{RB} = \underline{0}$). Table 5.2 shows that for $\underline{\omega}_{RB} = \underline{0}$ the hybrid algorithm analog section has a drift rate (for the set of system parameters chosen in Chapter 5) given by

$$|\delta\dot{\underline{\sigma}}(\underline{n})| = 1.4\text{x}10^{-5} \text{ rad/sec}$$

Since $\delta\dot{\underline{\sigma}} \neq \underline{0}$, $\underline{\phi}_{RB}$ is therefore not zero. This shows that when $\underline{\omega}_{RB}$ is small, hybrid system performance is inferior to pure digital system performance. There are two ways to reduce this hybrid error. Since it is directly proportional to $k_{\dot{\sigma}}$, the noncommutativity correction scale factor, reducing the scaling of $\underline{e}_{\dot{\sigma}}$ would reduce this error proportionately. The second way is to create an open circuit in the $\dot{\underline{\sigma}}$ signal path when $|\dot{\underline{\sigma}}|$ is smaller than some predetermined value. These suggestions were not implemented in the experiment.

The algorithm drift rate given in columns 3-6 refers to drift generated by the dominant source of error in each algorithm, exclusive of round-off error, where drift is defined as the magnitude of the orientation vector relating the actual Navigation Frame to the computed Navigation Frame. Column 4 shows the algorithm drift rate in radians per second as a function of angular velocity and update frequency for the case of fixed angular velocity, exclusive of quantization effects. Column 5 gives the algorithm drift rate for coning motion at

coning frequencies that can be discerned by the rate approximation process performed by the algorithm. Column 6 is the algorithm drift rate for coning frequencies above those that can be discerned. This is the full kinematic rate $\dot{\sigma}_z$ (Eq.(6.4) for small ϕ) generated by the coning process.

The drift rate for the hybrid method is the sum of the drift rate (1.4×10^{-5} rad/sec) generated by the multiplier noise (Table 5.2) and a function of angular velocity. When $\underline{1}_\omega$ = constant, the dominant error function of angular velocity is $k_\omega \delta\dot{\sigma}_{rms}(\delta e_\omega)$ as given by Eq. (5.46). (The choice of parameters was $k_\omega \delta e_\omega = 0.01\omega$ and $t = \frac{1}{2}$.) In the case of coning, the dominant error is caused by the roll off in the frequency response of the gyro. This is described by the amplitude response function

$$\omega^* = \frac{1}{\left(1+\tau_g^2\omega_c^2\right)^{1/2}} \omega_{IA}$$

where ω^* is the measured value of ω_{IA}

τ_g is the time constant of the gyro

ω_c is the coning angular frequency.

At low coning angular frequencies, $\omega^* \cong \omega_{IA}$ and the coning correction is accurately generated. At high coning angular frequencies, the coning drift rate is the coning rate minus the coning correction. This is given by

$$\omega_c = \frac{1}{2}\phi^2\omega_c - \left(\frac{1}{1+\tau_g^2\omega_c^2}\right)\frac{1}{2}\phi^2\omega_c$$

$$= \left(\frac{\tau_g^2\omega_c^2}{1+\tau_g^2\omega_c^2}\right)\frac{1}{2}\phi^2\omega_c$$

In Figure 6.1, relative computer loading is plotted against $|\underline{\omega}_{RB}|$ for four algorithms. Drift rate is held constant at the drift rate of the hybrid algorithm which was arbitrarily assigned a relative computer loading of unity. (In all figures, only the drift rate attributable to the dominant kinematic algorithm deficiency is plotted.) The curves above $|\underline{\omega}_{RB}| = \omega_{max}$ are immaterial, since ω_{max} is the maximum angular velocity to which the system may be subjected. They are drawn to show that the hybrid algorithm is superior to the 2nd order Runge-Kutta algorithm over a very wide dynamic range. For a given vehicle and its dynamic motion specification, the hybrid system is scaled to favorably locate the range of $|\underline{\omega}_{RB}|$ in which it offers better performance than the all digital algorithms. The scale factor to change to relocate the region of superior performance is $k_{\dot{\sigma}}$ where

$$\underline{\dot{\sigma}} = k_{\dot{\sigma}} \underline{e}_{\dot{\sigma}}$$

Figure 6.2 is a plot of algorithm drift rate vs $|\underline{\omega}_{RB}|$ for equal computer loading. The computer loading of the hybrid algorithm was taken as the standard. For a certain range of $|\underline{\omega}_{RB}|$, the superiority of the hybrid method is quite clear.

In Figure 6.3, the coning performance of the hybrid computation is compared with that of the 2nd order Runge-Kutta algorithm (the most efficient of the digital algorithms). The solid diagonal lines show relative loading for constant coning amplitude. These lines show that the hybrid algorithm is more efficient at higher coning frequencies than the digital algorithm. This is misleading however, since the power required to generate

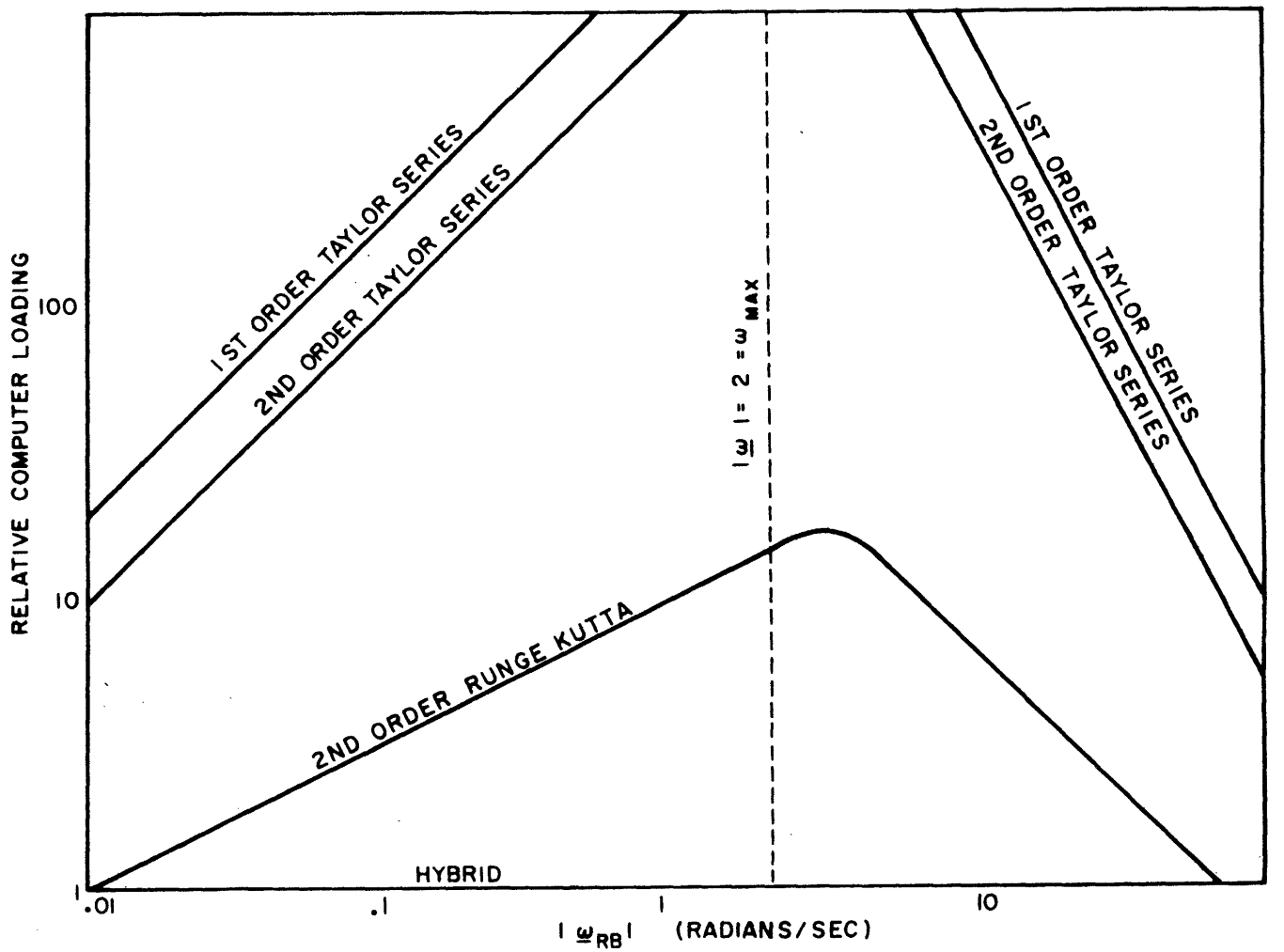


Figure 6.1.- $\underline{1}_{\omega}$ constant relative loading for equal accuracy

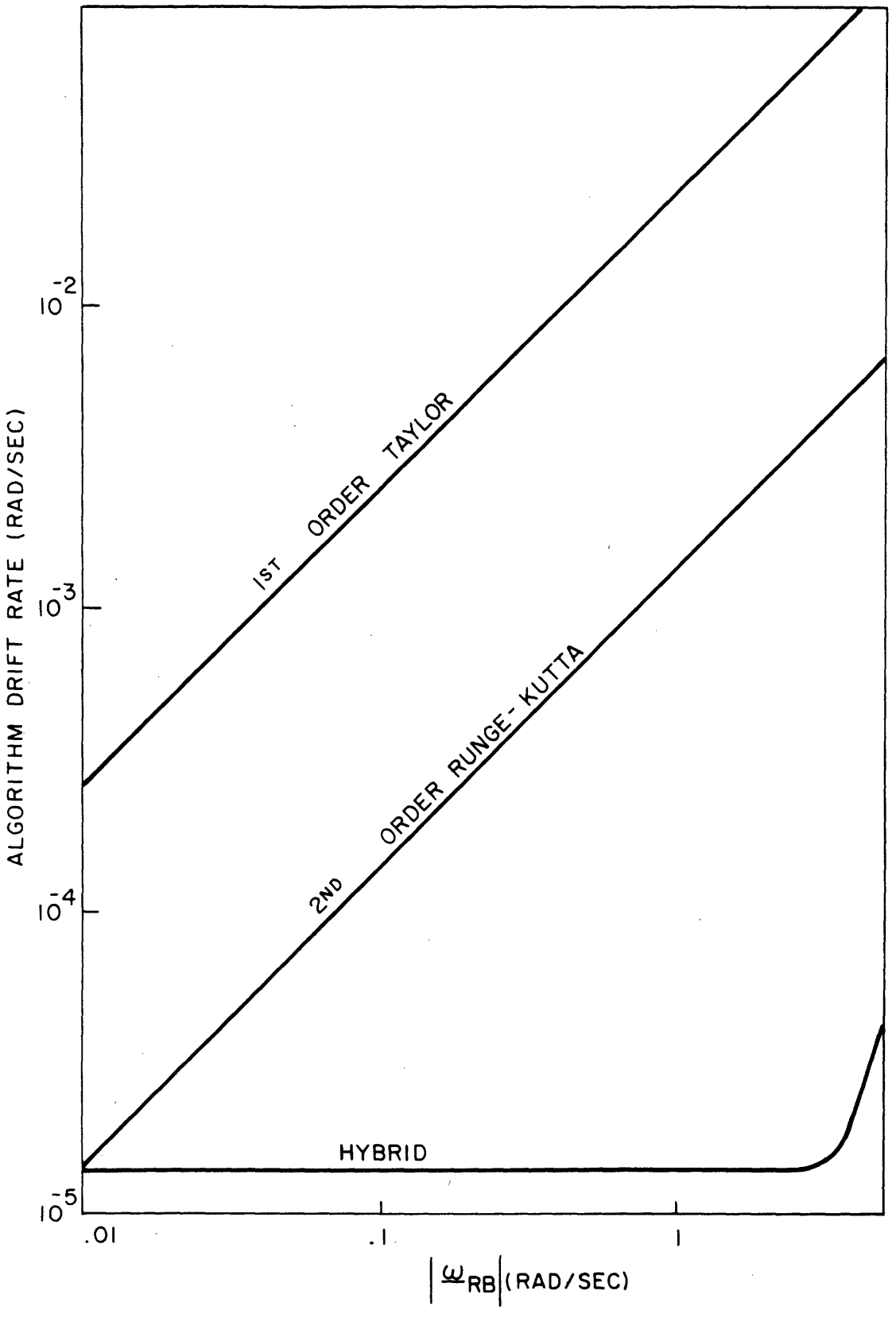


Figure 6.2.- Drift rate vs angular velocity magnitude for equal computer load

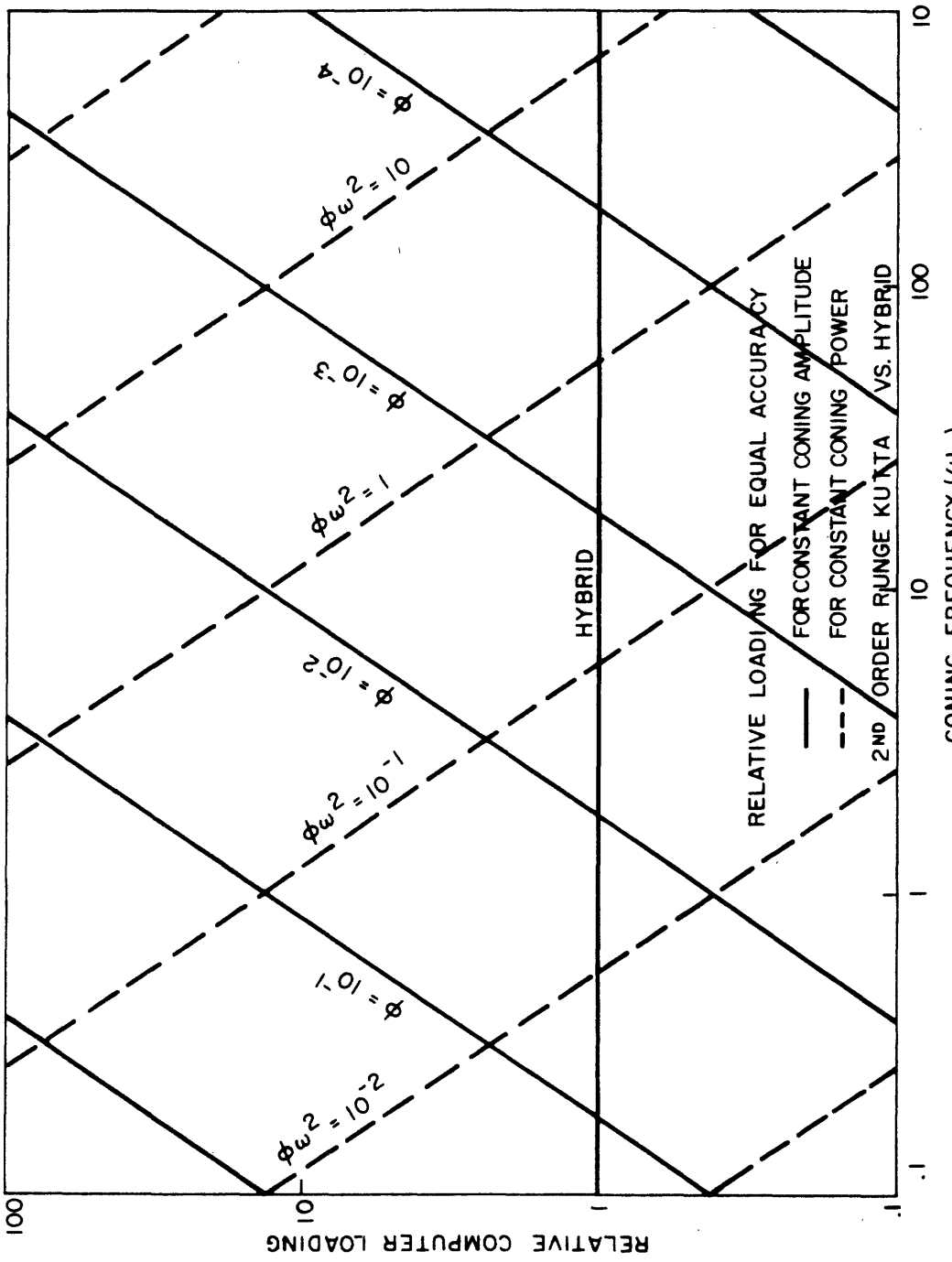


Figure 6.3.- Coning motion

coning motion increases as the square of the coning frequency. It is more reasonable to assume that for a given vehicle, the coning power will not change very much since the vehicle power plant has a fixed upper limit on the power that it can generate. The more efficient the engine and vehicle, the less the power that goes into generating coning motion. The dashed lines in Figure 6.3 show relative computer load for equal accuracy at constant coning power where $\phi\omega^2$ is taken to be a measure of coning power. This graph shows that for constant coning power, the hybrid algorithm is more efficient at low frequencies and the digital algorithm is more efficient at high frequencies. This is because at high frequencies, there is less coning drift than at low frequencies, given constant coning power.

Figure 6.4 is a plot of algorithm drift rate vs coning angular frequency at the same computer loading for the hybrid algorithm and the 2nd order Runge-Kutta algorithm. As in the case of the constant angular velocity, there is a region in which the hybrid algorithm exhibits the lower drift rate. This region is bounded on either side by regions in which the digital algorithm gives the better performance.

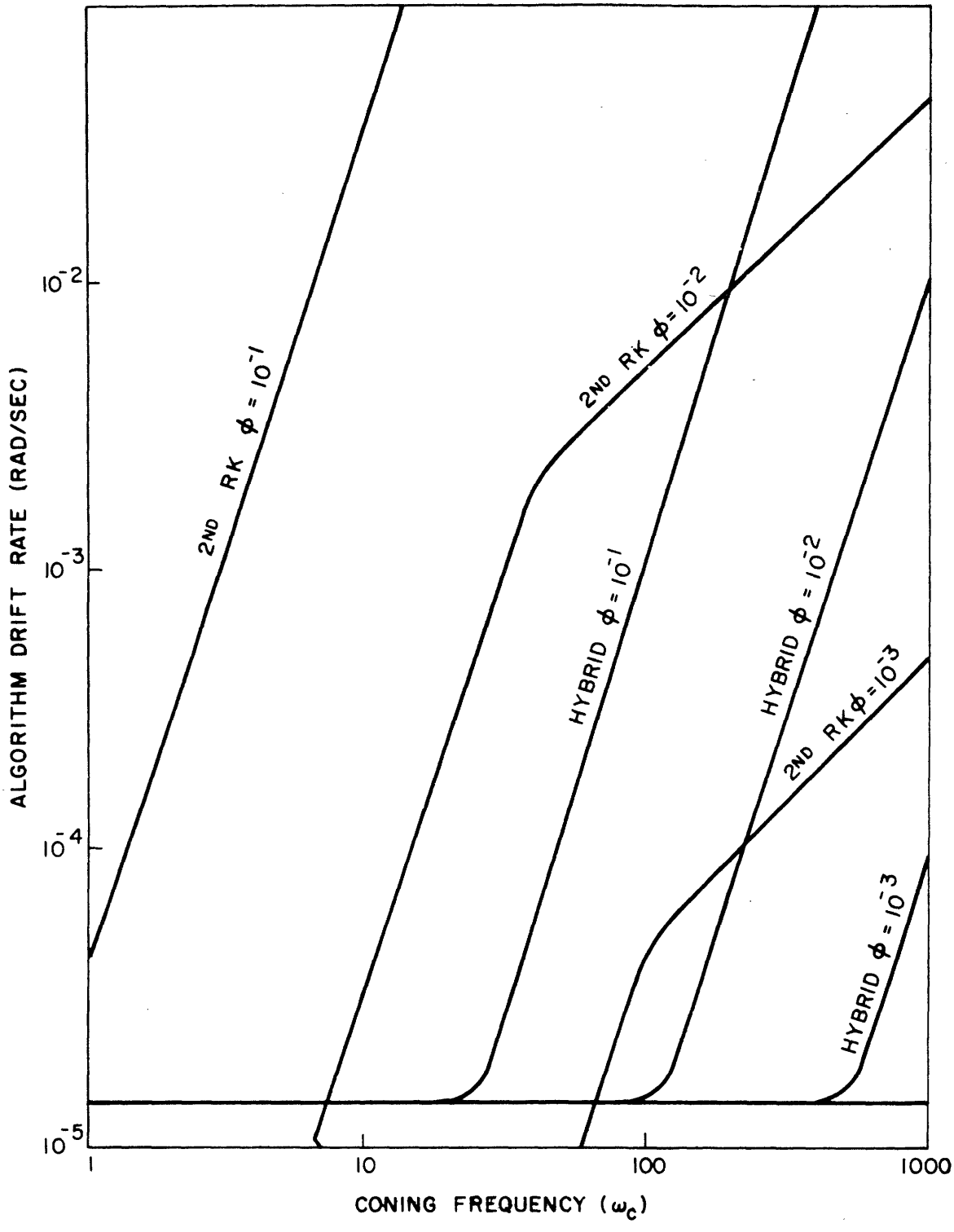


Figure 6.4.- Coning input algorithm drift at equal computer loading

CHAPTER 7

THE TREATMENT OF SPECIFIC FORCE

7.1 Vector Integration in a Rotating Space

In Chapter 3, a differential equation was derived for the rotation vector $\underline{\phi}_{RB}(t)$ relating the orientation of the Body Frame to the orientation of the Reference Frame at time t. One of the many forms that this equation may take is given by Eq. (3.38)

$$\frac{d\underline{\phi}}{dt_B} = \underline{\omega} + \frac{1}{2}\,\underline{\phi}\times\underline{\omega} + \frac{1}{\phi^2}\left(1 - \frac{\phi \sin \phi}{2(1-\cos \phi)}\right)\underline{\phi}\times(\underline{\phi}\times\underline{\omega}) \qquad (7.1)$$

where the symbol d/dt_B indicates that the derivative was taken with respect to the Body Frame.

A similar problem, that of finding an equation for $d^B\underline{v}_{rb}(t)/dt_B$ is considered in this chapter, where the vector $\underline{v}_{rb}(t)$ is the velocity of the origin b of the Body Frame at time t relative to the origin r of that Reference Frame with which the Body Frame was coincident at time t_o. It is necessary for a physically meaningful integration that the coordinate frame with respect to which the derivative (integrand) is taken, be the same as the coordinate frame in which the derivative (integrand) is coordinatized. For example, strapdown accelerometers sense (in a zero gravitation environment)

$$\underline{f}^B = - \frac{d^B \underline{v}_{rb}}{dt_R}$$

where $\underline{f}$, the specific force vector, is defined (ref. 1) as the reaction force per unit mass exerted by the accelerometer on its mounting structure. (The Reference Frame is always an inertial frame in this thesis even though it is redefined whenever $\underline{\phi}_{RB}$ is reset to zero.)

It is true that

$$\int_{t_o}^{t} \underline{f}^R(\tau)d\tau = \int_{t_o}^{t} - \frac{d^R \underline{v}_{rb}(\tau)}{d\tau_R} = -\underline{v}^R_{rb}(t)$$

but the integral

$$\underline{h}(t) = \int_{t_o}^{t} \underline{f}^B(\tau)d\tau = \int_{t_o}^{t} - \frac{d^B \underline{v}_{rb}(\tau)}{d\tau_R} \, d\tau$$

is a physically meaningless quantity if the B and R Frames have relative angular motion. Since strapdown accelerometers integrate $\underline{f}^B$, the conventional approach has been to approximate $\underline{v}^R_{rb}(t_o+n\Delta T)$ by

$$\underline{v}^R_{rb}(t_o+n\Delta T) \cong \sum_{i=0}^{n-1} C^{RB}(t_o+i\Delta T+\alpha) \int_{t_o+i\Delta T}^{t_o+(i+1)\Delta T} \underline{f}^B(t)\, dt \qquad (7.2)$$

$$0 \leq \alpha \leq \Delta T$$

$$n = 1, 2, 3, \ldots$$

where ΔT is the accelerometer sampling interval (A more complex, but more efficient, algorithm could be used.)

There are two problems with Eq. (7.2). The first is that the integrals in the summation are only approximations to $\underline{v}_{rb}^{B}$. The second is that C^{RB} must be computed and applied once during each integration cycle. Depending on the characteristics of the accelerometers and the transformation algorithm, the frequency of the transformation process may be greater than 10^3 cycles per second.

The Law of Coriolis provides a way to avoid the problems described in the preceding paragraph. Define $\underline{\nu}$ by the relationship

$$\frac{d^B \underline{\nu}}{dt_R} \equiv \underline{f}^B = \underline{G}^B - \frac{d^B \underline{v}_{rb}}{dt_R} \tag{7.3}$$

where $\underline{G}$ is the gravitation vector. Then by the Law of Coriolis [refs. (1) and (18)]

$$\frac{d^B \underline{\nu}}{dt_B} = \frac{d^B \underline{\nu}}{dt_R} - \underline{\omega}_{RB}^{B} \times \underline{\nu}^B \tag{7.4}$$

Since the coordinate frame with respect to which the derivative is taken is the same as that in which the components of the derivative on the left hand side of Eq. (7.4) are expressed, the left hand side may be integrated to get

$$\underline{\nu}^B = \int_{t_o}^{t} \frac{d^B \underline{\nu}}{d\tau_B} \, d\tau \tag{7.5}$$

It will be shown in the next paragraph that

$$\underline{\nu}^R = \int_{t_o}^{t} \underline{f}^R d\tau \tag{7.6}$$

The advantage of the formulation given by Eq. (7.4) is that Eq. (7.4) can be integrated entirely in Body Frame coordinates using a hybrid technique similar to that used to integrate Eq. (7.1). Then at a comparatively slow rate, the transformation

$$\underline{\nu}^R = C^{RB}\underline{\nu}^B \tag{7.7}$$

is performed and velocity is obtained by integrating gravitation $\underline{G}$ in a nonrotating coordinate frame and then using the relation

$$\underline{v}^R_{rb} = \int_{t_o}^{t} \underline{G}^R d\tau - \underline{\nu}^R \tag{7.8}$$

To verify Eq. (7.6), it is sufficient to show that the derivatives of both sides are always equal and that at some time, both sides have the same value. It is clear that

$$\frac{d}{dt_R}\int_{t_o}^{t} \underline{f}^R d\tau = \underline{f}^R \tag{7.9}$$

To eveluate the derivative on the right hand side of Eq. (7.6) use Eq. (7.7) and the Law of Coriolis

$$\frac{d}{dt_R}\underline{\nu}^R = \frac{d}{dt_B} C^{RB}\underline{\nu}^B - \left[\underline{\omega}^R_{RB} \times\right] C^{RB}\underline{\nu}^B$$

$$= \left(\frac{d}{dt_B} C^{RB}\right)\underline{\nu}^B + C^{RB}\frac{d\underline{\nu}_B}{dt_B} - \left[\underline{\omega}^R_{RB} \times\right] C^{RB}\underline{\nu}^B \tag{7.10}$$

Since

$$\frac{d}{dt_B} C^{RB} = C^{RB}\left[\underline{\omega}^B_{RB} \times\right]$$

Eq. (7.10) becomes

$$\frac{d^R}{dt_R} \underline{v}^R = C^{RB}\left[\underline{\omega}^B_{RB} \times\right]\underline{v}^B + C^{RB} \frac{d}{dt_B} \underline{v}^B$$

$$- \left[\underline{\omega}^R_{RB} \times\right] C^{RB}\underline{v}^B \qquad (7.11)$$

By premultiplying the last term on the right hand side of Eq. (7.11) by

$$I = C^{RB}C^{BR}$$

and recognizing the similarity transform

$$\underline{\omega}^B_{RB} = C^{BR}\left[\underline{\omega}^B_{RB} \times\right]C^{RB}$$

Eq. (7.11 becomes

$$\frac{d}{dt_R} \underline{v}^R = C^{RB}\left[\underline{\omega}^B_{RB} \times\right]\underline{v}^B + C^{RB} \frac{d}{dt_B} \underline{v}^B$$

$$- C^{RB} \left[\underline{\omega}_{RB} \times\right] \underline{v}^B$$

$$= C^{RB} \frac{d}{dt_B} \underline{v}^B \qquad (7.12)$$

but by Eq. (7.3),

$$\frac{d}{dt_R} \underline{v}^R = C^{RB}\underline{f}^B$$

and so

$$\frac{d}{dt_R} \underline{v}^R = \underline{f}^R \qquad (7.13)$$

By comparing Eqs. (7.9) and (7.13), it is seen that the derivatives of both sides of Eq. (7.6) are identical. If the initial

conditions

$$\underline{v}^B(t_o) \equiv \underline{0} \tag{7.14}$$

are arbitrarily assigned, then Eq. (7.6) is established since both sides are equal at $t = t_o$.

