PROGRAMMED COMPUTER CONTROL FOR LASER TRACKING

16 March 1967 - 22 April 1968

J.J. O'Donnell
J. Glass
S. Gajwani

Final Report
Contract No.: NAS8-21089

May 1968

Prepared for

National Aeronautics and Space Administration
George C. Marshall Space Flight Center
Huntsville, Alabama 35812

Applied Research Laboratory
Sylvania Electronic Systems
Sylvania Electric Products Inc.
40 Sylvan Road, Waltham, Massachusetts 02154

PROGRAMMED COMPUTER CONTROL FOR LASER TRACKING

16 March 1967 - 22 April 1968

J.J. O'Donnell
J. Glass
S. Gajwani

Final Report
Contract No.: NAS8-21089

May 1968

Prepared for

National Aeronautics and Space Administration
George C. Marshall Space Flight Center
Huntsville, Alabama 35812

Applied Research Laboratory
Sylvania Electronic Systems
Sylvania Electric Products Inc.
40 Sylvan Road, Waltham, Massachusetts 02154

ABSTRACT

This final report documents the design, implementation, and testing of a digital computer program, and the specification of the necessary interface hardware for the control of an early-launch, laser tracking system. The control computer is to accept angular pointing error information from a star tracker tube, phase error information from a laser ranging system, and mode control from a reacquisition system. During the tracking mode the computer will provide drive signals for the tracking telescope in order to null angular pointing errors, and will provide phase shift commands to the digital oscillator in the ranging system in order to null the phase and, hence, range error. In the reacquisition mode the control computer will output commands to the telescope drive and the range system oscillator, based on extrapolated target motion, so as to minimize initial errors when the track mode is reentered after target reacquisition. In addition to its control functions, the program computes target position, velocity, and acceleration in a launch-site-centered coordinate system. Provision is also made for interfacing with printing, plotting, and magnetic tape units in order to display and store the tracking information.

PRECEDING PAGE BLANK NOT FILMED.

TABLE OF CONTENTS

Section Page

TABLE OF CONTENTS (cont)

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS (cont)

SECTION 1

INTRODUCTION AND SUMMARY

This final report documents the design, implementation, and testing of a digital computer program, and the specification of the necessary interface hardware for the control of an early-launch, laser tracking system. The control computer is to accept angular pointing error information from a star tracker tube, phase error information from a laser ranging system, and mode control from a reacquisition system. During the tracking mode the computer will provide drive signals for the tracking telescope in order to null angular pointing errors, and will provide phase shift commands to the digital oscillator in the ranging system in order to null the phase and, hence, range error. In the reacquisition mode the control computer will output commands to the telescope drive and the range system oscillator, based on extrapolated target motion, so as to minimize initial errors when the track mode is reentered after target reacquisition. In addition to its control functions, the program computes target position, velocity, and acceleration in a launch-site-centered coordinate system. Provision is also made for interfacing with printing, plotting, and magnetic tape units in order to display and store the tracking information.

The main body of the report is organized, as nearly as possible, according to what is considered a typical procedure for the design of a digital computer-based, real-time, control system. In detail, the following are considered to be the major distinguishable stages of a thorough design process.

1. An input-output performance-oriented specification leading to a functional design in engineering terms (e.g., the setting of servo loop bandwidths to achieve desirable position and velocity tracking errors).

2. The translation of the design into an algorithmic form (or the devising of an algorithm to realize the desired operations) suitable to implementation in a digital system.

3. Coding of the algorithm in a programming language.

4. Simulation and test of the individual subsystems in order to assess performance and to effect any necessary modifications.

5. Integration of all subsystems into the overall system which is then computer tested to assure satisfactory data transfer and real-time performance.

In the final program there are four major subroutines, for each of which the above five stages are described in Sections 2 through 4, with supplementary appendices. Stages 1 and 2 are covered in Section 2. Stages 3 and 4 are described in Section 3; and stage 5 is the subject of Section 4. Descriptive listings of all subroutines will be found in Appendix A.

SECTION 2

SYSTEM DESIGN

2.1 PHILOSOPHY OF SYSTEM OPERATION

The philosophy to be dispensed in this section will be limited to the way in which the basic angular control system will function in the three allowable system modes. These three modes relate to the acquisition, tracking, and reacquisition functions of the overall system. It is best to begin with a discussion of the track mode, which is the steady-state mode for the system program.

A functional block diagram is presented in Figure 2-1. The salient feature of this diagram is the way in which the error signals are obtained for the mount angle servo. Because of the presence of the target dynamics estimator it might be thought desirable to interpose the estimator between the star tracker output (the raw angular tracking error data) and the position error input point of the angle servo, in order to smooth the noise on the star tracker output signal. However, the time constant of the estimator would introduce an undesirable lag. In the absence of any filtering by the estimator, the proportional plus integral compensation block in Figure 2-2 must include any noise filtering which has to be done. Of course this noise filtering decreases the bandwidth of the position loop, usually below a value considered satisfactory. The classical tradeoff between noise attenuation and a wide bandwidth tracking loop can be avoided by using the estimator to provide a velocity feed-forward signal. By this means good dynamic tracking performance can be obtained while position error measurement noise is substantially reduced.

When the return signal from the target is lost, a signal from the reacquisition hardware will cause the tracking system to enter the reacquire mode. The basic feature of operation in this mode is the provision of all command signals to the angle servos by the target dynamics extrapolator. The philosophy here is that loss of signal does not indicate a sudden change in target motion, causing a loss of track, but only an obstruction to transmission of the laser

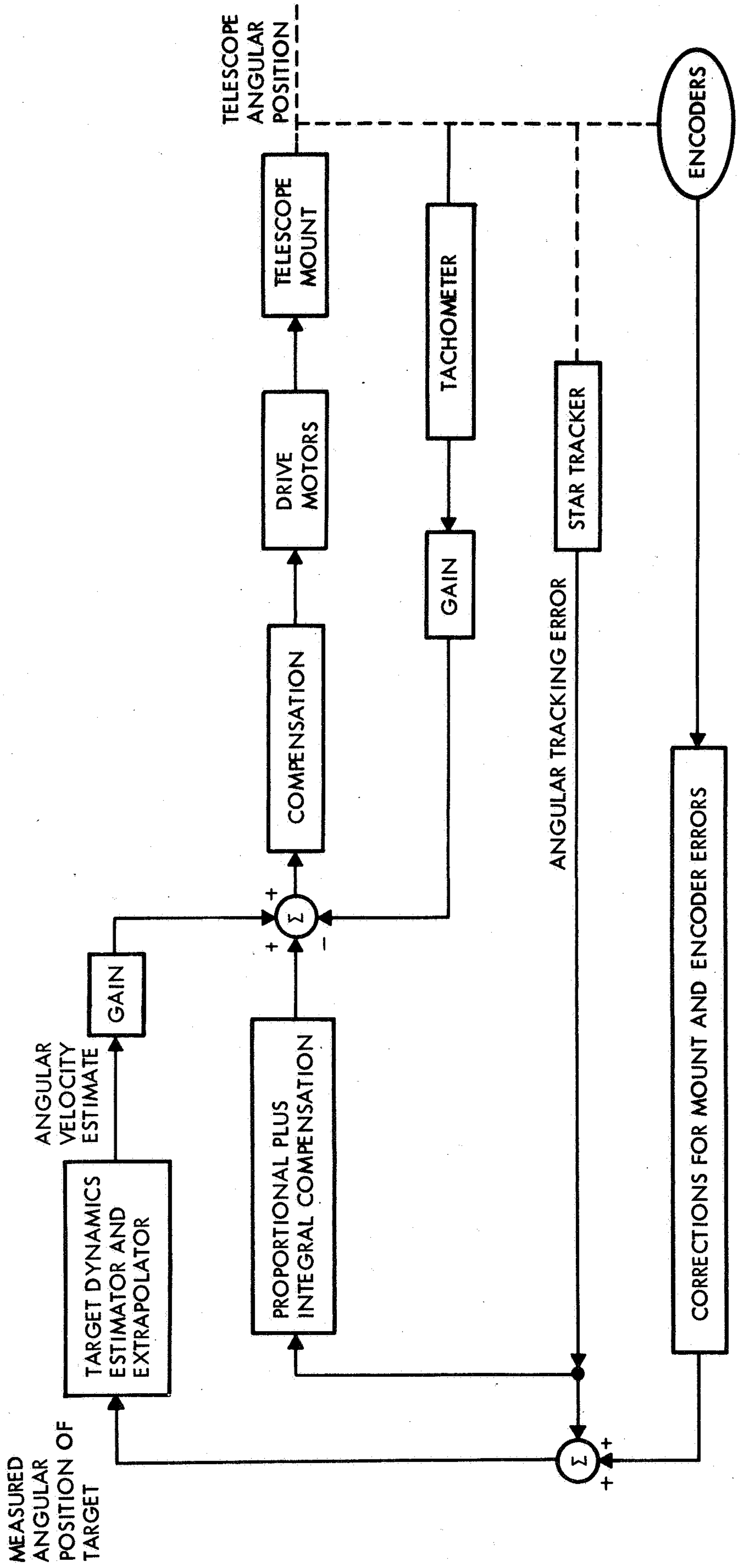

MEASURED ANGULAR POSITION OF TARGET
TARGET DYNAMICS ESTIMATOR AND EXTRAPOLATOR
ANGULAR VELOCITY ESTIMATE
GAIN
PROPORTIONAL PLUS INTEGRAL COMPENSATION
Σ
+
+
−
COMPENSATION
DRIVE MOTORS
TELESCOPE MOUNT
TELESCOPE ANGULAR POSITION
TACHOMETER
GAIN
STAR TRACKER
ANGULAR TRACKING ERROR
ENCODERS
CORRECTIONS FOR MOUNT AND ENCODER ERRORS
Σ
+
+

Figure 2-1. Angular Control System Configuration in Track Mode

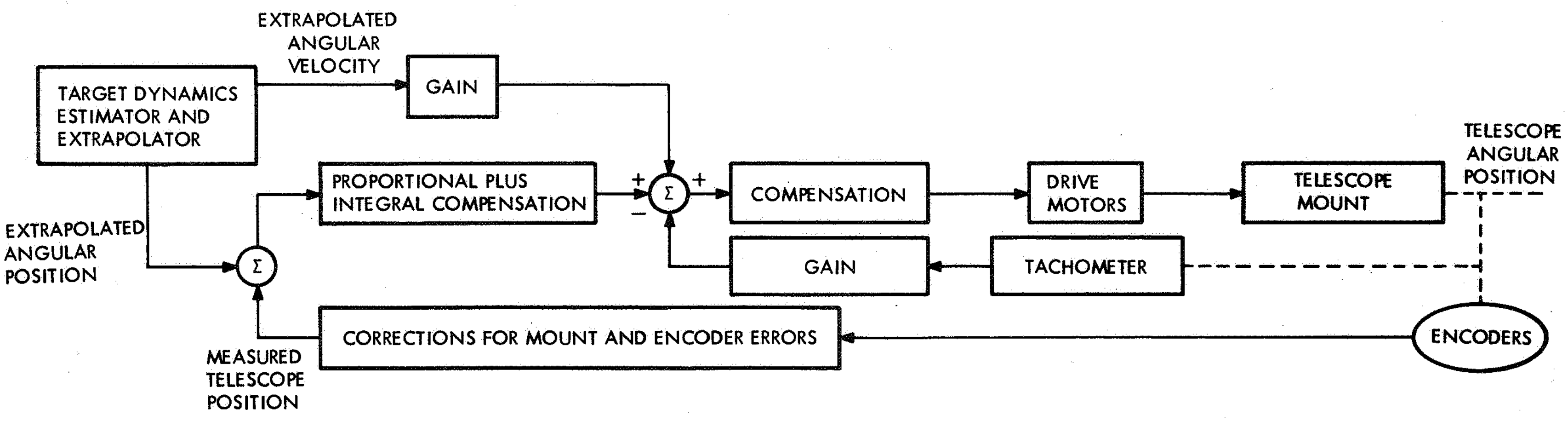

TARGET DYNAMICS ESTIMATOR AND EXTRAPOLATOR
EXTRAPOLATED ANGULAR VELOCITY
GAIN
EXTRAPOLATED ANGULAR POSITION
Σ
PROPORTIONAL PLUS INTEGRAL COMPENSATION
+
−
Σ
+
+
COMPENSATION
DRIVE MOTORS
TELESCOPE MOUNT
TELESCOPE ANGULAR POSITION
GAIN
TACHOMETER
CORRECTIONS FOR MOUNT AND ENCODER ERRORS
MEASURED TELESCOPE POSITION
ENCODERS

Figure 2-2. Angular Control System Configuration in Reacquire Mode

beam (typically a cloud or vapor). Hence while the reacquisition system is searching for a return signal, the telescope will be kept pointed at the expected target position as provided by the extrapolator. Provided the loss of signal condition does not prevail for too long a time period (less than 1.5 seconds, say), the angular tracking error may not be greater than 25 to 30 arc seconds, depending upon the target dynamics at the time of loss of signal.

Finally, it is necessary to mention system operation in the acquire mode. Reference to Figure 2-3 shows that basically the telescope mount is following a programmed target position--which could be a single position, as in calibration tests and initial launch vehicle acquisition, or a sequence of positions, as in acquiring a satellite. The reason for leaving the estimator in is to minimize the velocity and acceleration estimation transient following switching to the track mode.

2.2 DESIGN OF SERVOS FOR CONTROL OF MOUNT ANGULAR MOTION

The hour angle and declination axes of the tracking telescope are individually controlled by identical servos. Because these high-performance servos are implemented in a digital computer, much flexibility in design is available. In particular it now becomes feasible to implement a procedure for compensating for servo saturation during large signal transients. The existence of the target dynamics estimator allows the use of velocity feedforward, permitting, in turn, the smoothing of position error signals without loss of dynamic performance. In other respects the design is straightforward.

The full design details will be found in Appendix B.

2.3 DESIGN OF RANGING SERVO LOOP

The range hardware external to the computer provides a three-channel output for phase error signals and requires, as input, a phase-shifting command signal to control the digital local oscillator. It is the function of the software portion of the range servo loop to process the phase error signals both to close the loop via the input command to the digital oscillator,

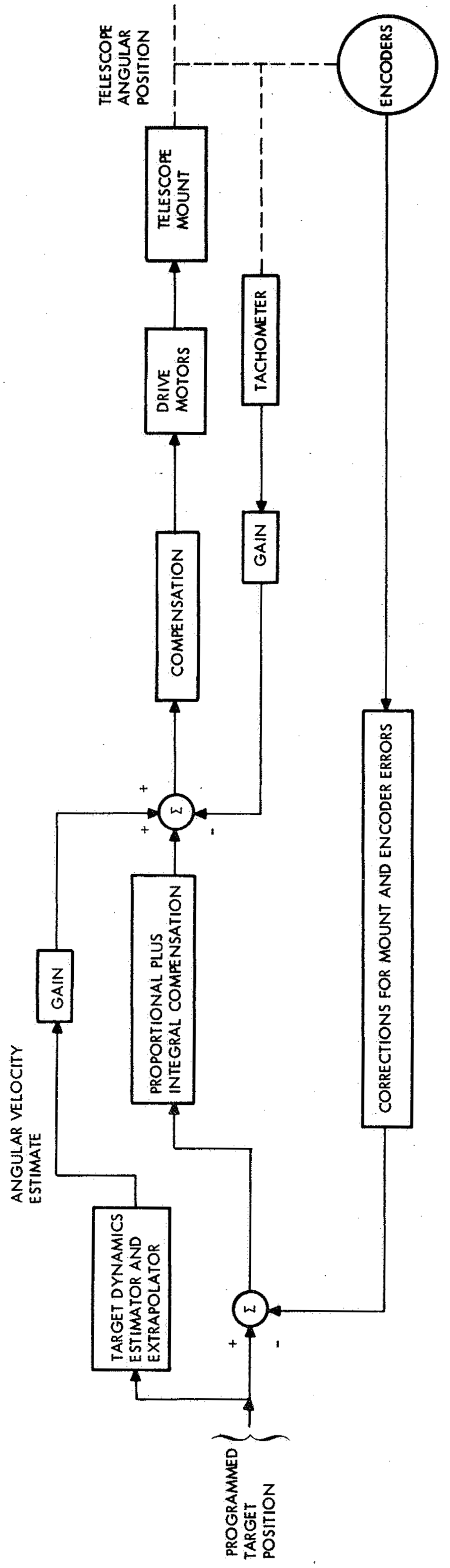

PROGRAMMED TARGET POSITION
TARGET DYNAMICS ESTIMATOR AND EXTRAPOLATOR
ANGULAR VELOCITY ESTIMATE
GAIN
+
-
Σ
PROPORTIONAL PLUS INTEGRAL COMPENSATION
CORRECTIONS FOR MOUNT AND ENCODER ERRORS
+
+
-
Σ
COMPENSATION
GAIN
DRIVE MOTORS
TACHOMETER
TELESCOPE MOUNT
TELESCOPE ANGULAR POSITION
ENCODERS

Figure 2-3. Angular Control System Configuration in Acquire Mode

and to accumulate phase error inputs so as to provide the instantaneous range value. Hence, the design problems are those of a standard servo (compensation to assure stability and reasonable tracking errors) with the exception of choosing a word size for the oscillator phase-shift signal.

As detailed in Appendix C, proportional plus integral compensation was chosen to eliminate velocity error and limit the acceleration error to less than one centimeter at a maximum range acceleration of 10 m/sec^2. Choosing the feedback word (phase shift command to digital oscillator) to contain 10 bits (plus sign) keeps the range error to less than 0.5 cm for a maximum range rate of 360 m/sec. An internal repetition rate of 256/sec is sufficient to implement the design.

2.4 PROCEDURES FOR COORDINATE CONVERSION AND REFRACTION CORRECTION

A detailed description of the coordinate systems and coordinate conversion equations used is given in Appendix D. The refraction correction equations are given in paragraph 3.5. A brief verbal description of the procedures will be given here.

Initial target position information is obtained in an equatorial coordinate system. The first step is to convert to coordinates relative to a rectangular topocentric system, at the tracker site. The standard Apollo topocentric system is used with axes directed to the south, to the east, and along the local vertical. At this point a second, but partial, conversion is performed, solely to allow refraction corrections to be made; the elevation angle (above the local horizon) is computed from the rectangular topocentric coordinates. This elevation angle is corrected for atmospheric refraction, and the rectangular topocentric coordinates are then corrected through a back conversion. The next coordinate system change is from rectangular topocentric at the tracker to rectangular topocentric at the launch site. Finally, the rectangular topocentric coordinates at the launch site are converted to spherical coordinates.

It should be apparent that the final data accuracy (for target position in spherical topocentric coordinates at the launch site) depends critically upon the accuracy of the transformation from the tracker site to the launch site. The quantities influencing this accuracy are five in number: they are the absolute geodetic latitude of the launch site, the height of the origin of coordinates above the reference ellipsoid for both tracker and launch sites, and the differences in both geodetic latitude and longitude between the two sites.

2.5 PRINCIPLES OF OPERATION OF TARGET DYNAMICS ESTIMATOR AND EXTRAPOLATOR

A detailed explanation of the theory of the estimator algorithm is given in Appendix E. In this section a nonmathematical and heuristic discussion is given, in an attempt to set forth the basic operating principles. Before proceeding to that discussion, a few remarks should be made about other estimators which were studied and the reasons why they were found unsuitable. The most general linear least-squares estimator is the Kalman-Bucy filter which requires a full description of the target dynamics and expected noise. For a Saturn V launch vehicle the equations of motion are far too complex to allow a real-time estimator to be realized on a medium-scale computer. In addition, the straight Kalman-Bucy filter has a rather long time constant (due, mainly, to the assumed completeness of the signal and noise dynamical models) which would not allow accurate enough tracking during a loss-of-engine incident. A short-time-constant, or fast responding, estimator must be designed if good tracking data is to be realized following loss of one of the five first-stage engines. Accordingly, a modified Kalman-Bucy filter was designed and implemented; the modification was achieved by recycling the error covariance matrix periodically in order to increase the apparent gain of the estimator and so allow any change in the data trend to influence the estimator output. Unfortunately the discontinuity in the velocity signal at the instant of recycling resulted in too high a noise level to provide a useful velocity feed-forward signal for the angular control system.

To achieve fast response and to reduce periodic transients in the estimator output, it was decided to perform a polynomial fit to short data spans (nominally, 90 data points with a 1/64 second spacing for a total period of 1.4 seconds) and to overlap these fitting polynomials. In choosing the degree of the fitting polynomials, consideration must be given to the nature of the acceleration estimate. Thus a quadratic would yield a piecewise constant acceleration, any change occurring only when the output is derived from the next-in-sequence polynomial. From a consideration of the nominal tracking data (refer to paragraph 3.2) a quadratic is eliminated so that a cubic polynomial must be used. The question still remains as to how many overlapping polynomials should be used; two constitue the minimum number with no clear limit to the maximum. An answer to this question must await an elucidation of the intended purpose of each of the overlapping polynomials.

To fit a polynomial to a large number of measured data points is but one aspect of least-squares parameter estimation. This parameter estimation may be accomplished in two ways in the time domain: by waiting until all measurements have been made and then estimating the parameters via the classical Gauss-Markov formula; or by updating the parameter estimate as each new measurement is obtained, using the more modern recursive equivalent of the Gauss-Markov formula. With obvious justification, the former procedure will henceforth be labelled batch processing, and the latter procedure either real-time or sequential estimation. To aid further discussion an additional piece of terminology must be introduced: because of the use of overlapping fitting polynomials, the complete estimator will actually consist of as many subunits as there are polynomials, each subunit being devoted to estimating the parameters, or coefficients, of one of the polynomials. Because the overall estimator is to operate in real time, it is necessary for at least one of the subunits to be performing sequential estimation; the remainder of the subunits can be doing batch processing. It is at this point that a decision can be made on the number of subunits or polynomials which must be used. The key question to be answered is how good must the estimate of the polynomial coefficients be before initiation of the sequential estimation phase? The reason for asking this question is that the current estimates of position, velocity, and acceleration will be derived from that subunit performing sequential estimation. Hence it is necessary that a reasonable number of data points will have been batch processed by the subunit before the beginning of the sequential estimation phase. With only two polynomials, or subunits, sequential estimation begins after half the data span has been batch processed; experience has indicated that the resulting sequential estimation produces large errors until more data points have been incorporated. Consequently, three or more polynomials must be used.

The final estimator structure is composed of three subunits; the use of four subunits was excluded as requiring too much computation time. The choice of 90 samples (with 1/64 second spacing) results from the necessity that the data span be divisible into three equal segments; 60 samples proved inadequate from an estimation error standpoint, and 120 samples would result in a time constant approaching 1.0 second--also deemed undesirable because of the results of the following analysis.

The time constant of the estimator will be taken as one half of the time duration of a subunit's processing cycle: for the 120 sample cycle the time constant would become 60/64 = 0.938 second; for the 90 sample cycle it would become 45/64 = 0.703 second. Using the nominal tracking data (see paragraph 3.2) the peak accelerations for range, hour angle, and declination angle are, respectively, 10^4 cm/sec^2, 457 arc sec/sec^2, and 230 arc sec/sec^2. If one of the five first-stage engines were to be shut down, there would be a nominal 20 percent decrease in acceleration. As a worst case situation, it will be assumed that this change in acceleration does not affect the estimator output until one time constant after its occurrence. In this case the use of the 120-sample cycle would result in net errors of 435 cm, 39.8 arc seconds, and 20 arc seconds, respectively, for range, hour angle, and declination.

Using the 90 sample cycle gives range, hour angle, and declination errors of 247 cm, 22.5 arc seconds, and 11.3 arc seconds, respectively. Thus use of 120 samples instead of 90 results in an almost twofold increase in tracking error in the event of engine malfunction, without a commensurate decrease in estimation error.

A timing diagram is shown in Figure 2-4; it is not intended to be indicative of the actual estimator program structure, only of its operating principles. In the actual implementation of the triply overlapping polynomial fit, only one set of estimator equations is programmed; this set of equations is time-shared by the three subunits to update the three parameter sets (see paragraph 3.6).

A detailed presentation of the mathematics underlying the estimator design is given in Appendix E.

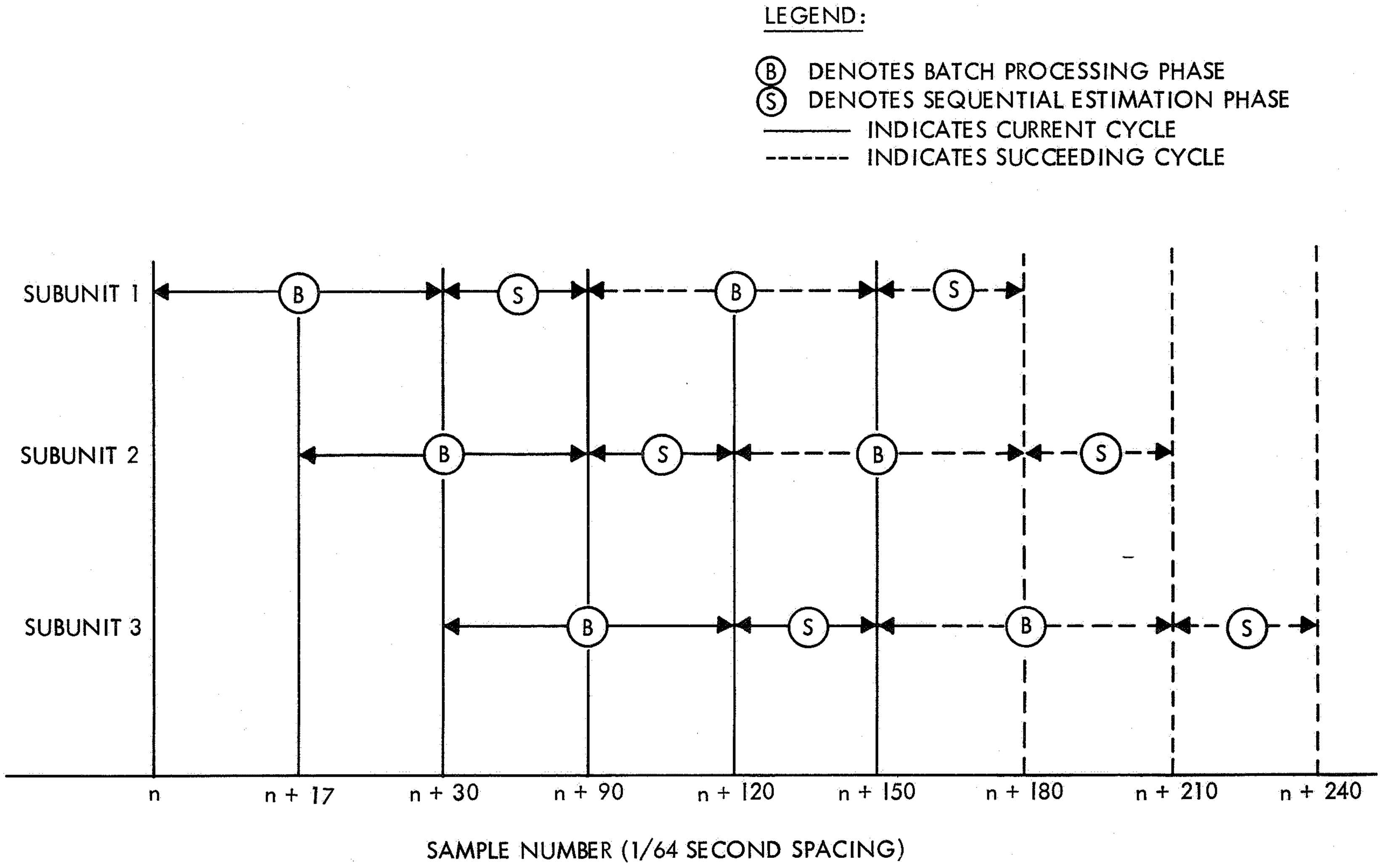

LEGEND:
Ⓑ DENOTES BATCH PROCESSING PHASE
Ⓢ DENOTES SEQUENTIAL ESTIMATION PHASE
——— INDICATES CURRENT CYCLE
------- INDICATES SUCCEEDING CYCLE
SUBUNIT 1
SUBUNIT 2
SUBUNIT 3
B
S
n
n + 17
n + 30
n + 90
n + 120
n + 150
n + 180
n + 210
n + 240
SAMPLE NUMBER (1/64 SECOND SPACING)

Figure 2-4. Operation Cycles of Estimator Subunits

2.6 SPECIFICATION OF SYSTEM HARDWARE

In the initial stages of this contract it was thought that a small-scale digital processor using fixed point arithmetic and a 16- or 18-bit word length would be adequate for handling the computational load. However, after the coordinate transformations were developed it became apparent that floating point arithmetic was extremely desirable both to avoid the awkward scaling problems and the use of double precision arithmetic in certain calculations. Accordingly, attention focussed on 32- or 36-bit machines that were supplied with floating point hardware.

Both the SIGMA 5 (Scientific Data Systems) and the PDP-10 (Digital Equipment Corporation) are considered adequate to handle the real-time computational and formatting requirements imposed by the system design. Both machines can be supplied with the complete line of interface and peripheral equipment necessary to perform all A/D and D/A conversions, data sampling functions, and control of printer, plotter, and magnetic tape units. The manufacturer-supplied system configurations and equipment specifications for the PDP-10 and SIGMA 5 are given in Figures 2-5 and 2-6.

The input/output data signal characteristics are specified below:

Input Signal Characteristics

A. Digital Signals

1. 2 angle encoder signals--22 bits each at 64/second

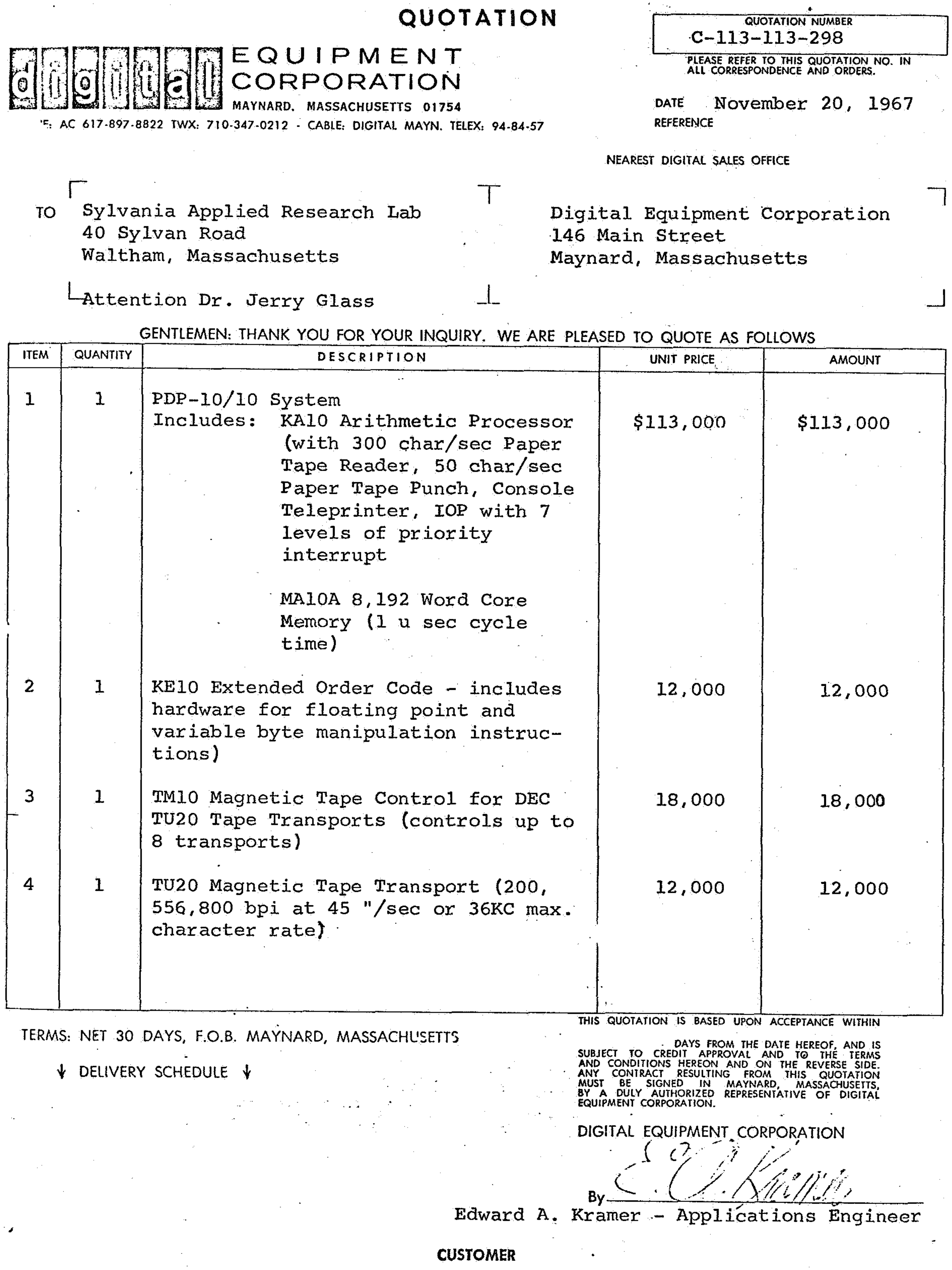
2. Clock signal

B. Analog Signals To Be Digitized

1. 2 star tracker signals--14 bits each, including sign, at 64/second

2. 3 range phase discriminator signals--14 bits each, including sign at 256/second

3. 2 tachometer signals--14 bits each, including sign, at 64/second



QUOTATION

digital EQUIPMENT CORPORATION
MAYNARD, MASSACHUSETTS 01754
TEL: AC 617-897-8822 TWX: 710-347-0212 · CABLE: DIGITAL MAYN. TELEX: 94-84-57

QUOTATION NUMBER
C-113-113-298
PLEASE REFER TO THIS QUOTATION NO. IN ALL CORRESPONDENCE AND ORDERS.

DATE November 20, 1967
REFERENCE

NEAREST DIGITAL SALES OFFICE

TO Sylvania Applied Research Lab
40 Sylvan Road
Waltham, Massachusetts

Attention Dr. Jerry Glass

Digital Equipment Corporation
146 Main Street
Maynard, Massachusetts

GENTLEMEN: THANK YOU FOR YOUR INQUIRY. WE ARE PLEASED TO QUOTE AS FOLLOWS

ITEM	QUANTITY	DESCRIPTION	UNIT PRICE	AMOUNT
1	1	PDP-10/10 System Includes: KA10 Arithmetic Processor (with 300 char/sec Paper Tape Reader, 50 char/sec Paper Tape Punch, Console Teleprinter, IOP with 7 levels of priority interrupt MA10A 8,192 Word Core Memory (1 u sec cycle time)	$113,000	$113,000
2	1	KE10 Extended Order Code - includes hardware for floating point and variable byte manipulation instructions)	12,000	12,000
3	1	TM10 Magnetic Tape Control for DEC TU20 Tape Transports (controls up to 8 transports)	18,000	18,000
4	1	TU20 Magnetic Tape Transport (200, 556,800 bpi at 45 "/sec or 36KC max. character rate)	12,000	12,000

TERMS: NET 30 DAYS, F.O.B. MAYNARD, MASSACHUSETTS

↓ DELIVERY SCHEDULE ↓

THIS QUOTATION IS BASED UPON ACCEPTANCE WITHIN DAYS FROM THE DATE HEREOF, AND IS SUBJECT TO CREDIT APPROVAL AND TO THE TERMS AND CONDITIONS HEREON AND ON THE REVERSE SIDE. ANY CONTRACT RESULTING FROM THIS QUOTATION MUST BE SIGNED IN MAYNARD, MASSACHUSETTS, BY A DULY AUTHORIZED REPRESENTATIVE OF DIGITAL EQUIPMENT CORPORATION.

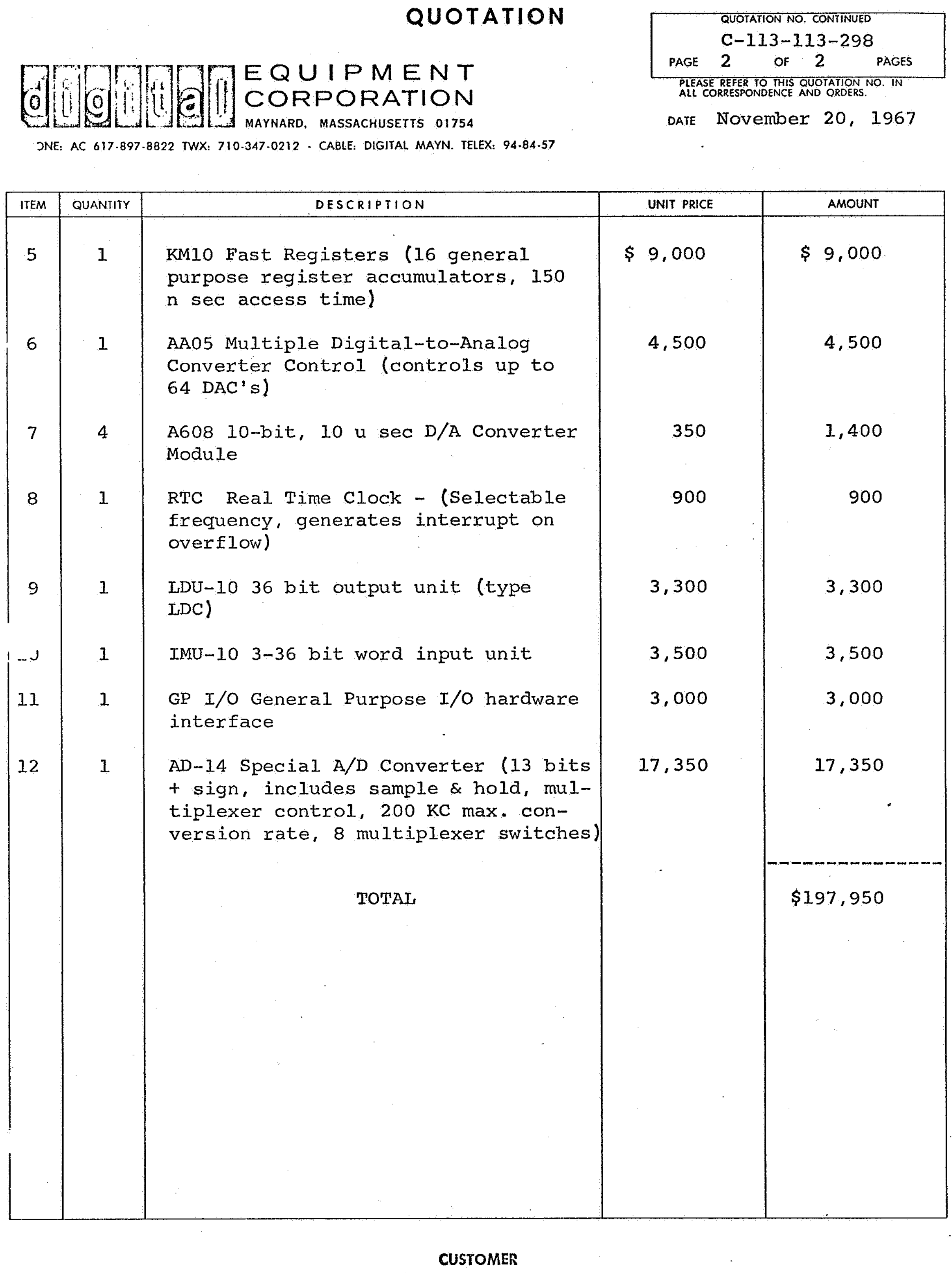
DIGITAL EQUIPMENT CORPORATION

By

Edward A. Kramer - Applications Engineer

CUSTOMER

Figure 2-5. Equipment Specifications for PDP-10. (Sheet 1 of 2)



QUOTATION

digital EQUIPMENT CORPORATION
MAYNARD, MASSACHUSETTS 01754
ONE: AC 617-897-8822 TWX: 710-347-0212 - CABLE: DIGITAL MAYN. TELEX: 94-84-57

QUOTATION NO. CONTINUED
C-113-113-298
PAGE 2 OF 2 PAGES

PLEASE REFER TO THIS QUOTATION NO. IN ALL CORRESPONDENCE AND ORDERS.

DATE November 20, 1967

ITEM	QUANTITY	DESCRIPTION	UNIT PRICE	AMOUNT
5	1	KM10 Fast Registers (16 general purpose register accumulators, 150 n sec access time)	$ 9,000	$ 9,000
6	1	AA05 Multiple Digital-to-Analog Converter Control (controls up to 64 DAC's)	4,500	4,500
7	4	A608 10-bit, 10 u sec D/A Converter Module	350	1,400
8	1	RTC Real Time Clock - (Selectable frequency, generates interrupt on overflow)	900	900
9	1	LDU-10 36 bit output unit (type LDC)	3,300	3,300
10	1	IMU-10 3-36 bit word input unit	3,500	3,500
11	1	GP I/O General Purpose I/O hardware interface	3,000	3,000
12	1	AD-14 Special A/D Converter (13 bits + sign, includes sample & hold, multiplexer control, 200 KC max. conversion rate, 8 multiplexer switches)	17,350	17,350
		TOTAL		$197,950

CUSTOMER

Figure 2-5. Equipment Specifications for PDP-10 (Sheet 2 of 2)

SDS SIGMA 5 SYSTEM

COMPUTER:

Qty	Model	Description	Purchase Price
1	8201	Sigma 5 CPU with integral I/O Processor, Two real time clocks, control panel and power supplies.	$ 50,000.00
1	8214	Memory Protect	4,000.00
1	8218	Floating Point Hardware	25,000.00
1	8251	Memory Module: 4096 Words (32 bits + parity)	33,000.00
1	8252	Memory Increment: 4096 Words (32 bits + parity)	17,500.00
1	8270	External Interface Feature	2,000.00
		Computer Price:	$131,500.00

PERIPHERALS:

Qty	Model	Description	Purchase Price
1	7010	Console keyboard/printer and controller (KSR35 teletype).	$ 6,000.00
1	7060	Paper tape reader 300 cps; paper tape punch 60 cps; spooler; cabinet and controller.	12,000.00
1	7361	Magnetic tape controller: 1 - 8 units	6,000.00
1	7362	Magnetic tape transports: 7 channels; 556 bpi; 37-1/2 ips.	19,000.00
1	7922	Analog and Digital Adapter	4,000.00
1	7930	Digital I/O Adapter	3,600.00
6	7950	Eight (8) Stored digital outputs.	708.00
1	7953	Eight (8) Pulsed Digital Outputs	75.00
		Peripheral Price:	$ 51,383.00

Figure 2-6. System Configuration and Equipment Specifications for Sigma 5 (Sheet 1 of 3)

ANALOG SECTION:

Qty	Model	Description	Purchase Price
1	8920	Cabinet	$ 1,000.00
1	ZX14	Blower Assembly	180.00
1	PT10	Power Supply	280.00
1	PX12	Power Supply	280.00
1	PX10	Power Supply	370.00
2	ET10-15	Cables	140.00
1	ET21	Cable	250.00
1	ET20	Cable	250.00
1	EZ28	Cable	20.00
1	EZ	Cable	250.00
1	DA35-15	D/A Controller	1,800.00
*4	DX16	D/A Converter: 11 bits + sign; ±10V	2,800.00
1	AD35	A/D Converter: 14 bits + sign	6,040.00
1	MU55	Multiplexer for up to 32 channels	2,450.00
2	SX40-2	Four Adjustable analog switches	360.00
1	SS10	Sample and Hold Controller	1,000.00
7	HX35	Sample and Hold Amplifier	3,500.00
		Analog Section:	$ 20,970.00
		SYSTEM TOTAL:	$203,853.00

*Two of the D/A converters are for a pen recorder. If a SDS model 7530 Graph Plotter (Calcomp) is substituted two of the DX16's can be eliminated. The price of the model 7530 (Plotter and Controller) is $13,000.00.

Figure 2-6. System Configuration and Equipment Specifications for Sigma 5 (Sheet 2 of 3)

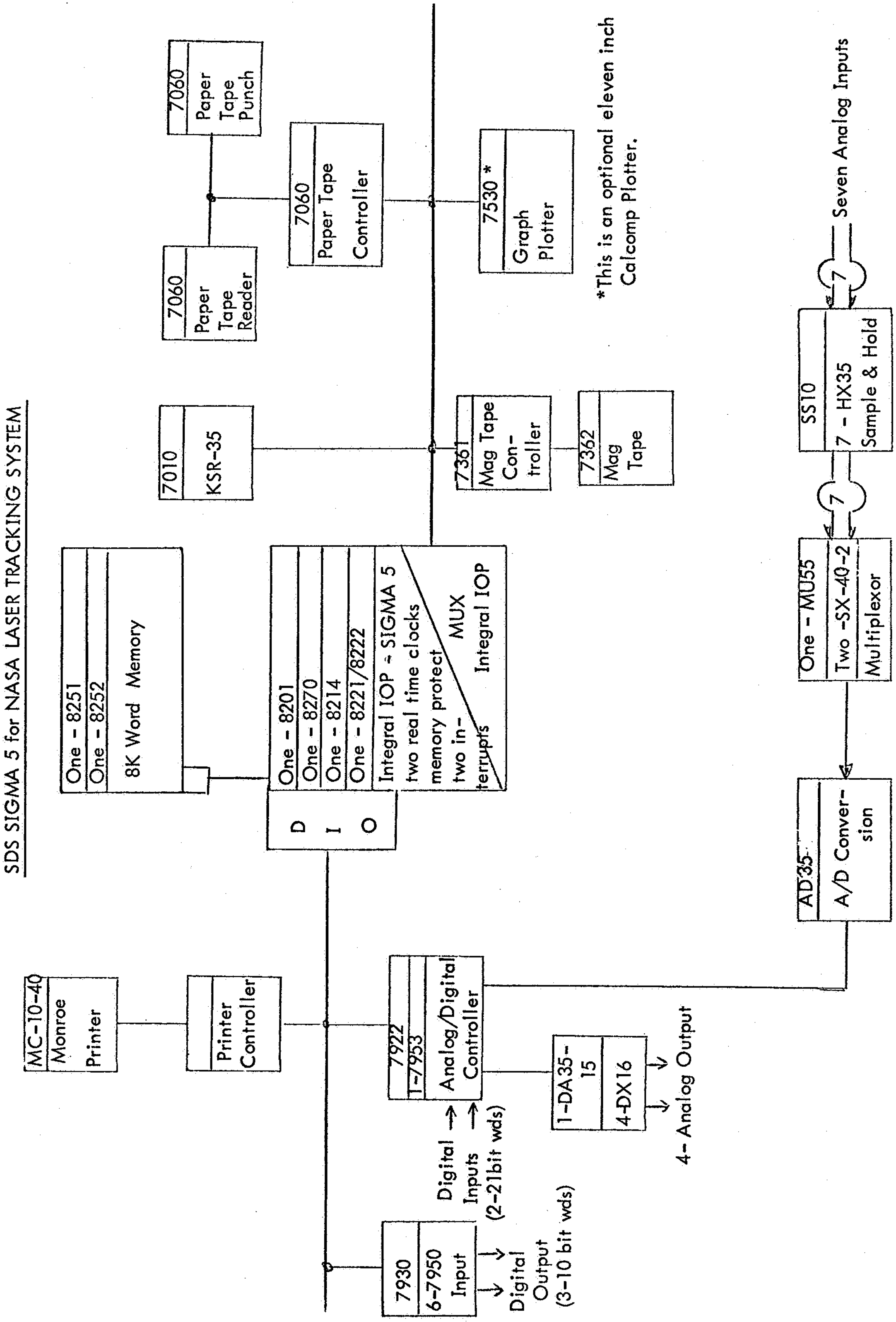


Figure 2-6. System Configuration and Equipment Specifications for Sigma 5 (Sheet 3 of 3)

Output Signal Characteristics

A. Digital Signals

1. 3 range servo signals--10 bits each at 256/second

2. 3 strobe signals to range servo--1 bit each at 256/second

3. Digital signals, as required to printer and magnetic tape

B. Digital Signals To Be Converted

1. 2 angle servo drive signals--10 bits each at 64/second

2. 2 signals to X-Y plotter--10 bits each at 64/second

As noted from the above listing, seven analog input signals must be sampled and digitized 64 times/second along with the sampled angle encoder values. Accuracy considerations, which required an ultimate angular resolution of 2^{-21} of a full circle, demanded that a tolerance be established for the maximum time skew of the sampled quantities. Assuming a worst case value for the angle rates (4 rad/sec), it was found that a time skew of 100 μsec for the sampled quantities was a sufficient margin to guard against inaccuracies being introduced at the input stage. In consultation with the computer manufacturer's representatives, it was decided to specify sample and hold circuitry in conjunction with the A/D converter.

SECTION 3

SOFTWARE DESIGN, IMPLEMENTATION, AND SIMULATION

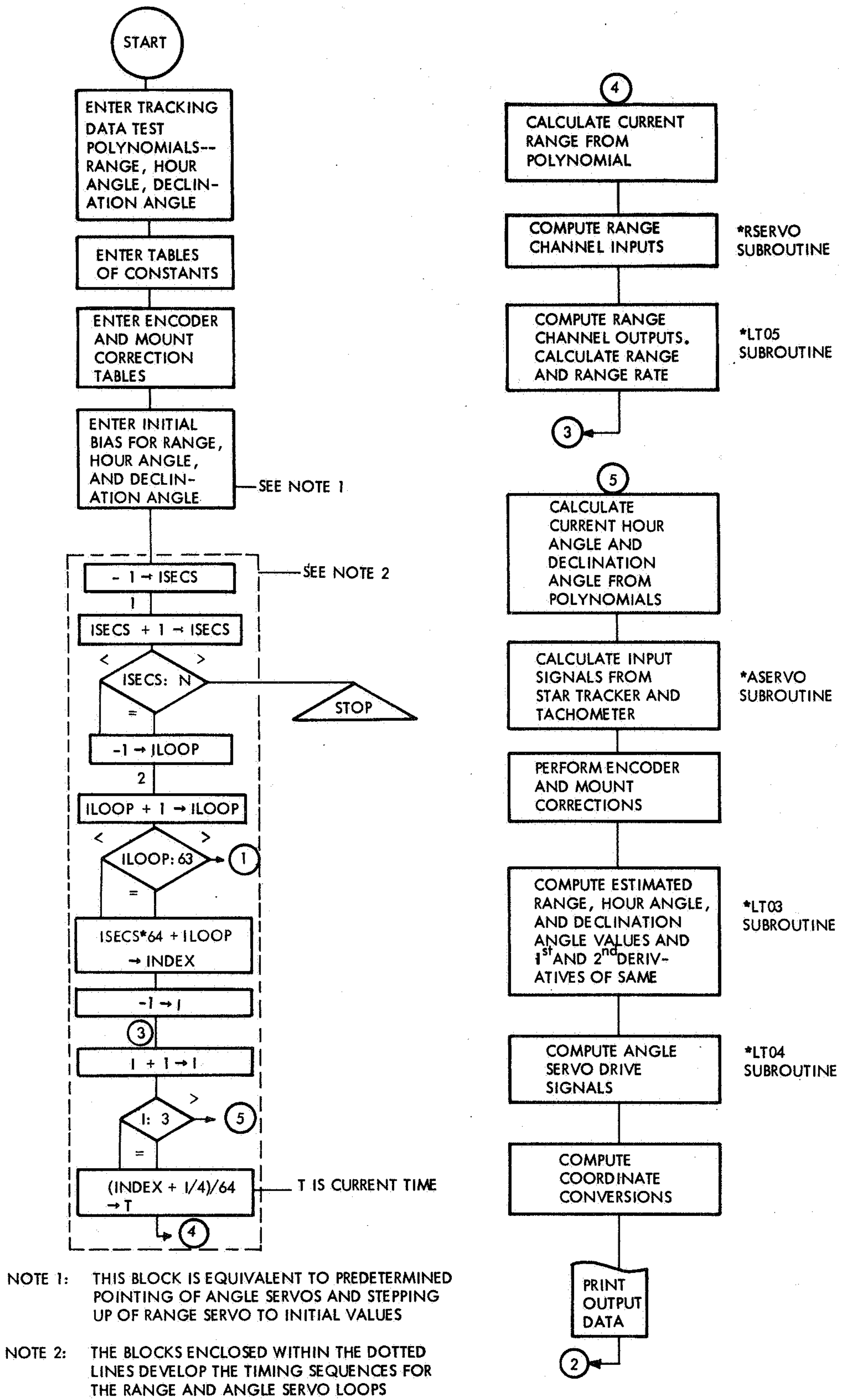
3.1 OBJECTIVES

The purpose of the system simulation was to exercise as many aspects of the system performance as possible in order to test the accuracy of the algorithms and the consistency of interfacing between the various subroutines. System simulation was only carried out for the tracking mode, for reasons to be stated. The flow chart in Figure 3-1 shows system operation, as simulated, blocked out in terms of subsystem functions. In some cases these blocks correspond directly to subroutines used in the simulation, such blocks being noted with an asterisk.

The basis for generating the polynomials to give range, hour angle, and declination angle data is explained in paragraph 3.2. In order to obtain realistic data for exercising the design algorithms it was necessary to simulate the actual range and angle servo hardware equipment. In this manner it was possible to obtain range channel, star tracker, and tachometer inputs consistent with the actual trajectory data and the response of the equipment to these inputs; thus the inclusion of the blocks "RSERVO" and "ASERVO" as shown in Figure 3-1.

A few words of explanation are in order concerning the blocks enclosed in dotted lines. These blocks develop the timing sequence for the operation. The outer loop (①) is incremented every second, the next loop (②) is incremented 64 times every second, and the innermost loop (③) is incremented 256 times every second. Thus the range servo loop is made to operate at four times the rate of the angle servo loop, the larger rate being required for achieving sufficient ranging accuracy. The entire system was simulated over a period of approximately 70 seconds, the length of time over which supplied trajectory data is available.

Each of the blocks labeled with an asterisk was tested individually before being combined into the overall system simulation. In the following paragraphs the structure of each of the subsystem subroutines is detailed.


START
ENTER TRACKING DATA TEST POLYNOMIALS-- RANGE, HOUR ANGLE, DECLINATION ANGLE
ENTER TABLES OF CONSTANTS
ENTER ENCODER AND MOUNT CORRECTION TABLES
ENTER INITIAL BIAS FOR RANGE, HOUR ANGLE, AND DECLINATION ANGLE
SEE NOTE 1
- 1 → ISECS
SEE NOTE 2
1
ISECS + 1 → ISECS
<
>
ISECS: N
=
STOP
-1 → ILOOP
2
ILOOP + 1 → ILOOP
<
>
ILOOP: 63
1
=
ISECS*64 + ILOOP → INDEX
-1 → I
3
I + 1 → I
>
I: 3
5
=
(INDEX + I/4)/64 → T
T IS CURRENT TIME
4
4
CALCULATE CURRENT RANGE FROM POLYNOMIAL
COMPUTE RANGE CHANNEL INPUTS
*RSERVO SUBROUTINE
COMPUTE RANGE CHANNEL OUTPUTS. CALCULATE RANGE AND RANGE RATE
*LT05 SUBROUTINE
3
5
CALCULATE CURRENT HOUR ANGLE AND DECLINATION ANGLE FROM POLYNOMIALS
CALCULATE INPUT SIGNALS FROM STAR TRACKER AND TACHOMETER
*ASERVO SUBROUTINE
PERFORM ENCODER AND MOUNT CORRECTIONS
COMPUTE ESTIMATED RANGE, HOUR ANGLE, AND DECLINATION ANGLE VALUES AND 1st AND 2nd DERIVATIVES OF SAME
*LT03 SUBROUTINE
COMPUTE ANGLE SERVO DRIVE SIGNALS
*LT04 SUBROUTINE
COMPUTE COORDINATE CONVERSIONS
PRINT OUTPUT DATA
2

NOTE 1: THIS BLOCK IS EQUIVALENT TO PREDETERMINED POINTING OF ANGLE SERVOS AND STEPPING UP OF RANGE SERVO TO INITIAL VALUES

NOTE 2: THE BLOCKS ENCLOSED WITHIN THE DOTTED LINES DEVELOP THE TIMING SEQUENCES FOR THE RANGE AND ANGLE SERVO LOOPS

Figure 3-1. System Simulation Flow Chart

The real-time projections for system performance are discussed in paragraph 4.1.

The reasoning for restricting the overall simulation testing to the tracking mode is as follows. In the track mode all basic subsystem functions are exercised; only the logical steps necessary to switch between modes are absent. If the various algorithms did not perform properly in the track mode then, in any case, they would be unsatisfactory. The only feature of system behavior to be tested in the reacquisition mode is the target state estimator. This estimator (see paragraph 3.6) has been designed on the basis of MSFC-supplied reference trajectory data and a pessimistic noise level. Thus it must be assumed that under normal signal dropout considerations, the estimator can provide usable output during the reacquisition phase based on extrapolation of the current best estimates of velocity and acceleration. Abnormal signal dropout conditions, that is, loss of signal for periods exceeding one or two seconds, cannot be accounted for in the design. In the acquisition mode the servos are to respond to pointing commands generated from pre-assigned coordinate information.

For the angle servos the pointing commands can be implemented directly, with no "stepping" increment necessary. For the range servos, however, it is necessary to use increments that are no greater than one half the coarsest range channel resolution to "step" the range servo up to its initial value and thereby avoid saturation.