By the laws of vector addition

$$\underline{v}^N_{nb}(t) = \underline{v}^N_{nr}(t) + \underline{v}^N_{rb}(t) \tag{7.15}$$

where n is the origin of the Navigation Frame in which the navigation equations are solved. Since

$$\underline{v}^N_{rb} = C^{NR}\underline{v}^R_{rb}$$

Eq. (7.15) can be written as

$$\underline{v}^N_{nb}(t) = \underline{v}^N_{nr}(t) + C^{NR}\underline{v}^R_{rb}$$

or using Eqs. (7.7) and (7.8), this becomes

$$\underline{v}^N_{nb}(t) = \underline{v}^N_{nr}(t) + C^{NR}(t)\int_{t_o}^{t} \underline{G}^R(\tau)d\tau - C^{NB}(t)\ \underline{v}^B(t) \tag{7.16}$$

From Eqs. (7.14) and (7.16), it can be seen that

$$\underline{v}_{nr}(t_o) = \underline{v}_{nb}(t_o)$$

If the Navigation Frame is an inertial frame, then

$$\underline{v}_{nr}(t) = \underline{v}_{nr}(t_o)$$

and

$$C^{NR}(t)\int_{t_o}^{t} \underline{G}^R(\tau)d\tau = \int_{t_o}^{t} C^{NR}(t_o)\ \underline{G}^R(\tau)d\tau = \int_{t_o}^{t} \underline{G}^N(\tau)d\tau$$

These relationships are substituted into Eq. (7.16) to get

$$\underline{v}^N_{nb}(t) = \underline{v}^N_{nb}(t_o) + \int_{t_o}^{t} \underline{G}^N(\tau)d\tau - C^{NB}(t)\ \underline{\nu}^B(t) \qquad (7.17)$$

7.2 System Mechanization

The system functional diagram is shown in Figure 7.1. The computer must solve either Eq. (7.16) or (7.17) given $\Delta\underline{\nu}^B$ as generated by the accelerometers and the analog Coriolis correction circuitry. The equation solved by this circuitry is found by combining Eqs. (7.3) and (7.4) to get

$$\dot{\underline{\nu}}^B \equiv \frac{d^B\underline{v}}{dt_B} = \underline{f}^B - \underline{\omega}^B_{RB} \times \underline{\nu}^B \qquad (7.18)$$

The pulse output from the accelerometer triad is

$$\Delta\underline{\nu}^B(t_o+n\Delta T) = \int_{t_o+(n-1)\Delta T}^{t_o+n\Delta T} \frac{d^B\underline{\nu}}{dt_B}\ dt$$

$$n = 1, 2, 3,...$$

The $\Delta\underline{\nu}^B$'s are accumulated in the digital computer to obtain $\underline{\nu}^B(t_o+n\Delta T)$. As in the case of the hybrid coordinate transformation computation, a filter is used to extract a continuous triad of signals representing $\underline{f}^B$ from accelerometer observables. An analog Coriolis correction, $\underline{\omega}_{RB} \times \underline{\nu}$ is generated and fed back through the accelerometers so that they then integrate and quantize $\dot{\underline{\nu}}^B$ instead of $\underline{f}^B$ as they normally would. A reset $\underline{\nu}$ signal is generated by the digital computer whenever $|\underline{\nu}|$ exceeds a predetermined value. Since the Coriolis correction $\underline{\omega}_{RB} \times \underline{\nu}^B$ is generated using analog circuitry, it

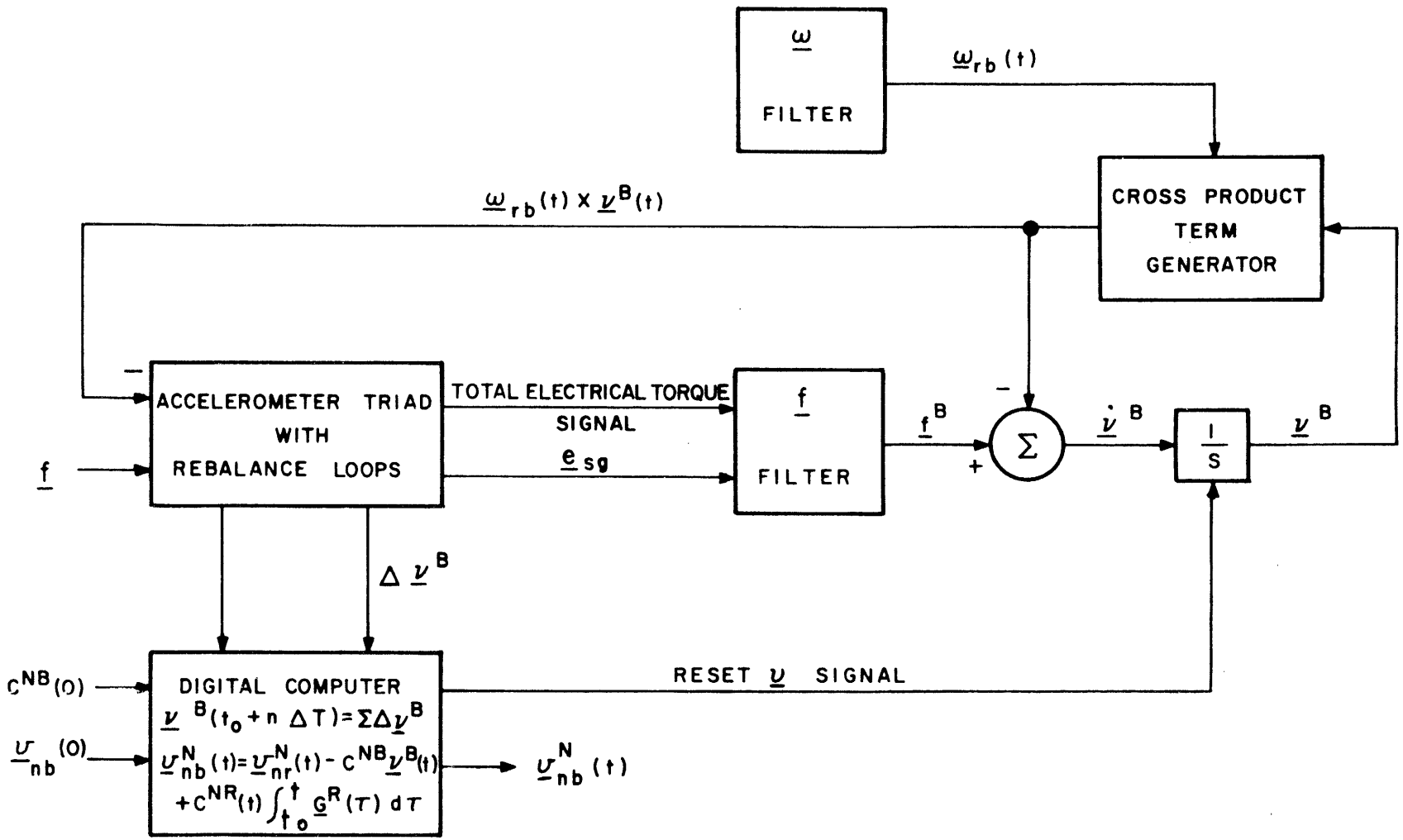


Figure 7.1.- Mechanization of solution for $\underline{v}_{nb}^{n}(t)$

must not be permitted to become significant compared with f^B_{MAX}, the maximum specific force magnitude that can be measured by the accelerometers. Thus, the criterion is imposed that when

$$|\underline{\omega}_{RB} x \underline{\nu}^B| \geq 0.1\ f^B_{MAX}$$

that $\underline{\nu}^B$ will be reset to zero. So $\underline{\nu}^B$ is reset when

$$|\underline{\nu}^B| \geq \frac{f^B_{MAX}}{|0.1\underline{\omega}_{RB}|_{MAX}} \tag{7.19}$$

Since the frequency at which C^{RB} is updated is predicated upon the uses which C^{RB} serves, an estimate of the frequency at which $\underline{\nu}^B$ is reset is required. This is because $\underline{\nu}^B$ is transformed by Eq. (7.7) just prior to being reset. The maximum rate of resetting $\underline{\nu}^B$ or the minimum time $t_{R,min}$ between resets occurs when

$$|\underline{f}^B| = f^B_{MAX}$$

Then at the time of reset,

$$|\underline{\nu}^B| = f^B_{max} t_{R,min}$$

This is used in Eq. (7.19) and the resulting equation is solved for $t_{R,min}$ to get

$$t_{R,min} = \frac{1}{|0.1\underline{\omega}_{RB}|_{MAX}}$$

or the maximum frequency $f_{R,MAX}$ of resetting $\underline{\nu}^B$ to zero is

$$f_{R,MAX} = |0.1\underline{\omega}_{RB}|_{MAX}$$

It was shown in Chapter 5 that there is significant analog computer error growth with time and with increasing magnitude of the integrated vector, in this case $\underline{v}^B$. For this reason, the frequency of resetting f_R would always be taken to be

$$f_R = f_{R,MAX}$$

$$= |0.1\underline{\omega}_{RB}|_{MAX} \qquad (7.20)$$

In the experimental system $|\underline{\omega}_{RB}|_{MAX} = 2$ rad/sec. If the specific force transformation were mechanized for this system, it would require a $\underline{v}^B$ resetting frequency of

$$f_R = 20 \text{ resets/sec}$$

7.3 Two Sample Problems

Example 1

For this example assume a vehicle in an environment with no gravitation and a specific force, angular velocity profile given by

$$\left.\begin{aligned} \underline{f}^B(t) &= -a\ \underline{1}_x^B \\ \underline{\omega}_{RB}^B(t) &= \underline{0} \end{aligned}\right\} \quad t_o < t \leq t_o + T$$

and

$$\left.\begin{aligned} \underline{f}^B(t) &= \underline{0} \\ \underline{\omega}_{RB}^B(t) &= \frac{2\pi}{T}\ \underline{1}_z^B \end{aligned}\right\} \quad t_o + T < t \leq t_o + 2T$$

From physical considerations, it is apparent that $\underline{v}^B_{rb}$ is as shown in Figure 7.2a and that $\underline{v}^R_{rb}$ is as shown in Figure 7.2b.

The navigation system would solve the problem by integrating Eq. (7.18) and solving Eq. (7.17). Assuming $\underline{v}^N_{nb}(t_o)=\underline{0}$, Eq. (7.17) becomes

$$\underline{v}^N_{nb}(t) = \underline{v}^R_{rb}(t) = -C^{RB}(t)\ \underline{\upsilon}^B(t) \tag{7.21}$$

On the time interval

$$t_o < t \leq t_o + T \tag{7.22}$$

then

$$\underline{\upsilon}^B(t) = -at\ \underline{1}^B_x \tag{7.23}$$

and

$$C^{RB}(t) = I \tag{7.24}$$

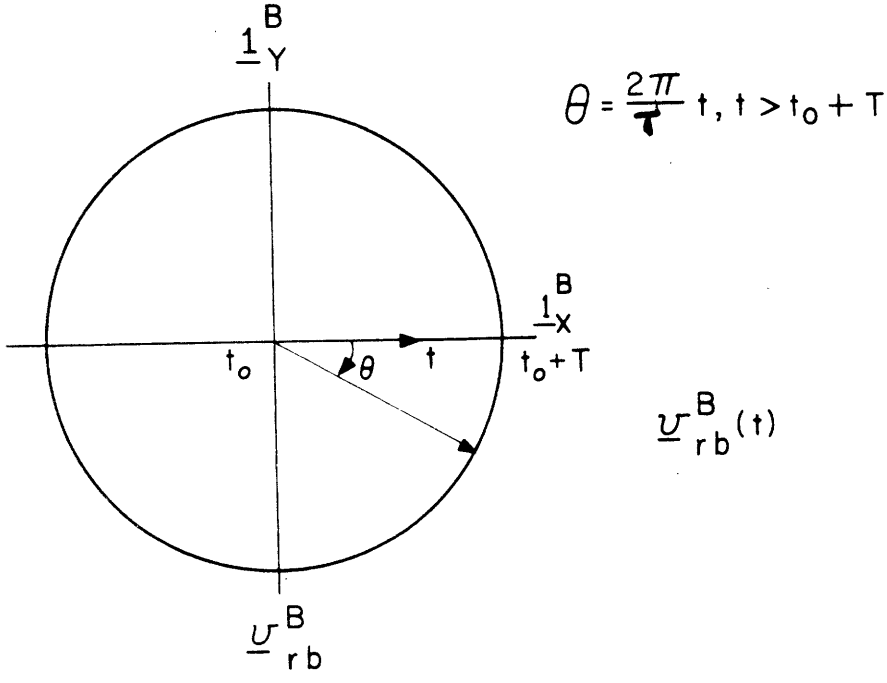


Figure 7.2(a).- $\underline{v}^B_{rb}$

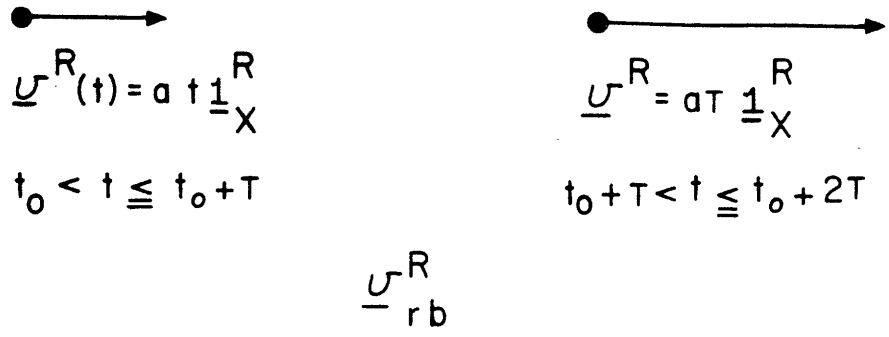


Figure 7.2(b).- $\underline{v}_{rb}^R$

since

$$\underline{\omega}_{RB} = \underline{0}$$

when the inequality (7.22) is satisfied. So during this time interval, the combination of Eqs. (7.21), (7.23), and (7.24) give

$$\underline{v}^R = at\,\underline{1}_x^R$$

On the time interval

$$t_o + T < t \leq t_o + 2T \tag{7.25}$$

Eq. (7.18) becomes

$$\dot{v}_x^B = -\frac{2\pi}{T} v_y^B \qquad v_x^B(t_o+T) = -aT$$

$$\dot{v}_y^B = \frac{2\pi}{T} v_x^B \qquad v_y^B(t_o+T) = 0$$

$$\dot{v}_z^B = 0 \qquad v_z^B(t_o+T) = 0$$

The solution to this equation set is

$$v_x^B(t) = -aT \cos \frac{2\pi}{T} (t-t_o)$$

$$v_y^B(t) = aT \sin \frac{2\pi}{T} (t-t_o)$$

$$v_z^B(t) = 0$$

and the direction cosine matrix is

$$C^{RB} = \begin{bmatrix} \cos \frac{2\pi}{T} (t-t_o) & -\sin \frac{2\pi}{T} (t-t_o) & 0 \\ \sin \frac{2\pi}{T} (t-t_o) & \cos \frac{2\pi}{T} (t-t_o) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Therefore

$$\underline{v}^R(t) = -C^{RB}(t)\ \underline{v}^B(t)$$

$$= aT\ \underline{1}_x^R$$

when the inequality (7.25) is satisfied. This is in agreement with the physically deduced result shown in Figure 7.2b.

Example 2

Although it is not a practical situation, imagine a strap-down inertial sensing unit whose center of mass is stationary on the (non-rotating) Earth. Assume that at time t = 0, the z accelerometer's sensitive axis is down and that the inertial sensing unit is rotating at ω rad/sec about its y-axis. Under these conditions Eq. (7.3) shows that

$$\underline{f}^B = \begin{bmatrix} -G \sin \omega t \\ 0 \\ G \cos \omega t \end{bmatrix} \qquad (7.26)$$

where G is the magnitude of the gravitational force. Also

$$\underline{\omega}^{R}_{RB} = \begin{bmatrix} 0 \\ \omega \\ 0 \end{bmatrix} \tag{7.27}$$

Assuming the Reference Frame z-axis is down, then

$$\int_0^t \underline{G}^R \, d\tau = Gt\underline{1}^R_z \tag{7.28}$$

The inertial sensing unit is stationary, so

$$\underline{v}_{rb}(t) \equiv \underline{0}$$

With the use of Eq. (7.28) Eq. (7.17) becomes

$$Gt\underline{1}^R_z = C^{RB}(t)\ \underline{\nu}^B(t) \tag{7.29}$$

Since

$$C^{RB}(t) = \begin{bmatrix} \cos\ \omega t & 0 & \sin\ \omega t \\ 0 & 1 & 0 \\ -\sin\ \omega t & 0 & \cos\ \omega t \end{bmatrix}$$

it is possible to solve Eq. (7.29) for $\underline{\nu}^B(t)$.

$$\underline{\nu}^B(t) = C^{BR}(t)\ Gt\underline{1}^R_z$$

$$\underline{\nu}^B(t) = \begin{bmatrix} -Gt\ \sin\ \omega t \\ 0 \\ Gt\ \cos\ \omega t \end{bmatrix} \tag{7.30}$$

The analog mechanization of Figure 7.1 must yield Eq. (7.30) as a solution of Eq. (7.18). If it can be shown that the derivative of Eq. (7.30) is always equal to the form Eq.(7.18) takes for this problem then the solution to Eq. (7.18) is indeed Eq. (7.30) if it can be shown also, that for one

instant of time, the solution to Eq. (7.18) is equal to Eq. (7.30). Note from Eq. (7.30) that

$$\underline{\nu}^B(0) = \underline{0}$$

and from Eq. (7.18)

$$\int_0^t \dot{\underline{\nu}}^B(\tau)d\tau = \underline{0}$$

Thus $\int \dot{\underline{\nu}}^B dt$ and $\underline{\nu}^B$ are equal for t = 0. Using Eqs. (7.26) and (7.27) in Eq. (7.18) gives

$$\dot{\underline{\nu}}^B = \underline{f}^B - \underline{\omega}^B_{RB} \times \underline{\nu}^B$$

$$= \begin{bmatrix} -G \sin \omega t \\ 0 \\ G \cos \omega t \end{bmatrix} - \begin{bmatrix} 0 \\ \omega \\ 0 \end{bmatrix} \times \begin{bmatrix} \nu^B_x \\ \nu^B_y \\ \nu^B_z \end{bmatrix}$$

Or using Eq. (7.30), this becomes

$$\dot{\underline{\nu}}^B = \begin{bmatrix} -G \sin \omega t \\ 0 \\ G \cos \omega t \end{bmatrix} - \begin{bmatrix} 0 \\ \omega \\ 0 \end{bmatrix} \times \begin{bmatrix} -Gt \sin \omega t \\ 0 \\ Gt \cos \omega t \end{bmatrix}$$

$$\dot{\underline{\nu}}^B = \begin{bmatrix} -G \sin \omega t - \omega Gt \cos \omega t \\ 0 \\ G \cos \omega t - \omega Gt \sin \omega t \end{bmatrix} \qquad (7.31)$$

This is Eq. (7.18) evaluated using the given $\underline{f}^B$ and $\underline{\omega}^B_{RB}$ and a guess, Eq. (7.30), for $\underline{\nu}^B$. Taking the derivative of Eq. (7.30) gives

$$\dot{\underline{\nu}}^B = \begin{bmatrix} -G \sin \omega t - \omega Gt \cos \omega t \\ 0 \\ G \cos \omega t - \omega Gt \sin \omega t \end{bmatrix}$$

Since this is identical to Eq. (7.31), it is concluded that the integration of Eq. (7.18) by the system mechanization shown in Figure 7.1 must give the same result, viz., Eq.(7.30), as was deduced from physical considerations.

CHAPTER 8

CONCLUSIONS AND RECOMMENDATIONS

8.1 General Discussion

In this thesis, a new concept for accomplishing the strapdown coordinate transformation computation was developed and tested. A vector differential equation for $\underline{\phi}_{RB}$, the argument of the coordinate transformation C^{RB}, is integrated by the gyros. Then the coordinate transformation $C^{RB}(\underline{\phi}_{RB})$ is evaluated as a matrix function of the argument $\underline{\phi}_{RB}$.

Analog computing elements generate a correction $\dot{\underline{\sigma}}_{RB}$ for the noncommutativity effect. The time rate of change of $\underline{\phi}_{RB}$ is

$$\dot{\underline{\phi}}_{RB} = \underline{\omega}_{RB} + \dot{\underline{\sigma}}_{RB} \tag{5.9}$$

Thus it is convenient to apply the analog signal $\dot{\underline{\sigma}}_{RB}$ to the gyro torque summing member and let the gyros themselves integrate and quantize $\dot{\underline{\phi}}_{RB}$. $\underline{\phi}_{RB}$ is maintained in the digital computer by counting the incremental outputs from the gyros. The only digital computation is the evaluation C^{RB} from $\underline{\phi}_{RB}$.

In the conventional strapdown techniques, the matrix differential equation

$$\dot{C}^{RB} = C^{RB}[\underline{\omega}_{RB}x] \tag{8.1}$$

is integrated numerically in the digital computer. Noncommutativity precludes a closed form solution to Eq. (8.1) except in the certain special cases (ref. 23). Consequently, the algorithm for integrating Eq. (8.1) must either approximate $\underline{\omega}_{RB}$ from the accumulated gyro pulses by a vector $\hat{\underline{\omega}}_{RB}$ that admits a closed form solution or it must employ some other approximation technique for solving Eq. (8.1).

The performance of three conventional algorithms was compared with that of the hybrid algorithm on the basis of accuracy, complexity and bandwidth (Chapter 6). The hybrid method was shown to offer a significant saving in digital computer loading. The point of diminishing returns for improving the accuracy of the coordinate transformation computation is generally taken to occur when the errors contributed by the computation process are smaller than the errors contributed by the gyros themselves. Computation errors and instrument errors are both frequency dependent. Conventional computation and hybrid computation alike serve well at low frequencies and each type can be made to do so at high frequencies. The state-of-the-art in analog computing elements is such that hybrid computational bandwidth can easily be made to exceed gyro bandwidth by an order of magnitude. In Chapter 6, it was seen that the bandwidth of efficient conventional algorithms for a given computational load was of the order of 10 rad/sec.* The bandwidth of

*Rectangular integration rules mechanized by DDA computation have much higher bandwidths, but the computational accuracy is comparatively poor as seen in Chapter 6. Also, a special purpose computer is required in addition to the navigation computer.

currently available strapdown gyros and therefore of the hybrid computation process for the same load is of the order of 1000 rad/sec. The bandwidth of the conventional algorithm is directly proportional to the computational load imposed by the algorithm. To double the computational bandwidth, the computational load must be doubled.

8.2 Gyro Quantization Level

A major consideration in the design of a pulse rebalanced strapdown gyro is output pulse quantization level. From a gyro designer's point of view, long sampling intervals are more attractive than short sampling intervals since the gyro operating frequencies can be made lower. This allows one cause of scale factor error (due to the uncertainty in the switching times of the torque pulses) to be more easily controlled. From the point of view of gyro dynamic errors, the gyro error model contains terms which are a function of float angle. These error terms increase as quantization level and hence the rms float angle increases. Conventional algorithms require fine quantization in order to achieve accuracy in both the coordinate transformation computation and the transformation of specific force. It can be seen from Section 5.3.3 that when using hybrid computation, quantization effects do not produce errors in the coordinate transformation with significant growth rate. Still the resolution of the coordinate transformation is limited by the quantization level. That is, the hybrid coordinate transformation is not degraded significantly in accuracy by coarse quantization, but its precision is a direct function of quantization level.

8.3 Computer Round-Off Error

The round-off error in the direction cosine matrix computation increases as the frequency of the update in conventional computation. In hybrid computation, it is not the frequency of the update that determines the growth of the round-off error, but the frequency with which $\underline{\phi}_{RB}$ is reset to zero. This is because the round-off error becomes permanent only when the initial condition matrix is multiplied by the computed matrix to form a new initial condition matrix (Chapter 6). This occurs at each update in conventional computation, but only when $\underline{\phi}_{RB}$ is reset in the hybrid technique.

The tendency of the computed coordinate transformation to become non-orthogonal is a problem that, like the round-off problem, grows more severe as the frequency of generating a new initial condition matrix increases. As in the case of round-off error, this tendency is much less pronounced in the hybrid computation.

8.4 Inertial Sensor Design Considerations

In the hybrid computation scheme, a continuous voltage representing the primary input must be generated from signals which can be measured at the sensor. That is, the sensor must be inherently, an analog device. (The Geiger counter, for example, is an inherently digital measurement device.) Many inertial sensors have a modulated output. The signal generator signal from the DDG 334A gyro used in the experiment is an amplitude modulated signal. The output signal from a vibrating string accelerometer is a frequency

modulated signal. Modulation itself does not effect the analog measurement as long as the modulation frequency is well above the sensor's first roll-off frequency, as is usually the case.

It is not necessary to be able to sum an externally generated correction signal with the input in the sensor itself, but if the sensor is pulse rebalanced, it is very convenient to do so. When the physical input signal and the externally generated correction signal are integrated and quantized separately, then there is a separate quantization error for each signal and this is to be avoided when convenient.

8.5 Recommendations

8.5.1 Filter Design

One of the most important links in the analog computation chain is the ω-Filter. In Section 5.4, it was seen that the magnitude of the noncommutativity correction $\underline{\dot{\sigma}}$ relative to the magnitude of $\underline{\omega}$ is

$$\dot{\sigma} \cong \frac{\phi}{2} \omega$$

For $\phi_{max} = 0.1$ rad and $\phi_{avg} = 0.05$ rad,

$$\dot{\sigma}_{avg} = 0.025 \ \omega$$

Care must therefore be taken to insure that the analog measurement of $\underline{\omega}$ is at least 1/40 as accurate as the basic measurement made by the gyro. The quality of the ω-Filter used in the experiment was marginal. The linearity, the quality of the

demodulation, and the quality of the torque signal cancellation could all be improved. Additional effort in ω-Filter design would be rewarding.

The frequency at which the response of the gyro filter ensemble begins to roll off is $1/\tau_g$ where τ_g is the gyro float time constant. This frequency determined the theoretical bandwidth of the hybrid computation. Perhaps lead-lag compensation could be introduced at the filter to extend the gyro-filter ensemble bandwidth, thus extending the dynamic range of the hybrid computation.

8.5.2 Cross Product Term Generation

The cross product term generation involves the subtraction of two relatively large numbers to obtain a relatively small one. In Section 5.4, it was seen that, as an approximation

$$\underline{\dot{\sigma}} = \frac{1}{2}\ \underline{\phi} \times \underline{\omega}$$

Suppose $\underline{\phi} \times \underline{\omega} = \underline{0}$ but that $\underline{\phi}$ and $\underline{\omega}$ both have rather large magnitudes. Now

$$\dot{\sigma}_i = \frac{1}{2}\ \phi_{i+1}\omega_{i-1} - \frac{1}{2}\ \phi_{i-1}\omega_{i+1} \qquad (8.2)$$

Each term on the right hand side is rather large for at least one component of σ, but since $\dot{\sigma}_i = 0$, the two terms on the right hand side of Eq. (8.2) must be equal. Perhaps a better mechanization of the cross product term could be found that would not have this undesirable feature in generating $\dot{\sigma}_i = 0$.

8.5.3 The Treatment of Specific Force

Chapter 7 describes the hybrid computation scheme for integrating and transforming the strapdown specific force measurement. Only a theoretical development is presented. An error analysis and an experimental verification of the method are recommended as the next steps in the process of demonstrating its operational feasibility.

8.5.4 Analog Computation Scaling

It was seen in Chapter 6, that there is a definite part of the input angular motion dynamic range in which the hybrid computation is superior to all digital computation, and there is a region in which it is not.

For any assumed mission and vehicle, the analog computer scaling could be chosen to most effectively place that portion of the dynamic range where the hybrid computation is the best. Perhaps, a systematic procedure for scaling the analog computation could be devised.

8.5.5 Analog Inertial Sensor Compensation

An error analysis of an analog compensation scheme for gyro and accelerometer dynamic errors and a performance comparison between analog and digital dynamic errors compensation schemes might reveal that analog compensation is a significantly superior method of gyro and accelerometer compensation. It is recommended that such a study be performed.

Appendix A

LANING'S THEORY

In the late 1940's the M.I.T. Instrumentation Laboratory was engaged in pioneering work in the area of fire-control. The paper, "The Vector Analysis of Finite Rotations and Angles" by J. Halcombe Laning, Jr. (ref. 12) was a consequence of his participation in that work. Because of its unique approach, it offers fresh insight into the dominant mathematical problem of strapdown inertial navigation. In this paper, Laning noted that, "The geometric problems of principal interest in the fire-control field are characterized more by complexity than by a high intrinsic level of mathematical difficulty ... The chief geometric difficulties are those which involve relating angles and space rotations, together with their time rates of change, to such kinematic quantities as angular velocities. Since angles and rotations possess direction and magnitude, and are not dependent for their definition upon a particular system of coordinates, a vector representation of these quantities seems natural. The principal obstacle in the path of such a representation is the fact that the natural laws of combination are not those of ordinary vector addition."

The main features of Laning's report will be presented here although this appendix will by no means be a complete exposition of that report. That which is presented will suit the present purpose. In order to develop an algebra for rotation vectors, it is useful to develop first an algebra in which the angles defined by two intersecting lines are considered as vectors. The mathematical relations which can be developed, then serve as the foundation upon which the algebra of rotation vectors can be built and understood.

Let $\underline{A}_{BC}$ denote the vector representing the angle between the directed line segments $\underline{B}$ and $\underline{C}$

$$\underline{A}_{BC} = \underline{l}_B \times \underline{l}_C \, q(A_{BC}) \tag{A.1}$$

where

$$\underline{B} = B\underline{l}_B$$

$$\underline{C} = C\underline{l}_C$$

$$q(A_{BC}) = \frac{A_{BC}}{\sin A_{BC}} \tag{A.2}$$

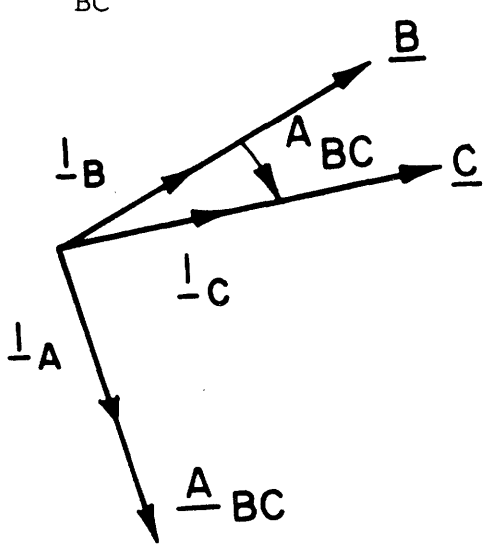


Figure A.1.- Geometry at angle vector

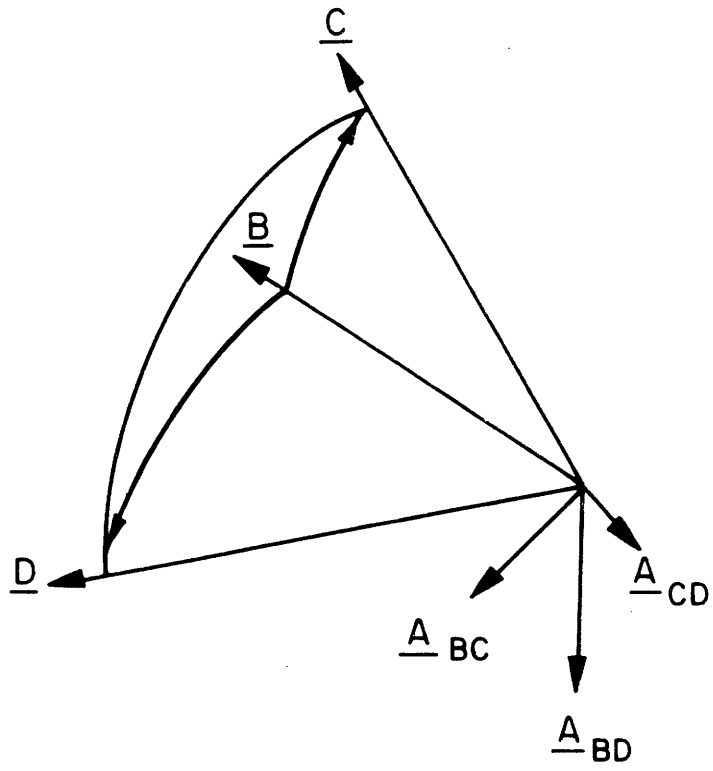


Figure A.2.- The angle sum - coincident sides

The vector $\underline{A}_{BC}$ (Figure A.1) has a magnitude equal to the radian measure of the angle from $\underline{B}$ to $\underline{C}$ and a direction perpendicular to the plane of $\underline{B}$ and $\underline{C}$ in a sense chosen by the right-hand rule. Without the factor $q(A_{BC})$, the magnitude of $\underline{A}_{BC}$ would be equal to sin A_{BC} and not to A_{BC}.

The angle vector is defined in terms of the vectors which form its sides. Thus when two angle vectors are added, the addition operation can be developed in terms of component sides of the two angle vectors. Before defining angle vector addition, note that there are many derived quantities defined as combinations of vectors and scalars, e.g., the product $m\underline{a}$ of a scalar m and a vector $\underline{a}$; the vector product $\underline{a} \times \underline{b}$, the difference $\underline{a} - \underline{b}$, etc. involving the vectors $\underline{a}$ and $\underline{b}$. It is important to note "that these definitions are made purely as a matter of convenience, because these combinations occur so

often in applications, and are in no way intrinsic to the concept of a vector."