3.2 COMPUTATION OF NOMINAL TRACKING DATA

In order to determine the tracking data characteristics it was necessary to write a special computer program to

1. Convert MSFC-supplied trajectory data to tracking data for a particular launch site--tracker site geometry,
2. Derive accelerations for the tracker-measured quantities, and
3. Do a least-squares fit of polynomials to the results so that simulation program inputs could be obtained.

The MSFC-supplied trajectory data were position and velocity, at one second intervals, in an inertial coordinate system which was earth-fixed and launch-derived. That is, the trajectory was known in a geocentric inertial coordinate system which was aligned with the launch azimuth and the geodetic vertical at the launch site. The tracker site was chosen as that one of the allowable tracker sites which was closest to the launch site; in this way, the maximum velocities and accelerations of the tracking data would be obtained. The results are tabulated in Figure 3-2; the acceleration information, which was not in the original trajectory data, was obtained by differencing the velocity data. The coefficients of sixth-degree, least-squares fitted polynomials are in Tables 3-1 and 3-2. Significant excerpts from the data, pertaining to maximum values, are given in Table 3-3. Note that the form of the fitted polynomial is

$$C_0 + C_1 t + C_2 t^2 + C_3 t^3 + C_4 t^4 + C_5 t^5 + C_6 t^6$$

Data on the coordinate conversions performed are in Appendix D.

3.3 MOUNT ANGLE SERVO SUBROUTINE

A full discussion of the software, and simulation results, is given in Appendix B.

TIME	RANGE ACCELERATION	HOUR ANGLE ACCELERATION	DECLINATION ACCELERATION
0	0	0	0
.3	7.69540499E-03	-1.44239590E-02	8.74001186E-03
1.0	1.89061192E-02	-3.44175529E-02	2.08425082E-02
2.0	3.26047251E-02	-5.02976487E-02	3.03432389E-02
3.0	4.56906789E-02	-5.17820124E-02	3.09559932E-02
4.0	6.71842869E-02	-5.33622486E-02	3.14301027E-02
5.0	9.69375740E-02	-5.51011083E-02	3.18042318E-02
6.0	1.35862321E-01	-5.70337050E-02	3.20738077E-02
7.0	1.84418975E-01	-5.91531691E-02	3.22168092E-02
8.0	2.42356375E-01	-6.14535256E-02	3.22267039E-02
9.0	3.10908660E-01	-6.39789362E-02	3.20824869E-02
10.0	3.90362042E-01	-6.67731762E-02	3.18045682E-02
11.0	4.80543951E-01	-6.98267400E-02	3.13740651E-02
12.0	5.85877702E-01	-7.32185564E-02	3.05918713E-02
13.0	6.98026835E-01	-7.66524770E-02	2.97937603E-02
14.0	8.10764902E-01	-7.99625365E-02	2.91979681E-02
15.0	9.27601979E-01	-8.32702323E-02	2.86220511E-02
16.0	1.05580863E 00	-8.67458658E-02	2.78891006E-02
17.0	1.19724361E 00	-9.04397184E-02	2.68973057E-02
18.0	1.35209189E 00	-9.43131185E-02	2.55560191E-02
19.0	1.52285055E 00	-9.83721635E-02	2.37637839E-02
20.0	1.71015640E 00	-1.02611276E-01	2.14874832E-02
21.0	1.91377331E 00	-1.06949986E-01	1.86931993E-02
22.0	2.13293883E 00	-1.11268697E-01	1.53590948E-02
23.0	2.36401137E 00	-1.15356803E-01	1.15095132E-02
24.0	2.61037688E 00	-1.19150480E-01	7.08177507E-03
25.0	2.86642484E 00	-1.22408573E-01	2.18836975E-03
26.0	3.13541577E 00	-1.24970363E-01	-3.23921529E-03
27.0	3.41100106E 00	-1.26532156E-01	-9.04908187E-03
28.0	3.69150562E 00	-1.26868190E-01	-1.51484067E-02
29.0	3.98282750E 00	-1.25744153E-01	-2.16222245E-02
30.0	4.27280088E 00	-1.22851044E-01	-2.80982987E-02
31.0	4.56740880E 00	-1.17953109E-01	-3.46111347E-02
32.0	4.85872447E 00	-1.10896483E-01	-4.08557680E-02
33.0	5.14962113E 00	-1.01535236E-01	-4.67774852E-02
34.0	5.43534635E 00	-8.98643996E-02	-5.21122487E-02
35.0	5.70914705E 00	-7.61526127E-02	-5.65370003E-02
36.0	5.97986109E 00	-6.04529330E-02	-6.01487546E-02
37.0	6.23848494E 00	-4.33528043E-02	-6.26148816E-02
38.0	6.48620825E 00	-2.53229652E-02	-6.39088455E-02
39.0	6.72004478E 00	-7.06709863E-03	-6.39665272E-02
40.0	6.94514767E 00	1.09947350E-02	-6.29410828E-02
41.0	7.15907836E 00	2.81822447E-02	-6.08603086E-02
42.0	7.35792793E 00	4.37288311E-02	-5.77892954E-02
43.0	7.55008455E 00	5.77460256E-02	-5.40306440E-02
44.0	7.72808149E 00	6.94525856E-02	-4.96279857E-02
45.0	7.89861884E 00	7.90881434E-02	-4.48433020E-02
46.0	8.05706391E 00	8.62812595E-02	-3.97986749E-02
47.0	8.21056756E 00	9.16212124E-02	-3.47034545E-02
48.0	8.35398993E 00	9.48044060E-02	-2.96546632E-02
49.0	8.49168143E 00	9.62981398E-02	-2.47869404E-02
50.0	8.62342251E 00	9.62631946E-02	-2.01874918E-02
51.0	8.75098511E 00	9.51686357E-02	-1.59054344E-02
52.0	8.87046556E 00	9.30052255E-02	-1.19920423E-02
53.0	8.98741861E 00	9.01519955E-02	-8.46269718E-03
54.0	9.10245035E 00	8.68003811E-02	-5.31817420E-03
55.0	9.21335196E 00	8.30082983E-02	-2.57166170E-03
56.0	9.32331198E 00	7.91035251E-02	-1.69944207E-04
57.0	9.43102032E 00	7.49802573E-02	1.85446598E-03
58.0	9.53666037E 00	7.09565784E-02	3.59032460E-03
59.0	9.63147310E 00	6.69953396E-02	5.03773810E-03
60.0	9.72438421E 00	6.31478284E-02	6.21866971E-03
61.0	9.81665996E 00	5.94798060E-02	7.17494990E-03
62.0	9.91561472E 00	5.58784970E-02	7.88507745E-03
63.0	1.00300815E 01	5.24402045E-02	8.40425547E-03
64.0	1.01554239E 01	4.91903025E-02	8.75320951E-03
65.0	1.02889759E 01	4.61410948E-02	8.96398552E-03
66.0	1.04409110E 01	4.32328134E-02	9.03737141E-03
67.0	1.06147145E 01	4.05147617E-02	9.02566073E-03
68.0	1.08063829E 01	3.79533336E-02	8.91753029E-03
69.0	1.10090410E 01	3.55918504E-02	8.74834593E-03
70.0	1.12297865E 01	3.34281102E-02	8.54134247E-03
71.0	1.14620263E 01	3.14501742E-02	8.28341694E-03
72.0	1.17040494E 01	2.96641560E-02	8.01490697E-03

Figure 3-2. Tracking Data as Obtained from Trajectory Data (Sheet 1 of 2)

```
LATITUDE OF TRACKER  2.85878162E 01          LONGITUDE DIFFERENCE  2.10187366E-02
GEOCENTRIC RADIUS AND DECLINATION  6.37331451E 06  2.84264328E 01
TRACKER IS    7519.3  FEET SOUTH AND    7670.7  FEET EAST OF LAUNCH SITE

TIME       RANGE            HOUR ANGLE        DECLINATION         RANGE           HOUR ANGLE        DECLINATION
                                                                   RATE              RATE              RATE

  0     3.07281394E 03    1.17318988E 02    4.11496766E 01                0                 0                 0
  .3    3.07281507E 03    1.17317060E 02    4.11508453E 01    7.69540499E-03   -1.64239590E-02    8.74001186E-03
 1.0    3.07282754E 03    1.17293905E 02    4.11648717E 01    2.66015247E-02   -4.88415120E-02    2.95825200E-02
 2.0    3.07286979E 03    1.17220540E 02    4.12092697E 01    5.92062493E-02   -9.91391607E-02    5.99257590E-02
 3.0    3.07294921E 03    1.17096762E 02    4.12839467E 01    1.04896928E-01   -1.50921173E-01    9.08817521E-02
 4.0    3.07308639E 03    1.16918364E 02    4.13910670E 01    1.72081215E-01   -2.04283422E-01    1.22311855E-01
 5.0    3.07330436E 03    1.16686768E 02    4.15292071E 01    2.69018789E-01   -2.59384530E-01    1.54116087E-01
 6.0    3.07363679E 03    1.16399343E 02    4.16991568E 01    4.04881110E-01   -3.16418235E-01    1.86189894E-01
 7.0    3.07413061E 03    1.16053283E 02    4.19016029E 01    5.89300085E-01   -3.75571404E-01    2.18406704E-01
 8.0    3.07483469E 03    1.15647519E 02    4.21359307E 01    8.31656660E-01   -4.37024930E-01    2.50633407E-01
 9.0    3.07581548E 03    1.15178857E 02    4.24025477E 01    1.14256512E 00   -5.01003866E-01    2.82715894E-01
10.0    3.07714820E 03    1.14643856E 02    4.27016711E 01    1.53292716E 00   -5.67777042E-01    3.14520463E-01
11.0    3.07891480E 03    1.14040892E 02    4.30322146E 01    2.01347111E 00   -6.37603782E-01    3.45894528E-01
12.0    3.08120853E 03    1.13368149E 02    4.33928676E 01    2.59934881E 00   -7.10822339E-01    3.76486399E-01
13.0    3.08414643E 03    1.12619446E 02    4.37842065E 01    3.29737565E 00   -7.87474816E-01    4.06280159E-01
14.0    3.08784620E 03    1.11790835E 02    4.42058623E 01    4.10814055E 00   -8.67437352E-01    4.35478127E-01
15.0    3.09240523E 03    1.10882679E 02    4.46553941E 01    5.03574253E 00   -9.50707584E-01    4.64100179E-01
16.0    3.09795640E 03    1.09889003E 02    4.51334498E 01    6.09155116E 00   -1.03745345E 00    4.91989279E-01
17.0    3.10463980E 03    1.08805791E 02    4.56393799E 01    7.28879477E 00   -1.12789317E 00    5.18886585E-01
18.0    3.11259153E 03    1.07631037E 02    4.61711914E 01    8.64088666E 00   -1.22220629E 00    5.44442604E-01
19.0    3.12197594E 03    1.06360397E 02    4.67275148E 01    1.01637372E 01   -1.32057845E 00    5.68206388E-01
20.0    3.13297486E 03    1.04989351E 02    4.73064847E 01    1.18738936E 01   -1.42318973E 00    5.89693871E-01
21.0    3.14578564E 03    1.03513247E 02    4.79057000E 01    1.37876669E 01   -1.53013971E 00    6.08387070E-01
22.0    3.16062560E 03    1.01927438E 02    4.85222128E 01    1.59206057E 01   -1.64140841E 00    6.23746165E-01
23.0    3.17749714E 03    1.00229723E 02    4.91517049E 01    1.82846171E 01   -1.75676521E 00    6.35255678E-01
24.0    3.19726774E 03    9.86134626E 01    4.97909534E 01    2.08949940E 01   -1.87591569E 00    6.42337453E-01
25.0    3.21956690E 03    9.64772171E 01    5.04346349E 01    2.37614188E 01   -1.99832427E 00    6.44525823E-01
26.0    3.24489004E 03    9.44153230E 01    5.10783457E 01    2.68968346E 01   -2.12329463E 00    6.41286608E-01
27.0    3.27348289E 03    9.22278416E 01    5.17158494E 01    3.03078357E 01   -2.24982678E 00    6.32237526E-01
28.0    3.30559136E 03    8.99160171E 01    5.23407247E 01    3.39993413E 01   -2.37669497E 00    6.17089119E-01
29.0    3.34157816E 03    8.74749738E 01    5.29477897E 01    3.79821688E 01   -2.50243913E 00    5.95466895E-01
30.0    3.38166111E 03    8.49114434E 01    5.35296441E 01    4.22549697E 01   -2.62529017E 00    5.67368596E-01
31.0    3.42617827E 03    8.22265568E 01    5.40802640E 01    4.68223785E 01   -2.74324328E 00    5.32757461E-01
32.0    3.47538014E 03    7.94284841E 01    5.45929540E 01    5.16811029E 01   -2.85413976E 00    4.91901693E-01
33.0    3.52961164E 03    7.65227392E 01    5.50619338E 01    5.68307241E 01   -2.95567500E 00    4.45124208E-01
34.0    3.58916867E 03    7.35196201E 01    5.54814979E 01    6.22660704E 01   -3.04553940E 00    3.93011959E-01
35.0    3.65422967E 03    7.04363147E 01    5.58465280E 01    6.79752175E 01   -3.12169201E 00    3.36474959E-01
36.0    3.72518382E 03    6.72826091E 01    5.61531858E 01    7.39550786E 01   -3.18214495E 00    2.76326204E-01
37.0    3.80224659E 03    6.40769723E 01    5.63983540E 01    8.01935635E 01   -3.22549775E 00    2.13711323E-01
38.0    3.88567924E 03    6.08367206E 01    5.65801350E 01    8.66797717E 01   -3.25082071E 00    1.49802477E-01
39.0    3.97566969E 03    5.75817821E 01    5.66979900E 01    9.33998165E 01   -3.25788781E 00    8.58359502E-02
40.0    4.07252202E 03    5.43279813E 01    5.67522305E 01    1.00344964E 02   -3.24689308E 00    2.28948674E-02
41.0    4.17647040E 03    5.10927433E 01    5.67443253E 01    1.07504043E 02   -3.21871083E 00   -3.79654411E-02
42.0    4.28759676E 03    4.78955968E 01    5.66773441E 01    1.14861970E 02   -3.17498200E 00   -9.57547366E-02
43.0    4.40625444E 03    4.47476595E 01    5.65540697E 01    1.22412055E 02   -3.11723598E 00   -1.49785381E-01
44.0    4.53250185E 03    4.16645249E 01    5.63791510E 01    1.30140137E 02   -3.04778339E 00   -1.99413366E-01
45.0    4.66660880E 03    3.86550766E 01    5.61567595E 01    1.38038755E 02   -2.96869525E 00   -2.44256668E-01
46.0    4.80860768E 03    3.57298161E 01    5.58924485E 01    1.46095819E 02   -2.88241399E 00   -2.84055343E-01
47.0    4.95883779E 03    3.28923356E 01    5.55904172E 01    1.54306387E 02   -2.79079278E 00   -3.18758798E-01
48.0    5.11731710E 03    3.01486549E 01    5.52563733E 01    1.62660377E 02   -2.69598837E 00   -3.48413461E-01
49.0    5.28421839E 03    2.75007000E 01    5.48951445E 01    1.71152058E 02   -2.59969023E 00   -3.73200401E-01
50.0    5.45962711E 03    2.49495696E 01    5.45116836E 01    1.79775481E 02   -2.50342704E 00   -3.93387893E-01
51.0    5.64379745E 03    2.24936554E 01    5.41098711E 01    1.88526466E 02   -2.40825840E 00   -4.09293328E-01
52.0    5.83672647E 03    2.01322441E 01    5.36943636E 01    1.97396931E 02   -2.31525317E 00   -4.21285370E-01
53.0    6.03858386E 03    1.78624536E 01    5.32686668E 01    2.06384350E 02   -2.22510118E 00   -4.29748067E-01
54.0    6.24954096E 03    1.56809162E 01    5.28358904E 01    2.15486800E 02   -2.13830080E 00   -4.35066241E-01
55.0    6.46959365E 03    1.35845576E 01    5.23994457E 01    2.24700152E 02   -2.05529250E 00   -4.37637903E-01
56.0    6.69900897E 03    1.15689693E 01    5.19613109E 01    2.34023464E 02   -1.97618897E 00   -4.37807847E-01
57.0    6.93749607E 03    9.63071641E 00    5.15244282E 01    2.43454485E 02   -1.90120872E 00   -4.35953381E-01
58.0    7.18593041E 03    7.76528180E 00    5.10900706E 01    2.52991145E 02   -1.83025214E 00   -4.32363057E-01
59.0    7.44371910E 03    5.96886555E 00    5.06601478E 01    2.62622618E 02   -1.76325680E 00   -4.27325318E-01
60.0    7.71116418E 03    4.23753023E 00    5.02359408E 01    2.72347002E 02   -1.70010897E 00   -4.21106649E-01
61.0    7.98846523E 03    2.56743070E 00    4.98182031E 01    2.82163662E 02   -1.64062916E 00   -4.13931699E-01
62.0    8.27554664E 03    9.55042497E-01    4.94082396E 01    2.92079277E 02   -1.58475067E 00   -4.06046621E-01
63.0    8.57261947E 03   -6.03206157E-01    4.90063915E 01    3.02109358E 02   -1.53231046E 00   -3.97642366E-01
64.0    8.87980611E 03   -2.11065661E 00    4.86130775E 01    3.12264782E 02   -1.48312016E 00   -3.88889156E-01
65.0    9.19723589E 03   -3.57044511E 00    4.82285851E 01    3.22553758E 02   -1.43697907E 00   -3.79925171E-01
66.0    9.52495082E 03   -4.98560391E 00    4.78532748E 01    3.32994669E 02   -1.39374625E 00   -3.70887800E-01
67.0    9.86328361E 03   -6.35883293E 00    4.74868110E 01    3.43609386E 02   -1.35323149E 00   -3.61862139E-01
68.0    1.02122929E 04   -7.69287065E 00    4.71293935E 01    3.54415747E 02   -1.31527816E 00   -3.52944609E-01
69.0    1.05721405E 04   -8.99022372E 00    4.67809392E 01    3.65424808E 02   -1.27968631E 00   -3.44196263E-01
70.0    1.09431850E 04   -1.02529937E 01    4.64409966E 01    3.76654594E 02   -1.24625820E 00   -3.35654920E-01
71.0    1.13255108E 04   -1.14834240E 01    4.61095652E 01    3.88116620E 02   -1.21480802E 00   -3.27371503E-01
72.0    1.17194934E 04   -1.26832095E 01    4.57861727E 01    3.99820670E 02   -1.18514387E 00   -3.19356596E-01
```

Figure 3-2. Tracking Data as Obtained from Trajectory Data (Sheet 2 of 2)

TABLE 3-1

COEFFICIENTS OF POLYNOMIAL FIT TO TRACKING DATA (POSITION)

	Range (meters)	Hour Angle (degrees)	Declination (degrees)
C_0	3.074285390E 03	1.174125936E 02	4.103893121E 01
C_1	-2.855444864E 00	-8.129312544E-02	2.129026631E-01
C_2	6.487582373E-01	-2.099399294E-02	-3.227090683E-02
C_3	-4.946418019E-02	8.142128362E-04	3.694090156E-03
C_4	1.930160010E-03	-8.858125812E-05	-1.181951605E-04
C_5	-1.928922545E-05	1.941353416E-06	1.480725462E-06
C_6	6.734844011E-08	-1.221808510E-08	-6.537293629E-09

TABLE 3-2

COEFFICIENTS OF POLYNOMIAL FIT TO TRACKING DATA (VELOCITY)

	Range Rate (m/sec)	Hour Angle Rate (^{o}/sec)	Declination Rate (^{o}/sec)
C_0	-5.053388666E-02	4.709042989E-02	7.323323364E-03
C_1	2.507330732E-02	-1.330068257E-01	2.449193662E-02
C_2	1.437644815E-02	1.715535798E-02	1.067491980E-04
C_3	-1.177956236E-03	-1.296911800E-03	1.081695380E-04
C_4	1.394900353E-04	3.784461609E-05	-7.557305137E-06
C_5	-2.590478355E-06	-4.654506566E-07	1.456644425E-07
C_6	1.457511935E-08	2.060032254E-09	-8.779346109E-10

TABLE 3-3

MAXIMUM VALUES OF TRACKING DATA QUANTITIES

	Value	Time of Occurrence
r	10.21 km	68 sec
$\dot{r}$	354.42 m/sec	68 sec
$\ddot{r}$	10.806 m/sec^2	68 sec
θ_{HA}	117.32 deg	0 sec
$\dot{\theta}_{HA}$	-3.258 deg/sec	39 sec
$\ddot{\theta}_{HA}$	-0.1269 deg/sec	28 sec
θ_{DEC}	56.75 deg	40 sec
$\dot{\theta}_{DEC}$	0.644 deg/sec	28 sec
$\ddot{\theta}_{DEC}$	-0.06397 deg/sec^2	39 sec

3.4 RANGE SERVO SUBROUTINE

The block diagram in Figure 3-3 shows the implementation of the Range Servo loop as simulated on the digital computer. The linearized and simplified representation of the hardware is simulated on the computer for testing purposes only; while the electronic controller represents a part of the software for the Laser Tracker.

To achieve the required accuracy of 0.5 cm in the output range word, a cycle time (TP) of 1/256 sec for the electronic controller was found necessary; a 9 bit A/D converter was found adequate and a bandwidth of 55 rad/sec for the servo loop was selected. Depending on the magnitude of the error signal in the closed loop, the range servo will track in either the fine, medium, or coarse mode. As seen from Figure 3-3 these channels differ in gain values only. A digital signal from the peripheral A/D converter, proportional to the error in the loop, serves as the input to the controller. For a phase error of 1/2 rad, which corresponds approximately to an error in range of 40 cm, the fine channel goes into saturation and the software switches the tracking operation from channel 3 to channel 2.

The range rate and the output range word are calculated numerically. Proportional plus integral compensation is included to eliminate velocity error, reduce acceleration errors, and assure good stability. It can be shown that the acceleration errors in such a system are given by

$$R_E = \frac{\ddot{R}}{\omega_{IP} \cdot \omega_{CP}}$$

where ω_{IP} is the lead corner frequency of proportional plus integral compensation (16.7 rad/sec) and ω_{CP} is the crossover frequency of the servo loop (55 rad/sec). The value of acceleration at each iteration is calculated by time differencing and averaging of available range rate words and past acceleration values. This value of acceleration is then used to compensate for the acceleration error in the output range word.

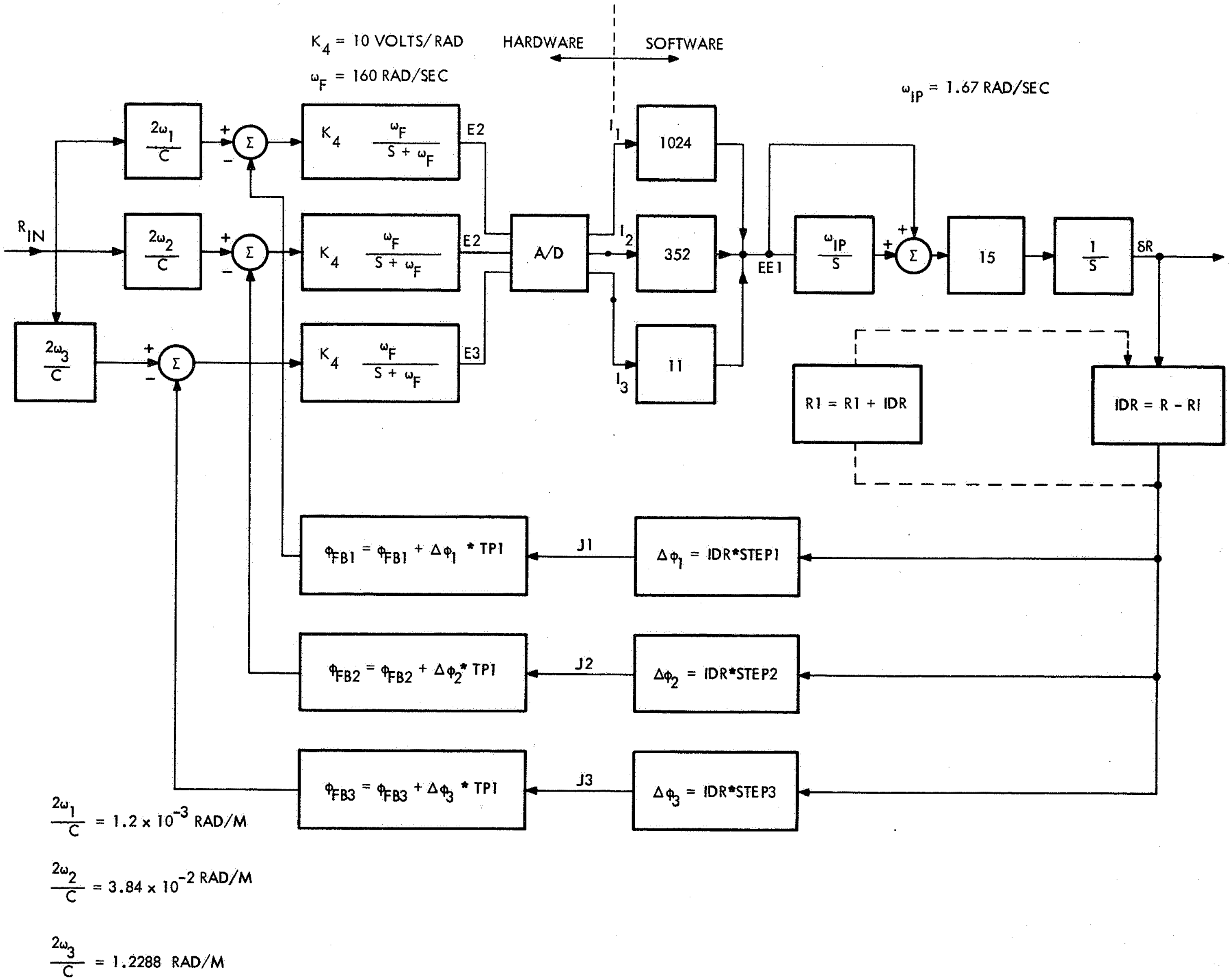


Figure 3-3. Block Diagram of Range Servo

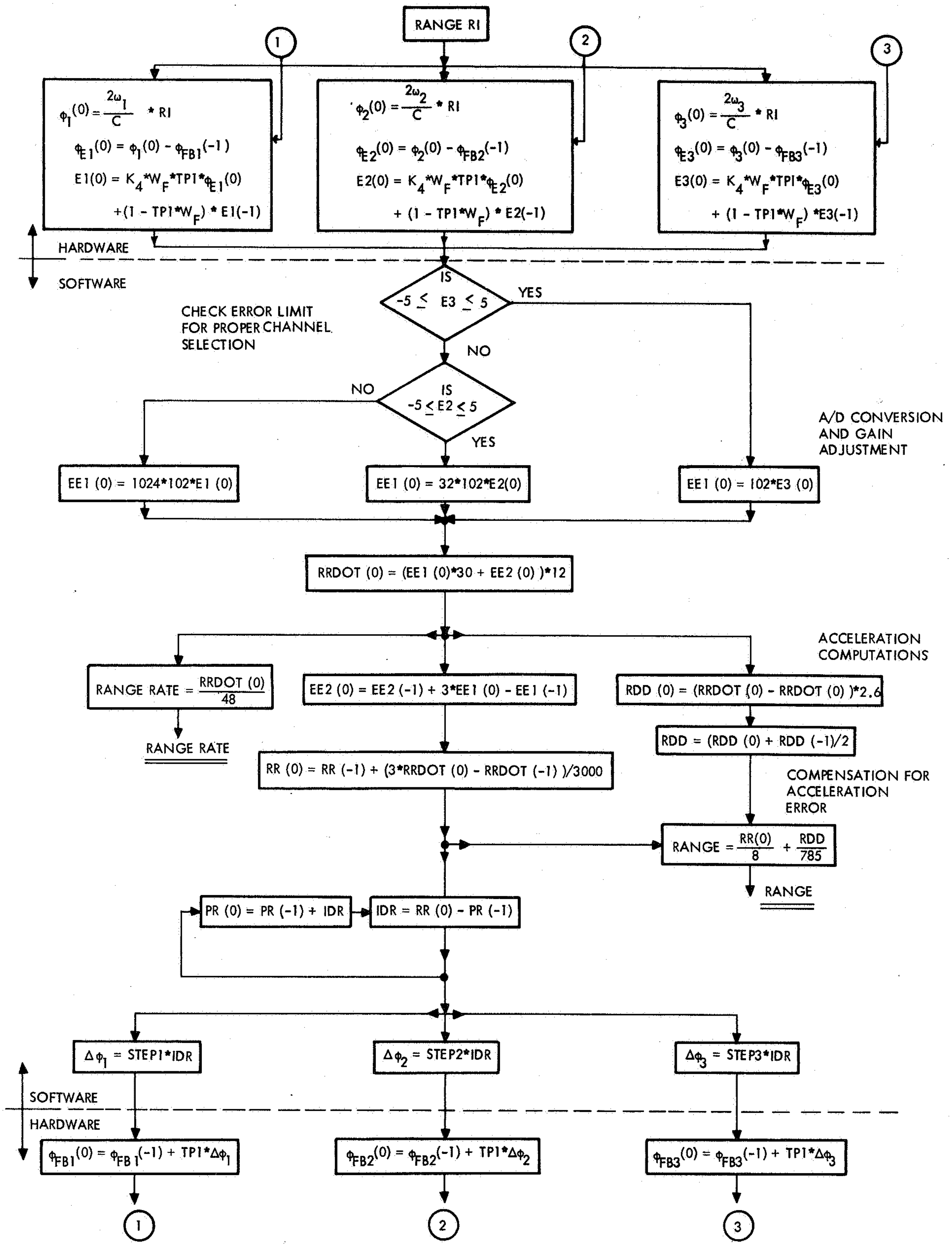

RANGE RI
1
2
3
φ1(0) = 2ω1/C * RI
φE1(0) = φ1(0) - φFB1(-1)
E1(0) = K4*WF*TP1*φE1(0) +(1 - TP1*WF) * E1(-1)
φ2(0) = 2ω2/C * RI
φE2(0) = φ2(0) - φFB2(-1)
E2(0) = K4*WF*TP1*φE2(0) + (1 - TP1*WF) * E2(-1)
φ3(0) = 2ω3/C * RI
φE3(0) = φ3(0) - φFB3(-1)
E3(0) = K4*WF*TP1*φE3(0) + (1 - TP1*WF) *E3(-1)
HARDWARE
SOFTWARE
IS -5 ≤ E3 ≤ 5
YES
NO
CHECK ERROR LIMIT FOR PROPER CHANNEL SELECTION
NO
IS -5 ≤ E2 ≤ 5
YES
A/D CONVERSION AND GAIN ADJUSTMENT
EE1 (0) = 1024*102*E1 (0)
EE1 (0) = 32*102*E2(0)
EE1 (0) = 102*E3 (0)
RRDOT (0) = (EE1 (0)*30 + EE2 (0))*12
ACCELERATION COMPUTATIONS
RANGE RATE = RRDOT (0)/48
RANGE RATE
EE2 (0) = EE2 (-1) + 3*EE1 (0) - EE1 (-1)
RDD (0) = (RRDOT (0) - RRDOT (0))*2.6
RDD = (RDD (0) + RDD (-1)/2
RR (0) = RR (-1) + (3*RRDOT (0) - RRDOT (-1))/3000
COMPENSATION FOR ACCELERATION ERROR
RANGE = RR(0)/8 + RDD/785
RANGE
PR (0) = PR (-1) + IDR
IDR = RR (0) - PR (-1)
Δφ1 = STEP1*IDR
Δφ2 = STEP2*IDR
Δφ3 = STEP3*IDR
SOFTWARE
HARDWARE
φFB1(0) = φFB 1(-1) + TP1*Δφ1
φFB2(0) = φFB2(-1) + TP1*Δφ2
φFB3(0) = φFB3(-1) + TP1*Δφ3
1
2
3

Figure 3-4. Flow Diagram of the Program for Range Servo (Sheet 1 of 2)

NOMENCLATURE	DESCRIPTION
RI	Simulated range input from nominal tracking data
X(0)	Value of X at this instant, t_n
X(-1)	Value of X at previous time, t_{n-1}
ϕ_1, ϕ_2, ϕ_3	Phase detector input to channels 1, 2, 3, respectively
ϕ_{E1}, ϕ_{E2}, ϕ_{E3}	Phase error in the loop of channels 1, 2, 3, respectively
ϕ_{FB1}, ϕ_{FB2}, ϕ_{FB3}	Feedback phase signal for channels 1, 2, 3, respectively
TP1	Time for one computation in hardware section corresponding to 20 kHz: $5\text{x}10^{-5}$ sec
TP	Time for one computer iteration--corresponding to 256 sec^{-1} or $4\text{x}10^{-3}$ sec
ω_F	Filter cutoff frequency = 160 rad/sec
K_4	Gain constant = 10 v/rad
E1, E2, E3	Output of filter in channels 1, 2, 3 respectively
IDR	Change in range in steps of 1/8 cm
RDD	Computed acceleration--for compensation of acceleration error
PR	Previous value of RR (i.e., 8*Range)
Step 1	$2\frac{\omega_1}{c} \text{ x } \frac{1}{8} \text{ x } \frac{1}{100} \text{ x } \frac{1}{TP} = 0.375 \text{ x } 10^{-3}$
Step 2	$2\frac{\omega_2}{c} \text{ x } \frac{1}{8} \text{ x } \frac{1}{100} \text{ x } \frac{1}{TP} = 1.2 \text{ x } 10^{-2}$
Step 3	$2\frac{\omega_3}{c} \text{ x } \frac{1}{8} \text{ x } \frac{1}{100} \text{ x } \frac{1}{TP} = 0.384$
$\Delta\phi_1$, $\Delta\phi_2$, $\Delta\phi_3$	Step increment in feedback phase in channels 1, 2, 3, respectively

Figure 3-4. Flow Diagram of the Program for Range Servo (Sheet 2 of 2)

The phase feedback in the loop is proportional to the range word calculated at the last iteration. This is updated by generating a digital word proportional to the change in range, which controls the number of pulses to be added to the train of clock pulses. This operation has been explained in detail in paragraph 2.3.

The range output, and hence the digital word proportional to the change in range, is calculated every 1/256 sec, while the hardware and the converting equipment operates at 20 kHz. The synchronization between these two is maintained in the software.

The flow chart for the computer program is shown in Figure 3-4. Figure 3-5 shows the response of the servo during "worst case" tracking of a Saturn V launch when error due to acceleration has not been compensated for; while Figure 3-6 shows the same case with the inclusion of compensation for acceleration error. It can be seen that error in range at the output stays well within allowable limits of 0.5 cm.

3.5 COORDINATE CONVERSION AND REFRACTION CORRECTION SUBROUTINE

A detailed explanation of the required coordinate conversions is contained in Appendix D. In the subroutine, the tracking data is converted into rectangular topocentric coordinates at the tracker via Eqs. (D-13) and (D-14). The result is the position vector $\underline{X}_{tt}$. The velocities and accelerations are converted to $\dot{\underline{X}}_{tt}$ and $\ddot{\underline{X}}_{tt}$ by expressions which are directly obtained by differentiation of Eqs. (D-13) and (D-14). The second step is a conversion to rectangular topocentric coordinates at the launch site via Eq. (D-10) which is valid for $\underline{X}_{tt}$, $\dot{\underline{X}}_{tt}$, and $\ddot{\underline{X}}_{tt}$ because $\Delta\underline{r}$ and $\underline{T}_{\ell t}$ are constant (i.e., the launch and tracker sites have a fixed relationship). The final step is to convert from the rectangular topocentric coordinates at the launch site to spherical coordinates via Eq. (D-12). Once again, straightforward differentiation supplies the expressions for obtaining the first and second derivatives of the spherical coordinates from $\underline{X}_{t\ell}$, $\dot{\underline{X}}_{t\ell}$. and $\ddot{\underline{X}}_{t\ell}$, the position, velocity, and acceleration, respectively, in rectangular topocentric coordinates at the launch site. Hence the conversion from tracker data to launch-site-centered data is straightforward. The coded

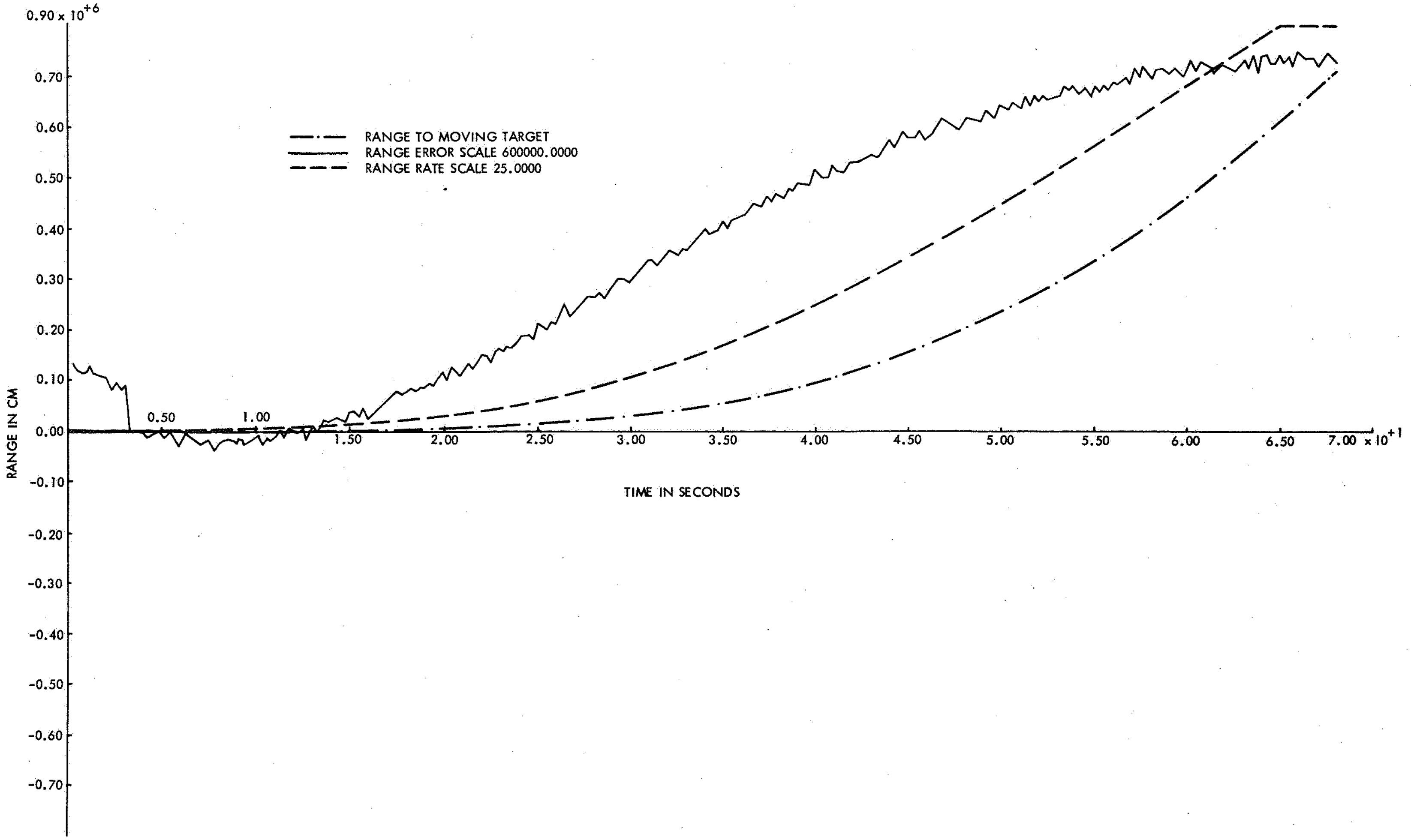


Figure 3-5. Range Servo Response to Worst-Case Tracking of Saturn V Launch (Compensation for Acceleration Error not Included)

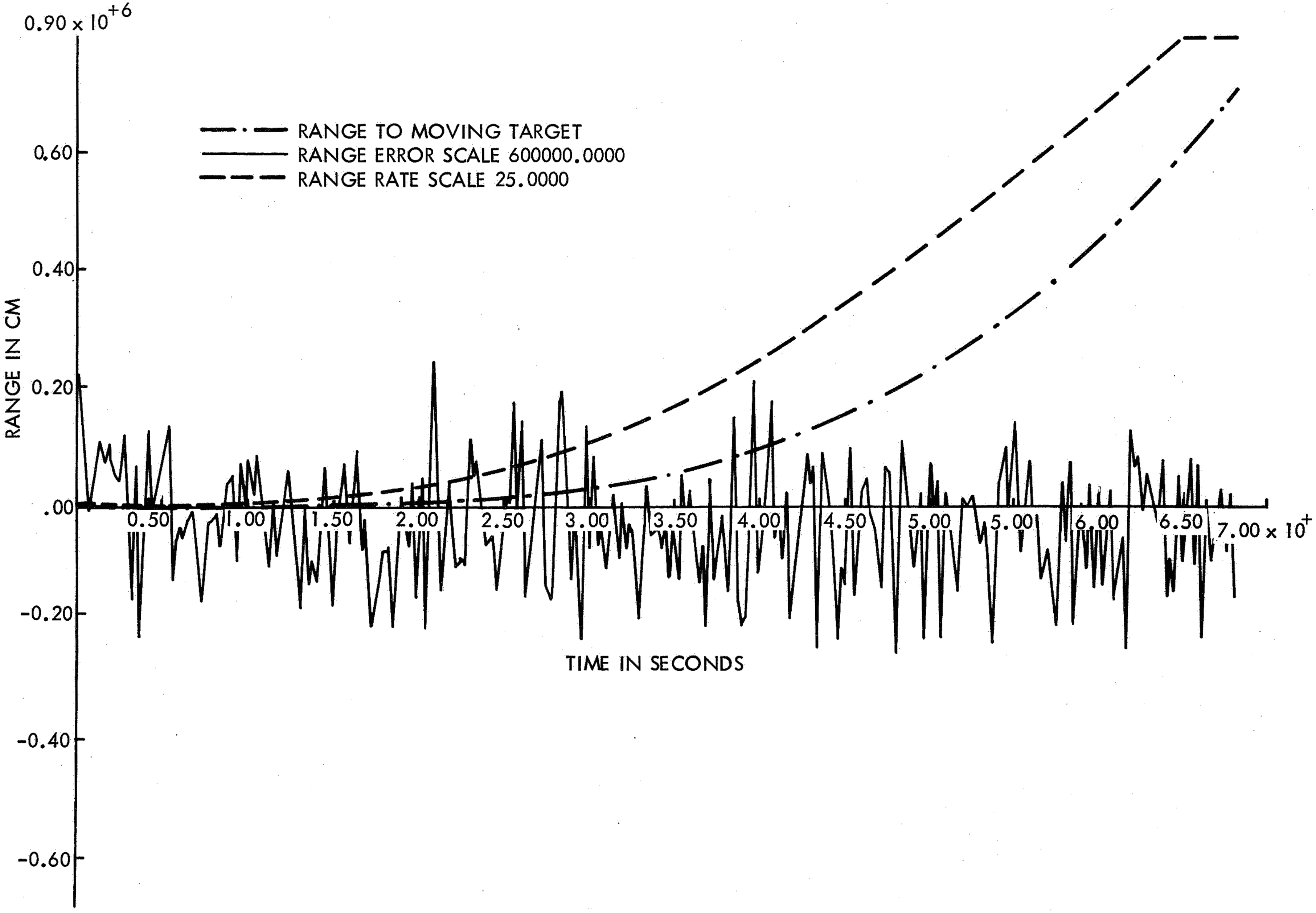


Figure 3-6. Range Servo Response to Worst Case Tracking of Saturn V Launch (Compensation for Acceleration Error Included)

coordinate conversion subroutine was checked out by inputting to it the nominal tracking data (paragraph 3.2).

The atmospheric refraction correction is made at an intermediate step of the coordinate conversion by using $\underline{X}_{tt}$, the measured target coordinates in the rectangular topocentric system, to compute the elevation and azimuth angles. Thus the result of combining Eqs. (D-13) and (D-14) may be written in the alternate form

$$\underline{X}_{tt} = r \begin{bmatrix} \sin \theta_E \\ \cos \theta_E \cos \theta_A \\ \cos \theta_E \sin \theta_A \end{bmatrix} = r \begin{bmatrix} S_1 \\ S_2 \\ S_3 \end{bmatrix} \qquad (3\text{-}1)$$

where θ_E and θ_A are elevation and azimuth in a tracker-centered topocentric system. Only the elevation angle is corrected (range correction is negligible). Letting the correction be $\Delta\theta_E$, the corrected coordinates are

$$\underline{X}'_{tt} \triangleq r \begin{bmatrix} S_1 + \Delta S_1 \\ S_2 + \Delta S_2 \\ S_3 + \Delta S_3 \end{bmatrix} \qquad (3\text{-}2)$$

where

$$\left.\begin{aligned} \Delta S_1 &\triangleq -\Delta\theta_E \cos \theta_E \\ \Delta S_2 &\triangleq \Delta\theta_E \sin \theta_E \cos \theta_A \\ \Delta S_3 &\triangleq \Delta\theta_E \sin \theta_E \sin \theta_A \end{aligned}\right\} \qquad (3\text{-}3)$$

This is a first-order correction valid for $|\Delta\theta_E| < 2 \times 10^{-4}$ radians, a bound which is not exceeded. Empirical fits to standard atmospheric refraction data yield

$$\Delta\theta_E \triangleq \frac{r \cdot B \cdot C \cdot \cos\theta_E}{1 + e^{B \cdot r \cdot \sin\theta_E}} \qquad (3\text{-}4)$$

where

$B = 1/7.61\ (\mathrm{km}^{-1})$

$C = 292 \times 10^{-6}$

r = range in kilometers

3.6 ESTIMATOR-EXTRAPOLATOR SUBROUTINE

Implementation of the triply overlapping polynomial fitting described in paragraph 2.5 requires the devising of an algorithm whereby a single set of estimator equations may be time-shared among the three subunits. This is done so as to eliminate triplication of the estimator equations and thus conserve memory space; an additional advantage is the construction of an algorithm which could be used for any number of subunits if desired. In addition, a strategy must be developed for the extrapolator operation during the reacquisition phase. These matters are discussed in paragraph 3.6.1

Using the nominal tracking data (see paragraph 3.2) and a noise generator, the estimator operation was simulated. A discussion of the simulation and the results is found in paragraph 3.6.2.

3.6.1 Software Description

The equations used during the batch and sequential estimation phases are developed in Appendix E. It should be noted that the computation during the batch processing phase amounts to the multiplication of a measurement vector by a matrix. In the computer program this matrix-vector multiplication is done piecewise by multiplying one column vector of the matrix and the corresponding element of the measurement vector, as this element of the measurement vector becomes available. It is possible to spread the batch processing computation over the entire batch processing interval since the parameter vector estimate generated is not used until the end of the batch processing interval. The computation during the sequential processing phase is similarly straightforward, consisting of the evaluation of a cubic polynomial to get the predicted measurement, and the modification of the parameter vector estimate by adding the current gain vector multiplied by the difference between the actual and predicted measurements. Hence the software for the actual mathematics of the estimator is simple and straightforward. The bulk of the programming is due to time-sharing of the estimator mathematics among the three subunits, using the same estimator for range, hour angle, and declination, and implementing an extrapolation function for use during the system reacquisition mode. The flow diagrams are given in Figures 3-7 and 3-8; a detailed FORTRAN listing is in Appendix A.

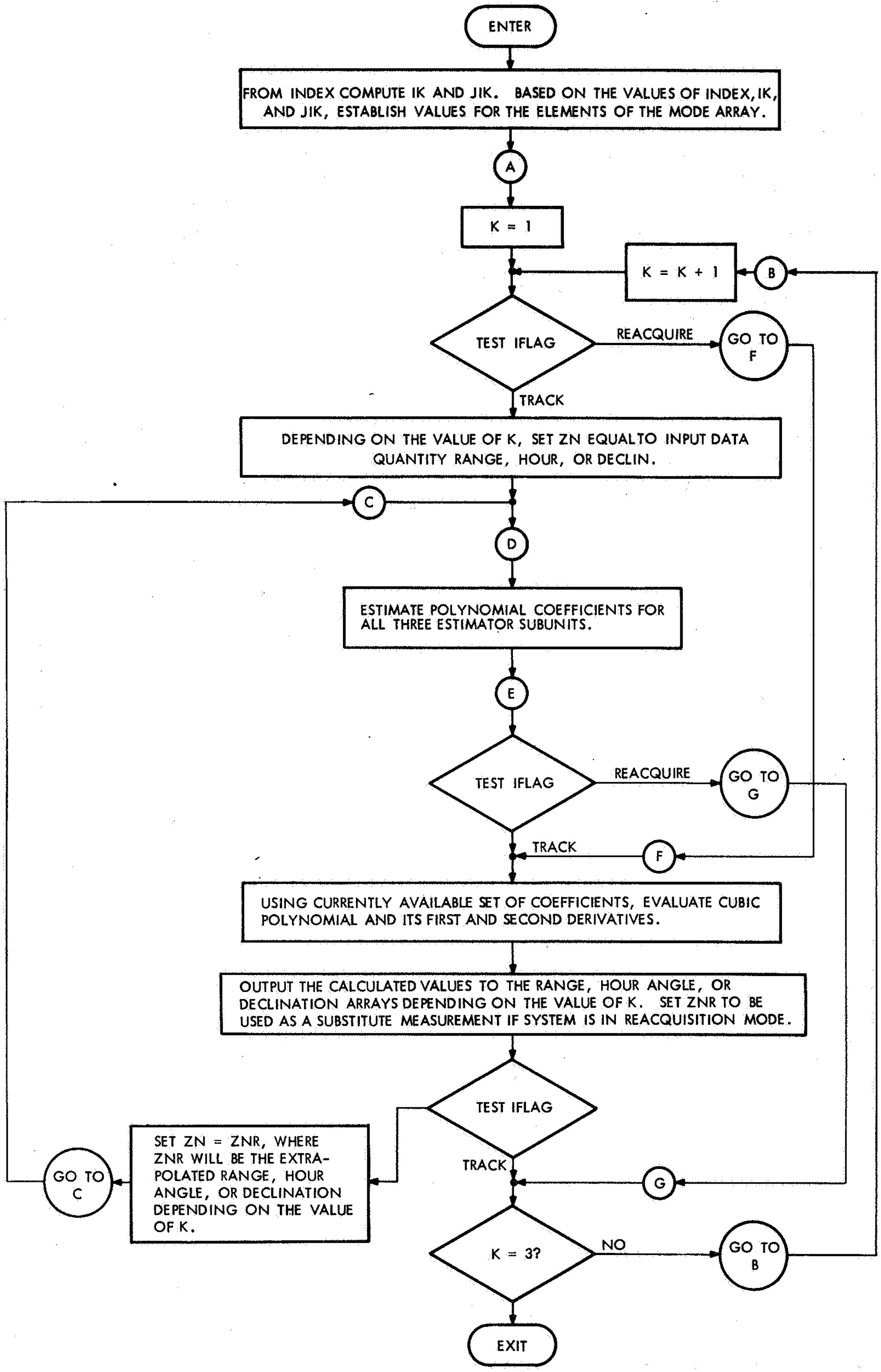

ENTER
FROM INDEX COMPUTE IK AND JIK. BASED ON THE VALUES OF INDEX, IK, AND JIK, ESTABLISH VALUES FOR THE ELEMENTS OF THE MODE ARRAY.
A
K = 1
K = K + 1
B
TEST IFLAG
REACQUIRE
GO TO F
TRACK
DEPENDING ON THE VALUE OF K, SET ZN EQUALTO INPUT DATA QUANTITY RANGE, HOUR, OR DECLIN.
C
D
ESTIMATE POLYNOMIAL COEFFICIENTS FOR ALL THREE ESTIMATOR SUBUNITS.
E
TEST IFLAG
REACQUIRE
GO TO G
TRACK
F
USING CURRENTLY AVAILABLE SET OF COEFFICIENTS, EVALUATE CUBIC POLYNOMIAL AND ITS FIRST AND SECOND DERIVATIVES.
OUTPUT THE CALCULATED VALUES TO THE RANGE, HOUR ANGLE, OR DECLINATION ARRAYS DEPENDING ON THE VALUE OF K. SET ZNR TO BE USED AS A SUBSTITUTE MEASUREMENT IF SYSTEM IS IN REACQUISITION MODE.
TEST IFLAG
SET ZN = ZNR, WHERE ZNR WILL BE THE EXTRAPOLATED RANGE, HOUR ANGLE, OR DECLINATION DEPENDING ON THE VALUE OF K.
GO TO C
TRACK
G
K = 3?
NO
GO TO B
EXIT

Figure 3-7. Functional Flow Diagram for Estimator--Extrapolator Subroutine

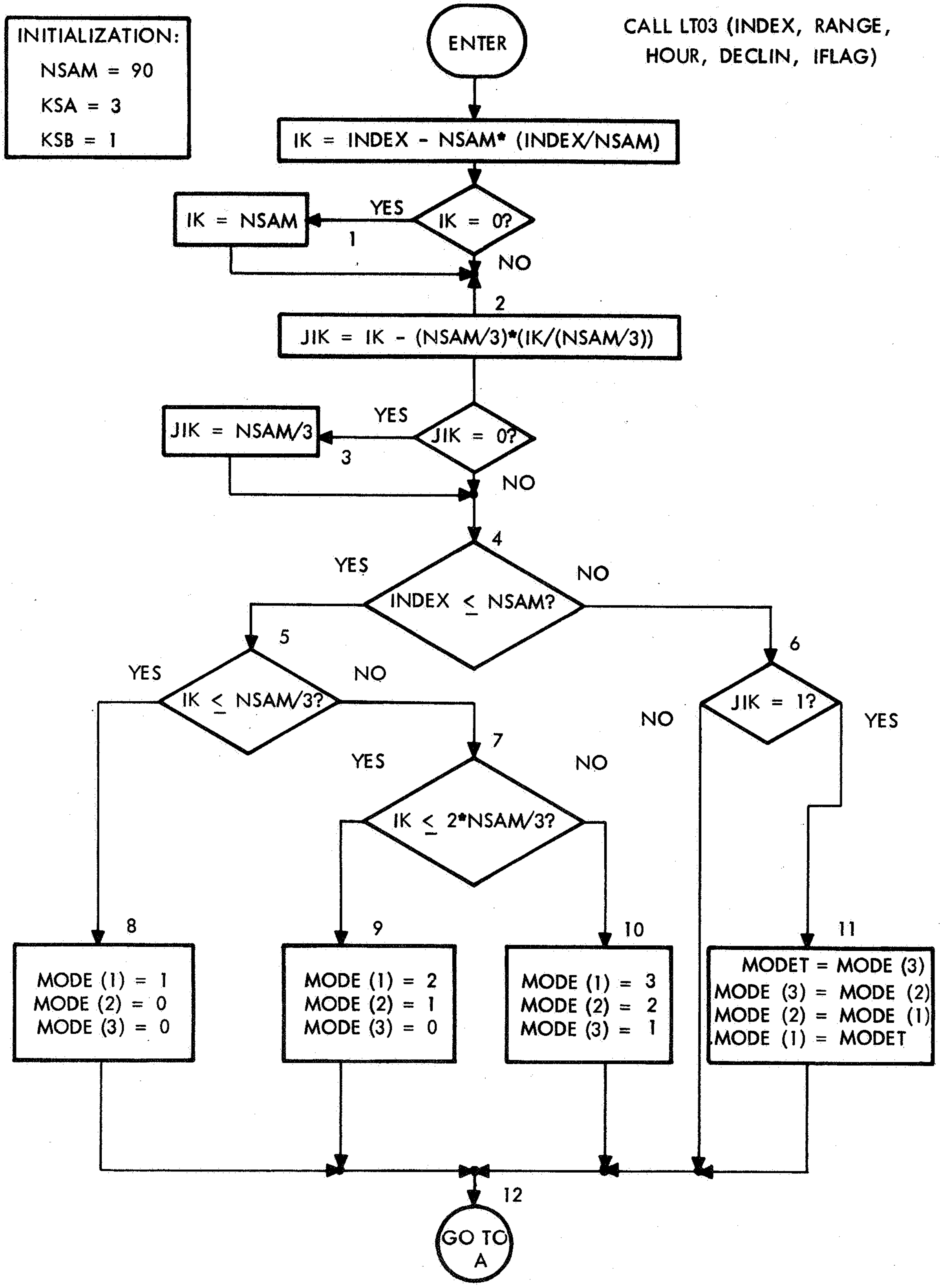


Figure 3-8. FORTRAN Flow Diagram for Estimator-Extrapoltor Subroutine (Sheet 1 of 4)