A.1 The Angle Sum of Two Vectors

A definition will now be given for the "angle sum" operation. First consider the case where the terminal side of one angle vector is the initial side of the other as shown in Figure A.2. Let $\underline{A}_{BC}$ and $\underline{A}_{CD}$ be the angle vectors defined by the intersecting vectors $\underline{B}$ and $\underline{C}$ and the intersecting vectors $\underline{C}$ and $\underline{D}$ respectively. The angle vector $\underline{A}_{BD}$, whose initial side is $\underline{B}$ and whose terminal side is $\underline{D}$, is defined to be

$$\underline{A}_{BD} = \underline{A}_{BC} (+) \underline{A}_{CD} \tag{A.3}$$

where the symbol (+) denotes the angle sum operation. Employing Eq. (A.1) gives

$$\underline{A}_{BD} = \underline{1}_B \times \underline{1}_D q(A_{BD}) \tag{A.4}$$

To express this in terms of $\underline{A}_{BC}$ and $\underline{A}_{CD}$, the vector identity

$$\underline{1}_B = \underline{1}_C (\underline{1}_B \cdot \underline{1}_C) + \underline{1}_C \times (\underline{1}_B \times \underline{1}_C) \tag{A.5}$$

is needed. Since

$$\underline{1}_B \cdot \underline{1}_C = \cos A_{BC}$$

and

$$\underline{1}_B \times \underline{1}_C = \frac{\underline{A}_{BC}}{q(A_{BC})}$$

Eq. (A.5) becomes

$$\underline{l}_B = \underline{l}_C \cos A_{BC} + \underline{l}_C \times \frac{\underline{A}_{BC}}{q(A_{BC})} \tag{A.6}$$

Similarly, it is found that

$$\underline{l}_D = \underline{l}_C \cos A_{CD} - \underline{l}_C \times \frac{\underline{A}_{CD}}{q(A_{CD})} \tag{A.7}$$

The combination of Eqs. (A.6) and (A.7) with Eq. (A.5) yields

$$\frac{\underline{A}_{BD}}{q(A_{BD})} = \underline{l}_B \times \underline{l}_D = -\frac{\cos A_{BC}}{q(A_{CD})} \underline{l}_C \times (\underline{l}_C \times \underline{A}_{CD}) + \frac{\cos A_{CD}}{q(A_{BC})} (\underline{l}_C \times \underline{A}_{BC}) \times \underline{l}_C - \frac{(\underline{l}_C \times \underline{A}_{BC}) \times (\underline{l}_C \times \underline{A}_{CD})}{q(A_{BC})\, q(A_{CD})} \tag{A.8}$$

The vector triple products can be reduced by means of Eq. (A.5) to get

$$-\underline{l}_C \times (\underline{l}_C \times \underline{A}_{CD}) = -\underline{l}_C (\underline{l}_C \cdot \underline{A}_{CD}) + \underline{A}_{CD} = \underline{A}_{CD} \tag{A.9}$$

$$(\underline{l}_C \times \underline{A}_{BC}) \times \underline{l}_C = \underline{A}_{BC} - \underline{l}_C (\underline{l}_C \cdot \underline{A}_{BC}) = \underline{A}_{BC} \tag{A.10}$$

Note from Figure A.2 that $\underline{l}_C$ is perpendicular to both $\underline{A}_{BC}$ and $\underline{A}_{CD}$. The vector quadruple product can be treated as follows

$$\begin{aligned} &-(\underline{l}_C \times \underline{A}_{BC}) \times (\underline{l}_C \times \underline{A}_{CD}) \\ &= -\underline{l}_C (\underline{l}_C \times \underline{A}_{BC} \cdot \underline{A}_{CD}) + \underline{A}_{CD} (\underline{l}_C \times \underline{A}_{BC} \cdot \underline{l}_C) \\ &= -\underline{l}_C (\underline{l}_C \cdot \underline{A}_{BC} \times \underline{A}_{CD}) \end{aligned} \tag{A.11}$$

The last step follows from the identities

$$\underline{a} \times \underline{b} \cdot \underline{c} \equiv \underline{a} \cdot \underline{b} \times \underline{c}$$

$$\underline{a} \times \underline{b} \cdot \underline{a} \equiv 0$$

Now it is true that

$$-\underline{1}_C(\underline{1}_C \cdot \underline{A}_{BC} \times \underline{A}_{CD}) = -\underline{A}_{BC} \times \underline{A}_{CD}$$

since the vector $\underline{A}_{BC} \times \underline{A}_{CD}$ has the same direction as $\underline{1}_C$ in as much as $\underline{A}_{BC}$ and $\underline{A}_{CD}$ both lie in a plane perpendicular to $\underline{1}_C$. Hence

$$-(\underline{1}_C x \underline{A}_{BC}) \times (\underline{1}_C x \underline{A}_{CD}) = -\underline{A}_{BC} \times \underline{A}_{CD} \qquad (A.12)$$

Now substitute Eqs. (A.9), (A.10) and (A.12) into Eq. (A.8) to get

$$\frac{\underline{A}_{BD}}{q(A_{BD})} = \frac{\underline{A}_{BC}}{q(A_{BC})} \cos A_{CD} + \frac{\underline{A}_{CD}}{q(A_{CD})} \cos A_{BC} - \frac{\underline{A}_{BC}}{q(A_{BC})} \times \frac{\underline{A}_{CD}}{q(A_{CD})} \qquad (A.13)$$

But according to Eq. (A.3)

$$\underline{A}_{BD} = \underline{A}_{BC}(+)\,\underline{A}_{CD}$$

so with this substitution, Eq. (A.13) becomes

$$\frac{\underline{A}_{BC}(+)\underline{A}_{CD}}{q|\underline{A}_{BC}(+)\underline{A}_{CD}|} = \frac{\underline{A}_{BC}}{q(A_{BC})} \cos A_{CD} + \frac{\underline{A}_{CD}}{q(A_{CD})} \cos A_{BC} - \frac{\underline{A}_{BC}}{q(A_{BC})} \times \frac{\underline{A}_{CD}}{q(A_{CD})} \qquad (A.14)$$

Eq. (A.14) is taken to be the basic algebraic definition of the angle sum (+) operation.

Now consider the case where the terminal side of the first angle vector is not coincident with the initial side of the second angle vector. This is shown in Figure A.3. Note that any simultaneous rotation of both vectors $\underline{P}$ and $\underline{Q}$ through the same angle in the plane of $\underline{P}$ and $\underline{Q}$ leaves the vector $\underline{A}_{PQ}$ unchanged. So if $\underline{B}$ and $\underline{C}$ are

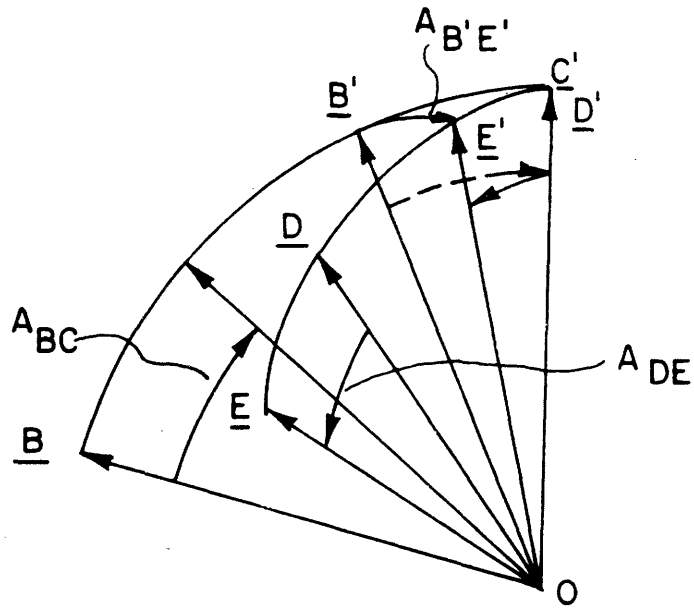


Figure A.3.- The angle sum - non-coincident sides

rotated together until $\underline{C}'$ coincides with the line of intersection of the $\underline{B}$-$\underline{C}$ plane with the $\underline{D}$-$\underline{E}$ plane and if $\underline{D}$ and $\underline{E}$ are rotated together until $\underline{D}'$ coincides with this same line, the vectors $\underline{A}_{BC}$ and $\underline{A}_{DE}$ are unchanged, but the terminal side $\underline{C}'$ of $\underline{A}_{BC}$ coincides with the initial side $\underline{D}'$ of $\underline{A}_{DE}$ and Eq.(A.14) applies.

Clearly Eq. (A.14) which defines the angle sum operation, does not rely on the initial side-terminal side visualization of a vector, but it may be used to define the angle sum operation for any two dimensionless vectors. Therefore

$$\frac{\underline{A}(+)\underline{B}}{q|\underline{A}(+)\underline{B}|} = \frac{\underline{A}}{q(A)} \cos B + \frac{\underline{B}}{q(B)} \cos A - \frac{\underline{A}}{q(A)} \times \frac{\underline{B}}{q(B)} \tag{A.15}$$

for any two dimensionless vectors $\underline{A}$ and $\underline{B}$. Let

$$\tilde{\underline{P}} = \frac{\underline{P}}{q(P)} \tag{A.16}$$

then Eq. (A.15) may be written as

$$\widetilde{A(+)B} = \tilde{A} \cos B + \tilde{B} \cos A - \tilde{A} \times \tilde{B} \tag{A.17}$$

Eq. (A.15) taken by itself is not entirely free of ambiguity since in this equation and in Eqs. (A.14) and (A.17), the magnitude of the vector $\underline{A}(+)\underline{B}$ has been replaced by the sine of the magnitude by division by $q|\underline{A}(+)\underline{B}|$. In cases of doubt, the formula

$$\cos |\underline{A}(+)\underline{B}| = \cos A \cos B - \tilde{A} \cdot \tilde{B} \tag{A.18}$$

may be used to resolve the ambiguity. Eq. (A.18) is derived from Figure A.2 from which it can be seen that

$$\cos |\underline{A}_{BC}(+)\underline{A}_{CD}| = \underline{1}_B \cdot \underline{1}_D$$

When Eqs. (A.6) and (A.7) are used, this becomes

$$\cos |\underline{A}_{BC}(+)\underline{A}_{CD}| = \cos A_{BC} \cos A_{CD} - (\underline{1}_C \times \tilde{A}_{BC}) \cdot (\underline{1}_C \times \tilde{A}_{CD})$$

$$= \cos A_{BC} \cos A_{CD} - \tilde{A}_{BC} \cdot \tilde{A}_{CD}$$

Eq. (A.18) is this result in terms of the two dimensionless vectors $\underline{A}$ and $\underline{B}$.

A.2 Algebraic Properties of the Angle Sum Operation

Commutativity

The presence of the cross product term, which is non-commutative with respect to its constituent vectors, makes it obvious that the angle sum operation is non-commutative, i.e.,

$$\underline{A}(+)\underline{B} \neq \underline{B}(+)\underline{A}$$

Associativity

To show that

$$[\underline{A}(+)\underline{B}] \ (+) \ \underline{C} = \underline{A}(+)[\underline{B}(+)\underline{C}] \tag{A.19}$$

Eq. (A.17) is used on the left hand side of Eq. (A.19) to get

$$\widetilde{[A(+)B] \ (+) \ C} = \widetilde{A(+)B} \cos C + \tilde{C} \cos |\underline{A}(+)\underline{B}| - \widetilde{A(+)B} \times \tilde{C}$$

Upon using Eqs. (A.17) and Eq. (A.18), this becomes

$$\begin{aligned}\widetilde{[A(+)B] \ (+) \ C} &= \tilde{A} \cos B \cos C + \tilde{B} \cos A \cos C + \tilde{C} \cos A \cos B \\ &\quad - \tilde{A} \times \tilde{C} \cos B - \tilde{B} \times \tilde{C} \cos A - \tilde{A} \times \tilde{B} \cos C \\ &\quad + \{(\tilde{A} \times \tilde{B}) \times \tilde{C} - \tilde{C} \ (\tilde{A} \cdot \tilde{B})\}\end{aligned} \tag{A.20}$$

Applying Eq. (A.17) to the right hand side of Eq. (A.19) gives

$$\begin{aligned}\widetilde{A(+)[B(+)C]} &= \tilde{A} \cos |\underline{B}(+)\underline{C}| + \widetilde{B(+)C} \cos \tilde{A} - A \times \widetilde{B(+)C} \\ &= \tilde{A} \cos B \cos C + \tilde{B} \cos A \cos C + \tilde{C} \cos A \cos B \\ &\quad -\tilde{A} \times \tilde{B} \cos C - \tilde{A} \times \tilde{C} \cos B - \tilde{B} \times \tilde{C} \cos A \\ &\quad + \{\tilde{A} \times (\tilde{B} \times \tilde{C}) - \tilde{A}(\tilde{B} \cdot \tilde{C})\}\end{aligned} \tag{A.21}$$

Eq. (A.21) is identical, term by term, with Eq. (A.20) except for the final bracketed quantity whose equality can be established by noting that

$$(\tilde{A} \times \tilde{B}) \times \tilde{C} = \tilde{B}(\tilde{A} \cdot \tilde{C}) - \tilde{A}(\tilde{B} \cdot \tilde{C})$$

and

$$\tilde{A} \times (\tilde{B} \times \tilde{C}) = \tilde{B}(\tilde{A} \cdot \tilde{C}) - \tilde{C}(\tilde{A} \cdot \tilde{B})$$

Thus it is proved that

$$\widetilde{[A(+)B] \ (+)C} = \widetilde{A(+)[B(+)C]}$$

However, this equality would hold even if $|[\underline{A}(+)\underline{B}]\ (+)\ \underline{C}|$, say, were in the first quadrant and $|\underline{A}(+)[\underline{B}(+)\underline{C}]|$ were in the second. To resolve this ambiguity, again resort to Eq.(A.18) to prove that

$$\cos|[\underline{A}(+)\underline{B}]\ (+)\ \underline{C}| = \cos\ |\underline{A}(+)[\underline{B}(+)\underline{C}]|$$

Making this substitution gives

$$\cos\ |\underline{A}(+)\underline{B}|\ \cos C - \widetilde{A(+)B} \cdot \tilde{C} = \cos A \cos\ |\underline{B}(+)\underline{C}| - \tilde{A} \cdot \widetilde{B(+)C}$$

and again gives

$$\cos A \cos B \cos C - \tilde{A} \cdot \tilde{B} \cos C - \tilde{A} \cdot \tilde{C} \cos B - \tilde{B} \cdot \tilde{C} \cos A + \tilde{A} \times \tilde{B} \cdot \tilde{C}$$
$$= \cos A \cos B \cos C - \cos A(\tilde{B} \cdot \tilde{C}) - \tilde{A} \cdot \tilde{B} \cos C - \tilde{A} \cdot \tilde{C} \cos B + \tilde{A} \cdot \tilde{B} \times \tilde{C}$$

which is an identity since

$$\tilde{A} \times \tilde{B} \cdot \tilde{C} = \tilde{A} \cdot \tilde{B} \times \tilde{C}.$$

Thus Eq. (A.19) and the associativity of the angle sum operation are established.

Further Algebraic Properties

For any vector $\underline{A}$, it is known that there exists a vector $\underline{B} = (-\underline{A})$ such that $\underline{A} + \underline{B} = \underline{0}$. To show that

$$\underline{A}(+)\ (-\underline{A}) = \underline{0} \qquad (A.22)$$

use Eq. (3.15 to get

$$\frac{\underline{A}(+)(-\underline{A})}{q|\underline{A}(+)(-\underline{A})|} = \frac{\underline{A}}{q(A)} \cos\ (-A) + \frac{(-\underline{A})}{q(-A)} \cos A\ \frac{\underline{A}}{q(A)} \times \frac{(-\underline{A})}{q(-A)}$$

Since

$$\cos(-A) = \cos A$$

$$q(-A) - \frac{\sin(-A)}{-A} = \frac{-\sin A}{-A} = q(A)$$

$$\underline{A} \times (-\underline{A}) = -\underline{A} \times \underline{A} = \underline{0}$$

this reduced to

$$\frac{\underline{A}(+)(-\underline{A})}{q|\underline{A}(+)(-\underline{A})|} = \frac{\underline{A}}{q(A)} \cos A - \frac{\underline{A}}{q(A)} \cos A = \underline{0}$$

In this equation, there is no ambiguity, so Eq. (A.22) is established.

Eq. (A.15) can be used to show that if $\underline{A}$ and $\underline{B}$ are parallel vectors, then

$$\underline{A}(+)\ \underline{B} = \underline{A} + \underline{B} \qquad \text{(A.23)}$$

To show this, write

$$\frac{\underline{A}(+)\underline{B}}{q|\underline{A}(+)\underline{B}|} = \frac{\underline{A}}{A} \sin A \cos B + \frac{\underline{B}}{B} \sin B \cos A - \frac{\underline{A}}{q(A)} \times \frac{\underline{B}}{q(B)}$$

Since $\underline{A}$ and $\underline{B}$ are parallel

$$\underline{A} \times \underline{B} = \underline{0}$$

so

$$\frac{\underline{A}(+)\underline{B}}{q|\underline{A}(+)\underline{B}|} = \underline{1}_A(\sin A \cos B + \sin B \cos A) = \underline{1}_A \sin(A+B)$$

$$= \frac{\underline{A} + \underline{B}}{q|\underline{A}+\underline{B}|}$$

establishing Eq. (A.23).

Finally, to show that

$$-[\underline{A}(+)\underline{B}] = (-\underline{B}) \ (+) \ (-\underline{A}) \tag{A.24}$$

let

$$\underline{x} = -[\underline{A}(+)\underline{B}] \tag{A.25}$$

Then by Eq. (A.22)

$$\underline{A}(+) \ \underline{B}(+) \ \underline{x} = \underline{0}$$

$$(-\underline{A})(+) \ \underline{A}(+) \ \underline{B}(+) \ \underline{x} = -\underline{A}$$

$$\underline{0}(+) \ \underline{B}(+) \ \underline{x} = -\underline{A}$$

The fact that

$$\underline{0}(+) \ \underline{B} = \underline{B}(+) \ \underline{0} = \underline{B} \tag{A.26}$$

can be readily verified using Eq. (A.15). Therefore

$$\underline{B}(+) \ \underline{x} = -\underline{A}$$

Further manipulations give

$$(-\underline{B})(+) \ \underline{B}(+)\underline{x} = (-\underline{B})(+)(-\underline{A})$$

$$\underline{0}(+)\underline{x} = (-\underline{B})(+)(-\underline{A})$$

$$\underline{x} = (-\underline{B})(+)(-A)$$

or in view of Eq. (A.24)

$$-[\underline{A}(+)\underline{B}] = (-\underline{B})(+)(-\underline{A})$$

which was to be proved.

A.3 The Rotation Sum of Two Vectors

What rotation vector $\underline{C}$ produces the same net effect as taking rotation $\underline{A}$ first and then rotation $\underline{B}$? In other words, an expression is sought for

$$\underline{C} = \underline{A} \# \underline{B} \tag{A.27}$$

where the symbol # denotes the rotation sum operation. Eq. (A.27) is to be read as "$\underline{C}$ equals $\underline{A}$ rotation summed with $\underline{B}$".

Consider the special case of a 180° rotation about one axis followed by a second rotation of 180° about a different and intersecting axis. Since the orientation of a rigid body is completely determined by the orientations of any two non-parallel lines within the body, it is sufficient to examine the motion of two lines only. A natural choice of these two lines is the set of axes about which the two 180° rotations are taken. In Figure A.4, these lines are shown as $\underline{M}$ and $\underline{N}$. A_{MN} is the angle from $\underline{M}$ to $\underline{N}$. First rotate the body through 180° about the $\underline{M}$ axis. The lines $\underline{M}$ and $\underline{N}$ are thus rotated into orientations denoted by $\underline{M}'$ and $\underline{N}'$. Next assume a rotation of the body through 180° about the fixed $\underline{N}$ axis. $\underline{M}'$ and $\underline{N}'$ are transformed into $\underline{M}''$ and $\underline{N}''$ by this rotation. The combined effect of the two rotations is to transform $\underline{M}$ and $\underline{N}$ into $\underline{M}''$ and $\underline{N}''$ respectively. Note that

$$A_1 = A_2 = A_3 = A_{MN}$$

It is evident that this transformation is equivalent to a single rotation of magnitude $2A_{MN}$ about an axis in the direction of $\underline{M} \times \underline{N}$. This rotation may be represented by the vector $2\underline{A}_{MN}$, since the orientation of the body is completely defined by the orientation of the lines $\underline{M}$ and $\underline{N}$.

In the general case, the two successive rotations are of arbitrary magnitude, but this same technique may be applied.

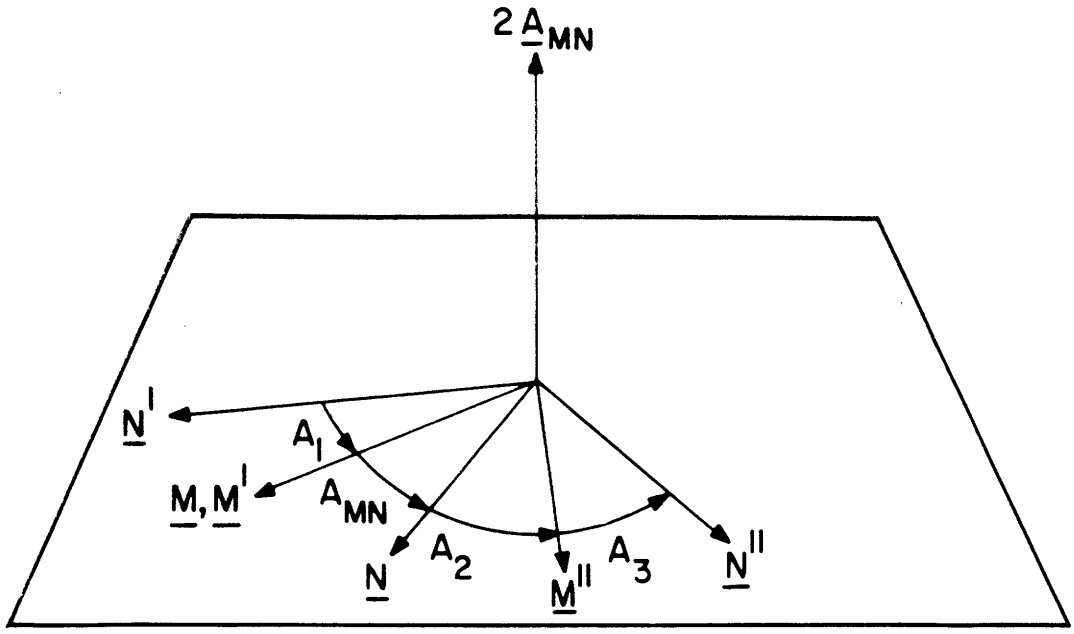


Figure A.4.- The rotation sum - special case

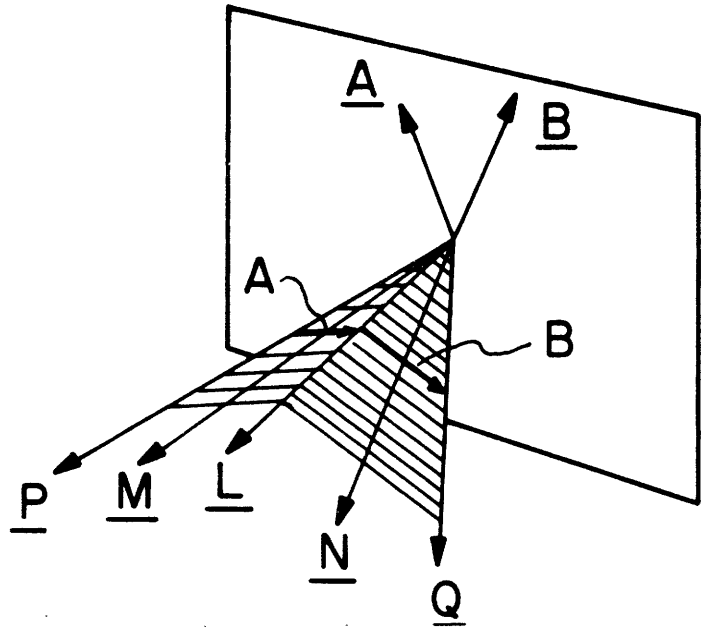


Figure A.5.- The rotation sum - general case

Let $\underline{A}$ and $\underline{B}$ be two vectors representing arbitrary rotations as shown in Figure A.5. Draw line $\underline{L}$ perpendicular to the vectors $\underline{A}$ and $\underline{B}$ as indicated in the figure.

There exists a unique line $\underline{P}$ which is carried into the orientation $\underline{L}$ by rotation $\underline{A}$. Likewise, there exists a unique line $\underline{Q}$ into which $\underline{L}$ is carried by rotation $\underline{B}$. A plane is constructed containing $\underline{P}$ and $\underline{L}$. The line $\underline{M}$ lies in the plane and bisects the angle A_{PL}. Similarly, the line $\underline{N}$ lies in the plane of $\underline{L}$ and $\underline{Q}$ and bisects the angle A_{LQ}. Thus

$$\underline{A} = \underline{A}_{PL} = 2\underline{A}_{ML} \tag{A.28}$$

$$\underline{B} = \underline{A}_{LQ} = 2\underline{A}_{LN} \tag{A.29}$$

By the preceding discussion, the rotation $\underline{A}$ is equivalent to two successive 180° rotations about $\underline{M}$ and $\underline{L}$. Similarly, $\underline{B}$ is equivalent to two successive rotations about $\underline{L}$ and $\underline{N}$. If these four rotations are performed consecutively, the result is equivalent to the combined effect of performing the rotations $\underline{A}$ and $\underline{B}$ in succession. The two intermediate rotations about $\underline{L}$ cancel, so the net result is equivalent to the first rotation of 180° about $\underline{M}$ followed by another rotation of 180° about $\underline{N}$. But these two rotations are equivalent to the single rotation represented by $2\underline{A}_{MN}$. So far it has been shown that

$$\underline{A} \# \underline{B} = 2\underline{A}_{MN} \tag{A.30}$$

From Figure A.5, it is seen that

$$\underline{A}_{MN} = \underline{A}_{ML} (+) \underline{A}_{LN}$$

but from Eqs. (A.28) and (A.29)

$$\underline{A}_{ML} = \frac{\underline{A}}{2}$$

$$\underline{A}_{LN} = \frac{\underline{B}}{2}$$

Using these last three equations in Eq. (3.30) gives

$$\underline{A} \# \underline{B} = 2\left(\frac{\underline{A}}{2} (+) \frac{\underline{B}}{2}\right) \tag{A.31}$$

This equation is the definition of the rotation sum operation in terms of the angle sum operation. From Eq. (A.15) it can be seen that the factor 2 can not be cancelled in Eq. (A.31).

A.4 Algebraic Properties of the Rotation Sum Operation

Since the angle sum operation is non-commutative, and since the rotation sum operation is defined in terms of the angle sum operation by Eq. (A.31), it follows that the rotation sum operation is also non-commutative.

The associativity of the rotation sum is readily established. It is desired to show that

$$[\underline{A}\#\underline{B}] \# \underline{C} = \underline{A} \# [\underline{B}\#\underline{C}] \tag{A.32}$$

Applying Eq. (A.31) twice to the left hand side of Eq. (A.32) gives

$$[\underline{A}\#\underline{B}] \# \underline{C} = 2\left(\frac{\underline{A}}{2} (+) \frac{\underline{B}}{2}\right) \# \underline{C}$$

$$= 2\left[\left(\frac{\underline{A}}{2} (+) \frac{\underline{B}}{2}\right) (+) \frac{\underline{C}}{2}\right] \tag{A.33}$$

Similarly the right hand side of Eq. (A.32) becomes

$$\underline{A} \# [\underline{B}\#\underline{C}] = 2\left[\frac{\underline{A}}{2} (+)\left(\frac{\underline{B}}{2} (+) \frac{\underline{C}}{2}\right)\right]$$

but by the associativity of the angle sum, this is equivalent to Eq. (A.33) and so the associativity of the rotation sum is proved.

The following algebraic properties of the rotation sum operation follow from Eqs. (A.31) and (A.32) and the similar properties of the angle sum operation.

There exists a rotation vector $-\underline{A}$ such that

$$\underline{A} \# (-\underline{A}) = \underline{0} \tag{A.34}$$

If $\underline{A}$ and $\underline{B}$ are parallel, then

$$\underline{A} \# \underline{B} = \underline{A}(+)\ \underline{B} = \underline{A} + \underline{B} \tag{A.35}$$

Also,

$$-[\underline{A}\#B] = (-\underline{B}) \# (-\underline{A}) \tag{A.36}$$

A.5 The Rotation Vector Differential Equation

The differential equation for the rotation vector will be derived next. Let $\underline{\phi}(t)$ be the value of a rotation vector at time t. Let $\underline{\phi}(t+\Delta t)$ be the value of this vector at time $t+\Delta t$. Define

$$\frac{d\underline{\phi}}{dt} = \lim_{\Delta t \to 0} \frac{[\underline{\phi}(t+\Delta t) - \phi(t)]}{\Delta t} \tag{A.37}$$

as the rate of change of the rotation vector $\underline{\phi}(t)$ with respect to time. This leads to the expression

$$\underline{\phi}(t+dt) = \underline{\phi}(t) + d\underline{\phi} \tag{A.38}$$

From the disembodied vector point of view, there are two ways in which an infinitesimal rotation vector, which will be symbolized as $d^{\#}\underline{\phi}$, may be rotation summed with the rotation

vector $\underline{\phi}$. It may be rotation summed from the left or from the right. These operations are

$$\underline{\phi}_{RB}(t+dt) = d\underline{\phi}_{L\,RB}^{\#} \# \underline{\phi}_{RB}(t) \tag{A.39}$$

where the symbol $d\underline{\phi}_{L\,RB}^{\#}$ is that infinitesimal rotation vector that must be rotation summed from the left with $\underline{\phi}_{RB}(t)$ to yield $\underline{\phi}_{RB}(t+dt)$, and

$$\underline{\phi}_{RB}(t+dt) = \underline{\phi}_{RB}(t) \# d\underline{\phi}_{R\,RB}^{\#} \tag{A.40}$$

where $d\underline{\phi}_{R\,RB}^{\#}$ is the right hand differential rotation vector.