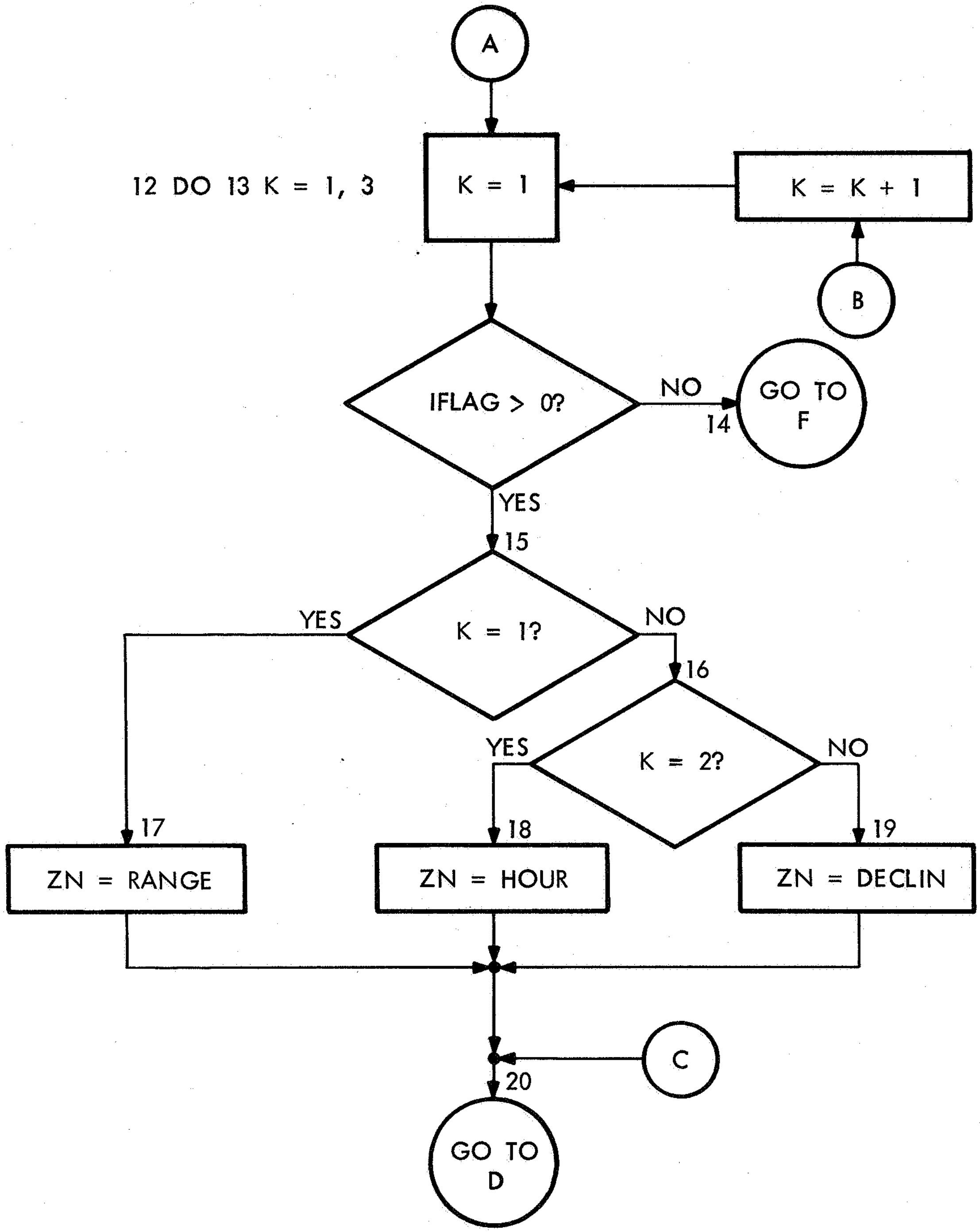


Figure 3-8. FORTRAN Flow Diagram for Estimator-Extrapolator Subroutine (Sheet 2 of 4)

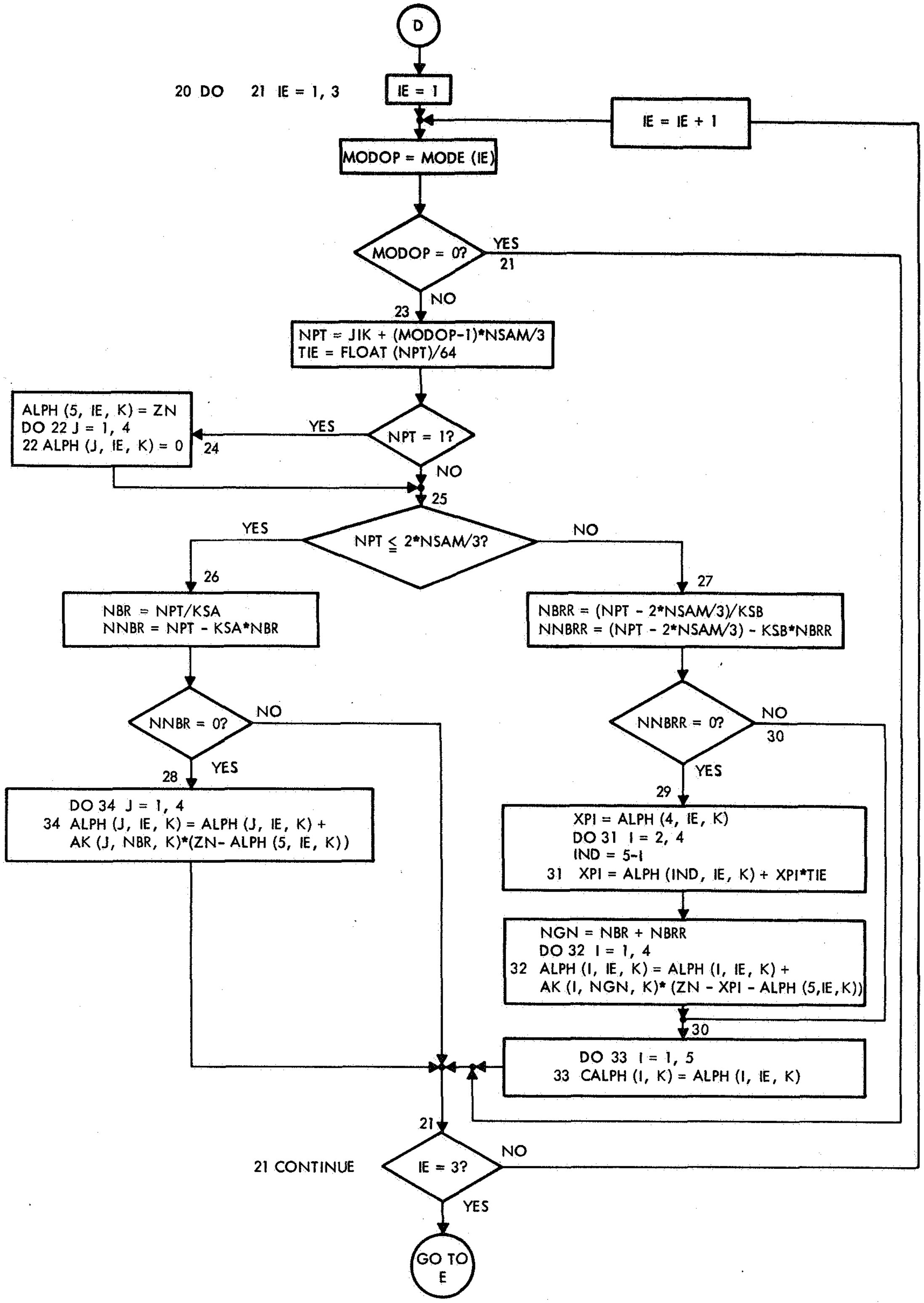


Figure 3-8. FORTRAN Flow Diagram for Estimator-Extrapolator Subroutine (Sheet 3 of 4)

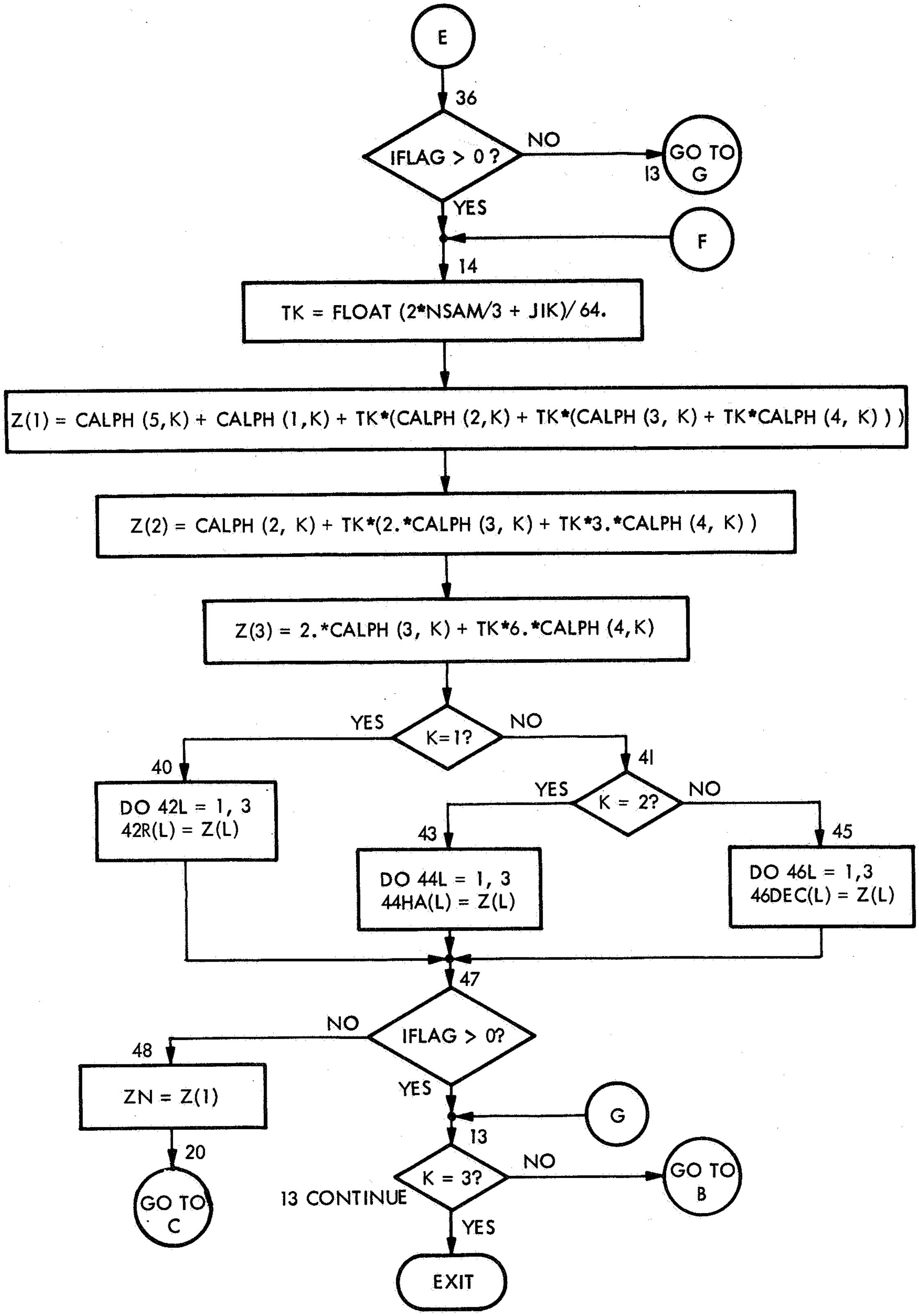

E
36
IFLAG > 0 ?
NO
13
GO TO G
YES
F
14
TK = FLOAT (2*NSAM/3 + JIK)/ 64.
Z(1) = CALPH (5,K) + CALPH (1,K) + TK*(CALPH (2,K) + TK*(CALPH (3, K) + TK*CALPH (4, K)))
Z(2) = CALPH (2, K) + TK*(2.*CALPH (3, K) + TK*3.*CALPH (4, K))
Z(3) = 2.*CALPH (3, K) + TK*6.*CALPH (4,K)
YES
K=1?
NO
40
DO 42L = 1, 3
42R(L) = Z(L)
41
YES
K = 2?
NO
43
DO 44L = 1, 3
44HA(L) = Z(L)
45
DO 46L = 1,3
46DEC(L) = Z(L)
47
NO
IFLAG > 0?
48
ZN = Z(1)
YES
G
20
GO TO C
13
K = 3?
NO
GO TO B
13 CONTINUE
YES
EXIT

Figure 3-8. FORTRAN Flow Diagram for Estimator-Extrapolator Subroutine (Sheet 4 of 4)

In paragraph 4.1 it will be noted that the present FORTRAN version of the estimator-extrapolator subroutine, LT03, has been timed at 17 milliseconds on the CDC-3200. The major reason for this long running time is the use of multiply-subscripted arrays; inspection of the assembly language program generated by the FORTRAN 3200 compiler discloses the large amount of arithmetic computation being done in order to convert, say, a triple subscript on a variable to a single address modifier number which then must be loaded into an index register before the actual program computation involving that triply subscripted variable can be done. A relatively conservative estimate of the time which is consumed in this subscript conversion is 6 milliseconds; hence actual computation within the LT03 subroutine consumes less than 11 milliseconds; how much less is not easily determined because of the time spent in indexing the DO loops on the subscripts plus the time spent in branching calculations necessitated by sharing of the estimator equations. However, an estimate of 9 milliseconds on the CDC-3200 for the estimator computations alone is, again, adequately conservative. On the PDP-10 or Sigma 5, the running time would be 6 milliseconds, within allowances for real-time operation (refer to paragraph 4.1). The key conclusion now is that the estimator would run fast enough if no subscripting, and hence no time-sharing of estimator equations, were allowed. To achieve this would require the triplicating of all groups of computation instructions. Again based on the CDC-3200 compiler results, the memory requirements for the estimator subroutine LT03 would increase from 685 words to 2661 words. Thus, halving the running time requires quadrupling the memory requirements.

Because of the above remarks it is likely that the subscripting presently in LT03 will be reduced by simply repeating instruction groups now indexed by a DO loop parameter. A detailed procedure for accomplishing this elimination of subscripts is given in Appendix E. Accordingly, the discussion in this section of the detailed flow diagram, Figure 3-8, will be abbreviated. Primarily the discussion will refer to the more functional flow diagram of Figure 3-7.

Note that in Figure 3-7 there is only one loop detailed, that one indexed by K. As K ranges from one to three, the estimator subroutine performs its computations for range, hour angle, and declination angle, respectively. The role of K is readily apparent from the detailed branching in Figure 3-8 (b). In the initial block in Figure 3-7, INDEX is a running index obtained from the main program and denotes the number of 1/64 second intervals which have elapsed since the main program first entered the track mode. IK and JIK are two counters derived from INDEX and range, respectively, from 1 to 90 and from 1 to 30. The IK counter is used only during the startup period, i.e., when the first ninety samples are being processed. The JIK counter is used continuously to control the estimator subunits and to select the proper one of the three sets of coefficients for input to the extrapolator section (see Figure 3-8(d)). After the initial block, the diagram in Figure 3-7 is largely self-explanatory except for the branching associated with the testing of IFLAG.

IFLAG within the LT03 subroutine corresponds to MODE in the main program; MODE values of -1, 0, +1 indicate, respectively, that the system is in the reacquire, acquire, or track mode. There is no estimator function during acquisition; during tracking the estimator operation is detailed in Figure 3-7 by the straight through flow of the diagram. The alternate path indicated in Figure 3-7 as that followed during reacquisition is based upon the following design philosophy.

It is assumed that the transition from one mode to another can occur only at the beginning of a 1/64 second interval. Hence the estimator will have finished processing, in the track mode, the last measurement before IFLAG indicates a transition to the reacquisition mode. When the system operation switches to target reacquisition, the estimator subroutine flow branches to F, which is the entry point for the evaluation of the fitting polynomials for range, hour angle, and declination angle. Since the current sets of coefficients are based on the previous tracking measurements, the net result is that of extrapolating forward by 1/64 second. If the flow now were to be an exit from the subroutine then all future entrys into the subroutine, while the system is still in the reacquire mode, would yield further extrapolations using the same set of polynomial coefficients. Such a procedure, however, is not advisable when extrapolating from a polynomial fit, except for extrapolation periods that are short compared with the fitting period. The reason is simply that beyond the region of fit the polynomial has no more "zero-crossings" relative to the data trend curve and its value will deviate arbitrarily far from the data trend curve. The avoidance of this problem is a simple matter: the extrapolated values are fed back into the estimator and used to modify the polynomial coefficients. In this way the extrapolated values are treated as measurements, the "zero-crossing" region of the fitting polynomials moves forward in time, and the extrapolated data trend will not deviate arbitrarily far from the actual data trend. In Figure 3-7 this strategy is indicated by the branching, after F, back to point C (to enter the estimator) and then to point G.

3.6.2 Simulation Results

The estimator performance has been simulated by processing measurements obtained from the nominal tracking data and simulated measurement noise. The nominal tracking data is discussed in paragraph 3.2, and the measurement noise model in Appendix E. The results of the simulation runs are presented in Table 3-4 and in Figures 3-9 to 3-14. The simulation process will be described first, followed by a discussion of the estimator performance.

Tracking data for range, hour angle, and declination were obtained by evaluating, at 1/64 second intervals, the sixth-degree polynomials (see paragraph 3.2) and adding to these values a measurement noise quantity (see Appendix E). The simulation covered a total tracking interval of approximately 69 seconds; the tracking interval was actually composed of 49 subintervals of 90 samples each for a total of 4410 measurement points. In addition, each 90-measurement interval was broken into three 30-measurement intervals, for each of which the rms estimation was evaluated. That is, since each estimator subunit is in the sequential estimation phase for 30/64 second, the estimation errors during that 30-measurement interval were combined to form an rms estimation error for the interval. The estimation errors were obtained by forming the difference between the position, velocity, and acceleration estimates and the true position, velocity, and acceleration as obtained by evaluating the nominal tracking data polynomials and their first and second derivatives. For comparison, the rms value of the measurement

TABLE 3-4

ESTIMATOR PERFORMANCE AS A FUNCTION OF MEASUREMENT AND NOISE PARAMETERS FOR FULL TRACKING PERIOD (0 to 68 SECONDS)

Data Quantity	Noise Correlation Time Constant (Second)	Measured RMS Noise	Measured RMS Position Error	Measured RMS Velocity Error	Measured RMS Acceleration Error
Range (cm, cm/sec, cm/sec^2)	1/14.76	1.04 2.06 2.96	1.05 2.13 2.94	8.87 17.3 24.2	39.0 75.5 104.0
	1/29.52	0.986 1.95 3.08	0.838 1.59 2.58	7.12 13.6 21.6	31.5 60.0 94.3
	1/44.28	1.0 2.0 3.0	0.764 1.51 2.17	6.48 12.9 18.1	28.7 57.9 80.1
Hour Angle (sec, sec/sec, sec/sec^2)	1/14.76	1.0 2.0 3.0	0.985 2.02 2.89	8.03 16.7 23.3	35.0 72.8 102.0
	1/29.52	1.0 2.0 3.0	0.827 1.61 2.41	6.88 13.7 20.2	29.8 60.2 89.2
	1/44.28	1.0 2.0 3.0	0.736 1.49 2.14	6.04 12.7 18.5	26.5 56.5 82.9
Declination (sec, sec/sec, sec/sec^2)	1/14.76	1.0 2.0 3.0	0.983 2.08 2.98	8.11 17.6 24.5	35.0 76.6 107.0
	1/29.52	1.0 2.0 3.0	0.852 1.66 2.42	7.19 13.3 20.6	31.7 57.0 91.4
	1/44.28	1.0 2.0 3.0	0.720 1.49 2.17	6.15 12.8 18.5	27.5 56.6 81.5

Note: Units of quantities in table are arc seconds, centimeters, and seconds.

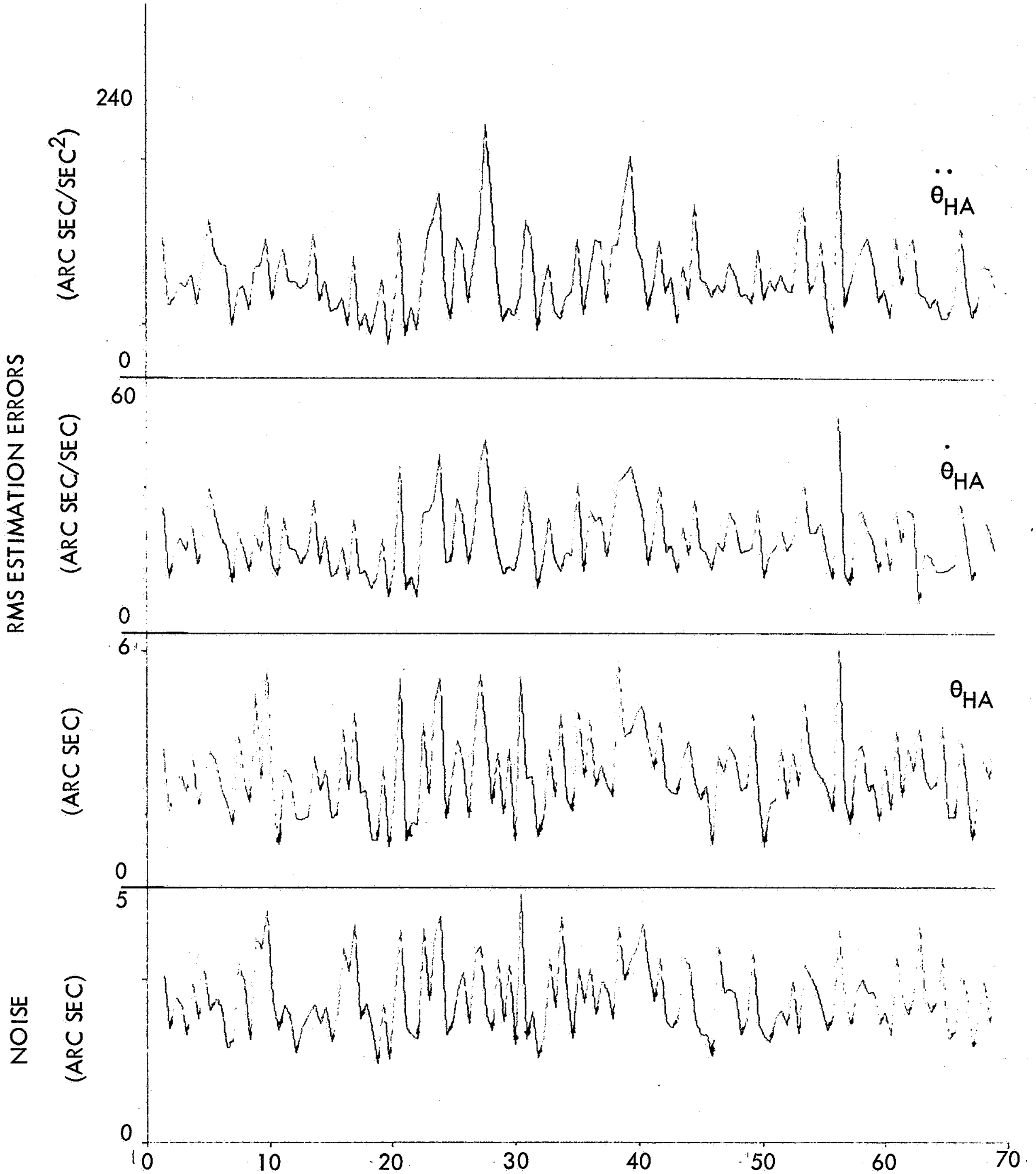


Figure 3-9. Estimator Simulation for Hour Angle (Sigma = 3, Tau = 14.76)

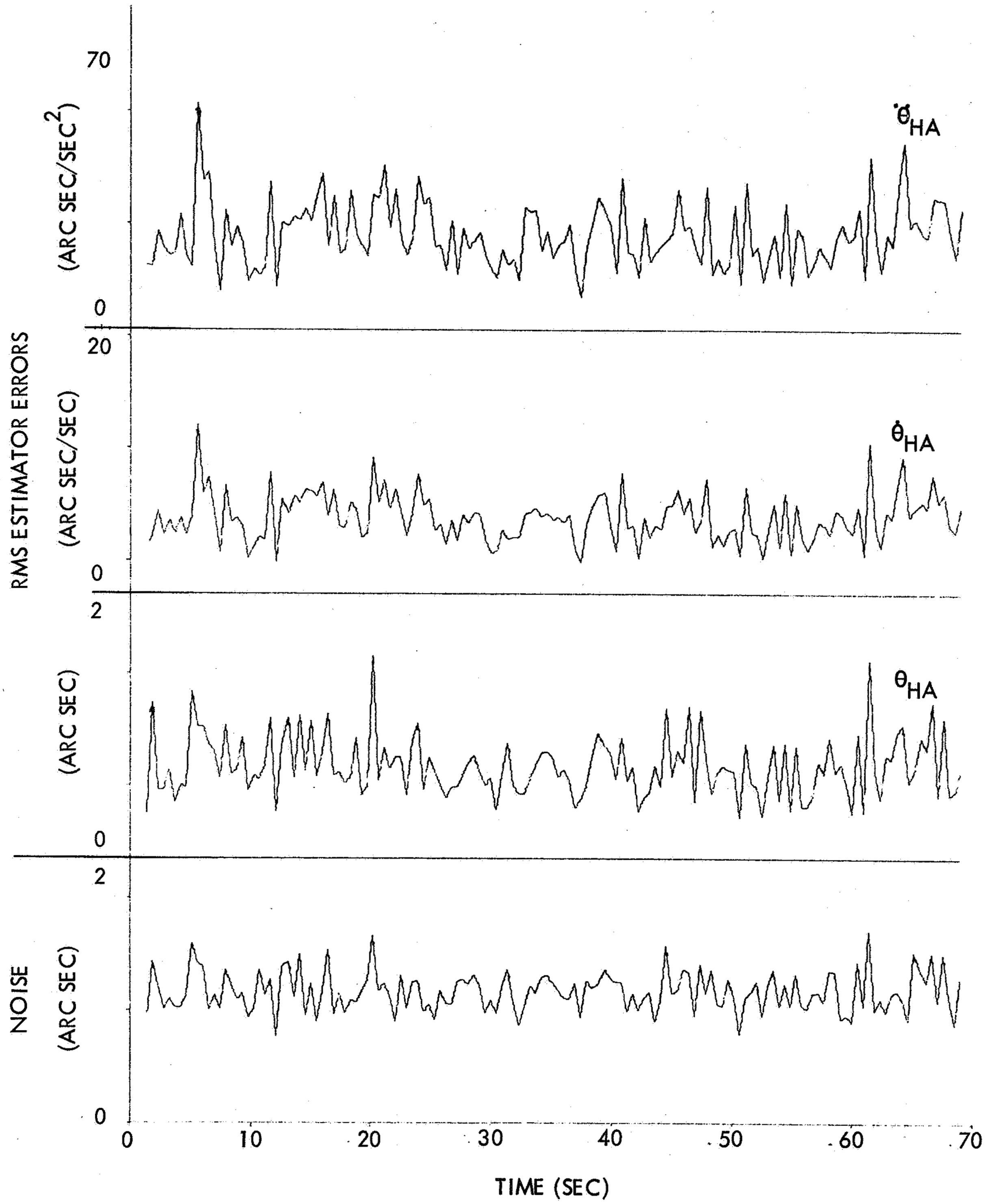


Figure 3-10. Estimator Simulation for Hour Angle (Sigma = 1, Tau = 44.28)

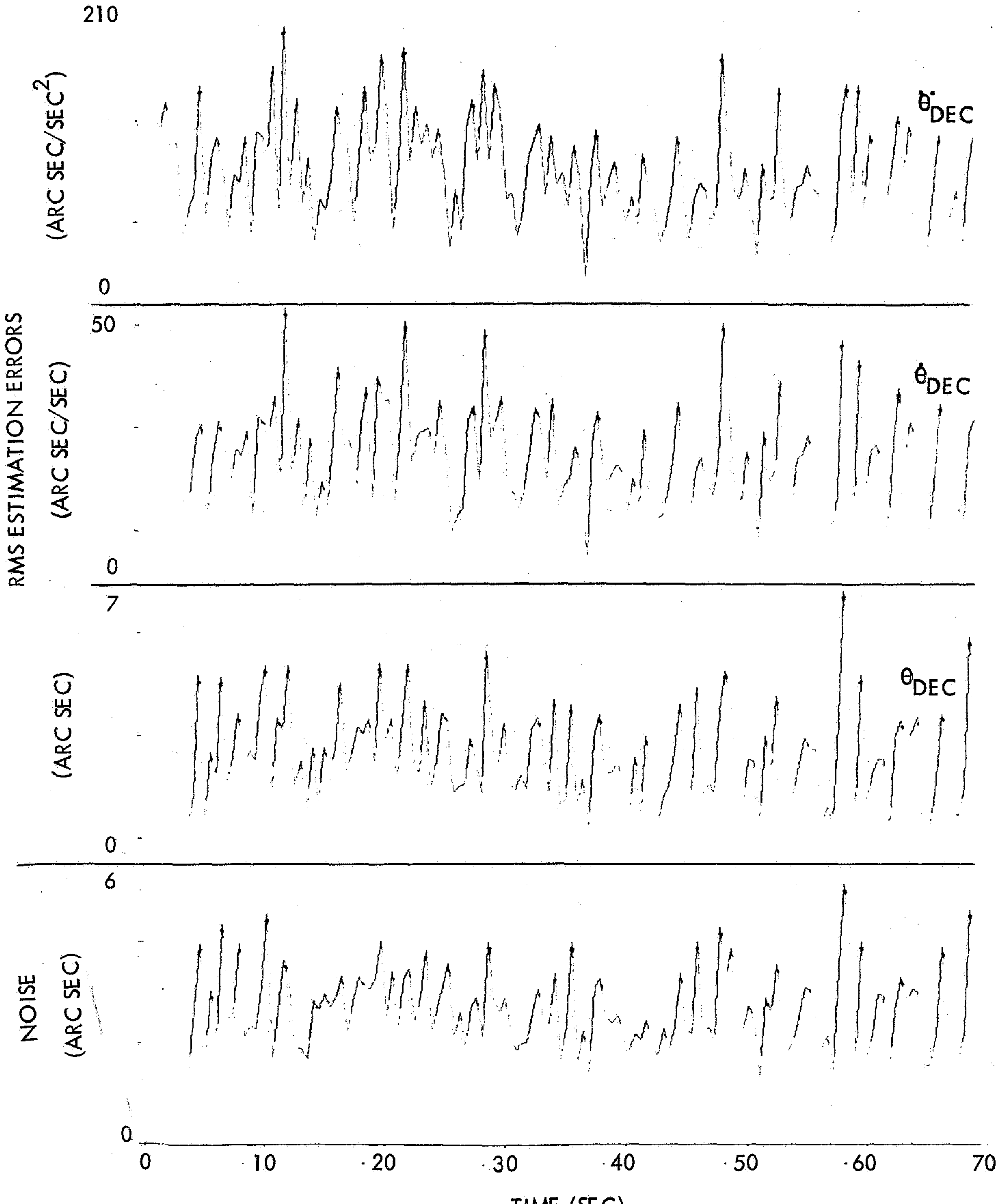


Figure 3-11. Estimator Simulation for Declination (Sigma = 3, Tau = 14.76)

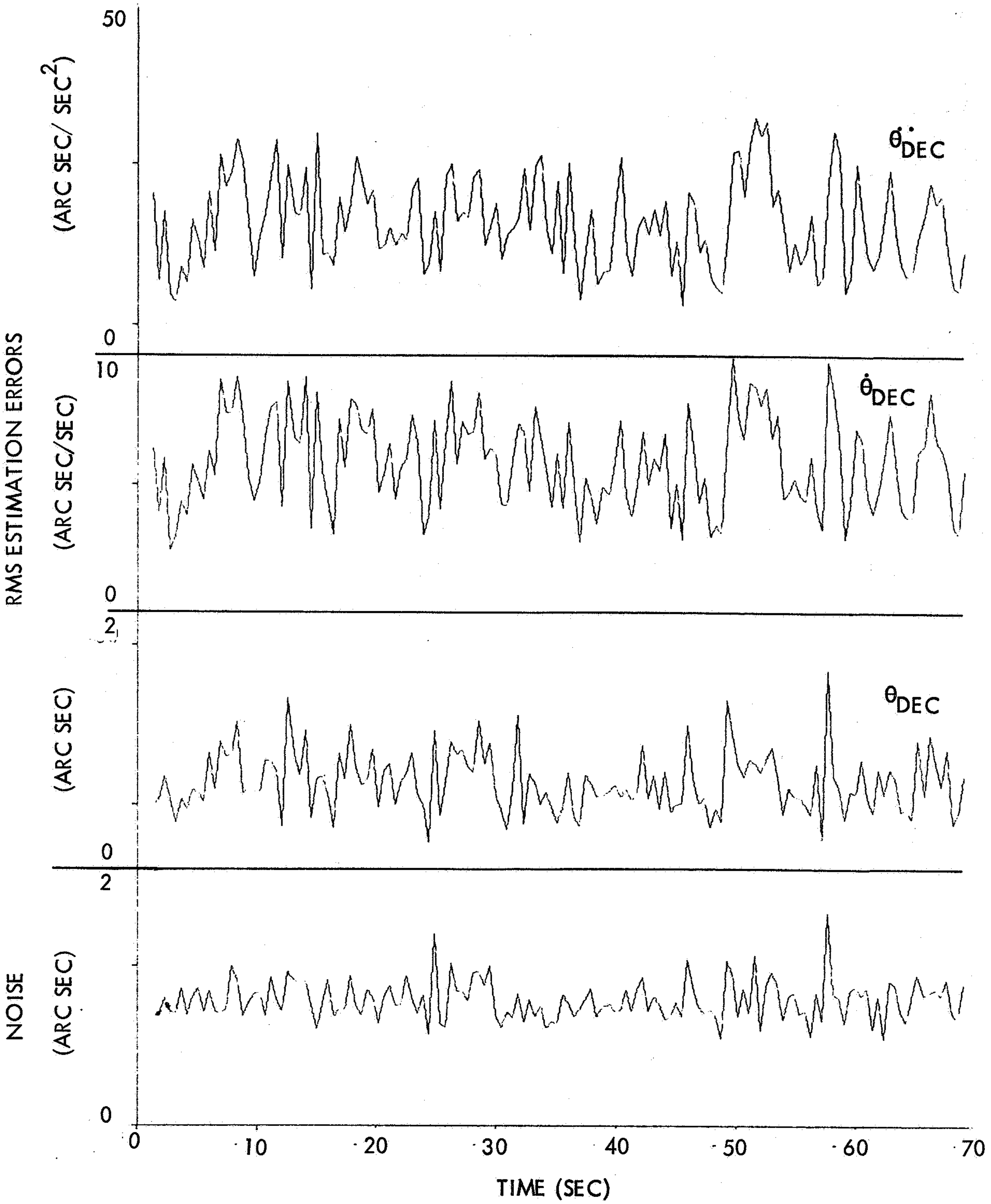


Figure 3-12. Estimator Simulation for Declination (Sigma = 1, Tau = 44.28)

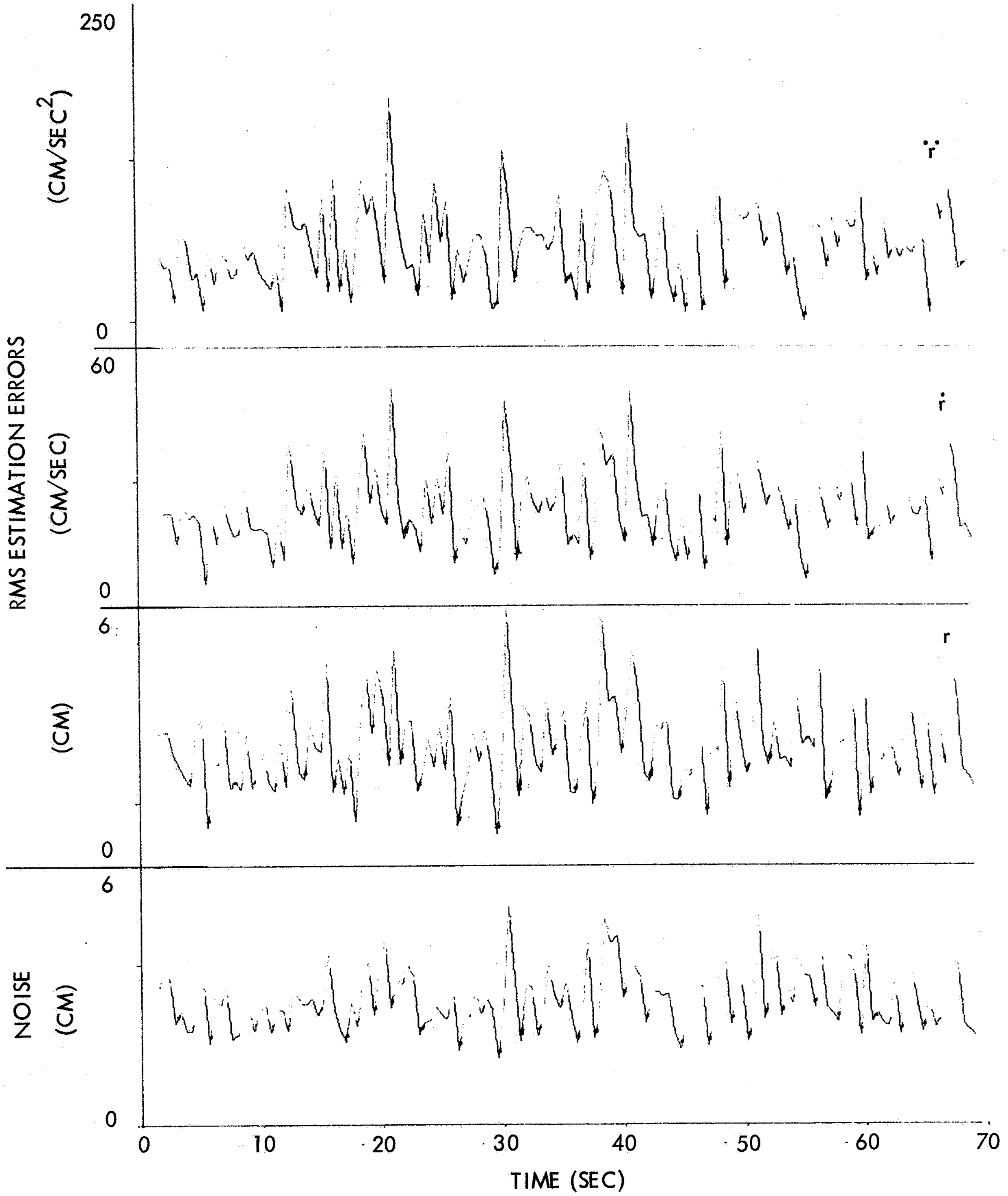


Figure 3-13. Estimator Simulation for Range (Sigma = 3, Tau = 14.76)

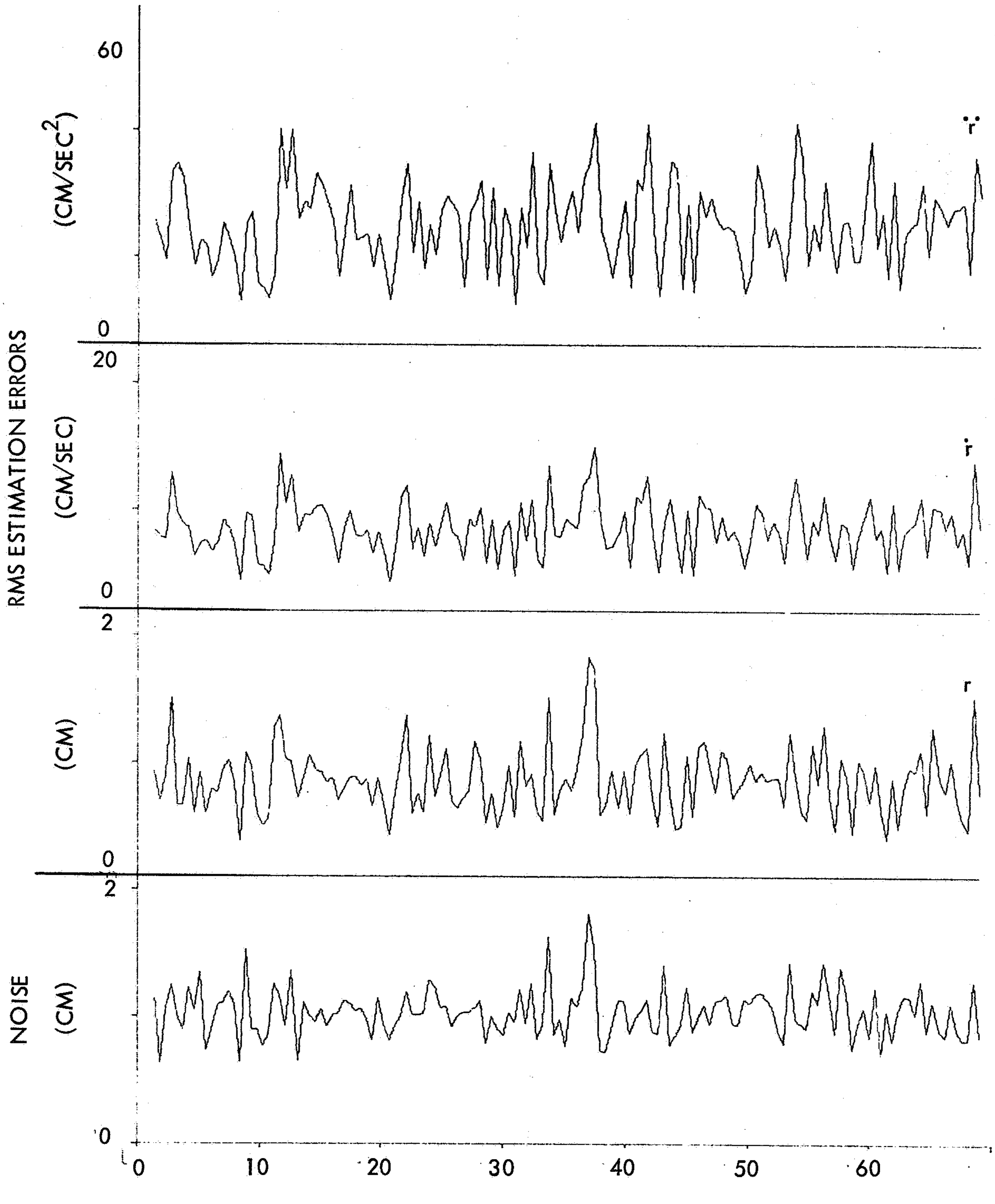


Figure 3-14. Estimator Simulation for Range (Sigma = 1, Tau = 44.28)

noise samples for each 30-sample interval was computed. The results of the simulation runs, for different measurement noise model assumptions, are in Figures 3-9 to 3-14. Note that the continuous curves were obtained by joining points spaced at 30/64 second intervals. As described in Appendix E, the measurement noise model is a stationary random process with autocorrelation function

$$\rho(t) = \sigma^2 e^{-\tau|t|} \qquad \left(\text{unit}^2\right)$$

and power spectral density

$$\Phi(j\omega) = \frac{2\sigma^2 \tau}{\tau^2 + \omega^2} \qquad \left(\text{unit}^2/\text{Hz}\right)$$

This power spectral density matches the 1/f trend observed in experimental determinatons of angle fluctuation spectra (see Appendix F for a treatment of the theoretical and experimental results on atmosphere-induced measurement noise). The estimation error curves were obtained for different σ and τ values. Increasing σ values indicate an increasing noise level, but increasing τ values indicate a decreasing correlation time. It is clear that decreasing σ and increasing τ will decrease the estimation errors.

The 147 sets of rms estimation errors on 30-sample intervals were also processed to yield rms estimation errors for the entire tracking interval. These total rms errors are arrayed in Table 3-4 as a function of σ and τ. From this table it is easy to see the extent to which the estimation errors are influenced by the measurement noise characteristics.

The values of $\sigma = 3$ and $\tau = 14.76$ correspond to the anticipated worst case conditions, when the elevation angle of the laser beam is less than 10 degrees above the local horizon. The actual noise conditions may be less severe. In any event, as the launch vehicle rises, σ will decrease significantly (see Appendix F, paragraph 3.1) and τ is expected to increase. The reasons are, simply, that the encountered turbulence is less severe at moderate elevation angles (σ decreases) and is less correlated (τ increases) as the target range increases. Hence it is definitely expected that the estimation errors will decrease as time from liftoff increases.

Prediction of the way in which the measurement noise will change with elevation angle and target range is a difficult task at present. As indicated in Appendix F, Hodara predicts that σ^2 will decrease exponentially with increasing altitude; the experimental results of Kurtz and Hayes indicate that τ will increase roughly proportionally to target range. However, experimental results for the tracter site will be required before any reasonable prediction of σ and τ variations can be made.

SECTION 4

SYSTEM SOFTWARE IMPLEMENTATION

4.1 REAL-TIME ASPECTS OF SOFTWARE OPERATION

As noted in paragraph 2.6 the real-time system implementation could be accomplished by use of either a PDP-10 or Sigma 5 computer system. The simulation effort (see Section 3) was aimed primarily at designing the various algorithms so as to ensure satisfactory system performance in terms of tracking accuracy requirements, and to guarantee interfacing compatibility among the several subroutines. Running time estimates were then made for each principal subroutine in order that real-time operation of the system program would be assured. The process used in the determination of these estimates will now be described.

Each of the principal subroutines involved in the system simulation effort was timed separately on the CDC-3200 by running the subroutine ten thousand times and noting the elapsed time by means of the computer's real-time clock. The resulting average running times are given in Table 4-1. Also included in the table is a projection for the running time on a PDP-10. The basis of this projection is the fact that arithmetic operation speeds for the PDP-10 are approximately one-third greater than those for the CDC-3200. Neglected is the fact that the PDP-10 is, in reality, a much more efficient machine than the CDC-3200, in that it has sixteen fast registers and a repertoire of 385 instructions. Thus it is expected that the realizable running times are, in fact, less than those indicated. The column labeled "perentage computation time" accounts for the percentage of time that must be devoted to each subroutine per clock cycle (1/64 sec). Note that the range servo operates at a rate of 256 sec^{-1} and thus operates four times in every clock cycle.

The overall system flow chart appears in Figure 4-1. This flow chart includes the program initialization and calibration steps necessary for proper program implementation. The key subroutines tested in the system simulation are labeled. Input and output quantities for each of these subroutines are given in Figure 4-2. Each of the subroutines is currently FORTRAN coded.

In implementing the real-time program it is not expected that any of the FORTRAN-coded subroutines need be converted to machine-language coding. In particular this is true for the PDP-10 system whose FORTRAN compiler has, on other projects at ARL, generated extremely tight machine coding, so that less than 10 percent reduction in running time can be achieved by recoding in assembly language.

TABLE 4-1

RUNNING TIMES FOR PRINCIPAL SUBROUTINES

Subroutine	Measured (CDC-3200)	Projected (PDP-10)	Percentage
Angle Servo	1.37 millisec	0.9 millisec	5.8%
Range Servo	1.035 millisec	0.675 millisec	17.3%
Coordinate Conversion	11.0 millisec	4.0 millisec	25.6%
*Estimator/Extrapolator	11.0 millisec	7.0 millisec	48%

*See paragraph 3.6 for detailed discussion of timing estimate for this subroutine.

4.2 SOFTWARE FOR OFF-LINE DATA REDUCTION

The real-time data processing is restricted to a cubic polynomial fit to a 90/64 second data span. In off-line processing no such restriction is necessary. FORTRAN programs have been provided which can accomplish weighted least-squares fitting of higher degree polynomials to longer data spans. These programs, in subroutine form, are described and listed in Appendix A.

The major subroutine is POLFITW, which accomplishes the weighted least-squares polynomial fit with the aid of GINV (for generalized inverse). CHOLSQR (Cholesky decomposition and inversion of lower triangular matrix), R (yielding the correlation, or weighting, matrix) and POLORT which orthogonalizes cubic polynomials on a particular time interval. The theory behind the use of these programs is in Appendix E.

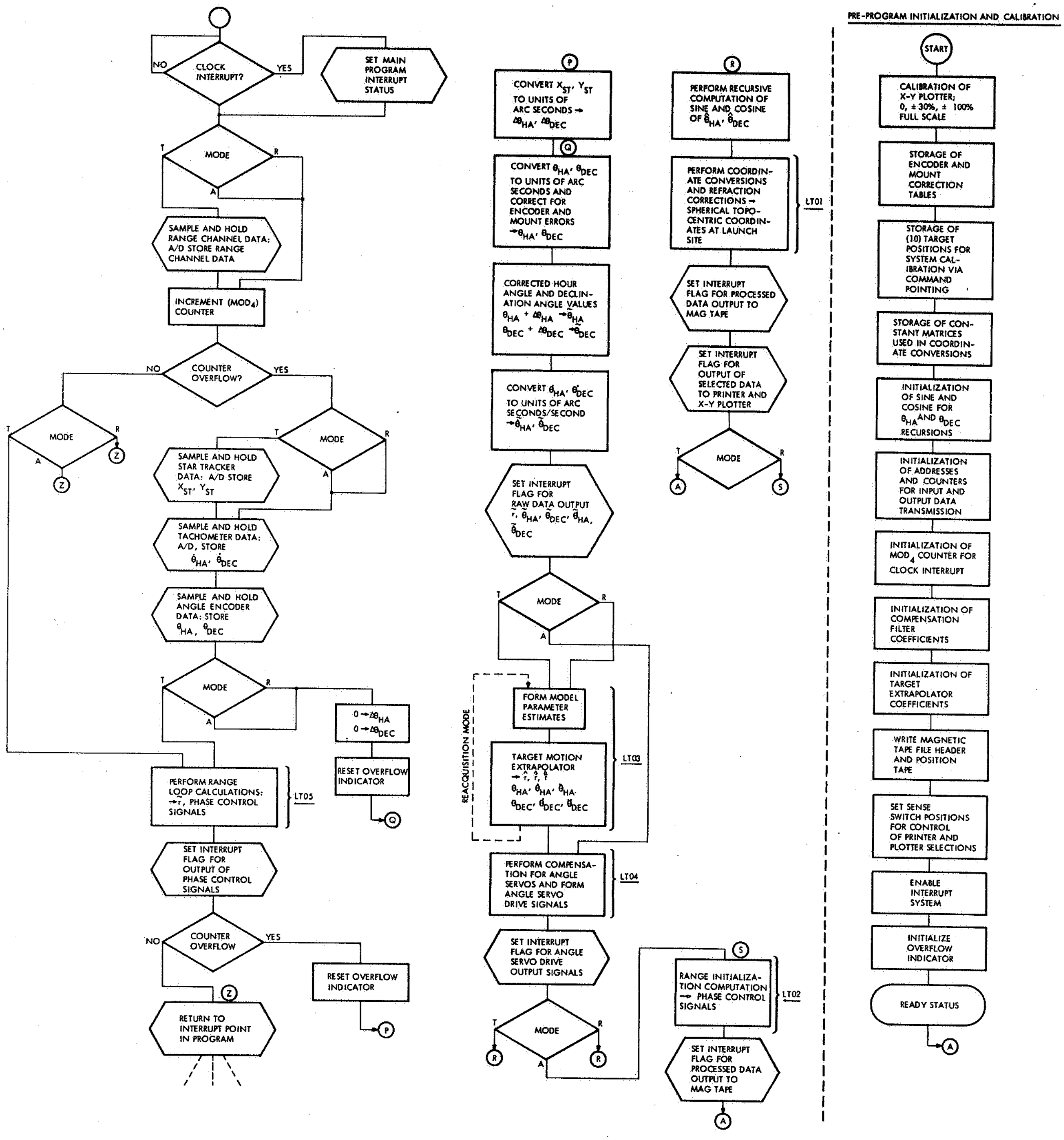

CLOCK INTERRUPT?
NO
YES
SET MAIN PROGRAM INTERRUPT STATUS
MODE
T
R
A
SAMPLE AND HOLD RANGE CHANNEL DATA: A/D STORE RANGE CHANNEL DATA
INCREMENT (MOD$_4$) COUNTER
COUNTER OVERFLOW?
NO
YES
MODE
T
R
A
Z
Z
MODE
T
R
A
SAMPLE AND HOLD STAR TRACKER DATA: A/D STORE X_{ST}, Y_{ST}
SAMPLE AND HOLD TACHOMETER DATA: A/D, STORE $\dot{\theta}_{HA}$, $\dot{\theta}_{DEC}$
SAMPLE AND HOLD ANGLE ENCODER DATA: STORE θ_{HA}, θ_{DEC}
MODE
T
R
A
$0 \rightarrow \Delta\theta_{HA}$
$0 \rightarrow \Delta\theta_{DEC}$
RESET OVERFLOW INDICATOR
Q
PERFORM RANGE LOOP CALCULATIONS: $\rightarrow \tilde{r}$, PHASE CONTROL SIGNALS
LT05
SET INTERRUPT FLAG FOR OUTPUT OF PHASE CONTROL SIGNALS
COUNTER OVERFLOW
NO
YES
RESET OVERFLOW INDICATOR
P
Z
RETURN TO INTERRUPT POINT IN PROGRAM
P
CONVERT X_{ST}, Y_{ST} TO UNITS OF ARC SECONDS $\rightarrow$ $\Delta\theta_{HA}$, $\Delta\theta_{DEC}$
Q
CONVERT θ_{HA}, θ_{DEC} TO UNITS OF ARC SECONDS AND CORRECT FOR ENCODER AND MOUNT ERRORS $\rightarrow \theta_{HA}$, θ_{DEC}
CORRECTED HOUR ANGLE AND DECLINATION ANGLE VALUES
$\theta_{HA} + \Delta\theta_{HA} \rightarrow \tilde{\theta}_{HA}$
$\theta_{DEC} + \Delta\theta_{DEC} \rightarrow \tilde{\theta}_{DEC}$
CONVERT $\dot{\theta}_{HA}$, $\dot{\theta}_{DEC}$ TO UNITS OF ARC SECONDS/SECOND $\rightarrow \tilde{\dot{\theta}}_{HA}$, $\tilde{\dot{\theta}}_{DEC}$
SET INTERRUPT FLAG FOR RAW DATA OUTPUT $\tilde{r}$, $\tilde{\theta}_{HA}$, $\tilde{\theta}_{DEC}$, $\tilde{\dot{\theta}}_{HA}$, $\tilde{\dot{\theta}}_{DEC}$
MODE
T
R
A
REACQUISITION MODE
FORM MODEL PARAMETER ESTIMATES
TARGET MOTION EXTRAPOLATOR $\rightarrow \hat{r}$, $\hat{\dot{r}}$, $\hat{\ddot{r}}$
θ_{HA}, $\dot{\theta}_{HA}$, $\ddot{\theta}_{HA}$
θ_{DEC}, $\dot{\theta}_{DEC}$, $\ddot{\theta}_{DEC}$
LT03
PERFORM COMPENSATION FOR ANGLE SERVOS AND FORM ANGLE SERVO DRIVE SIGNALS
LT04
SET INTERRUPT FLAG FOR ANGLE SERVO DRIVE OUTPUT SIGNALS
MODE
T
R
A
R
R
R
PERFORM RECURSIVE COMPUTATION OF SINE AND COSINE OF $\hat{\theta}_{HA}$, $\hat{\theta}_{DEC}$
PERFORM COORDINATE CONVERSIONS AND REFRACTION CORRECTIONS $\rightarrow$ SPHERICAL TOPOCENTRIC COORDINATES AT LAUNCH SITE
LT01
SET INTERRUPT FLAG FOR PROCESSED DATA OUTPUT TO MAG TAPE
SET INTERRUPT FLAG FOR OUTPUT OF SELECTED DATA TO PRINTER AND X-Y PLOTTER
MODE
T
R
A
S
S
RANGE INITIALIZATION COMPUTATION $\rightarrow$ PHASE CONTROL SIGNALS
LT02
SET INTERRUPT FLAG FOR PROCESSED DATA OUTPUT TO MAG TAPE
A
PRE-PROGRAM INITIALIZATION AND CALIBRATION
START
CALIBRATION OF X-Y PLOTTER; 0, ± 30%, ± 100% FULL SCALE
STORAGE OF ENCODER AND MOUNT CORRECTION TABLES
STORAGE OF (10) TARGET POSITIONS FOR SYSTEM CALIBRATION VIA COMMAND POINTING
STORAGE OF CONSTANT MATRICES USED IN COORDINATE CONVERSIONS
INITIALIZATION OF SINE AND COSINE FOR θ_{HA} AND θ_{DEC} RECURSIONS
INITIALIZATION OF ADDRESSES AND COUNTERS FOR INPUT AND OUTPUT DATA TRANSMISSION
INITIALIZATION OF MOD$_4$ COUNTER FOR CLOCK INTERRUPT
INITIALIZATION OF COMPENSATION FILTER COEFFICIENTS
INITIALIZATION OF TARGET EXTRAPOLATOR COEFFICIENTS
WRITE MAGNETIC TAPE FILE HEADER AND POSITION TAPE
SET SENSE SWITCH POSITIONS FOR CONTROL OF PRINTER AND PLOTTER SELECTIONS
ENABLE INTERRUPT SYSTEM
INITIALIZE OVERFLOW INDICATOR
READY STATUS
A

Figure 4-1. System Operation Flow Chart Subroutines Discussed in Text Are Labeled LT01 Through LT05

LT01 COORDINATE CONVERSIONS AND REFRACTION CORRECTIONS

INPUT DATA:

$$\left.\begin{array}{l}\hat{r},\ \hat{\dot{r}},\ \hat{\ddot{r}}\\[1em] \hat{\theta}_{HA},\ \hat{\dot{\theta}}_{HA},\ \hat{\ddot{\theta}}_{HA}\\[1em] \hat{\theta}_{DEC},\ \hat{\dot{\theta}}_{DEC},\ \hat{\ddot{\theta}}_{DEC}\end{array}\right\}\quad \text{FROM ESTIMATOR (LT03)}$$

$\mathrm{SIN}\ \hat{\theta}_{HA}$, $\mathrm{COS}\ \hat{\theta}_{HA}$, $\mathrm{SIN}\ \hat{\theta}_{DEC}$, $\mathrm{COS}\ \hat{\theta}_{DEC}$

OUTPUT DATA:

$$\left.\begin{array}{l}\theta,\ \dot{\theta},\ \ddot{\theta}\\[1em] \theta_{EL},\ \dot{\theta}_{EL},\ \ddot{\theta}_{EL}\\[1em] \theta_{AZ},\ \dot{\theta}_{AZ},\ \ddot{\theta}_{AZ}\end{array}\right\}\quad \begin{array}{l}\text{LAUNCH-SITE CENTERED}\\ \text{SPHERICAL TOPOCENTRIC}\\ \text{COORDINATES}\end{array}$$

LT02 RANGE SERVO INITIALIZATION AND CORRECTION

INPUT DATA:

SYSTEM MODE

PROGRAMMED RANGE, OR ESTIMATED RANGE

OUTPUT DATA:

THREE RANGE CHANNEL OUTPUTS (PHASE CONTROL SIGNALS)

Figure 4-2. Subroutine Input/Output Quantities for Simulation Program (Sheet 1 of 3)

LT03 ESTIMATOR/EXTRAPOLATOR

INPUT DATA:

SYSTEM MODE

$\tilde{r}, \tilde{\theta}_{HA}, \tilde{\theta}_{DEC}$

(EXTRAPOLATOR OUTPUT WHEN IN REACQUISITION MODE)

OUTPUT DATA:

$$\left.\begin{array}{l}\text{POLYNOMIAL}\\ \text{COEFFICIENTS}\\ \text{FOR GENERATING}\\ \text{RANGE AND ANGLE}\\ \text{ESTIMATES}\end{array}\right\}\begin{array}{l}\text{FROM}\\ \text{ESTIMATOR}\end{array}\quad \left.\begin{array}{lll}\hat{r}, & \hat{\dot{r}}, & \hat{\ddot{r}}\\ \hat{\theta}_{HA}, & \hat{\dot{\theta}}_{HA}, & \hat{\ddot{\theta}}_{HA}\\ \hat{\theta}_{DEC}, & \hat{\dot{\theta}}_{DEC}, & \hat{\ddot{\theta}}_{DEC}\end{array}\right\}\begin{array}{l}\text{FROM}\\ \text{EXTRAPOLATOR}\end{array}$$

LT04 ANGLE SERVO COMPENSATION AND CONTROL

INPUT DATA:

SYSTEM MODE

$\hat{\dot{\theta}}_{HA}, \hat{\dot{\theta}}_{DEC}$ (for velocity feed-forward) (TRACK, ACQUIRE)

$\hat{\theta}_{HA}, \hat{\theta}_{DEC}$ (REACQUIRE)

$\Delta\theta_{HA}, \Delta\theta_{DEC}$ (TRACK)

$\tilde{\dot{\theta}}_{HA}, \tilde{\dot{\theta}}_{DEC}$ (TRACK, ACQUIRE, REACQUIRE)

$\tilde{\theta}_{HA}, \tilde{\theta}_{DEC}$ (ACQUIRE, REACQUIRE)

PREPROGRAMMED $\theta_{HA}, \theta_{DEC}$ (ACQUIRE)

OUTPUT DATA:

HOUR ANGLE AND DECLINATION ANGLE
SERVO DRIVE SIGNALS

Figure 4-2. Subroutine Input/Output Quantities for Simulation Program (Sheet 2 of 3)

LT05 RANGE SERVO CONTROL

INPUT DATA:

THREE RANGE CHANNEL INPUTS

OUTPUT DATA:

THREE RANGE CHANNEL OUTPUTS (PHASE CONTROL SIGNALS)

$\tilde{r}$ (CALCULATED RANGE)

Figure 4-2. Subroutine Input/Output Quantities for Simulation Program (Sheet 3 of 3)

SECTION 5

CONCLUSIONS

The control computer program has been completely designed, and the operation of all major subsystems tested. Specifications have been set for all peripheral equipment.