By the definition of angular velocity, the incremental change in the rotation vector $\underline{\phi}_{RB}$ as seen by an observer, fixed with respect to the Reference Frame, is ω_{RB}^{R} dt. Thus

$$\underline{\phi}_{RB}^{R}(t+dt) = \underline{\phi}_{RB}^{R}(t) \# \underline{\omega}_{RB}^{R}\, dt \tag{A.41}$$

A comparison of Eqs. (A.40) and (A.41) shows that

$$(d\underline{\phi}_{R\,RB}^{\#})^{R} = \underline{\omega}_{RB}^{R}\, dt \tag{A.42}$$

In the next section, it will be shown that for an arbitrary vector $\underline{v}$,

$$\underline{v}^{R} = (-\underline{\phi}_{RB}) \# \underline{v}^{B} \# \underline{\phi}_{RB}$$

and therefore from Eq. (A.34),

$$\underline{v}^{B} = \underline{\phi}_{RB} \# \underline{v}^{R} \# (-\underline{\phi}_{RB}) \tag{A.43}$$

Eq. (A.41) can now be manipulated as follows:

$$\underline{\phi}^{R}_{RB}(t+dt) = \underline{\phi}^{R}_{RB}(t) \# \underline{\omega}^{R}_{RB}\, dt \# [-\underline{\phi}^{R}_{RB}(t)] \# \underline{\phi}^{R}_{RB}(t)$$

$$= \underline{\omega}^{B}_{RB}\, dt \# \underline{\phi}^{R}_{RB}(t) \qquad (A.44)$$

Now it is true that

$$\underline{\phi}^{R}_{RB}(t+dt) = \underline{\phi}^{B'}_{RB}(t+dt)$$

$$\underline{\phi}^{R}_{RB}(t) = \underline{\phi}^{B}_{RB}(t)$$

where the B' frame is the B frame at time t + dt, so Eq.(A.44) can be written

$$\underline{\phi}^{B'}_{RB}(t+dt) = \underline{\omega}^{B}_{RB}\, dt \# \underline{\phi}^{B}_{RB}(t) \qquad (A.45)$$

Comparison of Eqs. (A.39) and (A.45) shows that

$$(d^{\#}_{L}\underline{\phi}_{RB})^{B} = \underline{\omega}^{B}_{RB}\, dt$$

since Eq. (A.39) defines that variable $\underline{\phi}^{B'}_{RB}(t+dt)$ that results when the vectors $d^{\#}_{L}\underline{\phi}_{RB}$ and $\underline{\phi}_{RB}(t)$ are coordinatized in the Body Frame. By combining Eq. (A.38) with Eq. (A.45), there results

$$\underline{\omega}^{B}_{RB}\, dt \# \underline{\phi}^{B}_{RB} = \underline{\phi}^{B}_{RB} + (d\underline{\phi}_{RB})^{B}$$

Now, rotation sum $-\underline{\phi}_{RB}$ on the right on both sides (and suppress the superscript B under the understanding that it is implied unless otherwise stated) to get

$$\underline{\omega}^{B}_{RB}\, dt = (\underline{\phi}_{RB}+d\underline{\phi}_{RB}) \# (-\underline{\phi}_{RB})$$

It is convenient to introduce the scale change

$$\underline{\phi}_{RB} = 2\underline{\psi} \tag{A.46}$$

This gives

$$\underline{\omega}^{B}_{RB}\, dt = [2(\underline{\psi}+d\underline{\psi})] \# (-2\underline{\psi})$$

Apply Eq. (A.31) to get

$$\frac{1}{2}\,\underline{\omega}^{B}_{RB}\, dt = (\underline{\psi}+d\underline{\psi})(+)(-\underline{\psi})$$

and expand using Eq. (A.15).

$$\frac{1}{2}\,\underline{\omega}^{B}_{RB}\, dt = \frac{\underline{\psi}+d\underline{\psi}}{q|\underline{\psi}+d\underline{\psi}|}\cos\psi - \frac{\underline{\psi}}{q(\psi)}\cos|\underline{\psi}+d\underline{\psi}|$$

$$- \frac{\underline{\psi}+d\underline{\psi}}{q|\underline{\psi}+d\underline{\psi}|} \times \frac{\underline{\psi}}{q(\psi)} \tag{A.47}$$

If terms of order $|d\underline{\psi}|^2$ are neglected, the following relationships can be obtained.

$$|\underline{\psi}+d\underline{\psi}| = [(\underline{\psi}+d\underline{\psi}) \cdot (\underline{\psi}+d\underline{\psi})]^{1/2} = \psi + \frac{\underline{\psi} \cdot d\underline{\psi}}{\psi}$$

$$\cos|\underline{\psi}+d\underline{\psi}| = \cos\psi - \frac{\underline{\psi} \cdot d\underline{\psi}}{q(\psi)}$$

and

$$\frac{1}{q|\underline{\psi}+d\underline{\psi}|} = \frac{\sin\left(\psi + \dfrac{\underline{\psi} \cdot d\underline{\psi}}{\psi}\right)}{\psi + \dfrac{\underline{\psi} \cdot d\underline{\psi}}{\psi}} = \frac{1}{q(\psi)} + \frac{\underline{\psi} \cdot d\underline{\psi}}{\psi^2}\left(\cos\psi - \frac{1}{q(\psi)}\right)$$

When these last two expressions are inserted into Eq. (A.47),

it is found, after a slight manipulation, that

$$\frac{1}{2}\,\underline{\omega}^{B}_{RB}\,dt = \frac{\cos\,\psi}{q(\psi)}\,d\underline{\psi} + \frac{1}{\psi^2}\left(1 - \frac{\cos\,\psi}{q(\psi)}\right)(\underline{\psi}\cdot d\underline{\psi})\underline{\psi} - \frac{\underline{\psi}}{q(\psi)} \times \frac{d\underline{\psi}}{q(\psi)}$$

or, on dividing each side by dt,

$$\frac{1}{2}\,\underline{\omega}^{B}_{RB} = \frac{\cos\,\psi}{q(\psi)}\,\frac{d\underline{\psi}}{dt} + \frac{1}{\psi^2}\left(1 - \frac{\cos\,\psi}{q(\psi)}\right)\left(\underline{\psi}\cdot\frac{d\underline{\psi}}{dt}\right)\underline{\psi} - \frac{1}{q^2(\psi)}\,\underline{\psi} \times \frac{d\underline{\psi}}{dt} \tag{A.48}$$

Two intermediate results can be easily derived.

$$\frac{\cos\,\frac{\phi_{RB}}{2}}{q\left(\frac{\phi_{RB}}{2}\right)} = \frac{1}{q(\phi_{RB})} \tag{A.49}$$

and

$$\frac{1}{q^2\left(\frac{\phi_{RB}}{2}\right)} = \frac{2(1-\cos\,\phi_{RB})}{\phi^2_{RB}} \tag{A.50}$$

Using Eqs. (A.46), (A.49), and (A.50) in Eq. (A.48) gives

$$\underline{\omega}_{RB} = \frac{1}{q(\phi_{RB})}\,\frac{d\underline{\phi}_{RB}}{dt} + \frac{1}{\phi^2_{RB}}\left(1 - \frac{1}{q(\phi_{RB})}\right)\left(\underline{\phi}_{RB}\cdot\frac{d\underline{\phi}_{RB}}{dt}\right)\underline{\phi}_{RB} - \frac{1-\cos\,\phi_{RB}}{\phi^2_{RB}}\,\underline{\phi}_{RB} \times \frac{d\underline{\phi}_{RB}}{dt}$$

and since

$$\left(\underline{\phi}_{RB} \cdot \frac{d\underline{\phi}_{RB}}{dt}\right) \underline{\phi}_{RB} = \phi_{RB}^2 \frac{d\underline{\phi}_{RB}}{dt} + \underline{\phi}_{RB} \times \left(\underline{\phi}_{RB} \times \frac{d\underline{\phi}_{RB}}{dt}\right)$$

this becomes

$$\underline{\omega}_{RB} = \underline{\phi}_{RB} - \frac{1 - \cos \phi_{RB}}{\phi_{RB}^2} \underline{\phi}_{RB} \times \dot{\underline{\phi}}_{RB} + \frac{1}{\phi_{RB}^2}\left(1 - \frac{1}{q(\phi_{RB})}\right) \underline{\phi}_{RB} \times (\underline{\phi}_{RB} \times \dot{\underline{\phi}}_{RB}) \tag{A.51}$$

where $\dot{\underline{\phi}} = d\underline{\phi}/dt$. With $q(\phi_{RB})$ as defined in Eq. (A.2), it is apparent that Eq. (A.51) is exactly the same as Eq. (3.33) using the definitions of Eqs. (3.34) and (3.35).

From Eq. (A.51) it is clear that

$$\underline{\phi}_{RB} \cdot \underline{\omega}_{RB} = \underline{\phi}_{RB} \cdot \dot{\underline{\phi}}_{RB} \tag{A.52}$$

This fact was used in the derivation of Eq. (3.33).

A.6 The Coordinate Transformation

Eq. (A.43) is the rotation sum form of the coordinate transformation. To show that, write

$$\underline{v}^R = C^{RB} \underline{v}^B$$

Using Eq. (3.17) in the above equation and writing the result in vector format gives

$$\underline{v}^R = \underline{v}^B + \frac{\underline{\phi}_{RB}}{q(\phi_{RB})} \times \underline{v}^B + \frac{1 - \cos \phi_{RB}}{\phi_{RB}^2} \underline{\phi}_{RB} \times (\underline{\phi}_{RB} \times \underline{v}^B) \tag{A.53}$$

It will be shown that

$$\underline{v}^R = (-\underline{\phi}_{RB}) \# \underline{v}^B \# \underline{\phi}_{RB} \tag{A.54}$$

can be manipulated into the same form as Eq. (A.53). To do this, make the substitutions

$$2\underline{\psi} = \underline{\phi}_{RE} \tag{A.55}$$

$$2\underline{U} = \underline{V}^{B} \tag{A.56}$$

in Eq. (A.54) and expand using Eq. (A.31). Then Eq. (A.54) becomes

$$\frac{1}{2}\,\underline{V}^{R} = (-\underline{\psi})(+)\ \underline{U}(+)\underline{\psi} \tag{A.57}$$

According to Eqs. (A.16) and (A.20), this can be written as

$$\frac{1/2\ \underline{V}^{R}}{q(1/2V)} = \frac{\underline{U}}{q(U)}\cos^{2}\psi - 2\,\frac{\underline{U}}{q(U)} \times \frac{\underline{\psi}}{q(\psi)}\cos\psi$$

$$+\left\{-\left(\frac{\underline{\psi}}{q(\psi)} \times \frac{\underline{U}}{q(U)}\right) \times \frac{\underline{\psi}}{q(\psi)} + \frac{\underline{\psi}}{q(\psi)}\left(\frac{\underline{\psi}}{q(\psi)} \cdot \frac{\underline{U}}{q(U)}\right)\right\}$$

If the bracketed term is manipulated into the form

$$\left\{-\left(\frac{\underline{\psi}}{q(\psi)} \times \frac{\underline{U}}{q(U)}\right) \times \frac{\underline{\psi}}{q(\psi)} + \frac{\underline{\psi}}{q(\psi)}\left(\frac{\underline{\psi}}{q(\psi)} \cdot \frac{\underline{U}}{q(U)}\right)\right\}$$

$$= \frac{\underline{U}}{q(U)}\sin^{2}\psi - 2\left(\frac{\underline{\psi}}{q(\psi)} \times \frac{\underline{U}}{q(U)}\right) \times \frac{\underline{\psi}}{q(\psi)}$$

and if this is used in Eq. (A.58), then

$$\frac{1/2\ \underline{V}^{R}}{q(1/2V)} = \frac{\underline{\psi}}{q(U)} + 2\cos\psi\,\frac{\underline{\psi}}{q(\psi)} \times \frac{\underline{U}}{q(U)} + 2\,\frac{\underline{\psi}}{q(\psi)} \times \left(\frac{\underline{\psi}}{q(\psi)} \times \frac{\underline{U}}{q(U)}\right) \tag{A.59}$$

Taking the dot product of each side of Eq. (A.59) with itself yields

$$\left|\frac{1/2\ \underline{V}^{R}}{q(1/2V)}\right| = \left|\frac{\underline{U}}{q(U)}\right| \tag{A.60}$$

After two applications of Eq. (A.18) to Eq. (A.57), the result reduces to

$$\cos\left|\frac{1}{2}\,\underline{V}^{R}\right| = \cos U \tag{A.61}$$

From Eqs. (A.60) and (A.61) it is inferred that

$$\left|\frac{1}{2}\,\underline{V}^{R}\right| = \left|\underline{U}\right| \tag{A.62}$$

and as a result

$$q(1/2V) = q(U) \tag{A.63}$$

Eq. (A.63) allows the cancellation of the factor q(1/2V) on the left hand side of Eq. (A.59) with the factor q(U) on the right hand side. The result is

$$\underline{V}^{R} = 2\underline{U} + 2\cos\psi\,\frac{\underline{\psi}}{q(\psi)}\times 2\underline{U} + 2\,\frac{\underline{\psi}}{q(\psi)}\times\left(\frac{\underline{\psi}}{q(\psi)}\times 2\underline{U}\right) \tag{A.64}$$

Now make the inverse substitutions for $\underline{\psi}$ and $\underline{U}$ as given by Eqs. (A.55) and (A.56)

$$\underline{V}^{R} = \underline{V}^{B} + \cos\frac{\phi_{RB}}{2}\,\frac{\underline{\phi}_{RB}}{q\left(\frac{\phi_{RB}}{2}\right)}\times\underline{V}^{B} + \frac{1}{2}\,\frac{1}{q^{2}\left(\frac{\phi_{RB}}{2}\right)}\,\underline{\phi}_{RB}\times(\underline{\phi}_{RB}\times\underline{V}^{B}) \tag{A.65}$$

Finally, the use of Eqs. (A.49) and (A.50) in Eq. (A.65) gives

$$\underline{V}^R = \underline{V}^B + \frac{\underline{\phi}_{RB}}{q(\phi_{RB})} \times \underline{V}^B$$

$$+ \frac{1 - \cos \phi_{RB}}{\phi_{RB}^2} \underline{\phi}_{RB} \times (\underline{\phi}_{RB} \times \underline{V}^B) \qquad (A.66)$$

But Eq. (A.66) is identical to Eq. (A.53) and so Eq. (A.54) is established.

A.7 Other Forms of the $\dot{\phi}$ Equation

Eq. (A.51) may be written as

$$\underline{\dot{\phi}} = \underline{\omega} + B\underline{\phi} \times \underline{\dot{\phi}} - C\underline{\phi} \times (\underline{\dot{\phi}} \times \underline{\dot{\phi}}) \qquad (A.67)$$

where

$$B = \frac{1}{\phi^2} (1 - \cos \phi) \qquad (A.68)$$

$$C = \frac{1}{\phi^2} \left(1 - \frac{\sin \phi}{\phi}\right) \qquad (A.69)$$

This equation may be manipulated into several different forms, two of which are stated in Chapter 3 as Eq. (3.36), rewritten here for convenience

$$\dot{\phi} = \underline{\omega} + \frac{1}{2} \underline{\phi} \times \underline{\omega} + A\underline{\phi} \times (\underline{\phi} \times \underline{\omega}) \qquad (A.70)$$

where

$$A = \frac{1}{\phi^2} \left(1 - \frac{\phi \sin \phi}{2(1 - \cos \phi)}\right) \qquad (A.71)$$

and as Eq. (3.38) also rewritten here for convenience.

$$\underline{\dot{\phi}} = \underline{\omega} + \underline{\phi} \times \left(\frac{C}{B} \underline{\omega} + 2A\underline{\dot{\phi}}\right) \qquad (A.72)$$

Page 166 belongs between pages 158 and 159

$$\left(1 + \frac{C}{A} - \phi^2 C\right)(\underline{\dot{\phi}} - \underline{\omega}) = \underline{\phi} x \left(\frac{C}{2A}\,\underline{\omega} + B\underline{\dot{\phi}}\right) \tag{A.78}$$

It can be verified from Eqs. (A.68), (A.69) and (A.71) that

$$1 + \frac{C}{A} - \phi^2 C \equiv \frac{B}{2A}$$

Use this identity in Eq. (A.78) to get

$$\underline{\dot{\phi}} - \underline{\omega} = \underline{\phi} x \left(\frac{C}{B}\,\underline{\omega} + 2A\underline{\dot{\phi}}\right) \tag{A.79}$$

which when $\underline{\omega}$ is transferred to the right hand side is identical with Eq. (A.72).

Appendix B

ω-Filter Design Details

The ω-Filter is designed using operational amplifier RC active filter networks. The Fairchild Semiconductor μA 709 was the operational amplifier selected. It is described in Reference 19. The multiplier chosen was the GPS Instrument Company Mu40.

Demodulator

The demodulator consists of a gain of 25 preamplifier and a gain of 1/5 multiplier. The preamplifier is shown in Figure B.1. The multiplier connection is shown in Figure B.2.

Signal Generator Section

This section must implement the transfer function

$$F_{sg}(s) = \frac{4.17\left(\frac{s}{10^4}\right)}{\left(\frac{s}{10^4}\right)^2 + \sqrt{2}\left(\frac{s}{10^4}\right) + 1}$$

The circuit for accomplishing this was designed by a procedure given in the Burr-Brown "Handbook of Operational Amplifier Active RC Networks" (Reference 20, p. 78).

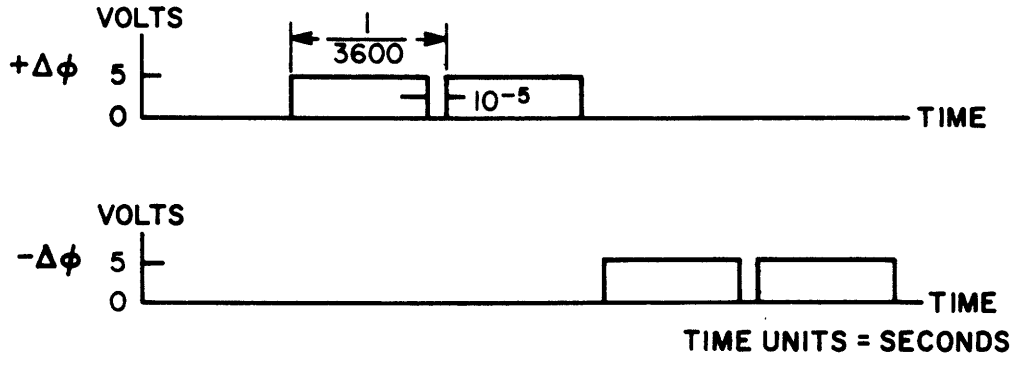
For practical reasons (reasonable resistance and capacitance values) implement the function

$$F_{sg}(s) = \frac{\frac{\sqrt{2}}{2}\left(\frac{s}{10^4}\right)}{\left(\frac{s}{10^4}\right)^2 + \sqrt{2}\left(\frac{s}{10^4}\right) + 1} \qquad (B.1)$$

The resulting circuit is shown in Figure B.2

Torque Generator Section

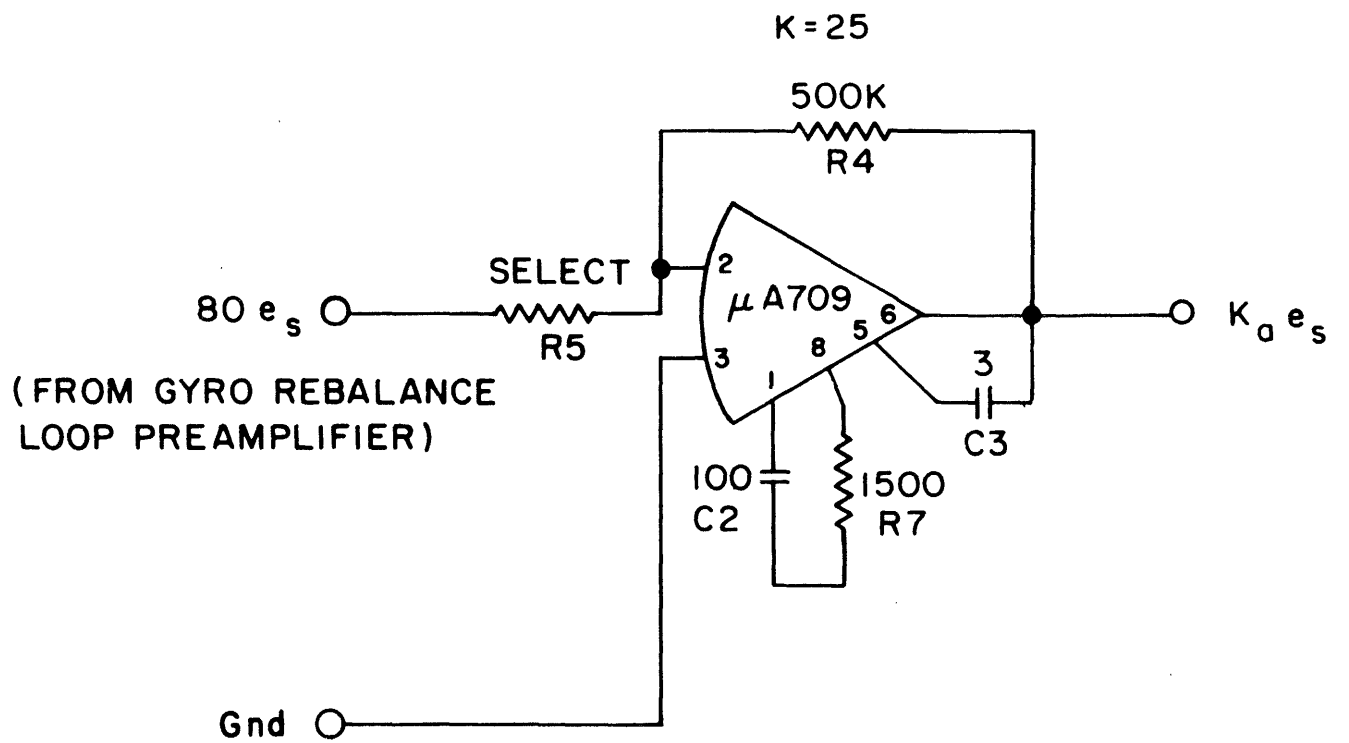
The data pulses ($\Delta\theta$ pulses) from the gyros have the following wave shape (GG 334A Gyro):



The wave shape and duration of the $+\Delta\phi$ pulse is the same as that of the positive torque pulse. The same comment applies to the $-\Delta\phi$ pulse except that the corresponding torque pulse is applied in a negative sense. The transfer function to be implemented by the torque generator section is

$$F_{tg}(s) = \frac{1}{\left(5\,\frac{s}{10^5} + 1\right)\left(5\,\frac{s}{10^4} + 1\right)\left(\frac{s^2}{10^8} + \sqrt{2}\,\frac{s}{10^4} + 1\right)} \qquad (B.2)$$

The resulting circuit is shown in Figure B.2.



ALL RESISTANCE IN OHMS
ALL CAPACITANCES IN PICOFARADS

+15 VOLTS TO PIN 7
-15 VOLTS TO PIN 4

Figure B.1.- Preamplifier

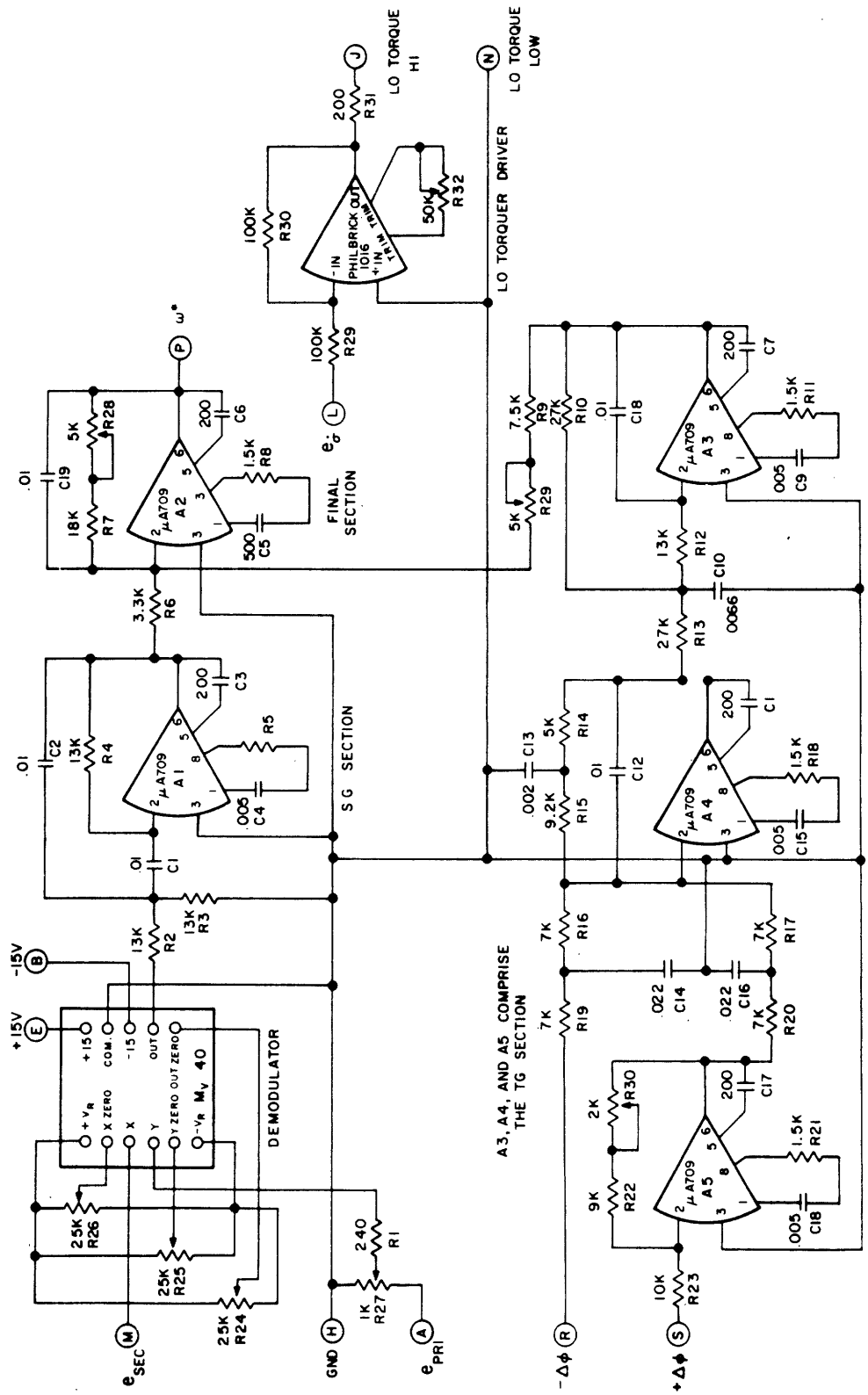


Figure B.2.- ω-filter

Final Section

By comparing Eqs. (4.25) and (B.1) it is seen that the gain of F_{sg} as given in Eq. (B.1) is down by a factor of 5.90 over the gain that is called out in Eq. (4.25). Hence the final section must restore this gain. That is

$$\frac{e_{\omega}}{e_{sgs}} = 5.9$$

Likewise, the gain of Eq. (B.2) differs from that of Eq. (4.27) by a factor of 2. Therefore

$$\frac{e_{\omega}}{e_{tgs}} = 2$$

The circuit diagram for the final section is shown in Figure B.2.

APPENDIX C

Numerical Integration of the ϕ Equation

Three experiments are described in this appendix for verifying the accuracy, stability, and performance of the $\underline{\phi}$ equation selected in Chapter 5. The equation chosen for system mechanization was

$$\dot{\underline{\phi}} = \underline{\omega} + \underline{\phi} \times \left(\frac{1}{3} \underline{\omega} + \frac{1}{6} \dot{\underline{\phi}} \right) \tag{5.13}$$

which is an approximation to

$$\dot{\underline{\phi}} = \underline{\omega} + \underline{\phi} \times \left(\frac{C(\phi)}{B(\phi)} \underline{\omega} + 2A(\phi) \dot{\underline{\phi}} \right) \tag{5.3}$$

where

$$A(\phi) = \frac{1}{\phi^2} \left(1 - \frac{\phi \sin \phi}{2(1 - \cos \phi)} \right)$$

$$B(\phi) = \frac{1}{\phi^2} (1 - \cos \phi)$$

$$C(\phi) = \frac{1}{\phi^2} \left(1 - \frac{\sin \phi}{\phi} \right)$$

A 4th order Runge-Kutta (ref. 21) integration scheme was chosen to perform the numerical integration of Eq. (5.3). The digital computer program for accomplishing this task is included at the end of this appendix. The three experiments are

Eq. (A.70) may be found by taking the cross product of $\underline{\phi}$ into both sides of Eq. (A.67) to get

$$\underline{\phi} \times \dot{\underline{\phi}} = \underline{\phi} \times \underline{\omega} + B\underline{\phi}\times(\underline{\phi}\times\dot{\underline{\phi}}) - C\underline{\phi}\times(\underline{\phi}\times(\underline{\phi}\times\dot{\underline{\phi}}))$$

or simplifying slightly

$$(1-\phi^2 C)\ \underline{\phi}\times\dot{\underline{\phi}} = \underline{\phi}\times\underline{\omega} + B\underline{\phi}\times(\underline{\phi}\times\dot{\underline{\phi}}) \tag{A.73}$$

Now take the cross product of $\underline{\phi}$ into both sides of Eq. (A.73).

$$(1-\phi^2 C)\ \underline{\phi}\times(\underline{\phi}\times\dot{\underline{\phi}}) = \underline{\phi}\times(\underline{\phi}\times\underline{\omega}) - B\phi^2\ \underline{\phi}\times\dot{\underline{\phi}} \tag{A.74}$$

If Eqs. (A.73) and (A.74) are solved simultaneously for $\underline{\phi}\times\dot{\underline{\phi}}$ and $\underline{\phi}\times(\underline{\phi}\times\dot{\underline{\phi}})$, the results are

$$\underline{\phi}\times\dot{\underline{\phi}} = \frac{1-\phi^2 C}{2B}\ \underline{\phi}\times\underline{\omega} + \frac{1}{2}\ \underline{\phi}\times(\underline{\phi}\times\underline{\omega}) \tag{A.75}$$

$$\underline{\phi}\times(\underline{\phi}\times\dot{\underline{\phi}}) = -\frac{\phi^2}{2}\ \underline{\phi}\times\underline{\omega} + \frac{1-\phi^2 C}{2B}\ \underline{\phi}\times(\underline{\phi}\times\underline{\omega}) \tag{A.76}$$

When Eqs. (A.75) and (A.76) are substituted into Eq. (A.67) and the result simplified then Eq. (A.70) is obtained.

To obtain Eq. (A.72), multiply Eq. (A.70) by C/A and add the result to Eq. (A.67) to get.

$$\left(1 + \frac{C}{A}\right)\dot{\underline{\phi}} = \left(1 + \frac{C}{A}\right)\underline{\omega} + \underline{\phi}\times\left(\frac{C}{2A}\ \underline{\omega} + B\dot{\underline{\phi}}\right) - C\underline{\phi}\times(\underline{\phi}\times(\dot{\underline{\phi}}+\underline{\omega})) \tag{A.77}$$

The last term in Eq. (A.77) can be reduced to $C\phi^2(\dot{\underline{\phi}}-\underline{\omega})$ by noting from Eq. (A.67) that $\underline{\phi}\cdot(\dot{\underline{\phi}}-\underline{\omega}) = 0$. Using this result in Eq. (A.77) and rearranging gives

described in the following sections.

Experiment 1

The purpose of this experiment is to verify the integration of Eq. (5.3) yields the correct answer in a familiar situation. The situation chosen to demonstrate the accuracy of Eq. (5.3) is taken from Section 3.3. It can be seen in Figure 3.2 that the initial condition

$$\underline{\phi}_{RB}(1) = \begin{bmatrix} \pi/2 \\ 0 \\ 0 \end{bmatrix} \text{ radians}$$

and an angular velocity

$$\underline{\omega}^{B}_{RB} = \begin{bmatrix} 0 \\ \pi/2 \\ 0 \end{bmatrix} \text{ rad/sec}$$

applied for 1 second should yield

$$\underline{\phi}_{RB}(2) = \frac{1}{3}\, 2\pi\, \frac{1}{\sqrt{3}} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} = 1.209200 \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

The results of the run are shown in printout labelled Experiment 1. It is seen that for dt = 0.01 the numerical integration is accurate to 1 part per million in each component of $\underline{\phi}_{RB}(2)$.

Experiment 2

Experiment 2 serves two purposes. The first is an accurate numerical integration of Eq. (5.3) under circumstances that can be matched by the experimental system. The results of this integration establish a performance bench mark against which the

EXPERIMENT 1 - PRINTOUT

```
APPENDIX C -- NUMERICAL INTEGRATION OF THE PHIDOT EQUATION
Experiment 1
1.570796327  .000000000  .000000000
 .000000000 1.570796327  .000000000
 .000000000  .000000000  .000000000
 .000000000  .000000000  .000000000
 .000000000  .000000000  .000000000
    .0001
   1.0000
      1
   2000
```

TIME	PHI	OMEGA
	1.5707963	.0000000
.00	.0000000	1.5707963
	.0000000	.0000000
	1.5566974	.0000000
.20	.2465556	1.5707963
	.2465577	.0000000
	1.5142225	.0000000
.40	.4920000	1.5707963
	.4920017	.0000000
	1.4428298	.0000000
.60	.7351583	1.5707963
	.7351593	.0000000
	1.3415979	.0000000
.80	.9747285	1.5707963
	.9747286	.0000000
	1.2091987	.0000000
1.00	1.2092003	1.5707963
	1.2091990	.0000000

```
*END-OF-FILE*
```

results of the experimental system can be judged. The conditions are as follows:

$$\underline{\phi}(t_o) = \begin{bmatrix} 0.1 \\ 0.0 \\ 0.0 \end{bmatrix}$$

$$\underline{\omega}_{RB}(t) = \begin{bmatrix} 0.0 \\ 0.1/t_f \\ 0.0 \end{bmatrix} \text{ rad/sec, } t_o < t \leq t_f$$

The result of the integration is

$$\underline{\phi}(t_f) = \begin{bmatrix} 0.099917 \\ 0.099917 \\ 0.005000 \end{bmatrix} \text{ rad}$$

This result is used in Table 5.3.

The second purpose is to demonstrate the stability of Eq. (5.3) when $\underline{\omega}_{RB}$ consists of a nominal value plus additive noise. The noise is taken to be an unbiased Gaussian white noise applied independently to each axis. The predicted error growth rate from passing Gaussian white noise through an integrator is proportional to the square root of time.