All software has been supplied, with the exception of some connective commands and some output commands which are best supplied when the actual computer system is obtained, and the nature of which is fully detailed in the flow charts in paragraph 4.1. The design of the angle and range servos has been optimized to the point that performance limitations will be due strictly to external hardware. In some measure the target dynamics estimator/extrapolator is performance-limited by the computer hardware, but it is also constrained by the time constant consideration (maintenance of a low error level in the event of an engine failure). More significantly, it is impossible to state just what the estimator performance will be, because of the generally confused state of knowledge concerning atmospheric noise properties. The type of estimator which has been implemented will perform very well in a wide variety of noise environments because its design is not tightly wedded to any specific measurement noise model.

When a computer system is obtained it must be expected that further debugging of the control program will be required. However, short of actually building the final system, the simulation and test procedures which have been followed in checking subroutine operations constitute the best that can be done before actual system implementation.

APPENDIX A

PROGRAM LISTINGS FOR SUBROUTINES

```
      PROGRAM TRACKER
C     LASER TRACKER PROGRAM SIMULATOR
C
      COMMON CONZ,CON(100),TZ,T(50),XT(3,3),XL(3,3),POLAR(3,3)
      COMMON IXT,IXTA,IXTB
      COMMON R(3),HA(3),DEC(3),S(3,3,2),DS(3,5),RR(3,2)
      COMMON NSAM,KSA,KSB
      COMMON/DATA/RK(4,50),HK(4,50),DK(4,50)
      COMMON IHEZ,IHE(3600),IDEZ,IDE(900),IHM(120),IDM(30)
      COMMON/DATA/RQ(7,3),HQ(7,3),DQ(7,3)
      DATA(((RQ(I,J),I=1,7),J=1,3)= 3.07428539E3, -2.855444864 ,
     X.6487582373,-.0494641801g, 1.93016001E-3,-1.928922545E-5,
     X6.734844011E-8, -.05053388666, .02507330732, .01437644815,
     X-1.177956236E-3, 1.394900353E-4, -2.590478355E-6, 1.457511935E-8,
     X.02507330732, .0287528963, -3.533868708E-3, 5.579601412E-4,
     X-1.2952391775E-5, 8.74507161E-8, 0.)
      DATA(((HQ(I,J),I=1,7),J=1,3)= 117.4125936, -.08129312544,
     X-.02099399294, 8.142128362E-4, -8.858125812E-5, 1.941353416E-6,
     X-1.22180851E-8, 4.709042989E-2,-.1330068257, .01715535798,
     X-.0012969118,   3.784461609E-5, -4.654506566E-7, 2.060032254E-9,
     X-.1330068257,   3.431071596E-2, -3.8907354E-3,   1.5137846436E-4,
     X-2.327253283E-6, 1.2360193524E-8, 0.)
      DATA(((DQ(I,J),I=1,7),J=1,3)= 41.03893121, .2129026631,
     X-3.227090683E-2, 3.694090156E-3, -1.181951605E-4, 1.480725462E-6,
     X-6.537293629E-9, 7.323323364E-3, 2.449193662E-2, 1.06749198E-4,
     X1.08169538E-4, -7.557305137E-6, 1.456644425E-7, -8.779346109E-10,
     X.02449193662, 2.13498396E-4, 3.24508614E-4, -3.0229220548E-5,
     X7.283222125E-7, -5.2676076654E-9, 0.)
      CALL SETDATA(DUM)
      MODE=1
      XKHST=XKDST=782.
      XKHC1=XKDC1=819.1
      PI=3.1415926536
      RANGE=RQ(1,1)
      DEGRAD=PI/180.
      RADSEC=3600./DEGRAD
      HAFAC=2.049236
      DEFAC=.7162645
      IHEZ=IDEZ=0
      DO 20 I=1,3600
   20 IHE(I)=0
      DO 21 I=1,900
   21 IDE(I)=0
      DO 22 I=1,120
   22 IHM(I)=0
      DO 23 I=1,30
   23 IDM(I)=0
      DO 5 I=1,3
      DO 5 J=1,5
    5 DS(I,J)=0.
      DO 6 I=1,3
      DO 6 J=1,2
    6 RR(I,J)=0.
C
C
      DO 1000 ISECS=0,60
      DO 1000 ILOOP=0,63
      INDEX=ISECS*64+ILOOP
      DO 1 I=0,3
      T1=(INDEX+I/4.)/64.
      T2=T1*T1
      T3=T2*T1
      T4=T3*T1
      T5=T4*T1
      T6=T5*T1
      RT=                  RQ(2,1)*T1+RQ(3,1)*T2+RQ(4,1)*T3
     *                    +RQ(5,1)*T4+RQ(6,1)*T5+RQ(7,1)*T6
      CALL RSERVO(INDEX,I,RT,J1,J2,J3,I1,I2,I3)
      CALL LT05(INDEX,I,I1,I2,I3,J1,J2,J3,RANGE,RGNRATE)
    1 CONTINUE
      GO TO (44,46),SSWTCHF(1)
   44 WRITE(61,950)ILOOP,RANGE
  950 FORMAT(7H ILOOP=I4,1X6HRANGE=E15.6)
   46 CONTINUE
C
      HT=                 +HQ(2,1)*T1+HQ(3,1)*T2+HQ(4,1)*T3
     *                    +HQ(5,1)*T4+HQ(6,1)*T5+HQ(7,1)*T6
      DT=                 +DQ(2,1)*T1+DQ(3,1)*T2+DQ(4,1)*T3
     *                    +DQ(5,1)*T4+DQ(6,1)*T5+DQ(7,1)*T6
      CALL ASERVO(INDEX,HT,DT,KH,KD,MHE,MDE,MH,MD,MDH,MDD)
      VHST=MHE
      VDST=MDE
      DHIN=MDH
      DDIN=MDD
      IH=MH*RADSEC/(XKHST*XKHC1)
      ID=MD*RADSEC/(XKDST*XKDC1)
C**** TEMPORARILY SKIP THE CORRECTIONS...
      IF(1.EQ.1)30,31
   31 CONTINUE
C     PERFORM THE ENCODER CORRECTIONS
      I=IH/360
      F=IH/360.-I
      IH=IH+IHEZ+IHE(I+1)+(IHE(I+2)-IHE(I+1))*F/360.
      I=ID/360
      F=ID/360.-I
      ID=ID+IDEZ+IDE(I+1)+(IDE(I+2)-IDE(I+1))*F/360.
C     PERFORM THE MOUNT CORRECTIONS
      I=IH/10800
      F=IH/10800.-I
      IH=IH+IHM(I+1)+(IHM(I+2)-IHM(I+1))*F/10800.
      I=ID/10800
      F=ID/10800.-I
      ID=ID+IDM(I+1)+(IDM(I+2)-IDM(I+1))*F/10800.
C**** TEMPORARILY SKIP THE CORRECTIONS...
   30 CONTINUE
      HOUR=IH/RADSEC+HAFAC
      DECLIN=ID/RADSEC+DEFAC
      HOUR=HOUR*RADSEC
      DECLIN=DECLIN*RADSEC
      RANGE=RANGE*100.
      INDEST=INDEX+1
      CALL LT03(INDEST,RANGE,HOUR,DECLIN,MODE)
      DO 778 IMS=1,3
      HA(IMS)=HA(IMS)/RADSEC
      DEC(IMS)=DEC(IMS)/RADSEC
      R(IMS)=R(IMS)/100.
  778 CONTINUE
      SH=SIN( HA(1))
      SD=SIN(DEC(1))
      CH=COS( HA(1))
      CD=COS(DEC(1))
      HA(1)=HA(1)-HAFAC
      DEC(1)=DEC(1)-DEFAC
C
      IF(MODE.LE.0)3,4
    3 VHST=XKHST*XKHC1*HA(1) -MH
      VDST=XKDST*XKDC1*DEC(1)-MD
    4 CONTINUE
      CALL LT04(INDEX,VHST,VDST,DHIN,DDIN,KH,KD,CD)
      HA(1)=HA(1)+HAFAC
      DEC(1)=DEC(1)+DEFAC
      CALL LT01(INDEX,SH,CH,SD,CD)
C
      IF(ILOOP.EQ.0)100,33
   33 IF(ILOOP.EQ.32)100,1000
  100 CONTINUE
      WRITE(61,10)
   10 FORMAT( /7X5HINDEX7X7HSIN(HA)8X7HCOS(HA)8X8HSIN(DEC)7X8HCOS(DEC))
      WRITE(61,11) INDEX,SH,CH,SD,CD
   11 FORMAT(XI11,4X4(1H/E14.6))
      WRITE(61,901) RANGE,HOUR,DECLIN
  901 FORMAT(X3(1H/E14.6))
      WRITE(61,12)((MX,NX=1,3),MX=1,3)
   12 FORMAT(/3(3X6HRANGE(I1,1H)9X3HHA(I1,1H)9X4HDEC(I1,1H)5X))
      WRITE(61,900)(R(MX),HA(MX),DEC(MX),MX=1,3)
  900 FORMAT(X9(1H/E14.6))
      WRITE(61,907)((MX,NX,MX=1,3),NX=1,3)
  907 FORMAT(/9(3X3HXT(,I1,1H,I1,1H)5X))
      WRITE(61,900)((XT(MX,NX),MX=1,3),NX=1,3)
      WRITE(61,908)((MX,NX,MX=1,3),NX=1,3)
  908 FORMAT(/9(3X3HXL(,I1,1H,I1,1H)5X))
      WRITE(61,900)((XL(MX,NX),MX=1,3),NX=1,3)
      WRITE(61,909)((MX,NX=1,3),MX=1,3)
  909 FORMAT(/3(5X4HRHO(I1,1H)9X3HEL(I1,1H)9X3HAZ(I1,1H)6X))
      WRITE(61,900)((POLAR(MX,NX),MX=1,3),NX=1,3)
 1000 CONTINUE
      STOP
      END
      SUBROUTINE SETDATA(DUM)
C     THIS ROUTINE PREPARES THE CONSTANT ARRAY -CON-
C
      COMMON CONZ,CON(100),TZ,T(50),XT(3,3),XL(3,3),POLAR(3,3)
      COMMON IXT,IXTA,IXTB
      COMMON R(3),HA(3),DEC(3),S(3,3,2),DS(3,5),RR(3,2)
      COMMON NSAM,KSA,KSB
      COMMON/DATA/RK(4,50),HK(4,50),DK(4,50)
C
      DATA(((RK(I,J),I=1,4),J=1,5)=
     *  1.1239E 00, -8.0322E 00,  1.5762E 01, -9.1377E 00,
     *  1.9054E-01, -2.8130E-01, -5.3558E-01,  6.9002E-01,
     *  7.7022E-02,  7.5374E-01, -2.7215E 00,  1.9947E 00,
     * -7.2046E-03,  1.4559E 00, -4.1077E 00,  2.7796E 00,
     * -6.5360E-02,  1.8654E 00, -4.7963E 00,  3.1142E 00)
      DATA(((RK(I,J),I=1,4),J=6,10)=
     * -1.0067E-01,  2.0223E 00, -4.8899E 00,  3.0676E 00,
     * -1.1635E-01,  1.9667E 00, -4.4905E 00,  2.7093E 00,
     * -1.1563E-01,  1.7387E 00, -3.7006E 00,  2.1084E 00,
     * -1.0173E-01,  1.3785E 00, -2.6226E 00,  1.3344E 00,
     * -7.7882E-02,  9.2620E-01, -1.3585E 00,  4.5633E-01)
      DATA(((RK(I,J),I=1,4),J=11,15)=
     * -4.7301E-02,  4.2188E-01, -1.0930E-02, -4.5633E-01,
     * -1.3212E-02, -9.4320E-02,  1.3180E 00, -1.3344E 00,
     *  2.1161E-02, -5.8227E-01,  2.5258E 00, -2.1084E 00,
     *  5.2594E-02, -1.0019E 00,  3.5104E 00, -2.7093E 00,
     *  7.7865E-02, -1.3130E 00,  4.1693E 00, -3.0676E 00)
      DATA(((RK(I,J),I=1,4),J=16,20)=
     *  9.3748E-02, -1.4755E 00,  4.4002E 00, -3.1142E 00,
     *  9.7022E-02, -1.4493E 00,  4.1009E 00, -2.7796E 00,
     *  8.4462E-02, -1.1942E 00,  3.1690E 00, -1.9947E 00,
     *  5.2846E-02, -6.7016E-01,  1.5021E 00, -6.9002E-01,
     * -2.2584E-01,  3.5646E 00, -1.1223E 01,  9.1377E 00)
      DATA(((RK(I,J),I=1,4),J=21,25)=
     * -1.0995E-01,  1.7565E 00, -5.6398E 00,  4.7230E 00,
     * -1.0828E-01,  1.6648E 00, -5.1031E 00,  4.0555E 00,
     * -1.1723E-01,  1.7480E 00, -5.1569E 00,  3.9176E 00,
     * -1.2834E-01,  1.8693E 00, -5.3554E 00,  3.9277E 00,
     * -1.3823E-01,  1.9761E 00, -5.5327E 00,  3.9482E 00)
      DATA(((RK(I,J),I=1,4),J=26,30)=
     * -1.4540E-01,  2.0458E 00, -5.6208E 00,  3.9240E 00,
     * -1.4938E-01,  2.0722E 00, -5.6007E 00,  3.8389E 00,
     * -1.5038E-01,  2.0584E 00, -5.4812E 00,  3.6972E 00,
     * -1.4897E-01,  2.0127E 00, -5.2841E 00,  3.5130E 00,
     * -1.4583E-01,  1.9446E 00, -5.0354E 00,  3.3025E 00)
      DATA(((RK(I,J),I=1,4),J=31,35)=
     * -1.4161E-01,  1.8632E 00, -4.7586E 00,  3.0806E 00,
     * -1.3686E-01,  1.7758E 00, -4.4724E 00,  2.8587E 00,
     * -1.3196E-01,  1.6879E 00, -4.1903E 00,  2.6446E 00,
     * -1.2721E-01,  1.6030E 00, -3.9210E 00,  2.4434E 00,
     * -1.2278E-01,  1.5233E 00, -3.6695E 00,  2.2575E 00)
      DATA(((RK(I,J),I=1,4),J=36,40)=
     * -1.1877E-01,  1.4500E 00, -3.4381E 00,  2.0877E 00,
     * -1.1523E-01,  1.3835E 00, -3.2276E 00,  1.9340E 00,
     * -1.1215E-01,  1.3240E 00, -3.0374E 00,  1.7956E 00,
     * -1.0954E-01,  1.2709E 00, -2.8663E 00,  1.6712E 00,
     * -1.0734E-01,  1.2240E 00, -2.7127E 00,  1.5597E 00)
      DATA(((RK(I,J),I=1,4),J=41,45)=
     * -1.0554E-01,  1.1825E 00, -2.5750E 00,  1.4597E 00,
     * -1.0409E-01,  1.1460E 00, -2.4515E 00,  1.3700E 00,
# -1,0295E-01,  1,1139E 00, -2,3406E 00,  1,2893E 00,
# -1,0207E-01,  1,0855E 00, -2,2409E 00,  1,2166E 00,
# -1,0142E-01,  1,0605E 00, -2,1509E 00,  1,1509E 00)
  DATA(((RK(I,J),I=1,4),J=46,50)=
# -1,0096E-01,  1,0383E 00, -2,0694E 00,  1,0913E 00,
# -1,0065E-01,  1,0185E 00, -1,9952E 00,  1,0372E 00,
# -1.0048E-01,  1.0007E 00, -1.9275E 00,  9.8774E-01,
# -1.0040E-01,  9.8464E-01, -1.8653E 00,  9.4241E-01,
# -1.0039E-01,  9.6993E-01, -1.8078E 00,  9.0070E-01)
  DATA(((HK(I,J),I=1,4),J=1,5)=
#  1,1239E 00, -8,0322E 00,  1,5762E 01, -9,1377E 00,
#  1,9054E-01, -2,8130E-01, -5,3558E-01,  6,9002E-01,
#  7,7022E-02,  7,5374E-01, -2,7215E 00,  1,9947E 00,
# -7,2046E-03,  1,4559E 00, -4,1077E 00,  2,7796E 00,
# -6,5360E-02,  1,8654E 00, -4,7963E 00,  3,1142E 00)
  DATA(((HK(I,J),I=1,4),J=6,10)=
# -1,0067E-01,  2,0223E 00, -4,8899E 00,  3,0676E 00,
# -1,1635E-01,  1,9667E 00, -4,4905E 00,  2,7093E 00,
# -1,1563E-01,  1,7387E 00, -3,7006E 00,  2,1084E 00,
# -1,0173E-01,  1,3785E 00, -2,6226E 00,  1,3344E 00,
# -7,7882E-02,  9,2620E-01, -1,3585E 00,  4,5633E-01)
  DATA(((HK(I,J),I=1,4),J=11,15)=
# -4,7301E-02,  4,2188E-01, -1,0930E-02, -4,5633E-01,
# -1,3212E-02, -9,4320E-02,  1,3180E 00, -1,3344E 00,
#  2,1161E-02, -5,8227E-01,  2,5258E 00, -2,1084E 00,
#  5,2594E-02, -1,0019E 00,  3,5104E 00, -2,7093E 00,
#  7,7865E-02, -1,3130E 00,  4,1693E 00, -3,0676E 00)
  DATA(((HK(I,J),I=1,4),J=16,20)=
#  9,3748E-02, -1,4755E 00,  4,4002E 00, -3,1142E 00,
#  9,7022E-02, -1,4493E 00,  4,1009E 00, -2,7796E 00,
#  8.4462E-02, -1.1942E 00,  3.1690E 00, -1.9947E 00,
#  5.2846E-02, -6.7016E-01,  1.5021E 00, -6.9002E-01,
# -2.2584E-01,  3.5646E 00, -1.1223E 01,  9.1377E 00)
  DATA(((HK(I,J),I=1,4),J=21,25)=
# -1,0995E-01,  1,7565E 00, -5,6398E 00,  4,7230E 00,
# -1,0828E-01,  1,6648E 00, -5,1031E 00,  4,0555E 00,
# -1,1723E-01,  1,7480E 00, -5,1569E 00,  3,9176E 00,
# -1,2834E-01,  1,8693E 00, -5,3554E 00,  3,9277E 00,
# -1,3823E-01,  1,9761E 00, -5,5327E 00,  3,9482E 00)
  DATA(((HK(I,J),I=1,4),J=26,30)=
# -1,4540E-01,  2,0458E 00, -5,6208E 00,  3,9240E 00,
# -1,4938E-01,  2,0722E 00, -5,6007E 00,  3,8389E 00,
# -1,5038E-01,  2,0584E 00, -5,4812E 00,  3,6972E 00,
# -1,4897E-01,  2,0127E 00, -5,2841E 00,  3,5130E 00,
# -1,4583E-01,  1,9446E 00, -5,0354E 00,  3,3025E 00)
  DATA(((HK(I,J),I=1,4),J=31,35)=
# -1,4161E-01,  1,8632E 00, -4,7586E 00,  3,0806E 00,
# -1,3686E-01,  1,7758E 00, -4,4724E 00,  2,8587E 00,
# -1,3196E-01,  1,6879E 00, -4,1903E 00,  2,6446E 00,
# -1,2721E-01,  1,6030E 00, -3,9210E 00,  2,4434E 00,
# -1,2278E-01,  1,5233E 00, -3,6695E 00,  2,2575E 00)
  DATA(((HK(I,J),I=1,4),J=36,40)=
# -1,1877E-01,  1,4500E 00, -3,4381E 00,  2,0877E 00,
# -1,1523E-01,  1,3835E 00, -3,2276E 00,  1,9340E 00,
# -1.1215E-01,  1.3240E 00, -3.0374E 00,  1.7956E 00,
# -1.0954E-01,  1.2709E 00, -2.8663E 00,  1.6712E 00,
# -1.0734E-01,  1.2240E 00, -2.7127E 00,  1.5597E 00)
  DATA(((HK(I,J),I=1,4),J=41,45)=
# -1,0554E-01,  1,1825E 00, -2,5750E 00,  1,4597E 00,
# -1,0409E-01,  1,1460E 00, -2,4515E 00,  1,3700E 00,
# -1,0295E-01,  1,1139E 00, -2,3406E 00,  1,2893E 00,
# -1,0207E-01,  1,0855E 00, -2,2409E 00,  1,2166E 00,
* -1.0142E-01,  1.0605E 00, -2.1509E 00,  1.1509E 00)
 DATA(((HK(I,J),I=1,4),J=46,50)=
* -1.0096E-01,  1.0383E 00, -2.0694E 00,  1.0913E 00,
* -1.0065E-01,  1.0185E 00, -1.9952E 00,  1.0372E 00,
* -1.0048E-01,  1.0007E 00, -1.9275E 00,  9.8774E-01,
* -1.0040E-01,  9.8464E-01, -1.8653E 00,  9.4241E-01,
* -1.0039E-01,  9.6993E-01, -1.8078E 00,  9.0070E-01)
 DATA(((DK(I,J),I=1,4),J=1,5)=
*  1.1239E 00, -8.0322E 00,  1.5762E 01, -9.1377E 00,
*  1.9054E-01, -2.8130E-01, -5.3558E-01,  6.9002E-01,
*  7.7022E-02,  7.5374E-01, -2.7215E 00,  1.9947E 00,
* -7.2046E-03,  1.4559E 00, -4.1077E 00,  2.7796E 00,
* -6.5360E-02,  1.8654E 00, -4.7963E 00,  3.1142E 00)
 DATA(((DK(I,J),I=1,4),J=6,10)=
* -1.0067E-01,  2.0223E 00, -4.8899E 00,  3.0676E 00,
* -1.1635E-01,  1.9667E 00, -4.4905E 00,  2.7093E 00,
* -1.1563E-01,  1.7387E 00, -3.7006E 00,  2.1084E 00,
* -1.0173E-01,  1.3785E 00, -2.6226E 00,  1.3344E 00,
* -7.7882E-02,  9.2620E-01, -1.3585E 00,  4.5633E-01)
 DATA(((DK(I,J),I=1,4),J=11,15)=
* -4.7301E-02,  4.2188E-01, -1.0930E-02, -4.5633E-01,
* -1.3212E-02, -9.4320E-02,  1.3180E 00, -1.3344E 00,
*  2.1161E-02, -5.8227E-01,  2.5258E 00, -2.1084E 00,
*  5.2594E-02, -1.0019E 00,  3.5104E 00, -2.7093E 00,
*  7.7865E-02, -1.3130E 00,  4.1693E 00, -3.0676E 00)
 DATA(((DK(I,J),I=1,4),J=16,20)=
*  9.3748E-02, -1.4755E 00,  4.4002E 00, -3.1142E 00,
*  9.7022E-02, -1.4493E 00,  4.1009E 00, -2.7796E 00,
*  8.4462E-02, -1.1942E 00,  3.1690E 00, -1.9947E 00,
*  5.2846E-02, -6.7016E-01,  1.5021E 00, -6.9002E-01,
* -2.2584E-01,  3.5646E 00, -1.1223E 01,  9.1377E 00)
 DATA(((DK(I,J),I=1,4),J=21,25)=
* -1.0995E-01,  1.7565E 00, -5.6398E 00,  4.7230E 00,
* -1.0828E-01,  1.6648E 00, -5.1031E 00,  4.0555E 00,
* -1.1723E-01,  1.7480E 00, -5.1569E 00,  3.9176E 00,
* -1.2834E-01,  1.8693E 00, -5.3554E 00,  3.9277E 00,
* -1.3823E-01,  1.9761E 00, -5.5327E 00,  3.9482E 00)
 DATA(((DK(I,J),I=1,4),J=26,30)=
* -1.4540E-01,  2.0458E 00, -5.6208E 00,  3.9240E 00,
* -1.4938E-01,  2.0722E 00, -5.6007E 00,  3.8389E 00,
* -1.5038E-01,  2.0584E 00, -5.4812E 00,  3.6972E 00,
* -1.4897E-01,  2.0127E 00, -5.2841E 00,  3.5130E 00,
* -1.4583E-01,  1.9446E 00, -5.0354E 00,  3.3025E 00)
 DATA(((DK(I,J),I=1,4),J=31,35)=
* -1.4161E-01,  1.8632E 00, -4.7586E 00,  3.0806E 00,
* -1.3686E-01,  1.7758E 00, -4.4724E 00,  2.8587E 00,
* -1.3196E-01,  1.6879E 00, -4.1903E 00,  2.6446E 00,
* -1.2721E-01,  1.6030E 00, -3.9210E 00,  2.4434E 00,
* -1.2278E-01,  1.5233E 00, -3.6695E 00,  2.2575E 00)
 DATA(((DK(I,J),I=1,4),J=36,40)=
* -1.1877E-01,  1.4500E 00, -3.4381E 00,  2.0877E 00,
* -1.1523E-01,  1.3835E 00, -3.2276E 00,  1.9340E 00,
* -1.1215E-01,  1.3240E 00, -3.0374E 00,  1.7956E 00,
* -1.0954E-01,  1.2709E 00, -2.8663E 00,  1.6712E 00,
* -1.0734E-01,  1.2240E 00, -2.7127E 00,  1.5597E 00)
 DATA(((DK(I,J),I=1,4),J=41,45)=
* -1.0554E-01,  1.1825E 00, -2.5750E 00,  1.4597E 00,
* -1.0409E-01,  1.1460E 00, -2.4515E 00,  1.3700E 00,
* -1.0295E-01,  1.1139E 00, -2.3406E 00,  1.2893E 00,
* -1.0207E-01,  1.0855E 00, -2.2409E 00,  1.2166E 00,
* -1.0142E-01,  1.0605E 00, -2.1509E 00,  1.1509E 00)
 DATA(((DK(I,J),I=1,4),J=46,50)=
     * -1.0096E-01,  1.0383E 00, -2.0694E 00,  1.0913E 00,
     * -1.0065E-01,  1.0185E 00, -1.9952E 00,  1.0372E 00,
     * -1.0048E-01,  1.0007E 00, -1.9275E 00,  9.8774E-01,