A 1000 second integration run was made using a 0.1 second integration step and the following nominal conditions:

$$\underline{\phi}(t_o) = \begin{bmatrix} 0.1 \\ 0.0 \\ 0.0 \end{bmatrix} \text{ rad}$$

$$\underline{\omega}_{RB}(t) = \begin{bmatrix} 0.0 \\ 0.005 \\ 0.0 \end{bmatrix} \text{ rad/sec}$$

$$\begin{matrix} 0<t\leq 20 \\ 20+40i<t\leq 60+40i \end{matrix} \qquad i=1,3,5...$$

and

$$\underline{\omega}_{RB}(t) = -\begin{bmatrix} 0.0 \\ 0.005 \\ 0.0 \end{bmatrix} \text{rad/sec}$$

$$20+40i < t \leq 60+40i \quad i=0,2,4...$$

Every 10 seconds, the result of the nominal integration was compared with the result of a similar integration in which each component of angular velocity was independently corrupted by additive Gaussian noise with a standard deviation of 0.0005 rad/sec. The results of this experiment are plotted in Figure C.1. The horizontal axis is the time axis. The vertical axis is

$$|\delta\underline{\phi}(t)| = \left\{[\underline{\phi}_n(t)-\underline{\phi}(t)] \cdot [\underline{\phi}_n(t)-\underline{\phi}(t)]\right\}^{1/2}$$

where

$\underline{\phi}_n(t)$ is the resulting vector when $\underline{\omega}_{RB}$ is corrupted by noise

$\underline{\phi}(t)$ is the uncorrupted vector

Also plotted in Figure C.1 is

$$\sigma_{\delta\phi} = \sqrt{\sigma^2_{\delta\phi}}$$

where $\sigma^2_{\delta\phi}$ is the variance parameter of $|\delta\underline{\phi}|$. This is found from

$$\sigma^2_{\delta\phi} = 3\sigma^2_{\omega}\Delta t \cdot t$$

where σ^2_{ω} is the variance parameter of the noise process $\underline{\omega}_n$, Δt is the integration time step, and t is the running time parameter. The factor 3 arises because $\underline{\phi}$ is a three dimensional vector. In Experiment 2, a time step of 0.1 second was used.

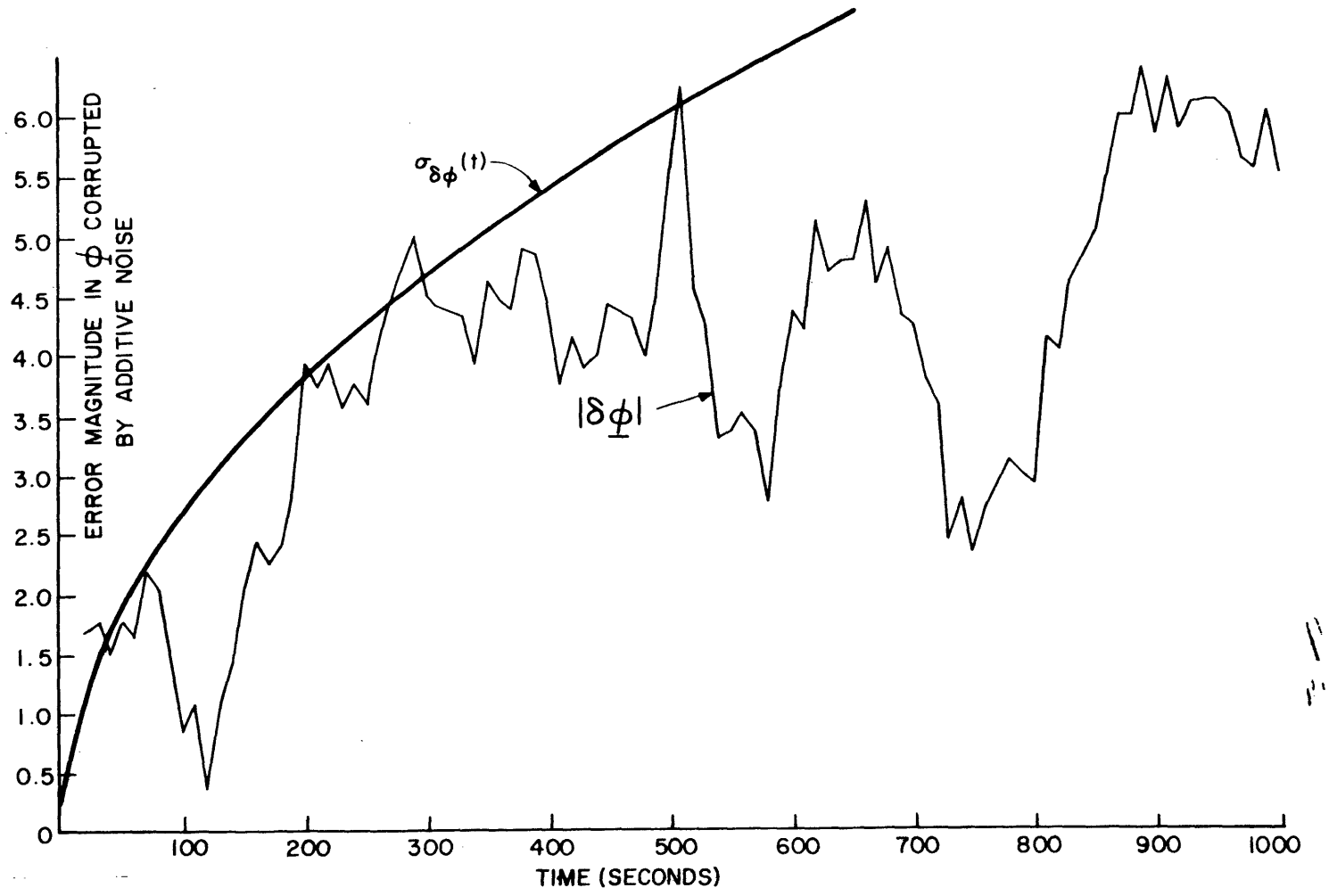


Figure C.1.- $|\delta\underline{\phi}(t)|$ vs time

So

$$\Delta t = 0.1$$

and from the given noise statistics

$$\sigma_{\omega} = 0.0005 \text{ rad/sec}$$

Therefore

$$\sigma_{\delta\phi} = (3\sigma_{\omega}^2 \Delta t \cdot t)^{1/2}$$

$$= 0.0005 \times \sqrt{0.3} \times t^{1/2}$$

$$= 2.7 \times 10^{-4} \, t^{1/2}$$

It can be seen that the results $|\delta\underline{\phi}(t)|$ of the sample run do not differ markedly from the plot of the standard deviation parameter $\sigma_{\delta\phi}(t)$. This parameter represents the expectation of the standard deviation over an infinite set of such numerical integrations.

Experiment 3

This experiment integrates the $\underline{\phi}$ equation for the case of the classical coning motion where

$$\underline{\omega}(t) = \begin{bmatrix} \theta\omega \cos \omega t \\ -\theta\omega \sin \omega t \\ 0 \end{bmatrix}$$

The theoretically predicted value for $\phi_z(t)$ (ref. 22) is

$$\phi_z(t) = \frac{1}{2} \theta^2 \omega t$$

The results of Experiment 3 for the conditions

$$\theta = 10^{-3} \text{ radians}$$

$$\omega = 20\pi \text{ rad/sec}$$

are shown in the printout labelled Experiment 3. The predicted value of $\phi_z(10)$ is

$$\phi_z(10) = \frac{1}{2} (10^{-3})^2 \times 2\pi \times 10$$

$$= \pi \times 10^{-4} \text{ radians}$$

It is seen that the numerical integrations of the $\dot{\phi}$ equation give results in excellent agreement with the predicted results.

APPENDIX C -- NUMERICAL INTEGRATION OF THE PHIDOT EQUATION

EXPERIMENT 3

TIME	PHI X	PHI Y	PHI Z
.0000	.0010000	.0000000	.0000000
1.0000	.0010000	.0000000	.0000314
2.0000	.0010000	.0000000	.0000628
3.0000	.0010000	.0000000	.0000942
4.0000	.0010000	.0000001	.0001257
5.0000	.0010000	.0000001	.0001571
6.0000	.0010000	.0000001	.0001885
7.0000	.0010000	.0000001	.0002199
8.0000	.0010000	.0000001	.0002513
9.0000	.0010000	.0000001	.0002827
10.0000	.0010000	.0000002	.0003142

END-OF-FILE

APPENDIX C -- NUMERICAL INTEGRATION OF THE PHIDOT EQUATION

```
AFORTRAN GO,LS
     1: C     MAIN PROGRAM
     2:       COMMON IREAD, IWRITE, DT, HDT
     3:       IREAD=5
     4:       IWRITE=6
     5:    20 READ(IREAD,1000) NEXPER
     6:       GO TO (40,60,80) NEXPER
     7:    40 CALL EXPER1
     8: C     EXPER1        OMEGA IS PIECEWISE CONSTANT -- NO NOISE
     9:       GO TO 20
    10:    60 CALL EXPER2
    11: C     EXPER2        OMEGA IS PIECEWISE CONSTANT -- ADDITIVE NOISE
    12:       GO TO 20
    13:    80 CALL EXPER3
    14: C     EXPER3        OMEGA PRODUCES CONING MOTION
    15:       GO TO 20
    16:  1000 FORMAT (I1)
    17:       END
```

APPENDIX C -- NUMERICAL INTEGRATION OF THE PHIDOT EQUATION

```
    1:        SUBROUTINE EXPER1
    2:        DIMENSION PHI(3), ANGVEL(3), COLD(3)
    3:        DIMENSION OMEGA(3,4)
    4:        COMMON IREAD, IWRITE, DT, HDT
    5:        READ(IREAD,1000) (PHI(I), I=1,3)
    6:        READ(IREAD,1000) (OMEGA(I,1), I=1,3)
    7:        READ(IREAD,1000) (OMEGA(I,2), I=1,3)
    8:        READ(IREAD,1000) (OMEGA(I,3), I=1,3)
    9:        READ(IREAD,1000) (OMEGA(I,4), I=1,3)
   10:        READ(IREAD,1050) DT
   11:        READ(IREAD,1050) TIME
   12:        READ(IREAD,1100) NINT
   13:        READ(IREAD,1100) NPRINT
   14:        WRITE(IWRITE,1200)
   15:        WRITE(IWRITE,1020) (PHI(I), I=1,3)
   16:        WRITE(IWRITE,1000) (OMEGA(I,1), I=1,3)
   17:        WRITE(IWRITE,1000) (OMEGA(I,2), I=1,3)
   18:        WRITE(IWRITE,1000) (OMEGA(I,3), I=1,3)
   19:        WRITE(IWRITE,1000) (OMEGA(I,4), I=1,3)
   20:        WRITE(IWRITE,1050) DT
   21:        WRITE(IWRITE,1050) TIME
   22:        WRITE(IWRITE,1100) NINT
   23:        WRITE(IWRITE,1150) NPRINT
   24:        HDT=DT/2.0
   25:        LINT =0
   26:        MINT=1
   27:        NDT=0
   28:        RUNTIME=0.0
   29:        SUBTIME=0.0
   30:        WRITE(IWRITE,1250)
   31:        WRITE(IWRITE,1350)          PHI(1),OMEGA(1,1)
   32:        WRITE(IWRITE,1400) RUNTIME,PHI(2),OMEGA(2,1)
   33:        WRITE(IWRITE,1450)          PHI(3),OMEGA(3,1)
   34:     20 J=1
   35:        GO TO 100
   36:     40 J=2
   37:        GO TO 100
   38:     60 J=3
   39:        GO TO 100
   40:     80 J=4
   41:        MINT=0
   42:    100 DO 120 I=1,3
   43:        ANGVEL(I)=OMEGA(I,J)
   44:    120 COLD(I)=OMEGA(I,J)
   45:    200 CALL RKSTEP (PHI,ANGVEL,ANGVEL,ANGVEL,COLD)
   46:        RUNTIME=RUNTIME+DT
   47:        SUBTIME=SUBTIME+DT
   48:        NDT=NDT+1
   49:        IF(NDT-NPRINT) 200,220,220
   50:    220 NDT=0
   51:        WRITE(IWRITE,1350)          PHI(1),ANGVEL(1)
   52:        WRITE(IWRITE,1400) RUNTIME,PHI(2),ANGVEL(2)
   53:        WRITE(IWRITE,1450)          PHI(3),ANGVEL(3)
```

```
APPENDIX C -- NUMERICAL INTEGRATION OF THE PHIDOT EQUATION
   54:        IF(SUBTIME-TIME+0.000001) 200,240,240
   55:   240 LINT=LINT+1
   56:        MINT=MINT+1
   57:        SUBTIME=0.0
   58:        IF(LINT-NINT) 260,280,280
   59:   260 GO TO (20,40,60,80) MINT
   60:   280 RETURN
   61:  1000 FORMAT (3F12.9)
   62:  1020 FORMAT (1H1/3F12.9)
   63:  1050 FORMAT (F10.4)
   64:  1100 FORMAT (I4)
   65:  1150 FORMAT (I4///)
   66:  1200 FORMAT (1H1/1X, 12HEXPERIMENT 1///)
   67:  1250 FORMAT (12X, 4HTIME, 14X, 3HPHI, 16X, 5HOMEGA///)
   68:  1350 FORMAT (26X, F10.7, 10X, F10.7)
   69:  1400 FORMAT (10X, F6.2, 2(10X, F10.7))
   70:  1450 FORMAT (26X, F10.7, 10X, F10.7//)
   71:        END
```

APPENDIX C -- NUMERICAL INTEGRATION OF THE PHIDOT EQUATION

```
    1:       SUBROUTINE EXPER2
    2:       DIMENSION PHI(3), ANGVEL(3), COLD(3)
    3:       DIMENSION PHIN(3), ANGVELN(3), COLDN(3), DELPHI(3)
    4:       DIMENSION OMEGA(3,4)
    5:       COMMON IREAD, IWRITE, DT, HDT
    6:       READ(IREAD,1000) (PHI(I), I=1,3)
    7:       READ(IREAD,1000) (OMEGA(I,1), I=1,3)
    8:       READ(IREAD,1000) (OMEGA(I,2), I=1,3)
    9:       READ(IREAD,1000) (OMEGA(I,3), I=1,3)
   10:       READ(IREAD,1000) (OMEGA(I,4), I=1,3)
   11:       READ(IREAD,1050) DT
   12:       READ(IREAD,1050) TIME
   13:       READ(IREAD,1100) NINT
   14:       READ(IREAD,1100) NPRINT
   15:       READ(IREAD,1070) SDNOISE
   16:       WRITE(IWRITE,1200)
   17:       WRITE(IWRITE,1020) (PHI(I), I=1,3)
   18:       WRITE(IWRITE,1000) (OMEGA(I,1), I=1,3)
   19:       WRITE(IWRITE,1000) (OMEGA(I,2), I=1,3)
   20:       WRITE(IWRITE,1000) (OMEGA(I,3), I=1,3)
   21:       WRITE(IWRITE,1000) (OMEGA(I,4), I=1,3)
   22:       WRITE(IWRITE,1050) DT
   23:       WRITE(IWRITE,1050) TIME
   24:       WRITE(IWRITE,1100) NINT
   25:       WRITE(IWRITE,1070) SDNOISE
   26:       WRITE(IWRITE,1150) NPRINT
   27:       HDT=DT/2.0
   28:       IX=32109
   29:       LINT=0
   30:       MINT=1
   31:       NDT=0
   32:       RMSNOISE=0.0
   33:       RUNTIME=0.0
   34:       SUBTIME=0.0
   35:       WRITE(IWRITE,1220)
   36:       WRITE(IWRITE,1230) SDNOISE
   37:       WRITE(IWRITE,1250)
   38:       WRITE(IWRITE,1350)          PHI(1),PHI(1),OMEGA(1,1)
   39:       WRITE(IWRITE,1400) RUNTIME,PHI(2),PHI(2),OMEGA(2,1),RMSNOISE
   40:       WRITE(IWRITE,1450)          PHI(3),PHI(3),OMEGA(3,1)
   41:       DO 10 I=1,3
   42:    10 PHIN(I)=PHI(I)
   43:    20 J=1
   44:       GO TO 100
   45:    40 J=2
   46:       GO TO 100
   47:    60 J=3
   48:       GO TO 100
   49:    80 J=4
   50:       MINT=0
   51:   100 DO 120 I=1,3
   52:       ANGVEL(I)=OMEGA(I,J)
   53:       COLDN(I)=OMEGA(I,J)
```

APPENDIX C -- NUMERICAL INTEGRATION OF THE PHIDOT EQUATION

```
   54:   120 COLD(I)=OMEGA(I,J)
   55:   160 DO 180 I=1,3
   56:       CALL GAUSS(IX,V)
   57:   180 ANGVELN(I)=ANGVEL(I)+SDNOISE*V
   58:   200 CALL RKSTEP (PHI,ANGVEL,ANGVEL,ANGVEL,COLD)
   59:       CALL RKSTEP (PHIN,ANGVELN,ANGVELN,ANGVELN,COLDN)
   60:       RUNTIME=RUNTIME+DT
   61:       SUBTIME=SUBTIME+DT
   62:       NDT=NDT+1
   63:       IF(NDT-NPRINT) 160,220,220
   64:   220 NDT=0
   65:       DO 230 I=1,3
   66:   230 DELPHI(I)=PHIN(I)-PHI(I)
   67:       RMSNOISE=SQRT(DELPHI(1)**2+DELPHI(2)**2+DELPHI(3)**2)
   68:       WRITE(IWRITE,1350)          PHI(1),PHIN(1),ANGVEL(1)
   69:       WRITE(IWRITE,1400) RUNTIME,PHI(2),PHIN(2),ANGVEL(2),RMSNOISE
   70:       WRITE(IWRITE,1450)          PHI(3),PHIN(3),ANGVEL(3)
   71:       IF(SUBTIME-TIME+0.000001) 160,240,240
   72:   240 LINT=LINT+1
   73:       MINT=MINT+1
   74:       SUBTIME=0.0
   75:       IF(LINT-NINT) 260,250,250
   76:   260 GO TO(20,40,60,80) MINT
   77:   280 RETURN
   78:  1000 FORMAT (3F12.9)
   79:  1020 FORMAT (1H1/3F12.9)
   80:  1050 FORMAT (F10.4)
   81:  1070 FORMAT (F10.7)
   82:  1100 FORMAT (I4)
   83:  1150 FORMAT (I4///)
   84:  1200 FORMAT (1H1/1X, 12HEXPERIMENT 2///)
   85:  1220 FORMAT ($ PHI  WAS GENERATED FROM OMEGA WITHOUT NOISE$)
   86:  1230 FORMAT (84H PHIN WAS GENERATED WITH OMEGA CORRUPTED BY WHITE NOI
   87:      1 WITH A STANDARD DEVIATION OF, F10.7//)
   88:  1250 FORMAT ( 6X, 4HTIME, 8X, 3HPHI, 11X, 4HPHIN, 9X, 5HOMEGA, 8X, 9H
   89:      1S NOISE///)
   90:  1350 FORMAT (10X, 3(4X, F10.7))
   91:  1400 FORMAT ( 3X, F7.2, 3(4X, F10.7), 4X, E12.5)
   92:  1450 FORMAT (10X, 3(4X, F10.7)//)
   93:       END
```

APPENDIX C -- NUMERICAL INTEGRATION OF THE PHIDOT EQUATION

```
      SUBROUTINE EXPER3
      DIMENSION PHI(3),COLD(3),ANGVELA(3),ANGVELB(3),ANGVELC(3)
      COMMON IREAD, IWRITE, DT, HDT
      READ(IREAD,1000) OMEGA
      READ(IREAD,1000) CONINGAMP
      READ(IREAD,1000) DT
      READ(IREAD,1000) RUNTIME
      READ(IREAD,1100) NPRINT
      WRITE(IWRITE,1050) OMEGA
      WRITE(IWRITE,1000) CONINGAMP
      WRITE(IWRITE,1000) DT
      WRITE(IWRITE,1000) RUNTIME
      WRITE(IWRITE,1100) NPRINT
      A=CONINGAMP*OMEGA
      HDT=DT/2.0
      NDT=0
      PHI(1)=CONINGAMP
      PHI(2)=0.0
      PHI(3)=0.0
      TIME=0.0
      WRITE(IWRITE,1200)
      WRITE(IWRITE,1250)
      WRITE(IWRITE,1350) TIME, (PHI(I), I=1,3)
      ANGVELA(1)=0.0
      ANGVELA(2)=A
      ANGVELA(3)=0.0
      COLD(1)=0.0
      COLD(2)=A
      COLD(3)=0.0
      GO TO 60
   20 DO 40 I=1,3
   40 ANGVELA(I)=ANGVELC(I)
   60 ANGVELB(1)=-A*SIN(OMEGA*(TIME+HDT))
      ANGVELB(2)= A*COS(OMEGA*(TIME+HDT))
      ANGVELB(3)=0.0
      ANGVELC(1)=-A*SIN(OMEGA*(TIME+DT))
      ANGVELC(2)= A*COS(OMEGA*(TIME+DT))
      ANGVELC(3)=0.0
      CALL RKSTEP (PHI,ANGVELA,ANGVELB,ANGVELC,COLD)
      TIME=TIME+DT
      NDT=NDT+1
      IF(NDT-NPRINT) 20,80,80
   80 NDT=0
      WRITE(IWRITE,1350) TIME, (PHI(I), I=1,3)
      IF(TIME-RUNTIME+0.000001) 20,100,100
  100 RETURN
 1000 FORMAT (F13.8)
 1050 FORMAT (1H1/F13.8)
 1100 FORMAT (I4)
 1200 FORMAT (1H1///1X, 12HEXPERIMENT 3///)
 1250 FORMAT (8X, 4HTIME, 14X, 5HPHI X, 10X, 5HPHI Y, 10X, 5HPHI Z///)
 1350 FORMAT (5X, F8.4, 5X, 3(5X, F10.7))
      END
```

```
APPENDIX C -- NUMERICAL INTEGRATION OF THE PHIDOT EQUATION                    DA
    1:        SUBROUTINE RKSTEP (FEE,DA,DB,DC,DD)
    2:        DIMENSION FEE(3),DA(3),DB(3),DC(3),CA(3),CB(3),CC(3),CD(3),SIGMA
    3:       1),FOE(3)
    4:        COMMON IREAD, IWRITE, DT, HDT
    5:        CALL XPROD(DA,FEE,CD,SIGMA)
    6:        DO 10 I=1,3
    7:     10 CA(I)=DA(I)+SIGMA(I)
    8: C      OBTAIN AN INITIAL CA.
    9:        CALL XPROD(DA,FEE,CA,SIGMA)
   10:        DO 20 I=1,3
   11:        CA(I)=DA(I)+SIGMA(I)
   12: C      OBTAIN A FINAL CA.
   13:     20 FOE(I)=FEE(I)+HDT*CA(I)
   14:        CALL XPROD(DB,FOE,CA,SIGMA)
   15:        DO 40 I=1,3
   16:        CB(I)=DB(I)+SIGMA(I)
   17:     40 FOE(I)=FEE(I)+HDT*CB(I)
   18:        CALL XPROD(DB,FOE,CB,SIGMA)
   19:        DO 60 I=1,3
   20:        CC(I)=DB(I)+SIGMA(I)
   21:     60 FOE(I)=FEE(I)+ DT*CC(I)
   22:        CALL XPROD(DC,FOE,CC,SIGMA)
   23:        DO 80 I=1,3
   24:        CD(I)=DC(I)+SIGMA(I)
   25:     80 FEE(I)=FEE(I)+DT*(CA(I)+2*(CB(I)+CC(I))+CD(I))/6.0
   26:        RETURN
   27:        END
```

APPENDIX C -- NUMERICAL INTEGRATION OF THE PHIDOT EQUATION

```
    1:        SUBROUTINE XPROD (OGA,PHIE,PHIDOT,SIGA)
    2:        DIMENSION OGA(3),PHIE(3),PHIDOT(3),SIGA(3)
    3:        FIESQR= PHIE(1)*PHIE(1)+PHIE(2)*PHIE(2)+PHIE(3)*PHIE(3)
    4:        IF(FIESQR-0.0001) 20,20,40
    5:     20 CA=(1.0+FIESQR/30.0+FIESQR*FIESQR/840.0)/3.0
    6:        CB=(1.0+FIESQR/60.0+FIESQR*FIESQR/2520.0)/6.0
    7:        GO TO 60
    8:     40 FIE=SQRT(FIESQR)
    9:        A=1.0-COS(FIE)
   10:        B=1.0-SIN(FIE)/FIE
   11:        CA=B/A
   12:        C=FIE*SIN(FIE)
   13:        CB=2.0*(1.0-C/(2.0*A))/FIESQR
   14:     60 SIGA(1)=PHIE(2)*(CA*OGA(3)+CB*PHIDOT(3))
   15:        SIGA(1)=SIGA(1)-PHIE(3)*(CA*OGA(2)+CB*PHIDOT(2))
   16:        SIGA(2)=PHIE(3)*(CA*OGA(1)+CB*PHIDOT(1))
   17:        SIGA(2)=SIGA(2)-PHIE(1)*(CA*OGA(3)+CB*PHIDOT(3))
   18:        SIGA(3)=PHIE(1)*(CA*OGA(2)+CB*PHIDOT(2))
   19:        SIGA(3)=SIGA(3)-PHIE(2)*(CA*OGA(1)+CB*PHIDOT(1))
   20:        RETURN
   21:        END
```

APPENDIX C -- NUMERICAL INTEGRATION OF THE PHIDOT EQUATION

```
      SUBROUTINE GAUSS(IX,V)
      A=0.0
      DO 50 I=1,12
      CALL RANDU(IX,IY,Y)
      IX=IY
   50 A=A+Y
      V=A-6.0
C     V IS NORMALLY DISTRIBUTED WITH ZERO MEAN AND UNITY STAN. DEV.
      RETURN
      END
```

```
'ENDIX C -- NUMERICAL INTEGRATION OF THE PHIDOT EQUATION
  1:        SUBROUTINE RANDU (IX,IY,YFL)
  2: C      YFL IS A UNIFORMLY  DISTRIBUTED RANDOM VARIABLE ON (0,1).
  3:        IY=IX*1953125
  4:        IF (IY) 5, 6, 6
  5:      5 IY=IY+8388607+1
  6:      6 YFL=IY
  7:        YFL=YFL**.11920928955E-6
  8:        RETURN
  9:        END
```

APPENDIX D

HYBRID COMPUTER ANALOG PROGRAM

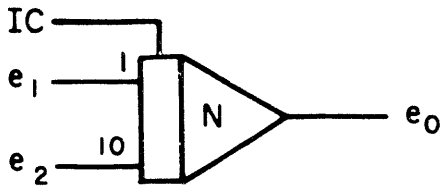
The amalog computer used in the thesis experiment was a Beckman 2200 analog computer. The operation and patching of the Beckman 2200 are described in Reference 23.

The analog computer symbols used in Appendix D are shown in Figure D.1. Tables D.2 and D.3 are computer set-up check lists. Figure D.2 is self explanatory.

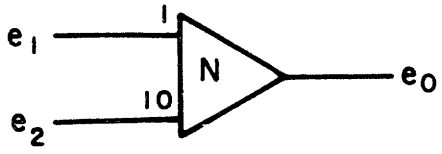
The complementary $\dot{\phi}$ integrators discussed in Chapter 5 are shown in Figures D.3 and D.4. In normal operation, integrators 72, 73, and 74 are in the integrate mode, switches 24, 25, and 26 are closed; integrators 76, 77, and 78 are in initial condition mode, and switches 36, 37, and 38 are open. When $\dot{\phi}$ is reset, the roles integrate and initial condition, open and closed are interchanged.

The cross product term generation discussed in Chapter 5 is shown in Figure D.5. Figure D.6 is self explanatory.

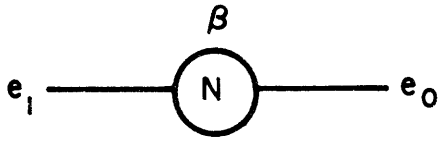
Figures D.7, D.8, and D.9 are discussed in Appendix E for the most part. The function of the circuitry in Figure D.7 not discussed in Appendix E is to generate the timing sequences shown in Figure D.8.



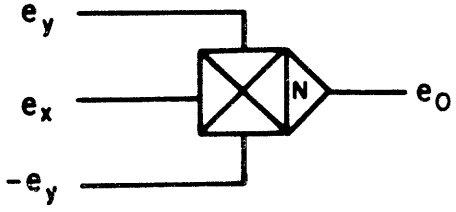
INTEGRATOR $e_0 = -\int_0^t (1 \cdot e_1 + 10 \cdot e_2)\, dt + IC$



SUMINING AMPLIFIER $e_0 = -(1 \cdot e_1 + 10 \cdot e_2)$



POTENTIOMETER $e_0 = \beta\, e_1$



MULTIPLIER $e_0 = \frac{e_x\, e_y}{100}$

TRUNK LINE

A = T TRUNK ≈ ANALOG BD-OUTSIDE
A = P TRUNK ≈ ANALOG BD-CONTROL BD
A = E TRUNK ≈ CONTROL BD-LOGIC BD

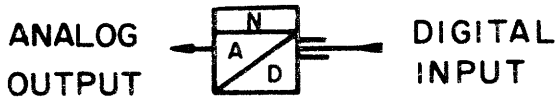


Figure D.1.- Analog computer symbols

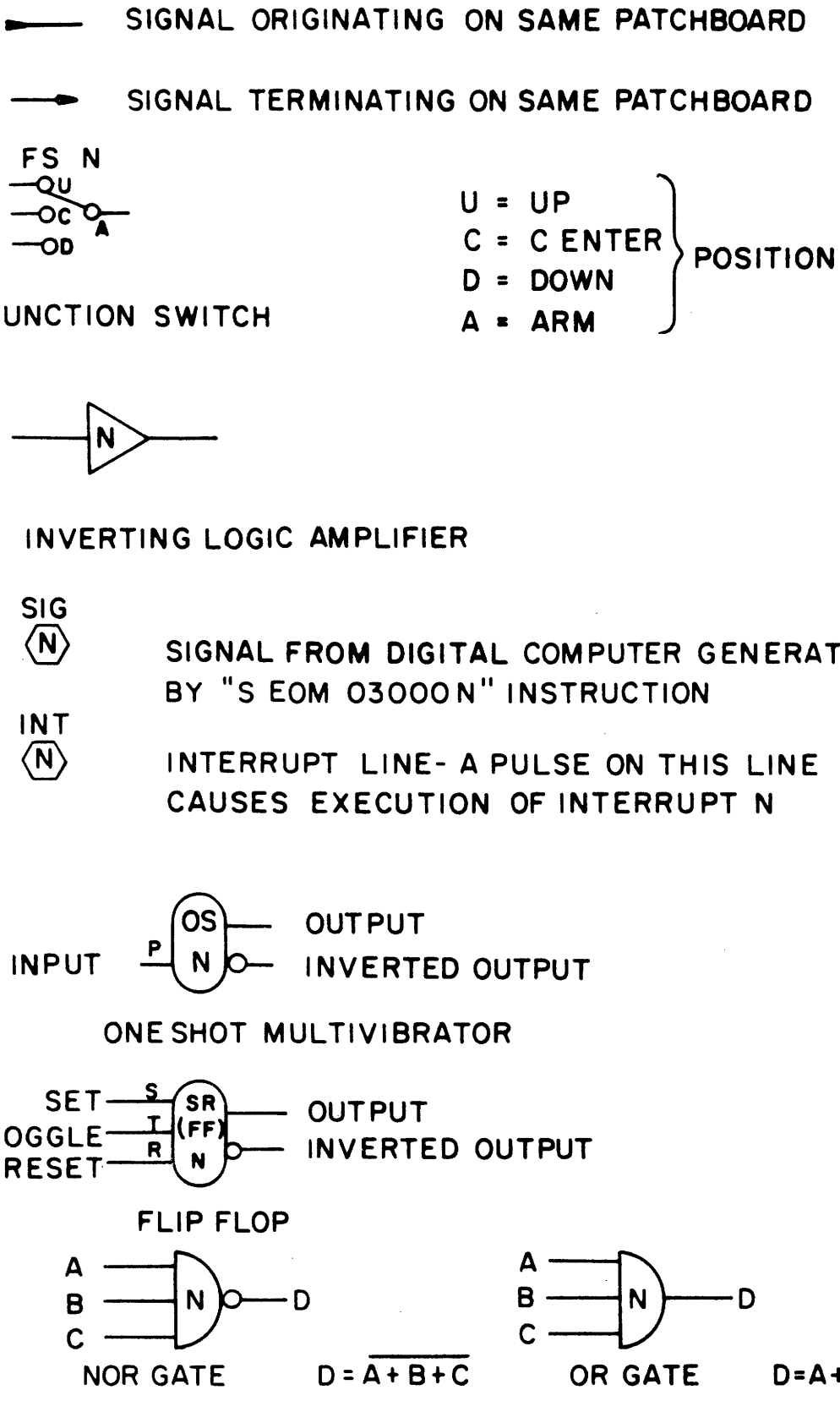


Figure D.1(cont).- Analog computer symbols

1.	ISU Cable		Disconnect
2.	Console Switch		1A
3.	9300 Console		
	Sense Switches 2 & 4		Set
	Others		Reset
4.	Logic Board FF00, SR00		Set
5.	Function Switches	6,7,8	Down
6.	Perform Anacheck	3,4,5,9,10,11	Up
		12,13,14	Center
7.	ISU Cable		Connect
8.	Function Switches	6,7,8	Center
9.	Adjust R32 (See Figure B2) on each ω-filter for stationary Butterfly Pattern		
10.	Adjust R412, R414, R416 to null ω reading (observe output of A015, A019, A023 respectively)		

Table D.1 Anacheck and Circuit Set Up

1.	Console Switch	1A		
2.	Preset Counter 2	36		
3.	One Shot Multivibrators			
	a. Con Bd	1	10 Sec	-5ccw
	b. Log Bd	0	100 "	-4 "
		1	10 "	-5 "
		2	10 "	-5 "
		3	10 "	-5 "
		4	10 "	-5 "
		5	10 "	-5 "
		10	10 "	-5 "
		12	10 "	-5 "
		13	10 "	-5 "
4.	Interrupt Switches	0,1,2,3,10,11		Up
		Others		Down
5.	Function Switches	12,13,14		Center
		3,4,5,6,7,8,9,10,11		Down

Table D.2 Analgo Computer Set Up

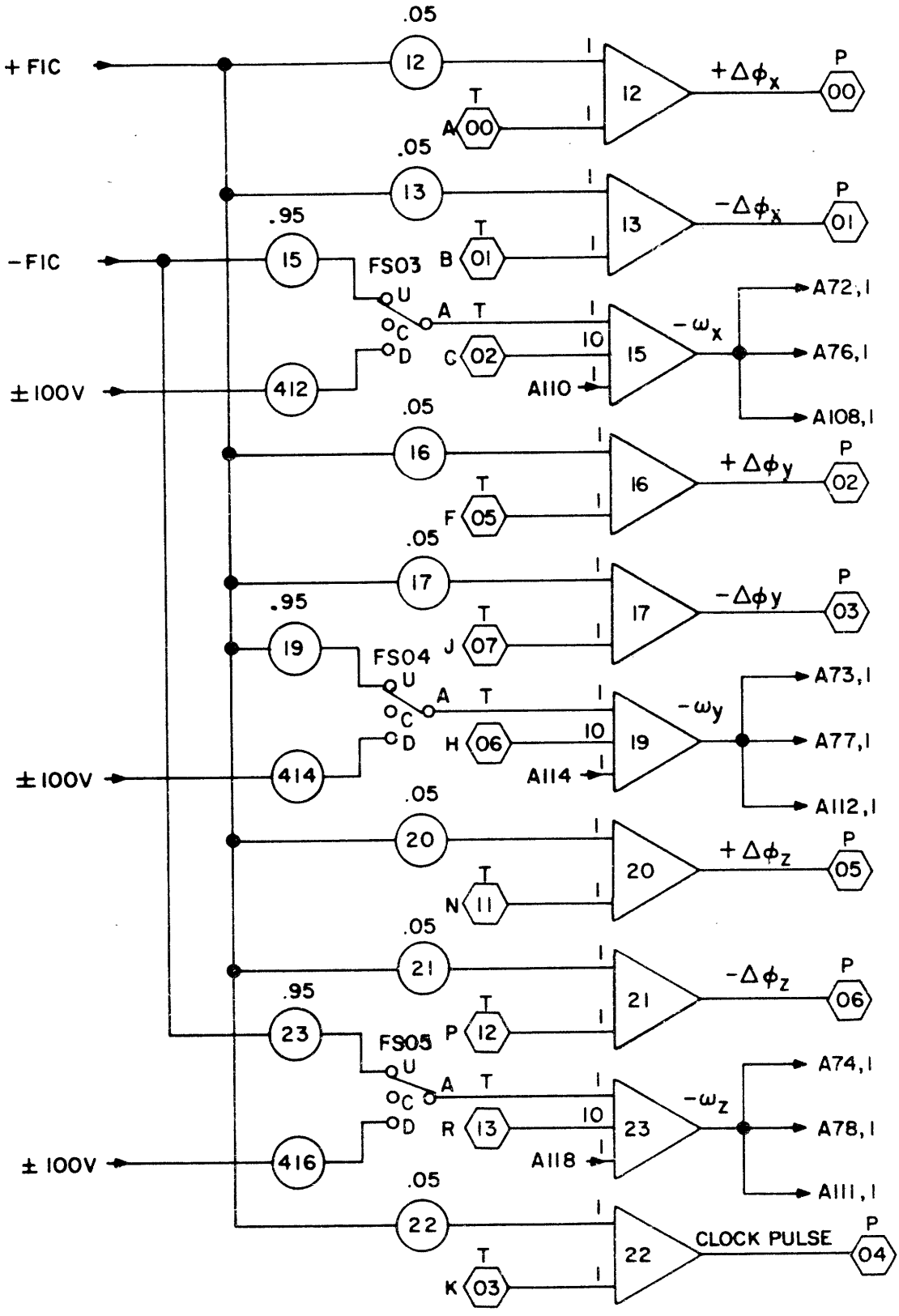


Figure D.2.- ISU input buffers

Note: Letter to left of T-Trunk symbol indicates connector pin designation

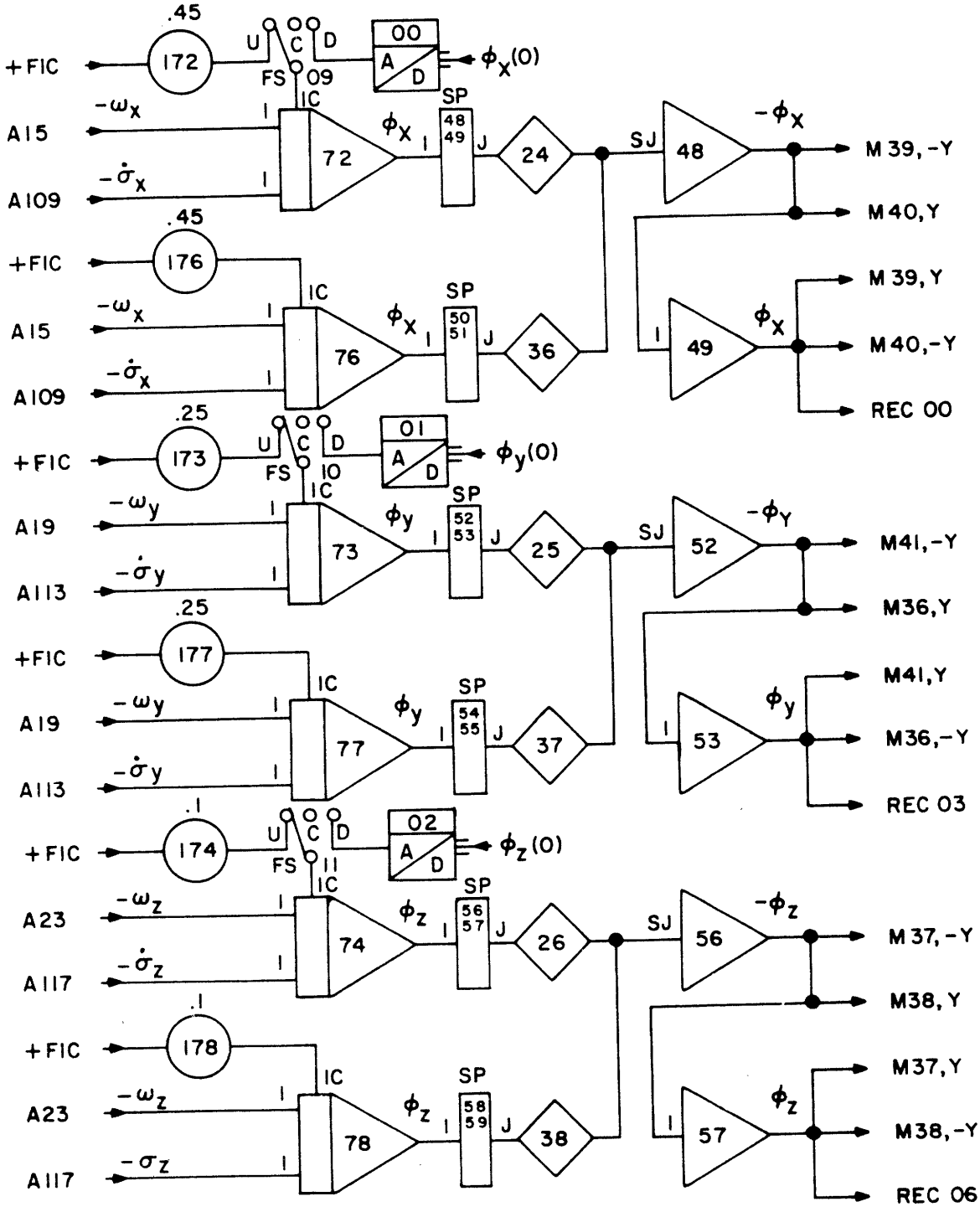


Figure D.3.- $\pm$ Integrators (excluding mode control)

SR 00	FF 00	I 72 I 73 I 74	I 76 I 77 I 77	CS 24 CS 25 CS 26	CS 36 CS 37 CS 38
RESET	RESET	COMP	I.C.	CLOSED	OPEN
RESET	SET	I.C.	COMP	OPEN	CLOSED
SET	RESET	COMP	COMP	OPEN	OPEN
SET	SET	I.C.	I.C.	CLOSED	CLOSED

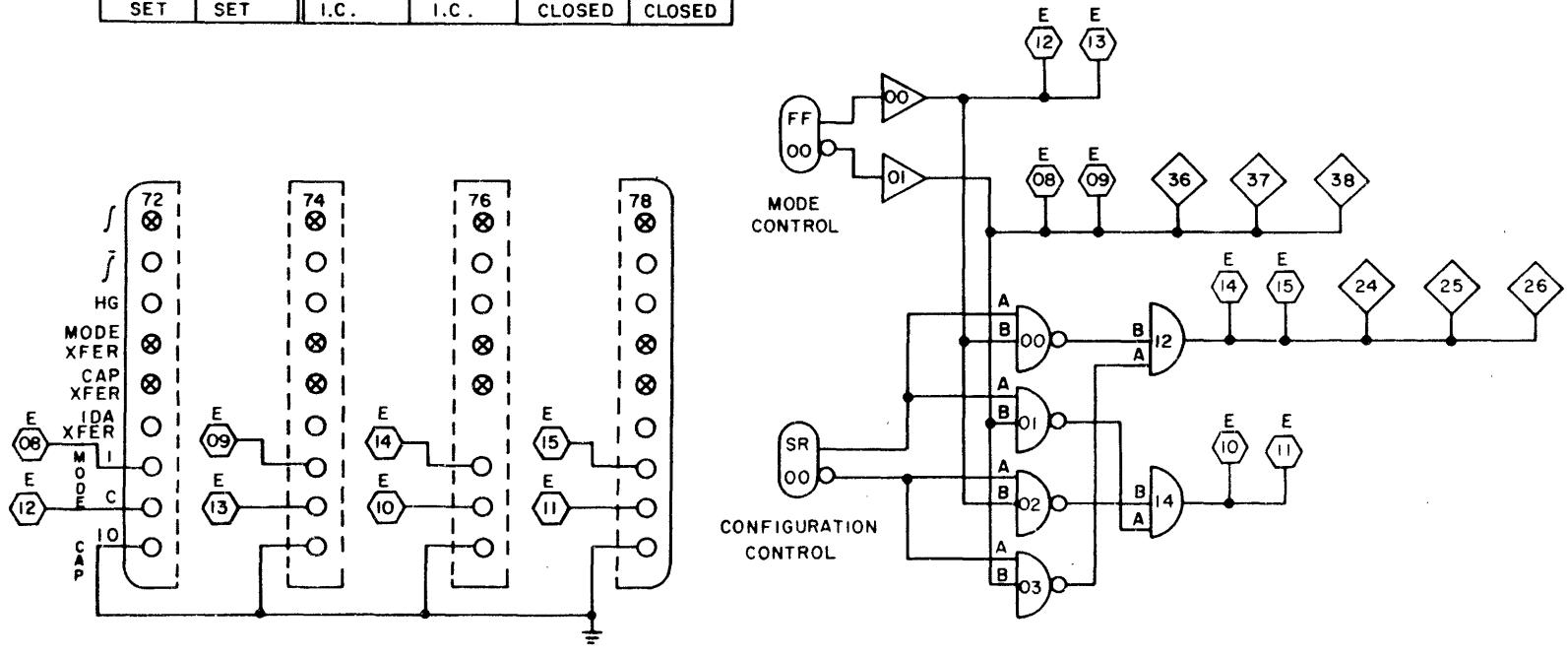


Figure D.4.- Integrator mode, configuration and switch control

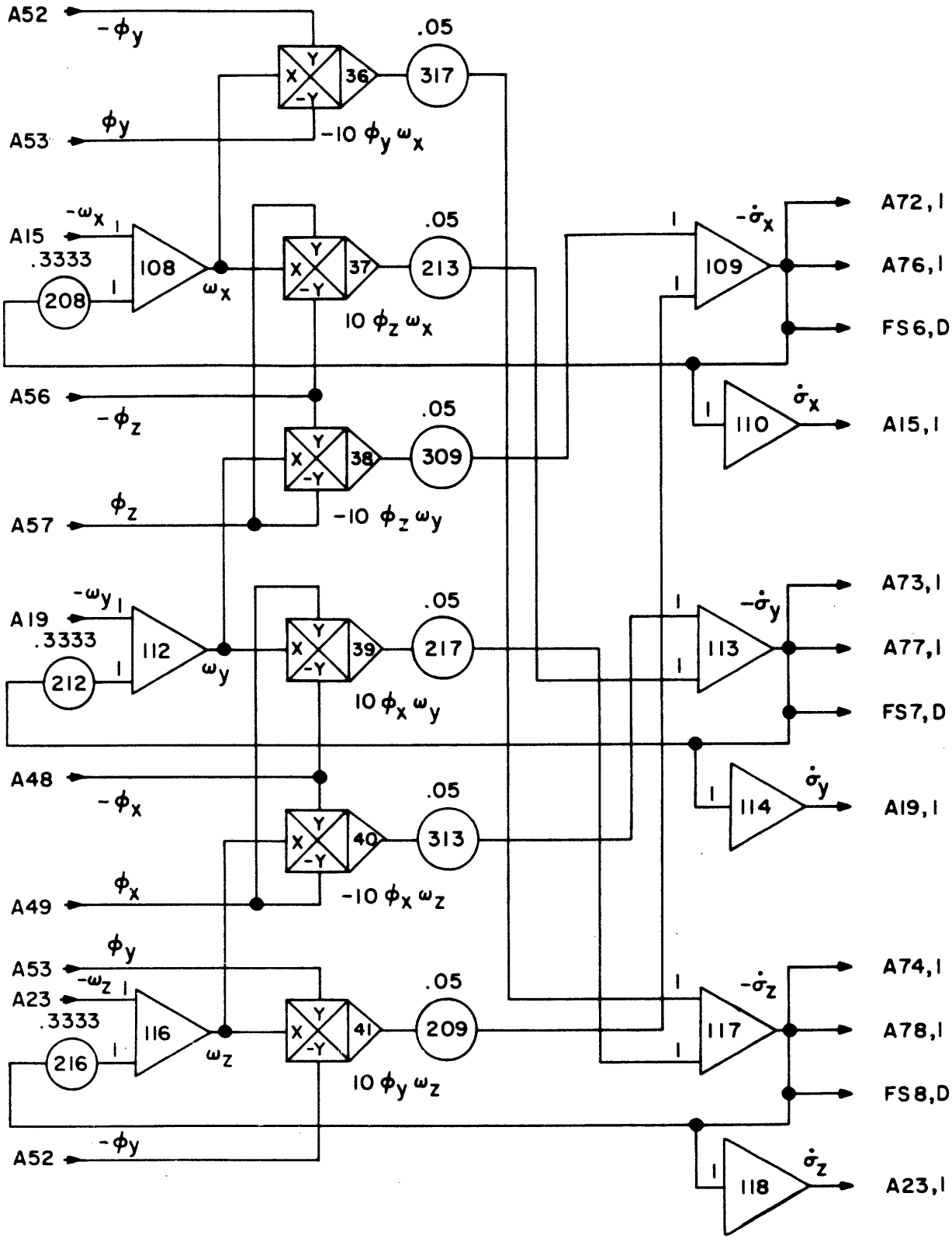


Figure D.5.- Cross-product term generator

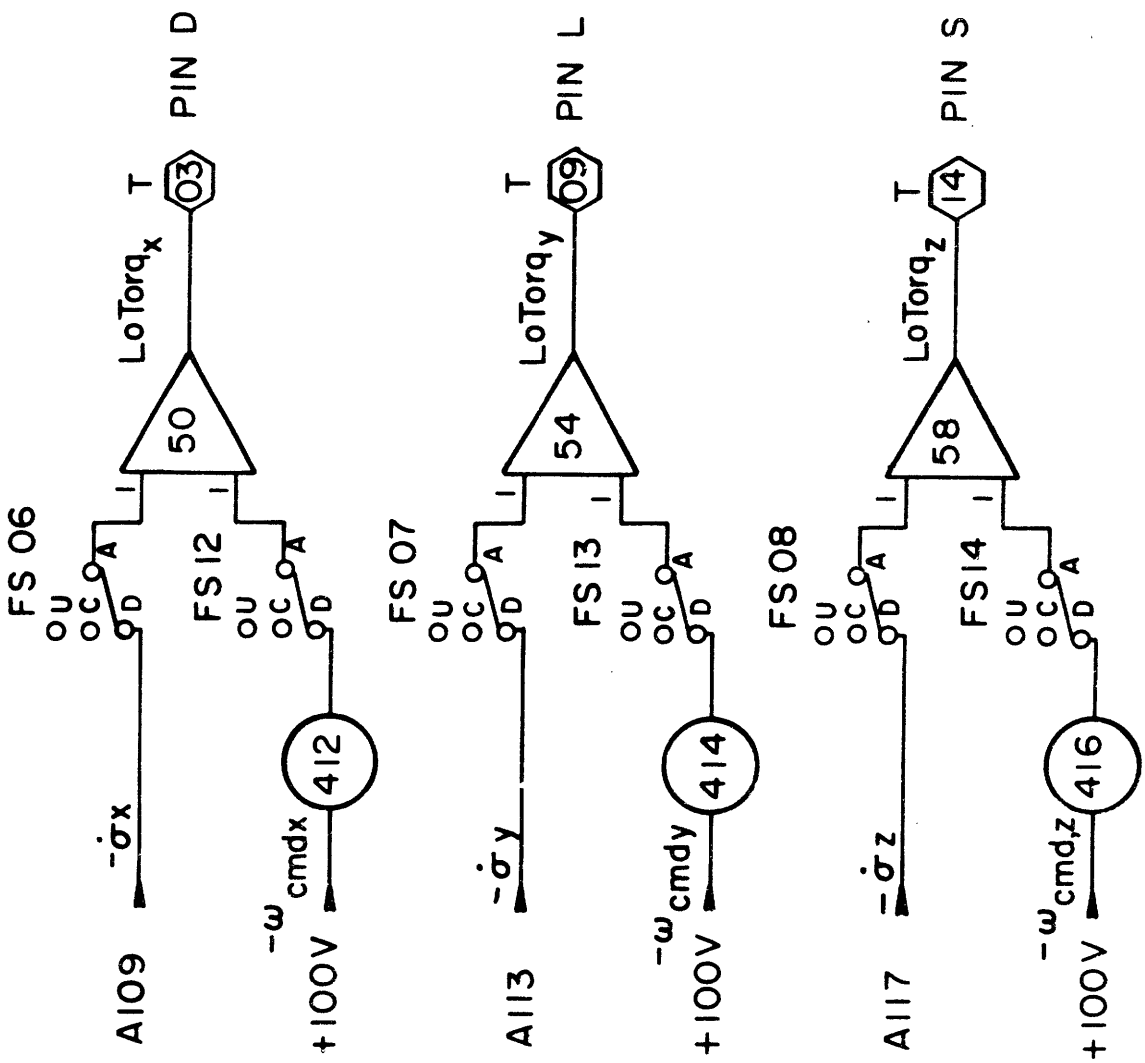
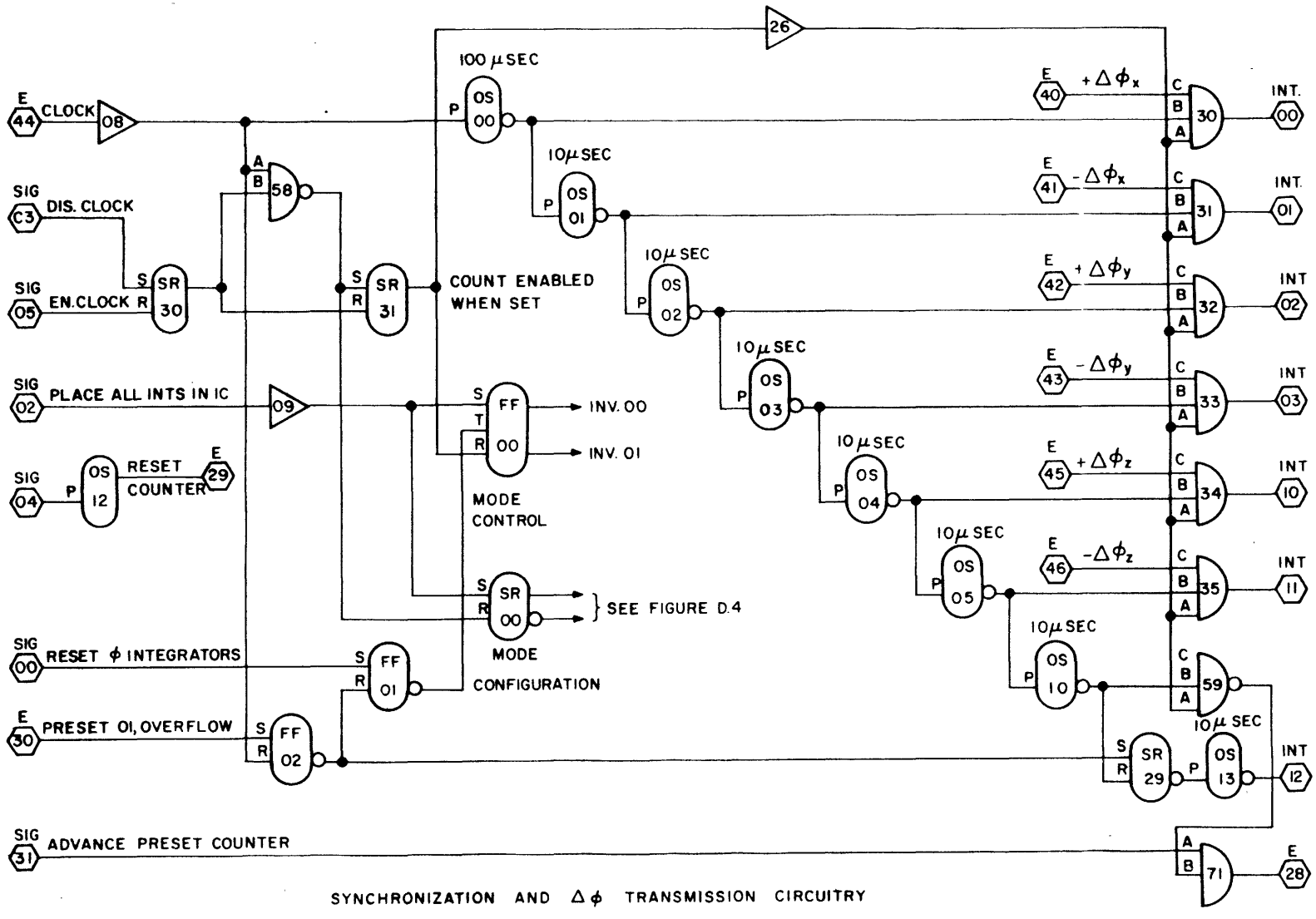


Figure D.6.- Gyro to torquer signal synthesis



SYNCHRONIZATION AND Δφ TRANSMISSION CIRCUITRY

Figure D.7.- Synchronization and Δφ transmission circuitry

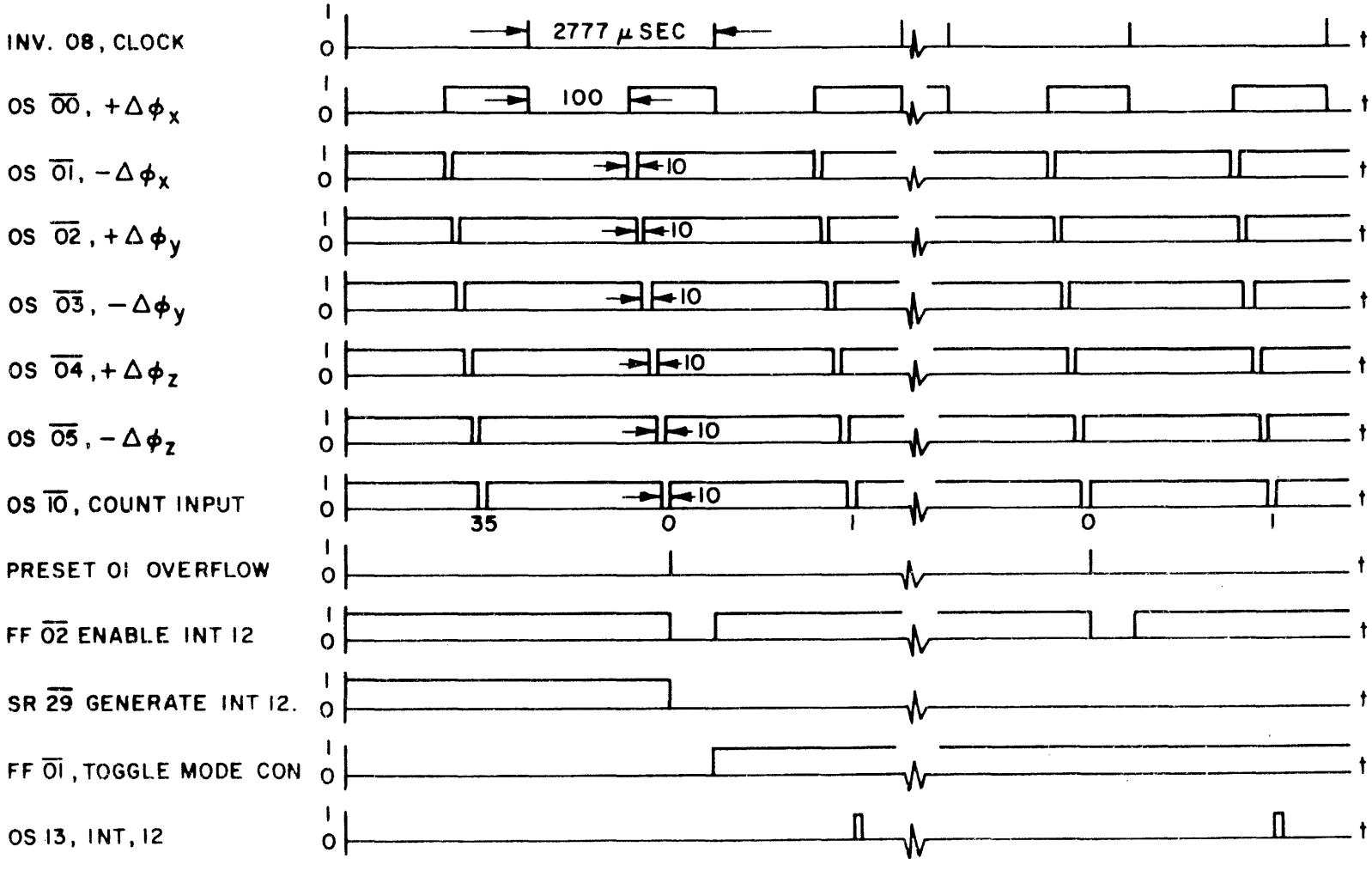
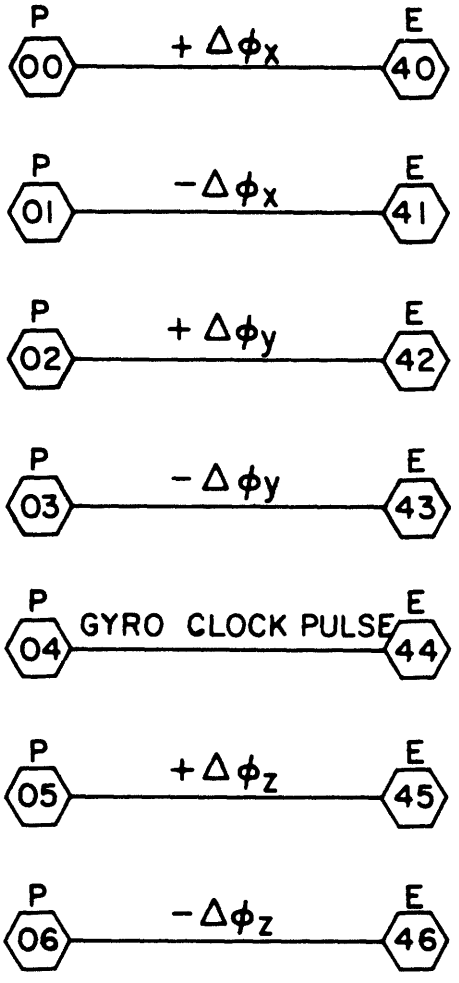


Figure D.8.- Interrupt signal timing diagram



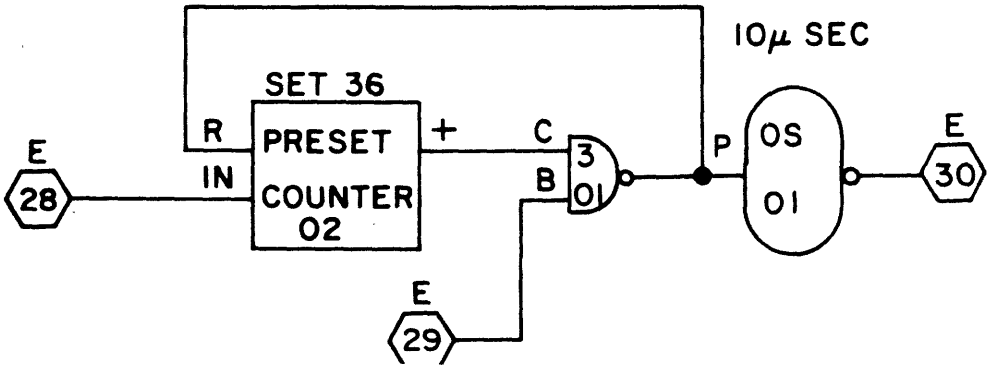


Figure D.9.- Clock pulse counter and trunks

APPENDIX E

HYBRID COMPUTER DIGITAL PROGRAM

An SDS 9300 digital computer was used to perform the digital part of the hybrid computation. The programming language is a combination of BECKTRAN (a Fortran II variant with hybrid computation statements) described in Reference 24 and machine language Symbol instructions described in Reference 25. The Symbol instructions are prefaced by the letter S in column 1.

Main Program

The Main Program accepts typewritten inputs for the printout title block. It then types "Ready to Run" and then "Pause". When the pause is cleared (by toggling Sense Switch 6), the Main Program calls a series of three subroutines to initialize the Hybrid computer. In the final subroutine in this series, Subroutine Run Start, the computer idles until a command is received to begin the run.

The runtime part of the Main Program is an idle loop in which the computer awaits either a command to end the run or to update the direction cosine matrix. In a navigation computer, this idle time would be used for the other required computations of the navigation system.

The flags ISS2OLD and ISS3OLD are used in Subroutine Data In. The flags IUPAA, IUPBB, and IEND are set in Subroutine

Increment Phi.

The Main Program flow chart is shown in Figure E.1.

Subroutine Data In

The functions of Sense Switches 2 and 3 and their associated programming is to call for new Run Parameters (total time, time between printouts, and ϕ_{MAX}) and for new Initial Conditions respectively whenever their settings are changed. The heading is printed on the printout page. FEESQMAX, NPERPR, and NOUT are used in Subroutine Increment Phi.

Subroutine Computer Set Up

The Symbol instruction EOM 030003 (Sig 03)* sets the flip-flop SR30 (Figure D.7). This action resets flip-flop SR31 disabling gates 58, 30-35, and 59. This in turn blocks both the interrupt signals and the clock pulse which advances Preset Counter 02 (Figure D.9). Preset Counter 02 divides the gyro clock pulse by 36. The Symbol instruction EOM 030004 (Sig 04) pulses the one shot multivibrator 0S12 (Figure D.7) which then resets Preset Counter 02 (Figure D.9).

The various flags, registers, and the direction cosine matrix are initialized.

The initial value of $\underline{\phi}$ (from Subroutine Data In) is transmitted to the integrators (Figure D.3) by the CALL DAL instruction. This initial value is then incorporated into the CZERO

*Sig 03 refers to the analog computer designation for the Logic Board patch point on which the pulse generated by EOM 030003 arrives.

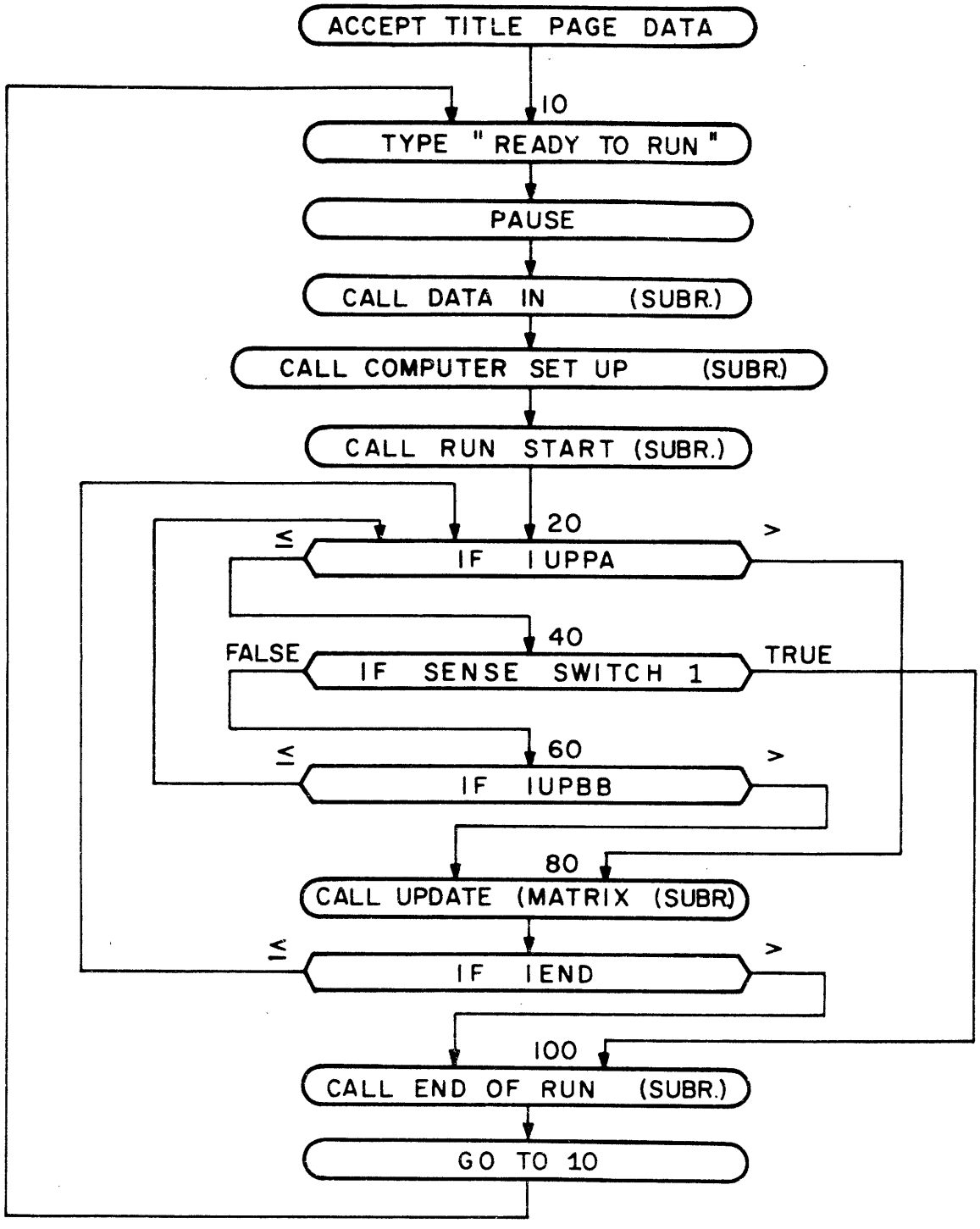


Figure E.1.- Main program synoptic flow chart

matrix which serves as C^{NR} in the equation

$$C^{NB} = C^{NR}C^{RB} \tag{6.1}$$

Each of the next six sets of four Symbol instructions sets up a "Single Instruction Interrupt". A typical set of four instructions is

```
S70     LDA     80S
S       STA     043
S       BRU     90S
S80     MPO     NYNEG
```

Instruction S70 loads the instruction MPO NYNEG to be executed by Interrupt 03 into the accumulator. The instruction MPO means "memory plus one". Therefore NYNEG, the number of $-\Delta\phi_y$ pulses that have occurred, is incremented by one. The next instruction stores the MPO NYNEG instruction in memory location 43. (When Single Instruction Interrupt n occurs it causes the execution of the instruction stored in memory location 40+n.) The instruction

```
S       BRU     90S
```

then causes the program to branch unconditionally to instruction S90. The Symbol instruction NOP (no operation) is the Symbol equivalent of the Fortran CONTINUE statement.

Symbol instruction EOM 030001 (Sig 01) causes a pulse which passes OR Gate 71 (Figure D.7) and advances the Preset Counter 02 (Figure D.9) to an initial count of 1. The need for this is seen in Figure D.8 which shows that Interrupt 12 (which calls Subroutine Increment Phi) does not occur until Preset Counter 02 has been advanced to 1.

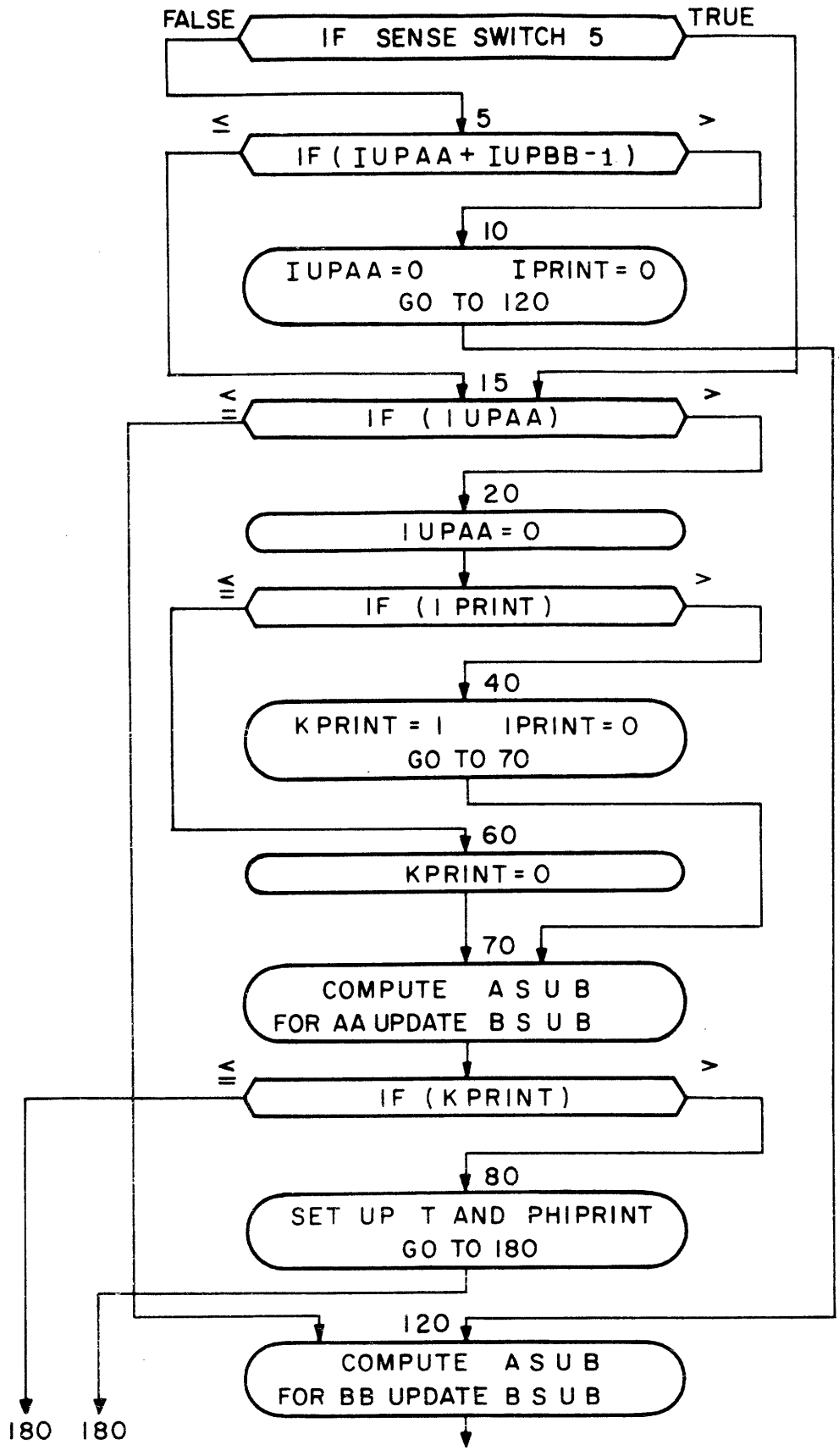


Figure E.2.- Subroutine update C matrix synoptic flow chart

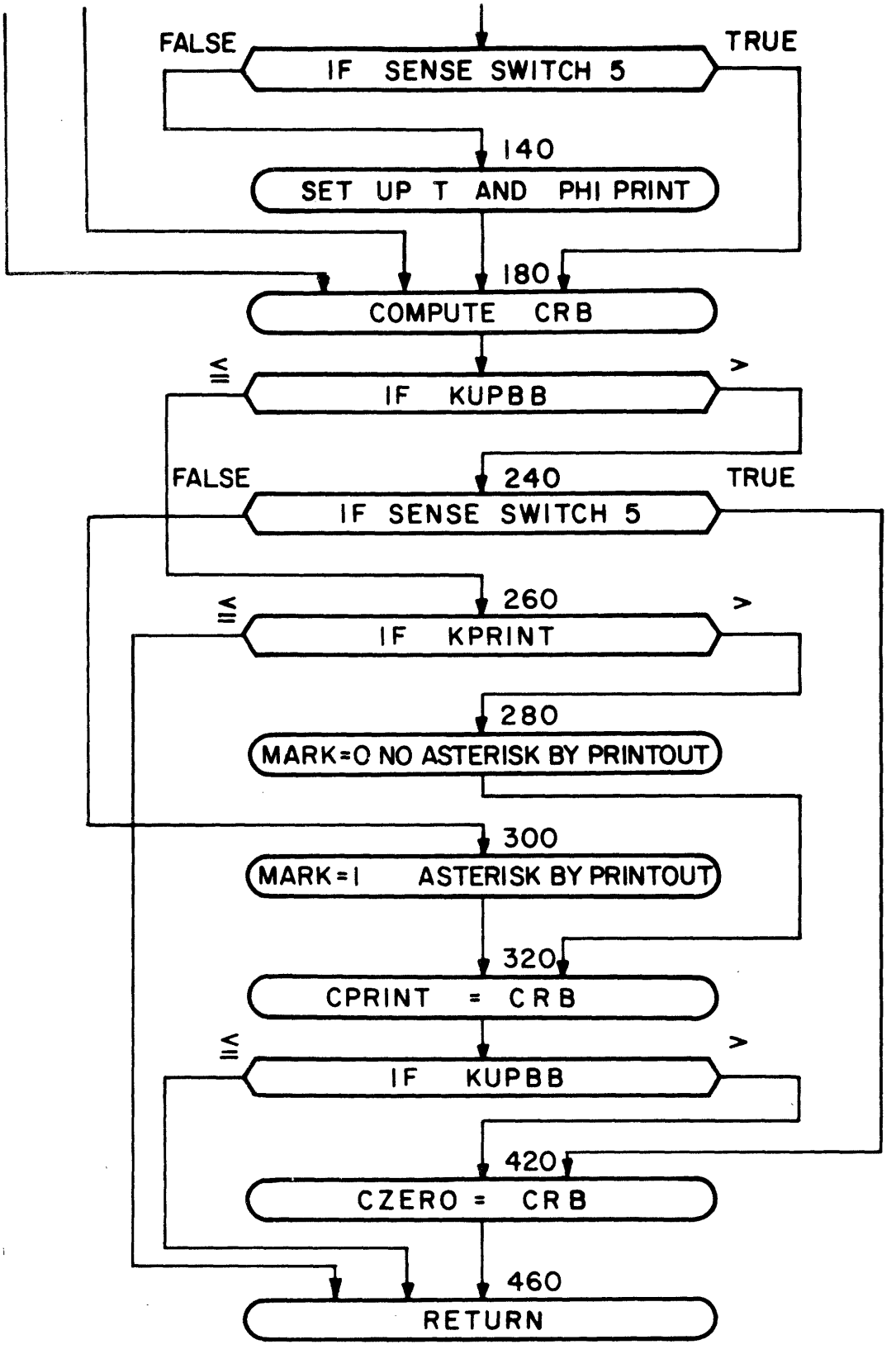


Figure E.2(cont.).- Subroutine update C matrix synoptic flow chart

Symbol instruction EOM 030002 (Sig 02) sets both the Mode Control flip-flop FF00 and the Mode Configuration flip-flop SR00 (Figure D.7). This places both sets of integrators in the initial condition mode thus preventing any integrator from overloading before the run begins.

Subroutine Run Start

The instructions involving Sense Switch 4 form an idle loop that awaits a change in the condition of Sense Switch 4. Symbol instruction EOM 030005 (Sig 05) resets the flip-flop SR30 (Figure D.7). This opens gate 58 which sets flip-flops SR31 and resets the Mode Configuration flip-flop SR00. When SR31 is set, gates 30-35 and 59 allow generation of Interrupts 00-03, 10, and 11, the generation of the pulse which increments Preset Counter 02 (Figure D.9). When SR00 is reset, the integrators are in normal complementary operation.

Subroutine Update C Matrix

A synoptic flow chart for the Update CMatrix Subroutine is shown in Figure E.2. This subroutine evaluates

$$C^{NB} = C^{NR}C^{RB} \tag{6.1}$$

where

C^{RB} is generated as CMAT

C^{NR} is remembered as CZERO

C^{NB} is formed by the operation symbolized as CRB = CZERO * CMAT (C^{NB} is called CRB in this subroutine.)

If a printout is requested either because ϕ was reset to zero or because the time till the next printout has elapsed,

then CRB and the time at which CRB was generated are stored as CPRINT and T until the run is over. Then Subroutine End of Run causes the stored matrices to be printed out. Setting Sense Switch 5 causes the printing of CRB when ϕ is reset to zero to be suppressed.

The flags IUPAA, IUPBB, IPRINT, and IEND are set in Subroutine Increment Phi.

Subroutine Increment Phi

The first set of instructions increments the $\underline{\phi}$ vector and zeros the registers which are incremented by the $\Delta\phi$ interrupts. Then $|\underline{\phi}|$ is tested to see if it exceeds the magnitude which calls for $\underline{\phi}$ to be reset to zero. If this is the case, the timing logic on the Logic Patch Board (Figure D.7) is set up so that $\underline{\phi}$ is reset to zero just prior to the next time that the Increment Phi interrupt is energized. The flag MUPBB is set so that at the next Increment Phi interrupt, $\underline{\phi}$ will be reset to zero in the digital computer as well as in the analog computer. Then the flag IUPBB is set so that Subroutine Update CMatrix updates the CMatrix and establishes a new CZERO matrix.

The TESTFEE cycle occurs every 0.01 second. Statement 140 counts 10 of these cycles before calling for a normal CMatrix update by setting flag IUPAA. Statement 220 counts the number, NUPAA, of normal update cycles and calls for a printout, IPRINT = 1, when NUPAA = NPERPR. The value NPERPR, the number of updates per printout, was set in Subroutine Data In. When the number of printouts, NPRINT, equals the

total number, NOUT, allotted in Subroutine Data In for the run, IEND is set to 1 and this calls for Subroutine End of Run in the Main Program.

Subroutine End of Run

The printout of the stored CMatrices and the stored vectors occurs in this subroutine.

```
C     JACK BORTZ
C     MAY 2, 1969
C     THESIS LISTING -- ISU SYSTEM
C
C
C                    SENSE SWITCH 1       SET TO END RUN
C                    SENSE SWITCH 2       TOGGLE TO CHANGE RUN PARAMETERS
C                    SENSE SWITCH 3       TOGGLE TO CHANGE PHI(O)
C                    SENSE SWITCH 4       TOGGLE TO START RUN
C                    SENSE SWITCH 5       SET TO DELETE FEE RESET PRINTOUT
C
C
C     MAIN PROGRAM
      DIMENSION CZERO(3,3), CRB(3,3), CMAT(3,3), ASUB(3,3), BSUB(3,3)
      DIMENSION PHI(3), PHIAA(3), PHIBB(3), T(200), TEND(200)
      DIMENSION CPRINT(600,3), MARK(200), PHIPRINT(600), PHIO(3)
      COMMON ISS2OLD, ISS2NEW, ISS3OLD, ISS3NEW, CPRNT TIME, PHIO, NXPOS
      COMMON NXNEG, NYPOS, NYNEG, NZPOS, NZNEG, NRUN, FEEMAX, TOTAL TIME
      COMMON CRB, CZERO, CPRINT, DPHI, IEND, INDEX, MARK, FEESQMAX, NOUT
      COMMON FEESAA, FEESBB, PHIAA, PHIBB, IUPAA, IUPBB, NTESTFEE, KUPBB
      COMMON IUPAA, IUPBB, NTIME, IDAY, NWRITE, PHIPRINT, PHI, IPRINT, T
      COMMON NUPAA, ICALLEND, NPRINT, NPERPR, MONTH, MUPBB
      ISS2OLD=2
      ISS3OLD=2
      TYPE 1000
      ACCEPT 1050, IDAY
      TYPE 1070
      ACCEPT 1080, MONTH
      TYPE 1100
      ACCEPT 1150, NRUN
   10 TYPE 1200
      PAUSE
      CALL DATA IN
      CALL COMPUTER SET UP
      CALL RUN START
   20 IF(IUPAA) 40,40,80
C     AN UPDATE INTERVAL HAS ELAPSED. UPDATE THE CMATRIX.
   40 IF(SENSE SWITCH 1) 100,60
C     SET SENSE SWITCH 1 TO END RUN
   60 IF(IUPBB) 20,20,80
C     FEE HAS BEEN RESET. UPDATE CMATRIX AND CZERO.
   80 CALL UPDATE CMATRIX
      IF(IEND) 20,20,100
C     IEND=1 IMPLIES TOTAL RUN TIME HAS ELAPSED.
  100 CALL END OF RUN
      GO TO 10
      CALL REQUEST
C     THE REQUEST PACKAGE IS LOADED INTO MEMORY FOR POSSIBLE LATER USE.
 1000 FORMAT ($ TYPE IN DAY OF MONTH$)
 1050 FORMAT (I2)
 1070 FORMAT ($ TYPE IN MONTH$)
 1080 FORMAT (A4)
 1100 FORMAT ($ TYPE IN NRUN$)
 1150 FORMAT (I4)
 1200 FORMAT ($ READY TO RUN$)
      END

SUBROUTINES REQUIRED
  \TYPE    \READKB    \PAUSE    DATAIN    COMPUTER    RUNSTART    \IFSNSW    UPDATE
  ENDOFRUN  REQUEST    \STOP
```

```
      SUBROUTINE DATA IN
      DIMENSION CZERO(3,3), CRB(3,3), CMAT(3,3), ASUB(3,3), BSUB(3,3)
      DIMENSION PHI(3), PHIAA(3), PHIBB(3), T(200), TEND(200)
      DIMENSION CPRINT(600,3), MARK(200), PHIPRINT(600), PHI0(3)
      COMMON ISS2OLD, ISS2NEW, ISS3OLD, ISS3NEW, CPRNT TIME, PHI0, NXPOS
      COMMON NXNEG, NYPOS, NYNEG, NZPOS, NZNEG, NRUN, FEEMAX, TOTAL TIME
      COMMON CRB, CZERO, CPRINT, DPHI, IEND, INDEX, MARK, FEESQMAX, NOUT
      COMMON FEESAA, FEESBB, PHIAA, PHIBB, TUPAA, TUPBB, NTESTFEE, KUPBB
      COMMON IUPAA, IUPEB, NTIME, IDAY, NWRITE, PHIPRINT, PHI, IPRINT, T
      COMMON NUPAA, ICALLEND, NPRINT, NPERPR, MONTH, MUPBB
      DPHI=0.0005555555555
      INDEX=200
C     INDEX IS THE NUMBER OF CMATRICES THAT CAN BE STORED FOR PRINTING.
      UPDATE TIME = 0.1
      IF (SENSE SWITCH 2) 5,10
C     TOGGLE SENSE SWITCH 2 TO CHANGE RUN PARAMETERS.
    5 ISS2NEW=1
      GO TO 15
   10 ISS2NEW=0
   15 IF(ISS2NEW-ISS2OLD) 20,25,20
   20 ISS2OLD=ISS2NEW
      TYPE 1000
      ACCEPT 1050, TOTAL TIME
      TYPE 1100
      ACCEPT 1050, CPRNT TIME
      TYPE 1150
      ACCEPT 1050, FEEMAX
   25 IF (SENSE SWITCH 3) 30,35
C     TOGGLE SENSE SWITCH 3 TO ENTER NEW PHI(0)
   30 ISS3NEW=1
      GO TO 40
   35 ISS3NEW=0
   40 IF(ISS3NEW-ISS3OLD) 45,50,45
   45 ISS3OLD=ISS3NEW
      TYPE 1200
      ACCEPT 1050, PHI0(1)
      TYPE 1220
      ACCEPT 1050, PHI0(2)
      TYPE 1250
      ACCEPT 1050, PHI0(3)
   50 PRINT 1300
      PRINT 1350, MONTH,IDAY
      PRINT 1400, NRUN
      PRINT 1550, TOTAL TIME
      PRINT 1600, CPRNT TIME
      PRINT 1650, FEEMAX
      FEESQMAX=FEEMAX**2
      NPERPR=CPRNT TIME/UPDATE TIME +0.001
      NOUT=TOTAL TIME/CPRNT TIME
C     NOUT IS THE NUMBER OF PRINTOUTS CALLED FOR BY THE LENGTH OF THE UPDATE
C     INTERVAL ALONE. THERE MAY BE OTHER PRINTOUTS DUE TO FEE BEING RESET.
      RETURN
 1000 FORMAT ($ TYPE IN TOTAL TIME$)
 1050 FORMAT (F9.3)
 1100 FORMAT ($ TYPE IN CPRNT TIME$)
   50 FORMAT ($ TYPE IN FEEMAX$)
 1200 FORMAT ( $ TYPE IN PHI(1)$)
 1220 FORMAT ( $ TYPE IN PHI(2)$)
 1250 FORMAT ( $ TYPE IN PHI(3)$)
 1300 FORMAT ($1JACK BORTZ --- THESIS PROBLEM $///)
 1350 FORMAT (22X, A9, 1X, I2, 6H, 1969//)
```

```
 1400 FORMAT (22X, 7HRUN NO., 2X, I4/)
 1550 FORMAT (22X, 14HTOTAL RUN TIME,33X, F9.3, 8H SECONDS)
 1600 FORMAT (22X, 31HTIME INTERVAL BETWEEN PRINTOUTS,18X, F7.3, 8H SECO
     1NDS)
    0 FORMAT (22X, 5CHFEE IS RESET WHEN ITS MAGNITUDE EQUALS OR EXCEEDS
     1, F6.3, 8H RADIANS////)
      END

SUBROUTINES REQUIRED
  DATAIN    \IFSNSW    \TYPE      \READKB    \PRINT     \FPOWER    \STOP

COMMON STORAGE
  77705 CZERO     77727 CRB       66025 PHI       70334 PHIAA     70326 PHIBB
  65204 T         70665 CPRINT    70351 MARK      66033 PHIPRINT  77764 PHIO
  77777 ISS2OLD   77776 ISS2NEW   77775 ISS3OLD   77774 ISS3NEW   77772 CPRNTT
  77763 NXPOS     77762 NXNEG     77761 NYPOS     77760 NYNEG     77757 NZPOS
  77756 NZNEG     77755 NRUN      77753 FEEMAX    77751 TOTALTIM  70663 DPHI
  70662 IEND      70661 INDEX     70347 FEESQMAX  70346 NOUT      70344 FEESAA
  70342 FEESBB    70324 IUPAA     70322 IUPBB     70321 NTESTFEE  70320 KUPBB
  70317 IUPAA     70316 IUPBB     70315 NTIME     70314 IDAY      70313 NWRITE
  66024 IPRINT    65203 NUPAA     65202 ICALLEND  65201 NPRINT    65200 NPERPR
  65177 MONTH     65176 MUPBB

NON-RECURSIVE STORAGE
  00000 CMAT      00022 ASUB      00044 BSUB      00066 TEND

END OF COMPILATION
```

```
      SUBROUTINE COMPUTER SET UP
      DIMENSION CZERO(3,3), CRB(3,3), CMAT(3,3), ASUB(3,3), BSUB(3,3)
      DIMENSION PHI(3), PHIAA(3), PHIBB(3), T(200), TEND(200)
      DIMENSION CPRINT(600,3), MARK(200), PHIPRINT(600), PHIO(3)
      COMMON ISS2OLD, ISS2NEW, ISS3OLD, ISS3NEW, CPRNT TIME, PHIO, NXPOS
      COMMON NXNEG, NYPOS, NYNEG, NZPOS, NZNEG, NRUN, FEEMAX, TOTAL TIME
      COMMON CRB, CZERO, CPRINT, DPHI, IEND, INDEX, MARK, FEESQMAX, NOUT
      COMMON FEESAA, FEESBB, PHIAA, PHIBB, TUPAA, TUPBB, NTESTFEE, KUPBB
      COMMON IUPAA, IUPBB, NTIME, IDAY, NWRITE, PHIPRINT, PHI, IPRINT, T
      COMMON NUPAA, ICALLEND, NPRINT, NPERPR, MONTH, MUPBB
      CALL CONDITION
C     INITIALIZE THE HYBRID INTERFACE.
      CALL MUX TRACK
C     PUT DVM INTO TRACK MODE.
      CONNECT INCREMENT PHI, 12
C     THE GYRO TIME BASE TRIGGERS THESE TWO INTERRUPTS.
      CALL CONSOLE (1)
C     SELECT CONSOLE NUMBER 1.
S     EOM 030003
C     DISABLE THE CLOCK PULSE.
      PRINT 1000
S     EOM 030004
C     RESET THE DT COUNTER.
      CALL SYSTEM ARM
C     ENABLE ANALOG COMPUTER TO SEND INTERRUPTS.
      CALL STANDBY
      ICALLEND=0
      IEND=0
      IPRINT=1
      IUPAA=1
      IUPBB=0
      KUPBB=0
      MUPBB=0
      NPRINT=0
      NRUN=NRUN+1
      NTESTFEE=0
      NTIME=0
      NUPAA=0
      NWRITE=0
      NXPOS=0
      NXNEG=0
      NYPOS=0
      NYNEG=0
      NZPOS=0
      NZNEG=0
      TUPAA=0.0
      CZERO(1,1)=1.0
      CZERO(1,2)=0.0
      CZERO(1,3)=0.0
      CZERO(2,1)=0.0
      CZERO(2,2)=1.0
      CZERO(2,3)=0.0
      CZERO(3,1)=0.0
      CZERO(3,2)=0.0
      CZERO(3,3)=1.0
      DO 10 I=1,3
   10 PHI(I)=PHIO(I)
      CALL DAL (0, -PHI(1), 1000.0, -PHI(2), 1000.0, -PHI(3), 1000.0)
      PHIAA(1)=PHI(1)
      PHIAA(2)=PHI(2)
      PHIAA(3)=PHI(3)
```

```
      FEESAA=PHI(1)**2 + PHI(2)**2 + PHI(3)**2
      CALL UPDATE CMATRIX
C     THE NEXT 4 INSTRUCTIONS CAUSE INTERRUPT  0 TO ADD 1 TO NXPOS.
S     LDA 20S
S     STA 040
S     BRU 30S
S20   MPO NXPOS
C     THE NEXT 4 INSTRUCTIONS CAUSE INTERRUPT  1 TO ADD 1 TO NXNEG.
S30   LDA 40S
S     STA 041
S     BRU 50S
S40   MPO NXNEG
C     THE NEXT 4 INSTRUCTIONS CAUSE INTERRUPT  2 TO ADD 1 TO NYPOS.
S50   LDA 60S
S     STA 042
S     BRU 70S
S60   MPO NYPOS
C     THE NEXT 4 INSTRUCTIONS CAUSE INTERRUPT  3 TO ADD 1 TO NYNEG.
S70   LDA 80S
S     STA 043
S     BRU 90S
S80   MPO NYNEG
C     THE NEXT 4 INSTRUCTIONS CAUSE INTERRUPT 10 TO ADD 1 TO NZPOS.
S90   LDA 100S
S     STA 052
S     BRU 110S
S100  MPO NZPOS
C     THE NEXT 4 INSTRUCTIONS CAUSE INTERRUPT 11 TO ADD 1 TO NZNEG.
S     LDA 120S
S     STA 053
S     BRU 130S
S120  MPO NZNEG
S130  NOP
S     EOM 030001
C     INITIALIZE PRESET COUNTER TO A COUNT OF 1.
C
S     EOM 030002
C     PLACE ALL INTEGRATORS IN THE INITIAL CONDITION MODE.
      RETURN
 1000 FORMAT (26X, 4HTIME, 40X, 8HC MATRIX, 32X, 3HPHI//)
      END

SUBROUTINES REQUIRED
  COMPUTER  CONDITIO  MUXTRACK  INCREMEN  \CONNECT  CONSOLE   \PRINT    SYSTEM
  STANDBY   DAL       \FPOWER   UPDATECM  \STOP

COMMON STORAGE
  77705 CZERO     77727 CRB       66025 PHI       70334 PHIAA     70326 PHIBB
  65204 T         70665 CPRINT    70351 MARK      66033 PHIPRINT  77764 PHIO
  77777 ISS2OLD   77776 ISS2NEW   77775 ISS3OLD   77774 ISS3NEW   77772 CPRNTT
  77763 NXPOS     77762 NXNEG     77761 NYPOS     77760 NYNEG     77757 NZPOS
  77756 NZNEG     77755 NRUN      77753 FEEMAX    77751 TOTALTIM  70663 DPHI
  70662 IEND      70661 INDEX     70347 FEESQMAX  70346 NOUT      70344 FEESAA
  70342 FEESBB    70324 TUPAA     70322 TUPBB     70321 NTESTFEE  70320 KUPBB
   0317 IUPAA     70316 IUPBB     70315 NTIME     70314 IDAY      70313 NWRITE
  66024 IPRINT    65203 NUPAA     65202 ICALLEND  65201 NPRINT    65200 NPERPR
  65177 MONTH     65176 MUPBB

NON-RECURSIVE STORAGE
  00000 CMAT      00022 ASUB      00044 BSUB      00066 TEND
```

```
      SUBROUTINE RUN START
      DIMENSION CZERO(3,3), CRB(3,3), CMAT(3,3), ASUB(3,3), BSUB(3,3)
      DIMENSION PHI(3), PHIAA(3), PHIBB(3), T(200), TEND(200)
      DIMENSION CPRINT(600,3), MARK(200), PHIPRINT(600), PHI0(3)
      COMMON ISS2OLD, ISS2NEW, ISS3OLD, ISS3NEW, CPRNT TIME, PHI0, NXP0S
      COMMON NXNEG, NYP0S, NYNEG, NZP0S, NZNEG, NRUN, FEEMAX, T0TAL TIME
      COMMON CRB, CZERO, CPRINT, DPHI, IEND, INDEX, MARK, FEESQMAX, N0UT
      COMMON FEESAA, FEESBB, PHIAA, PHIBB, TUPAA, TUPBB, NTESTFEE, KUPBB
      COMMON IUPAA, IUPBB, NTIME, IDAY, NWRITE, PHIPRINT, PHI, IPRINT, T
      COMMON NUPAA, ICALLEND, NPRINT, NPERPR, M0NTH, MUPBB
      IF(SENSE SWITCH 4) 20,40
   20 IF(SENSE SWITCH 4) 20,60
   40 IF(SENSE SWITCH 4) 60,40
C     T0GGLE SENSE SWITCH 4 T0 BEGIN RUN
   60 CALL COMPUTE
      CALL ARM (0, 1, 2, 3, 10, 11, 12)
C     SET DIGITAL COMPUTER T0 ACCEPT SPECIFIC INTERRUPTS.
S     E0M 030005
C     ENABLE THE CLOCK PULSE. RESET PHI M0DE CONTR0L FLIP-FL0P.
      RETURN
      END

SUBROUTINES REQUIRED
  RUNSTART  \IFSNSW   COMPUTE   ARM       \ST0P

COMMON STORAGE
  77705 CZERO     77727 CRB       66025 PHI       70334 PHIAA     70326 PHIBB
  65204 T         70665 CPRINT    70351 MARK      66033 PHIPRINT  77764 PHI0
  77777 ISS2OLD   77776 ISS2NEW   77775 ISS3OLD   77774 ISS3NEW   77772 CPRNTT
  77763 NXP0S     77762 NXNEG     77761 NYP0S     77760 NYNEG     77757 NZP0S
   7756 NZNEG     77755 NRUN      77753 FEEMAX    77751 T0TALTIM  70663 DPHI
  70662 IEND      70661 INDEX     70347 FEESQMAX  70346 N0UT      70344 FEESAA
  70342 FEESBB    70324 TUPAA     70322 TUPBB     70321 NTESTFEE  70320 KUPBB
  70317 IUPAA     70316 IUPBB     70315 NTIME     70314 IDAY      70313 NWRITE
  66024 IPRINT    65203 NUPAA     65202 ICALLEND  65201 NPRINT    65200 NPERPR
  65177 M0NTH     65176 MUPBB

NON-RECURSIVE STORAGE
  00000 CMAT      00022 ASUB      00044 BSUB      00066 TEND

END OF COMPILATION
```

```
      SUBROUTINE UPDATE CMATRIX
      DIMENSION CZERO(3,3), CRB(3,3), CMAT(3,3), ASUB(3,3), BSUB(3,3)
      DIMENSION PHI(3), PHIAA(3), PHIBB(3), T(200), TEND(200)
      DIMENSION CPRINT(600,3), MARK(200), PHIPRINT(600), PHI0(3)
      COMMON ISS2OLD, ISS2NEW, ISS3OLD, ISS3NEW, CPRNT TIME, PHI0, NXPOS
     COMMON NXNEG, NYPOS, NYNEG, NZPOS, NZNEG, NRUN, FEEMAX, TOTAL TIME
     COMMON CRB, CZERO, CPRINT, DPHI, IEND, INDEX, MARK, FEESQMAX, NOUT
     COMMON FEESAA, FEESBB, PHIAA, PHIBB, TUPAA, TUPBB, NTESTFEE, KUPBB
     COMMON IUPAA, IUPBB, NTIME, IDAY, NWRITE, PHIPRINT, PHI, IPRINT, T
     COMMON NUPAA, ICALLEND, NPRINT, NPERPR, MONTH, MUPBB
      IF (SENSE SWITCH 5) 15,5
    5 IF(IUPAA+IUPBB-1) 15,15,10
C     IF BOTH IUPAA AND IUPBB ARE SET, ONLY DO A TYPE BB UPDATE.
   10 IUPAA=0
      IPRINT=0
      GO TO 120
   15 IF(IUPAA) 120,120,20
C     IUPAA=1 INDICATES A C MATRIX UPDATE INTERVAL HAS ELAPSED
   20 IUPAA=0
      IF(IPRINT) 60,60,40
   40 KPRINT=1
C     K-FLAGS ARE SET UP SINCE THE I-FLAGS ARE SUBJECT TO CHANGE IN THE
C     INCREMENT PHI INTERRUPT SUBROUTINE.
      IPRINT=0
      GO TO 70
   60 KPRINT=0
   70 FEE4AA=FEESAA*FEESAA
      FEE6AA=FEE4AA*FEESAA
      P=1.0-FEESAA/6.0+FEE4AA/120.0-FEE6AA/5040.0
      PSQ=P*P
      Q=1.0/(2.0-FEESAA/2.0+FEE4AA/24.0-FEE6AA/720.0)
      PHI2AAX=PHIAA(1)*PHIAA(1)
      PHI2AAY=PHIAA(2)*PHIAA(2)
      PHI2AAZ=PHIAA(3)*PHIAA(3)
      ASUB(1,1)= 1.0
      ASUB(1,2)=-P*PHIAA(3)
      ASUB(1,3)= P*PHIAA(2)
      ASUB(2,1)= P*PHIAA(3)
      ASUB(2,2)= 1.0
      ASUB(2,3)=-P*PHIAA(1)
      ASUB(3,1)=-P*PHIAA(2)
      ASUB(3,2)= P*PHIAA(1)
      ASUB(3,3)= 1.0
      BSUB(1,1)=-Q*PSQ*(PHI2AAZ+PHI2AAY)
      BSUB(1,2)= Q*PSQ*PHIAA(1)*PHIAA(2)
      BSUB(1,3)= Q*PSQ*PHIAA(1)*PHIAA(3)
      BSUB(2,1)= Q*PSQ*PHIAA(2)*PHIAA(1)
      BSUB(2,2)=-Q*PSQ*(PHI2AAX+PHI2AAZ)
      BSUB(2,3)= Q*PSQ*PHIAA(2)*PHIAA(3)
      BSUB(3,1)= Q*PSQ*PHIAA(3)*PHIAA(1)
      BSUB(3,2)= Q*PSQ*PHIAA(3)*PHIAA(2)
      BSUB(3,3)=-Q*PSQ*(PHI2AAX+PHI2AAY)
      IF(KPRINT) 180,180,80
   80 IROW=3*NPRINT
      NPRINT=NPRINT+1
C     NPRINT IS THE TOTAL NUMBER OF PRINTOUT MATRICES CALLED FOR SO FAR.
      T(NPRINT)=TUPAA/3600.0
      DO 100 M=1,3
      L=M+IROW
  100 PHIPRINT(L)=PHIAA(M)
      GO TO 180
```

```
  120 KUPBB=1
      IUPBB=0
      FEE4BB=FEESBB*FEESBB
      FEE6BB=FEE4BB*FEESBB
      P=1.0-FEESBB/6.0+FEE4BB/120.0-FEE6BB/5040.0
      PSQ=P*P
      Q=1.0/(2.0-FEESBB/2.0+FEE4BB/24.0-FEE6BB/720.0)
      PHI2BBX=PHIBB(1)*PHIBB(1)
      PHI2BBY=PHIBB(2)*PHIBB(2)
      PHI2BBZ=PHIBB(3)*PHIBB(3)
      ASUB(1,1)= 1.0
      ASUB(1,2)=-P*PHIBB(3)
      ASUB(1,3)= P*PHIBB(2)
      ASUB(2,1)= P*PHIBB(3)
      ASUB(2,2)= 1.0
      ASUB(2,3)=-P*PHIBB(1)
      ASUB(3,1)=-P*PHIBB(2)
      ASUB(3,2)= P*PHIBB(1)
      ASUB(3,3)= 1.0
      BSUB(1,1)=-Q*PSQ*(PHI2BBZ+PHI2BBY)
      BSUB(1,2)= Q*PSQ*PHIBB(1)*PHIBB(2)
      BSUB(1,3)= Q*PSQ*PHIBB(1)*PHIBB(3)
      BSUB(2,1)= Q*PSQ*PHIBB(2)*PHIBB(1)
      BSUB(2,2)=-Q*PSQ*(PHI2BBX+PHI2BBZ)
      BSUB(2,3)= Q*PSQ*PHIBB(2)*PHIBB(3)
      BSUB(3,1)= Q*PSQ*PHIBB(3)*PHIBB(1)
      BSUB(3,2)= Q*PSQ*PHIBB(3)*PHIBB(2)
      BSUB(3,3)=-Q*PSQ*(PHI2BBX+PHI2BBY)
      IF (SENSE SWITCH 5) 180,140
C     SET SENSE SWITCH 5 TO SUPRESS FEE RESET PRINTOUT.
  140 KPRINT=1
      IR8W=3*NPRINT
      NPRINT=NPRINT+1
      T(NPRINT)=TUPBB/3600.0
      DO 160 M=1,3
      L=M+IR8W
  160 PHIPRINT(L)=PHIBB(M)
  180 DO 200 M=1,3
      DO 200 N=1,3
  200 CMAT(M,N)=ASUB(M,N)+BSUB(M,N)
      DO 220 M=1,3
      DO 220 N=1,3
      CRB(M,N)=CMAT(M,1)*CZERO(1,N)+CMAT(M,2)*CZERO(2,N)+CMAT(M,3)*CZERO
     1(3,N)
  220 CONTINUE
      IF(KUPBB) 260,260,240
  240 IF (SENSE SWITCH 5) 420,300
  260 IF(KPRINT) 460,460,280
  280 MARK(NPRINT)=0
C     THIS CAUSES PRINTOUT NOT TO PRINT AN ASTERISK BY THIS MATRIX
      GO TO 320
  300 MARK(NPRINT)=1
C     PRINT AN ASTERISK BY THE MATRIX PRINTED OUT AT THIS TIME.
C     THIS INDICATES ON PRINTOUT THAT FEE WAS RESET
  320 DO 340 M=1,3
      DO 340 N=1,3
      L=M+IR8W
  340 CPRINT(L,N)=CRB(M,N)
      IF(NPRINT-INDEX) 400,360,360
  360 IF(IEND) 380,380,400
  380 ICALLEND=1
```

```
C     ALL THE CPRINT STORAGE HAS BEEN USED.
      CALL END OF RUN
  400 KPRINT=0
      IF(KUPBB) 460,460,420
  420 DO 440 M=1,3
      DO 440 N=1,3
  440 CZERO(M,N)=CRB(M,N)
C     THE UPDATE WAS CALLED BY RESETTING FEE TO ZERO.
C     CZERO MUST BE BROUGHT UP TO DATE.
      KUPBB=0
  460 RETURN
      END

SUBROUTINES REQUIRED
  UPDATECM  \IFSNSW   ENDOFRUN  \STOP

COMMON STORAGE
  77705 CZERO     77727 CRB       66025 PHI       70334 PHIAA     70326 PHIBB
  65204 T         70665 CPRINT    70351 MARK      66033 PHIPRINT  77764 PHIO
  77777 ISS2OLD   77776 ISS2NEW   77775 ISS3OLD   77774 ISS3NEW   77772 CPRNTT
  77763 NXPOS     77762 NXNEG     77761 NYPOS     77760 NYNEG     77757 NZPOS
  77756 NZNEG     77755 NRUN      77753 FEEMAX    77751 TOTALTIM. 70663 DPHI
  70662 IEND      70661 INDEX     70347 FEESQMAX  70346 NOUT      70344 FEESAA
  70342 FEESBB    70324 TUPAA     70322 TUPBB     70321 NTESTFEE  70320 KUPBB
  70317 IUPAA     70316 IUPBB     70315 NTIME     70314 IDAY      70313 NWRITE
  66024 IPRINT    65203 NUPAA     65202 ICALLEND  65201 NPRINT    65200 NPERPR
  65177 MONTH     65176 MUPBB

NON-RECURSIVE STORAGE
   00000 CMAT     00022 ASUB      00044 BSUB      00066 TEND

END OF COMPILATION
```

```
      SUBROUTINE INCREMENT PHI
      DIMENSION CZERO(3,3), CRB(3,3), CMAT(3,3), ASUB(3,3), BSUB(3,3)
      DIMENSION PHI(3), PHIAA(3), PHIBB(3), T(200), TEND(200)
      DIMENSION CPRINT(600,3), MARK(200), PHIPRINT(600), PHI0(3)
      COMMON ISS2OLD, ISS2NEW, ISS3OLD, ISS3NEW, CPRNT TIME, PHI0, NXPOS
      COMMON NXNEG, NYPOS, NYNEG, NZPOS, NZNEG, NRUN, FEEMAX, TOTAL TIME
      COMMON CRB, CZERO, CPRINT, DPHI, IEND, INDEX, MARK, FEESQMAX, NOUT
      COMMON FEESAA, FEESBB, PHIAA, PHIBB, TUPAA, TUPBB, NTESTFEE, KUPBB
      COMMON IUPAA, IUPBB, NTIME, IDAY, NWRITE, PHIPRINT, PHI, IPRINT, T
      COMMON NUPAA, ICALLEND, NPRINT, NPERPR, MONTH, MUPBB
      DX=NXPOS-NXNEG
      DY=NYPOS-NYNEG
      DZ=NZPOS-NZNEG
      NXPOS=0
      NXNEG=0
      NYPOS=0
      NYNEG=0
      NZPOS=0
      NZNEG=0
      PHI(1)=PHI(1)+DX*DPHI
      PHI(2)=PHI(2)+DY*DPHI
      PHI(3)=PHI(3)+DZ*DPHI
      NTIME=NTIME+36
      NTESTFEE=NTESTFEE+1
      FEESQR=PHI(1)*PHI(1)+PHI(2)*PHI(2)+PHI(3)*PHI(3)
      IF (MUPBB) 100,100,130
  100 IF(FEESQR-FEESQMAX) 140,120,120
  120 MUPBB=1
S     ESM 030000
C     MODE CONTROL FLIP-FLOP IS RESET JUST PRIOR TO NEXT INTERRUPT.
      GO TO 140
   30 IUPBB=1
C     FEE WAS JUST RESET. UPDATE CMATRIX.
      MUPBB=0
      TUPBB=NTIME
      FEESAA=FEESQR
      FEESBB=FEESQR
      PHIAA(1)=PHI(1)
      PHIAA(2)=PHI(2)
      PHIAA(3)=PHI(3)
      PHIBB(1)=PHI(1)
      PHIBB(2)=PHI(2)
      PHIBB(3)=PHI(3)
      PHI(1)=0.0
      PHI(2)=0.0
      PHI(3)=0.0
  140 IF(NTESTFEE-10) 160,180,180
  160 RETURN
  180 NTESTFEE=0
C     UPDATE TIME HAS ELAPSED. UPDATE CMATRIX.
      NUPAA=NUPAA+1
      IUPAA=1
      TUPAA=NTIME
      IF (IUPBB) 200,200,220
  200 PHIAA(1)=PHI(1)
      PHIAA(2)=PHI(2)
      PHIAA(3)=PHI(3)
      FEESAA=FEESQR
  220 IF (NUPAA-NPERPR) 240,260,260
  240 RETURN
  260 NUPAA=0
```

```
C     PRINT TIME HAS ELAPSED. PRINTOUT CMATRIX.
      NWRITE=NWRITE+1
      IPRINT=1
      IF (NWRITE-NOUT) 280,300,300
  280 RETURN
    0 IEND=1
C     TOTAL TIME HAS ELAPSED. END THE RUN.
      RETURN
      END

SUBROUTINES REQUIRED
  INCREMEN  \STOP

COMMON STORAGE
  77705 CZERO     77727 CRB       66025 PHI       70334 PHIAA     70326 PHIBB
  65204 T         70665 CPRINT    70351 MARK      66033 PHIPRINT  77764 PHIO
  77777 ISS2OLD   77776 ISS2NEW   77775 ISS3OLD   77774 ISS3NEW   77772 CPRNTT
  77763 NXPOS     77762 NXNEG     77761 NYPOS     77760 NYNEG     77757 NZPOS
  77756 NZNEG     77755 NRUN      77753 FEEMAX    77751 TOTALTIM  70663 DPHI
  70662 IEND      70661 INDEX     70347 FEESQMAX  70346 NOUT      70344 FEESAA
  70342 FEESBB    70324 TUPAA     70322 TUPBB     70321 NTESTFEE  70320 KUPBB
  70317 IUPAA     70316 IUPBB     70315 NTIME     70314 IDAY      70313 NWRITE
  66024 IPRINT    65203 NUPAA     65202 ICALLEND  65201 NPRINT    65200 NPERPR
  65177 MONTH     65176 MUPBB

NON-RECURSIVE STORAGE
  00000 CMAT      00022 ASUB      00044 BSUB      00066 TEND

END OF COMPILATION
```

```
      SUBROUTINE END OF RUN
      DIMENSION CZERO(3,3), CRB(3,3), CMAT(3,3), ASUB(3,3), BSUB(3,3)
      DIMENSION PHI(3), PHIAA(3), PHIBB(3), T(200), TEND(200)
      DIMENSION CPRINT(600,3), MARK(200), PHIPRINT(600), PHIB(3)
      COMMON ISS2OLD, ISS2NEW, ISS3OLD, ISS3NEW, CPRNT TIME, PHIB, NXPOS
      COMMON NXNEG, NYPOS, NYNEG, NZPOS, NZNEG, NRUN, FEEMAX, TOTAL TIME
      COMMON CRB, CZERO, CPRINT, DPHI, IEND, INDEX, MARK, FEESQMAX, NOUT
      COMMON FEESAA, FEESBB, PHIAA, PHIBB, IUPAA, IUPBB, NTESTFEE, KUPBB
      COMMON IUPAA, IUPEB, NTIME, IDAY, NWRITE, PHIPRINT, PHI, IPRINT, T
      COMMON NUPAA, ICALLEND, NPRINT, NPERPR, MONTH, MUPBB
S     EOM 030003
C     DISABLE THE CLOCK PULSE.
      CALL DISARM (0,1,2,3,10,11,12)
      CALL SYSTEM DISARM
      CALL STANDBY
      ICOUNT=0
      ISKIP=1
      M=0
S     EOM 030002
C     PLACE ALL INTEGRATORS IN THE INITIAL CONDITION MODE.
   20 N=3*M
      L=M
      M=M+1
C     M NUMBERS THE PRINT OUT MATRICES SERIALLY
      K=N+1
      PRINT 1000, CPRINT(K,1), CPRINT(K,2), CPRINT(K,3), PHIPRINT(K)
      K=K+1
      IF(MARK(M)) 60,60,40
   40 PRINT 1050, L, T(M), CPRINT(K,1), CPRINT(K,2), CPRINT(K,3), PHIPRI
     1NT(K)
C     PRINTOUT REQUIRED BECAUSE FEE WAS RESET. PRINT AN ASTERISK.
      GO TO 80
   60 PRINT 1100, L, T(M), CPRINT(K,1), CPRINT(K,2), CPRINT(K,3), PHIPRI
     1NT(K)
   80 K=K+1
      PRINT 1150, CPRINT(K,1), CPRINT(K,2), CPRINT(K,3), PHIPRINT(K)
      IF(M-NPRINT) 85,110,110
C     WHEN M=NPRINT, ALL STORED MATRICES HAVE BEEN PRINTED.
   85 ICOUNT=ICOUNT+1
      IF(ICOUNT-6) 20,90,90
   90 ICOUNT=0
      PRINT 1170
      ISKIP=ISKIP+1
      IF(ISKIP-2) 95,100,100
   95 GO TO 20
  100 ISKIP=0
      GO TO 20
  110 IF(SENSE SWITCH 5) 140,120
  120 PRINT 1200
  140 IF(ICALLEND) 180,180,160
  160 PRINT 1250
  180 IF(ISKIP) 220,220,200
  200 PRINT 1170
  220 RETURN
 1000 FORMAT (52X, 2(F11.8, 5X), F11.8, 10X, F11.8)
 1050 FORMAT (12X, 1H*, 2X, I3, 5X, F9.5, 4H SEC, 16X, 2(F11.8, 5X), F11
     1.8, 10X, F11.8)
 1100 FORMAT (15X,          I3, 5X, F9.5, 4H SEC, 16X, 2(F11.8, 5X), F11
     1.8, 10X, F11.8)
 1150 FORMAT (52X, 2(F11.8, 5X), F11.8, 10X, F11.8///)
 1170 FORMAT (1H1)
```

```
 1200 FORMAT (1H AN ASTERISK (*) MEANS THIS PRINTOUT WAS CALLED BY FEE BE
     1ING RESET TO ZERO $///)
 1250 FORMAT (1H THERE WAS NOT ENOUGH STORAGE ALLOTTED IN CPRINT$///)
      END

SUBROUTINES REQUIRED
  ENDOFRUN  DISARM    SYSTEMDI  STANDBY   \PRINT    \IFSNSW   \STOP

COMMON STORAGE
  77705 CZERO     77727 CRB       66025 PHI       70334 PHIAA     70326 PHIBB
  65204 T         70665 CPRINT    70351 MARK      66033 PHIPRINT  77764 PHIO
  77777 ISS2OLD   77776 ISS2NEW   77775 ISS3OLD   77774 ISS3NEW   77772 CPRNTT
  77763 NXPOS     77762 NXNEG     77761 NYPOS     77760 NYNEG     77757 NZPOS
  77756 NZNEG     77755 NRUN      77753 FEEMAX    77751 TOTALTIM  70663 DPHI
  70662 IEND      70661 INDEX     70347 FEESQMAX  70346 NOUT      70344 FEESAA
  70342 FEESBB    70324 TUPAA     70322 TUPBB     70321 NTESTFEE  70320 KUPBB
  70317 IUPAA     70316 IUPBB     70315 NTIME     70314 IDAY      70313 NWRITE
  66024 IPRINT    65203 NUPAA     65202 ICALLEND  65201 NPRINT    65200 NPERPR
  65177 MONTH     65176 MUPBB

NON-RECURSIVE STORAGE
  00000 CMAT      00022 ASUB      00044 BSUB      00066 TEND

END OF COMPILATION
```

APPENDIX F

NOTATION CONVENTIONS

F.1 Matrices

Matrices are represented by a capital Greek or Roman letters. In particular

C = direction cosine matrix

Coordinate transformation matrices bear a pair of capital letter superscripts: the first indicating the coordinate frame of the transformed vector, the second indicating the coordinate frame of the vector to be transformed. For example

C^{MN} = the direction cosine transformation from the N frame to the M frame

(The coordinate frame indices always occur in pairs on coordinate transformation operators.) Other matrix conventions are

A^T = the transpose of A

$(T^{PQ})^{-1}$ = the inverse of T^{PQ}

F.2 Vectors

Vectors are designated by a subwritten bar. For example:

$$\underline{a} = \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix}$$

As in the case of matrices,

$\underline{a}^T$ = the transpose of $\underline{a}$

The coordinate frame in which the components of a vector are expressed is indicated by a capital letter superscript. For example:

$\underline{r}^S$ = the vector $\underline{r}$ coordinatized in the S frame

A component of a vector is indicated by a lower case subscript. For example:

v_y^I = the y component in the I frame coordinate system of $\underline{V}$

The special symbol $\underline{1}$ is used for denoting a unit vector in the direction of a coordinate frame coordinate axis. For example:

$\underline{1}_x^R$ = a unit vector in the direction of the R frame x axis.

F.3 The Cross Product Operator

The notation

$$A = [\underline{a}\ x]$$

is used to denote the matrix equivalent of taking the cross product of the vector $\underline{a}$ into another vector. For example:

$$A\ \underline{b} = [\underline{a}\ x]\ \underline{b} \approx \underline{a} \times \underline{b}$$

If the components of $\underline{a}$ (necessarily in the same coordinate frame as the components of $\underline{b}$) are a_x, a_y, and a_z, then

$$A = [\underline{a}\ x] = \begin{bmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{bmatrix}$$

F.4 Coordinate Frames and the Time Derivative

From time to time it is necessary to indicate both the coordinate frame with respect to which the time derivative of a vector is taken and the coordinate frame in which the components of the time derivative are expressed. The notation

$$\frac{d\underline{v}}{dt_Q}$$

indicates that the derivative of the vector $\underline{v}$ is taken with respect to the Q coordinate frame. The notation

$$\frac{d^P\underline{v}}{dt}$$

indicates that the components of the derivative are given with respect to the P coordinate frame. For example

$$\frac{d^P\underline{v}}{dt_Q} = \frac{d^P\underline{v}}{dt_P} - \underline{\omega}^P_{PQ} \times \underline{v}^P$$

is the Law of Coriolis applied to the vector $\underline{v}$ where all vectors are coordinatized in the P frame, the derivative on the left-hand side is taken with respect to the Q frame, and the derivative on the right-hand side is taken with respect to the P frame.

BIOGRAPHY

John E. Bortz was born March 12, 1935 in Reading, Pennsylvania. He attended high school in Petersburg, Virginia graduating from Petersburg High School in 1953. Mr. Bortz then attended the United States Naval Academy at Annapolis, Maryland and graduated with distinction in 1957.

Upon graduation, he accepted a commission in the United States Air Force and served for four years as a navigator in the Strategic Air Command. In 1961, Mr. Bortz left active duty with the Air Force to work as an electronics engineer at Bell Aerosystems Company in Buffalo. He retains his Air Force affiliation as a navigator in the 731st Military Airlift Squadron (Reserve) at L. G. Hanscom Field at Bedford, Massachusetts. During his two years with Bell, Mr. Bortz completed the formal course work at the University of Buffalo towards a master's degree in electrical engineering.

In 1963, he joined the staff of the M.I.T. Instrumentation Laboratory. In 1964, he accepted a research assistantship in the Department of Aeronautics and Astronautics and received the S.M. degree in Instrumentation in June 1965. Following a year of study at M.I.T. under a National Science Foundation fellowship, Mr. Bortz joined the NASA Electronics Research Center in Cambridge in 1966. There he accomplished his thesis research under the sponsorship of the Guidance Laboratory.

Mr. Bortz resides in Acton, Massachusetts with his wife Elizabeth, their four daughters, Sandra, Melanie, Tamara, and Christina, and their son, John Jr.

REFERENCES

1. Wrigley, Walter; Hollister, Walter M.; and Denhard, William S.: Gyroscopic Theory, Design, and Instrumentation. MIT Press, 1969.

2. Halfman, Robert L.: Dynamics. Vol. I, Addison-Wesley Pub. Co., 1962.

3. Miller, John E.: Space Navigation Guidance and Control, Technivision Limited, Maidenhead, England, 1966.

4. Broxmeyer, Charles: Inertial Navigation Systems. McGraw-Hill Book Co., Inc., 1964.

5. McKern, Richard A. : A Study of Transformation Algorithms for Use in a Digital Computer. MIT/IL Report No. T-493, Massachusetts Institute of Technology, 1969.

6. Wierner, Thomas F.: Theoretical Analysis of Gimballess Inertial Reference Equipment Using Delta-Modulated Instruments. T-300, MIT Instrumentation Laboratory, Massachusetts Institute of Technology, 1962.

7. Goldstein, Herbert: Classical Mechanics. Addison-Wesley Publishing Co., Inc., 1951.

8. Jackson, Albert S.: Analog Computation. McGraw-Hill Book Co., Inc., 1960.

9. Bekey, George A.; and Karplus, Walter J.: Hybrid Computation. John Wiley and Sons, Inc., 1968.

10. Kuo, Benjamin C.: Analysis and Synthesis of Sampled-Data Control Systems. Prentice-Hall, Inc., 1963.

11. Hildebrand, Francis B.: Methods of Applied Mathematics. Prentice-Hall, Inc., 1952.

12. Laning, J. Halcombe, Jr.,: The Vector Analysis of Finite Rotations and Angles. MIT/IL Special Report 6398-S-3, Massachusetts Institute of Technology, 1949.

13. Goodman, L. E.; and Robinson, A.R.: Effects of Finite Rotations on Gyroscope Sensing Devices. J. Appl. Mech., vol. 25, June, 1958.

14. Hildebrand, Francis B.: Advanced Calculus for Applications. Prentice-Hall Inc., 1962.

15. Ball, William F.: Strapped-Down Inertial Guidance, The Coordinate Transformation Problem, NOTS Tech. Pub. 4267, Weapons Development Department, Naval Ordinance Test Station, China Lake, California, April 1967.

16. IBM, V/STOL Inertial Navigation with Radar Update Capability, IBM No. 69-NC7-027, International Business Machines Corporation. Gaithersburg, Md., 1969.

17. Laning, J. Halcombe, Jr.: amd Battin, Richard J.: Random Processes in Automatic Control. McGraw-Hill Book Co., Inc. 1956.

18. Jordan, John W.: Direction Cosine Computational Error. NASA TR R-304, 1969.

19. Giles, James N.: Fairchild Semiconductor Linear Integrated Circuits Applications Handbook. Fairchild Semiconductor, 1967.

20. Burr-Brown Research Corp.: Handbook of Operational Amplifier Active RC Networks. Burr-Brown Research Corp., 1966.

21. Kuo, Shan S.: Numerical Methods and Computers. Addison-Wesley Publishing Co., 1965.

22. Koenke, Edmund J.; and Dawning, David R.: Evaluating Strapdown Algorithms: A Unified Approach. Fourth Inertial Guidances Test Symposium Proceedings, vol. I, MDC TR 68-76, Air Force Missile Development Center, Holloman AFB, 1968.

23. Beckman Instruments Inc.: 2200 Series Operators Manual. Beckman Instruments,Inc.

24. Beckman Instruments, Inc.: Beckman 2200/SDS 9300 Hybrid Operating Systems Manual. Beckman Instruments, Inc.

25. Scientific Data Systems: SDS 9300 Computer Reference Manual. Scientific Data Systems, 1967.

26. Bumstead, Robert M.; and Vander-Velde, Wallace E.: Navigation and Guidance Systems Employing a Gimballess IMU. R-412A, MIT Instrumentation Laboratory, 1963.