     * -1.0040E-01,  9.8464E-01, -1.8653E 00,  9.4241E-01,
     * -1.0039E-01,  9.6993E-01, -1.8078E 00,  9.0070E-01)
C
      CONZ=2.
      CON(1)=.0001314
      CON( 2)=.000292
      CON( 3)=2.7182818
      CON( 4)=.000001
      CON( 5)=1.0
      CON( 6)=.015625
      CON( 7)=.000244141
      CON( 8)=.0078125
      CON(9)=-22.32509986
      CON(10)=2283.338565
      CON(11)=2056.1279001
      CON(12)=.99999988
      CON(13)=3.595758511E-04
      CON(14)=3.2212200927E-04
      CON(15)=-3.596324088E-04
      CON(16)=.99999992
      CON(17)=1.7553777069E-04
      CON(18)=-3.2205886434E-04
      CON(19)=-1.7565359575E-04
      CON(20)=.99999993
      CON(21)=.8780847482
      CON(22)=.4785051463
      CON(23)=100.
      CON(24)=20.
      CON(25)=600.
      CON(26)=120.
      CON(27)=640000.
      CON(28)=3200000.
      CON(29)=.170
      CON(30)=.034
      CON(31)=55600.
      CON(32)=5880.
      CON(33)=.3
      CON(34)=819.1
      CON(35)=4700.
      CON(36)=6.8
      CON(37)=782.
      CON(38)=20.
      CON(39)=.015625
      CON(40)=300.
      CON(41)=18.
      CON(42)=6.
      CON(43)=18.
      CON(44)=10.
      CON(45)=.087
      CON(46)=3.1415926536/(180.*3600.)
      NSAM = 90
      KSA=3
      KSB=1
      RETURN
      END

        3200 FORTRAN DIAGNOSTIC RESULTS - FOR   SETDATA
```

```
                                    31/32/3300 FORTRAN (2.2)/MSOS   04/25/68
      SUBROUTINE ASERVO(INDEX,HT,DT,IH,ID,MHE,MDE,MH,MD,MDH,MDD)
C     LASER TRACKER ANGLE SERVO SIMULATOR
C     THIS PROGRAM CALCULATES THE ANGLE,ANGLE ERROR AND ANGLE RATE
C        QUANTITIES GENERATED FROM THE HARDWARE EQUIPMENT
C     INPUTS ARE-
C        1. IH- HOUR ANGLE SERVO DRIVE OUTPUT
C        2. ID- DECLINATION ANGLE SERVO DRIVE OUTPUT
C        3 HT- TARGET HOUR ANGLE
C        4.- TARGET DECLINATION ANGLE
C     OUTPUTS ARE-
C        1.MHE- STAR TRACKER HOUR ANGLE SIGNAL
C        2. MDE- STAR TRACKER DECLINATION ANGLE SIGNAL
C        3. MDH- TACHOMETER HOUR ANGLE SIGNAL
C        4. MDD- TACHOMETER DECLINATION ANGLE SIGNAL
C        5. MH- HOUR ANGLE
C        6. DECLINATION ANGLE
C
      IF(INDEX.EQ.0)1,2
    1 PI=3.1415926536
      DEGRAD=PI/180.
      TP2=1./1024.
      WME=400.
      XKHA=XKDA=.3
      BH=100.
      BD=20.
      XJH=600.
      XJD=120.
      XKHST=XKDST=782.
      XKHT=XKDT=20.
      XKHC1=XKDC1=819.1
      XKHC2=XKDC2=4700.
      XKHC3=.170
       XKDC3=.034
C
      IH=ID=0
      HTMC=DTMC=0.
      HYTMC=DYTMC=0.
      DHI=DDI=HI=DI=0.
C
    2 CONTINUE
      DO 7 K=0,15
       HXTMC=WME*(XKHA*IH*XKHC3-HTMC)
      HTMC=HTMC+TP2*(3.*HXTMC-HYTMC)*.5
      HYTMC=HXTMC
      DHI=DHI+TP2*(HTMC-DHI*BH)/XJH
      HI=HI+DHI*TP2
C
      DXTMC=WME*(XKDA*ID*XKDC3-DTMC)
      DTMC=DTMC+TP2*(3.*DXTMC-DYTMC)*.5
      DYTMC=DXTMC
      DDI=DDI+TP2*(DTMC-DDI*BD)/XJD
      DI=DI+DDI*TP2
    7 CONTINUE
C
      HIN=XKHST*XKHC1*HI
      MH=HIN
      MHE=XKHST*XKHC1*DEGRAD*HT-HIN
      MDH=XKHT*XKHC2*DHI
      DIN=XKDST*XKDC1*DI
      MD=DIN
```

```
MDE=XKDST*XKDC1*DEGRAD*DT-DIN
MDD=XKDT*XKDC2*DDI
RETURN
END
```

```
                                    31/32/3300 FORTRAN (2.2)/MSOS   04/25/68
      SUBROUTINE RSERVO(INDEX,I,RT,J1,J2,J3,I1,I2,I3)
C     RANGE SERVO SIMULATOR
C     THIS PROGRAM CALCULATES THE RANGE CHANNEL PJASE SIGNALS GENERATED
C        FRCM THE RANGE SERVO EQUIPMENT
C     INPUTS ARE-
C        1. J1
C        2. J2
C        3. J3
C        THESE ARE THE RANGE CHANNEL ERROR SIGNALS
C        4. RT- RANGE OF TARGET IN METERS
C     OUTPUTS ARE-
C        1. I1
C        2. I2
C        3. I3
C        THESE ARE THE RANGE CHANNEL PHASE SIGNALS
CP
      IF(INCEX.EQ.0)1,3
    1 IF(I.EQ.0)2,3
    2 TP1=1/20000.
      XK4=10.
      WF=160.
      E1=E2=E3=0.
      FB1=FB2=FB3=0.0
      J1=J2=J3=0
    3 CA=XK4*WF*TP1
      CB=1.-TP1*WF
      C1=RT*.0012
      C2=C1*32.
      C3=C2*32.
C
      DO 7 K=0,79
      FB1=FB1+TP1* J1*0.384
      FB2=FB2+TP1* J2*0.384
      FB3=FB3+TP1* J3*0.384
       E1=CA*(C1-FB1)+CB* E1
       E2=CA*(C2-FB2)+CB* E2
       E3=CA*(C3-FB3)+CB* E3
    7 CONTINUE
       I1=102.*E1
       I2=102.*E2
       I3=102.*E3
      RETURN
      END

        3200 FORTRAN DIAGNOSTIC RESULTS - FOR   RSERVO

NO ERRORS
```

```
                                31/32/3300 FORTRAN (2.2)/MSOS   04/25/68
      SUBROUTINE LT01(INDEX,SH,CH,SD,CD)
C     VERSION -8-
C     PROGRAM LT01, CALCULATE LASER TRACKER COORDINATE CONVERSIONS
C
C     INPUTS ARE-
C     1. T=TEMPORARY STORAGE
C     2. SD=SINE OF DECLINATION ANGLE
C     3. CD=COSINE OF DECLINATION ANGLE
C     4. SH=SINE OF HOUR ANGLE
C     5. CH=COSINE OF HOUR ANGLE
C     6. CON=LIST OF CONSTANTS
C     7. INDEX=COUNT OF BASIC TIME INTERRUPTS-
C        (INCRAMENTS EVERY 64TH OF A SECOND, STARTS AT ZERO)
C     8. R=RANGE VECTOR (FROM ESTIMATOR- IN METERS)
C     9. HA=HOUR ANGLE VECTOR (FROM ESTIMATOR- IN RADIANS)
C     10. DEC=DECLINATION ANGLE VECTOR (FROM ESTIMATOR- IN RADIANS)
C     11. PLACE TO STORE XL,XT,POLAR COORDINATES.
C
C     OUTPUTS ARE-
C     1. XL=LAUNCH CENTERED COORDINATES
C     2. XT=TRACKER CENTERED COORDINATES
C     3. POLAR= VEHICLE POLAR COORDINATES
C
      COMMON CONZ,CON(100),TZ,T(50),XT(3,3),XL(3,3),POLAR(3,3)
      COMMON IXT,IXTA,IXTB
      COMMON R(3),HA(3),DEC(3),S(3,3,2),DS(3,5),RR(3,2)
      COMMON NSAM,KSA,KSB
      IF(INDEX.EQ.0)1,2
    1 I=J=1
      GO TO 10
    2 IF(AND(INDEX,1).EQ.0)3,4
    3 I=1
      J=2
      GO TO 10
    4 I=2
      J=1
   10 DO 5 K=1,3
    5 RR(K,I)=R(K)
      TZ=SD*SH
      T(1)=SD*CH
      T(2)=CD*SH
      T(3)=CD*CH
      T(4)=CON(22)*SD
      T(5)=CON(22)*CD
      T(6)=CON(21)*SD
      T(7)=CON(21)*CD
      T(8)=DEC(2)*DEC(2)
      T(9)=HA(2)*HA(2)
      T(10)=T(8)+T(9)
      T(11)=CONZ*HA(2)*DEC(2)
      T(12)=HA(2)*T(2)+DEC(2)*T(1)
      T(13)=HA(3)*T(2)+DEC(3)*T(1)+T(10)*T(3)+T(11)*TZ
      S(1,1,I)=CON(21)*T(3)+T(4)
      S(2,1,I)=CON(22)*T(3)-T(6)
      S(3,1,I)=-T(2)
      S(1,2,I)= DEC(2)*T(5)-CON(21)*T(12)
      S(2,2,I)=-DEC(2)*T(7)-CON(22)*T(12)
      S(3,2,I)=DEC(2)*TZ-HA(2)*T(3)
      S(1,3,I)=DEC(3)*T(5)-T(8)*T(4)-T(13)*CON(21)
      S(2,3,I)=T(8)*T(6)-DEC(3)*T(7)-T(13)*CON(22)
      S(3,3,I)=DEC(3)*TZ-HA(3)*T(3)+T(10)*T(2)+T(11)*T(1)
      TZ=SQRTF(S(2,1,I)*S(2,1,I)+S(3,1,I)*S(3,1,I))
      IF(TZ.LE.CON(4))6,7
    6 T(1)=T(2)=T(3)=0.
      GO TO 8
    7 T(4)=RR(1,I)*CON(1)
      T(5)=CON(5)+EXPF(T(4)*S(1,1,I))
      T(5)=T(4)*CON(2)/T(5)
      T(2)=T(5)*S(1,1,I)*S(2,1,I)
      T(1)=-T(5)*TZ*TZ
      T(3)=T(5)*S(1,1,I)*S(3,1,I)
    8 DS(1,5)=DS(1,4)
      DS(1,4)=DS(1,1)
      DS(1,1)=T(1)
      DS(2,5)=DS(2,4)
      DS(2,4)=DS(2,1)
      DS(2,1)=T(2)
      DS(3,5)=DS(3,4)
      DS(3,4)=DS(3,1)
      DS(3,1)=T(3)
      DS(1,2)=(DS(1,1)-DS(1,5))*CON(8)
      DS(2,2)=(DS(2,1)-DS(2,5))*CON(8)
      DS(3,2)=(DS(3,1)-DS(3,5))*CON(8)
      DS(1,3)=(DS(1,1)+DS(1,5)-CONZ*DS(1,4))*CON(7)
      DS(2,3)=(DS(2,1)+DS(2,5)-CONZ*DS(2,4))*CON(7)
      DS(3,3)=(DS(3,1)+DS(3,5)-CONZ*DS(3,4))*CON(7)
      XT(1,1)=RR(1,J)*S(1,1,J)
      XT(2,1)=RR(1,J)*S(2,1,J)
      XT(3,1)=RR(1,J)*S(3,1,J)
      XT(1,2)=RR(2,J)*S(1,1,J)+RR(1,J)*S(1,2,J)
      XT(2,2)=RR(2,J)*S(2,1,J)+RR(1,J)*S(2,2,J)
      XT(3,2)=RR(2,J)*S(3,1,J)+RR(1,J)*S(3,2,J)
      XT(1,3)=RR(3,J)*S(1,1,J)+RR(2,J)*S(1,2,J)*CONZ+RR(1,J)*S(1,3,J)
      XT(2,3)=RR(3,J)*S(2,1,J)+RR(2,J)*S(2,2,J)*CONZ+RR(1,J)*S(2,3,J)
      XT(3,3)=RR(3,J)*S(3,1,J)+RR(2,J)*S(3,2,J)*CONZ+RR(1,J)*S(3,3,J)
C
      XL(1,1)=CON(12)*XT(1,1)+CON(15)*XT(2,1)+CON(18)*XT(3,1)+CON(9)
      XL(2,1)=CON(13)*XT(1,1)+CON(16)*XT(2,1)+CON(19)*XT(3,1)+CON(10)
      XL(3,1)=CON(14)*XT(1,1)+CON(17)*XT(2,1)+CON(20)*XT(3,1)+CON(11)
      XL(1,2)=CON(12)*XT(1,2)+CON(15)*XT(2,2)+CON(18)*XT(3,2)
      XL(2,2)=CON(13)*XT(1,2)+CON(16)*XT(2,2)+CON(19)*XT(3,2)
      XL(3,2)=CON(14)*XT(1,2)+CON(17)*XT(2,2)+CON(20)*XT(3,2)
      XL(1,3)=CON(12)*XT(1,3)+CON(15)*XT(2,3)+CON(18)*XT(3,3)
      XL(2,3)=CON(13)*XT(1,3)+CON(16)*XT(2,3)+CON(19)*XT(3,3)
      XL(3,3)=CON(14)*XT(1,3)+CON(17)*XT(2,3)+CON(20)*XT(3,3)
C
      TZ=XL(1,1)*XL(1,1)
      T(1)=XL(2,1)*XL(2,1)
      T(2)=XL(3,1)*XL(3,1)
      T(3)=T(1)+T(2)
      T(4)=SQRTF(T(3))
      T(5)=TZ+T(1)+T(2)
      T(6)=XL(1,1)*XL(1,2)
      T(7)=XL(2,1)*XL(2,2)
      T(8)=XL(3,1)*XL(3,2)
      T(9)=T(7)+T(8)
      POLAR(1,1)=SQRTF(T(5))
      POLAR(2,1)=ATANQD(XL(1,1),T(4))
      POLAR(3,1)=ATANQD(XL(3,1),XL(2,1))
      POLAR(1,2)=(T(6)+T(9))/POLAR(1,1)
      POLAR(2,2)=(XL(1,2)*POLAR(1,1)-XL(1,1)*POLAR(2,1))
     */(POLAR(1,1)*T(4))
```

```
       POLAR(3,2)=(XL(2,1)*XL(3,2)-XL(3,1)*XL(2,2))/T(3)
       POLAR(1,3)=(XL(1,2)*XL(1,2)+XL(2,2)*XL(2,2)+XL(3,2)*XL(3,2)
      *+XL(1,1)*XL(1,3)+XL(2,1)*XL(2,3)+XL(3,1)*XL(3,3)
      *-POLAR(2,1)*POLAR(2,1))/POLAR(1,1)
       XX=T(4)*(POLAR(1,1)*XL(1,3)-POLAR(3,1)*XL(1,1))
       YY=POLAR(2,2)*(POLAR(1,1)*T(9)+POLAR(2,1)*T(3))
       POLAR(2,3)=(XX-YY)/(POLAR(1,1)*T(3))
       POLAR(3,3)=(XL(2,1)*XL(3,3)-XL(3,1)*XL(2,3)-CONZ*POLAR(2,3)*T(9))
      */T(3)
       RETURN
       END

          3200 FORTRAN DIAGNOSTIC RESULTS - FOR   LT01

NO ERRORS
```

```
      SUBROUTINE LT03(INDEX,RANGE,HOUR,DECLIN,IFLAG)
C
C     PROGRAM LT03--TARGET DYNAMICS ESTIMATOR/EXTRAPOLATOR
C
      COMMON CONZ,CON(100),TZ,T(50),XT(3,3),XL(3,3),POLAR(3,3),IXT,
     X      IXTA,IXTB
      COMMON R(3),HA(3),DEC(3),S(3,3,2),DS(3,5),RR(3,2),NSAM,KSA,KSB
      COMMON/DATA/ RK(4,50),HK(4,50),DK(4,50)
      DIMENSION ALPH(5,3,3),AK(4,50,3),CALPH(5,3),MODE(3),Z(3)
      EQUIVALENCE (RK(1),AK(1))
      IK=INDEX-NSAM*(INDEX/NSAM)
      IF(IK.EQ.0)1,2
    1 IK=NSAM
    2 JIK=IK-(NSAM/3)*(IK/(NSAM/3))
      IF(JIK.EQ.0)3,4
    3 JIK=NSAM/3
    4 IF(INDEX.LE.NSAM)5,6
    5 IF(IK.LE.NSAM/3)8,7
    6 IF(JIK.EQ.1)11,12
    7 IF(IK.LE.2*NSAM/3)9,10
    8 MODE(1)=1
      MODE(2)=MODE(3)=0
      GO TO 12
    9 MODE(1)=2
      MODE(2)=1
      MODE(3)=0
      GO TO 12
   10 MODE(1)=3
      MODE(2)=2
      MODE(3)=1
      GO TO 12
   11 MODET=MODE(3)
      MODE(3)=MODE(2)
      MODE(2)=MODE(1)
      MODE(1)=MODET
   12 DO 13 K=1,3
      IF(IFLAG.GT.0)15,14
   15 IF(K.EQ.1)17,16
   16 IF(K.EQ.2)18,19
   17 ZN=RANGE
      GO TO 20
   18 ZN=HOUR
      GO TO 20
   19 ZN=DECLIN
   20 DO 21 IE=1,3
      MODOP=MODE(IE)
      IF(MODOP.EQ.0)21,23
   23 NPT=JIK+(MODOP-1)*NSAM/3
      TIE=FLOAT(NPT)/64.
      IF(NPT.EQ.1)24,25
   24 ALPH(5,IE,K)=ZN
      DO 22 J=1,4
   22 ALPH(J,IE,K)=0.
   25 IF(NPT.LE.2*NSAM/3)26,27
   26 NBR=NPT/KSA
      NNBR=NPT-KSA*NBR
      IF(NNBR.EQ.0)28,21
   28 DO 34 J=1,4
   34 ALPH(J,IE,K)=ALPH(J,IE,K)+AK(J,NBR,K)*(ZN-ALPH(5,IE,K))
      GO TO 21
```

```
27 NBRR=(NPT-2*NSAM/3)/KSB
   NNBRR=(NPT-2*NSAM/3)-KSB*NBRR
   IF(NNBRR.EQ.0)29,30
29 XPI=ALPH(4,IE,K)
   DO 31 I=2,4
   IND=5-I
31 XPI=ALPH(IND,IE,K)+XPI*TIE
   NGN=NBR+NBRR
   DO 32 I=1,4
32 ALPH(I,IE,K)=ALPH(I,IE,K)+AK(I,NGN,K)*(ZN-XPI-ALPH(5,IE,K))
30 DO 33 I=1,5
33 CALPH(I,K)=ALPH(I,IE,K)
21 CONTINUE
36 IF(IFLAG.GT.0)14,13
14 TK=FLOAT(2*NSAM/3+JIK)/64.
   Z(1)=CALPH(5,K)+CALPH(1,K)+TK*(CALPH(2,K)+TK*(CALPH(3,K)+TK*CALPH
  X      (4,K)))
   Z(2)=CALPH(2,K)+TK*(2.*CALPH(3,K)+TK*3.*CALPH(4,K))
   Z(3)=2.*CALPH(3,K)+TK*6.*CALPH(4,K)
   IF(K.EQ.1)40,41
40 DO 42 L=1,3
42 R(L)=Z(L)
   GO TO 47
41 IF(K.EQ.2)43,45
43 DO 44 L=1,3
44 HA(L)=Z(L)
   GO TO 47
45 DO 46 L=1,3
46 DEC(L)=Z(L)
47 IF(IFLAG.GT.0)13,48
48 ZN=Z(1)
   GO TO 20
13 CONTINUE
   RETURN
   END

       3200 FORTRAN DIAGNOSTIC RESULTS - FOR   LT03


NULL STATEMENT NUMBERS
     36
```

```
                                   31/32/3300 FORTRAN (2.2)/MSOS   04/25/68
      SUBROUTINE LT04(INDEX,VHST,VDST,DHIN,DDIN,IHOUT,IDOUT,CD)
C     PROGRM LT04,CALCULATE LASER TRACKER ANGLE SERVO CONTROL.
C
C     INPUTS ARE-
C     1. CON=LIST OF CONSTANTS
C     2. INDEX=COUNT OF BASIC TIME INTERRUPTS-
C        (INCRAMENTS EVERY 64TH OF A SECOND, STARTS AT ZERO)
C     3. HA=HOUR ANGLE VECTOR (FROM ESTIMATOR-ARCSECS)
C     4. DEC= DECLINATION ANGLE VECTOR (FROM ESTIMATOR-ARCSECS)
C     6. VHST=RAW HOUR ANGLE STAR TRACKER INPUT (GAIN BITS)
C     7. VDST=RAW DECLINATION ANGLE STAR TRACKER INPUT(GAIN BITS)
C        (VHST OR VDST DIVIDED BY XKHST*XKHC1 GIVES RADIANS)
C     8. DHIN=RAW HOUR ANGLE TACHOMETER INPUT (GAIN BITS/SECOND)
C     9. DDIN=RAW DECLINATION ANGLE TACHOMETER INPUT (GAIN BITS/SECOND)
C        (DHIN OR DDIN DIVIDED BY XKHT*XKHC2 GIVES RADIANS/SECOND)
C
C     OUTPUTS ARE-
C     1. IH= HOUR ANGLE SERVO DRIVE OUTPUT  (BITS)
C     2.   ID= DECLINATTON ANGLE SERVO DRIVE OUTPUT (BITS)
C
      COMMON CONZ,CON(100),TZ,T(50),XT(3,3),XL(3,3),POLAR(3,3)
      COMMON IXT,IXTA,IXTB
      COMMON R(3),HA(3),DEC(3),S(3,3,2),DS(3,5),RR(3,2)
      COMMON NSAM,KSA,KSB
      EQUIVALENCE(CON(27),XKH1 )
      EQUIVALENCE(CON(28),XKD1 )
      EQUIVALENCE(CON(31),EHRL)
      EQUIVALENCE(CON(31),EDRL)
      EQUIVALENCE(CON(32),EHTL)
      EQUIVALENCE(CON(32),EDTL)
      EQUIVALENCE(CON(34),XKHC1)
      EQUIVALENCE(CON(34),XKDC1)
      EQUIVALENCE(CON(36),XKHR)
      EQUIVALENCE(CON(36),XKDR)
      EQUIVALENCE(CON(37),XKHST)
      EQUIVALENCE(CON(37),XKDST)
      EQUIVALENCE(CON(38),XKHT)
      EQUIVALENCE(CON(38),XKDT)
      EQUIVALENCE(CON(39),TP)
      EQUIVALENCE(CON(41),WHCP)
      EQUIVALENCE(CON(41),WDCP)
      EQUIVALENCE(CON(42),WHIP)
      EQUIVALENCE(CON(42),WDIP)
      EQUIVALENCE(CON(43),WHIR)
      EQUIVALENCE(CON(43),WDIR)
      EQUIVALENCE(CON(44),ALPHA)
      EQUIVALENCE(CON(46),ASTOR)
C
C
      IF(INDEX.EQ.0)21,25
C     INITIALIZATION IF INDEX IS ZERO
   21 HIT=HIV=0
      XVHST=EH1=EHR3=XEHR2=EHR2=0.
      XVDST=ED1=EDR3=XEDR2=EDR2=0.
C***  PROCESS THE HOUR ANGLE
   25 IF(HIT.NE.0.)1,3
    1 X=VHST-XVHST
      WHCPL=XKH1*TP/(ABSF(X)+1.)
      IF(WHCPL.GE.WHCP)3,4
    3 WHCP1=WHCP
```

```
      GO TO 5
    4 WHCP1=WHCPL
    5 EHRC=(EH1+WHCP1*VHST)/CD+XKHST*XKHC1*HA(2)
      CALL LIMIT(EHRC,EHRL,HIV)
C
      IF(HIV.EQ.0.)2,9
    9 EH2=0.
    2 EHR1=EHRC-XKHR*DHIN
      HOUT=EHR1+EH2
C
      CALL LIMIT(HOUT,EHTL,HIT)
      IF(HIT.EQ.0.)6,10
    6 IF(HIV.EQ.0.)7,8
    7 EH1=EH1+.5*WHCP*WHIP*TP*(3.*VHST-XVHST)
      XVHST=VHST
    8 EHR2=EHR1-EHR3
      EHR3=EHR3+.5*WHIR*TP*(3.*EHR2-XEHR2)/ALPHA
      XEHR2=EHR2
      EH2=(ALPHA-1.)*EHR3
   10 CONTINUE
C***  PROCESS THE DECLINATION ANGLE
      IF(DIT.NE.0.)11,13
   11 X=VDST-XVDST
      WDCPL=XKD1*TP/(ABSF(X)+1.)
      IF(WDCPL.GE.WDCP)13,14
   13 WDCP1=WDCP
      GO TO 15
   14 WDCP1=WDCPL
C
   15 EDRC=ED1+WDCP1*VDST +XKDST*XKDC1*DEC(2)
      CALL LIMIT(EDRC,EDRL,DIV)
      IF(DIV.EQ.0.)12,19
   19 ED2=0.
   12 EDR1=EDRC-XKDR*DDIN
      DOUT=EDR1+ED2
C
      CALL LIMIT(DOUT,EDTL,DIT)
      IF(DIT.EQ.0.)16,20
   16 IF(DIV.EQ.0)17,18
   17 ED1=ED1+.5*WDCP*WDIP*TP*(3.*VDST-XVDST)
      XVDST=VDST
   18 EDR2=EDR1-EDR3
      EDR3=EDR3+.5*WDIR*TP*(3.*EDR2-XEDR2)/ALPHA
      XEDR2=EDR2
      ED2=(ALPHA-1.)*EDR3
C
   20 CONTINUE
      IHOUT=HOUT
      IDOUT=DOUT
      RETURN
      END
```

```
          3200 FORTRAN DIAGNOSTIC RESULTS - FOR   LT04

NO ERRORS
```

```
                                        31/32/3300 FORTRAN (2.2)/MSOS   04/25/68
          SUBROUTINE LIMIT(VALUE,BOUND,FLAG)
    C     LIMIT-SUBROUTINE OF LT04
    C     FOR LASER TRACKER PROGRAM
    C
          IF(VALUE.GE.BOUND)1,2
        1 VALUE=BOUND
          FLAG=1.
          GO TO 10
        2 IF(VALUE+BOUND.GE.0.)3,4
        3 FLAG=0.
          GO TO 10
        4 VALUE=-BOUND
          FLAG=1.
       10 CONTINUE
          RETURN
          END

            3200 FORTRAN DIAGNOSTIC RESULTS - FOR   LIMIT

NO ERRORS
```

```
                                    31/32/3300 FORTRAN (2.2)/MSOS   04/25/68
      SUBROUTINE LT05(INDEX,IMAX3,I1,I2,I3,J1,J2,J3,RANGE,RGNRATE)
C
C     LT05-RANGE SERVO ROUTINE
C
C     THIS PROGRAM COMPUTES THE RANGE AND RANGE RATE AND THE RANGE SERVO
C        ERROR SIGNALS
C     INPUTS ARE-
C     1. INDEX=COUNT OF BASIC TIME INTERRUPTS-
C              (INCREMENTS EVERY 64TH OF A SECOND,STARTS AT ZERO)
C     2. IMAX3=COUNT OF RANGE INTERRUPTS SINCE LAST INDEX-
C              (STARTS AT ZERO,MAX SIZE IS 3,RESETS WHEN INDEX CHANGES)
C     3.  I1, I2, I3=THE THREE INPUTS FROM THE RANGE CHANNELS-
C
C     OUTPUTS ARE-
C     1.  J1, J2, J3=THE THREE OUTPUTS TO THE RANGE CHANNELS.
C     2. RANGE= THE CALCULATED RANGE IN METERS.
C     3. RGNRATE= THE CALCULATED RANGE RATE IN METERS / SEC.
C
C
      IF(INDEX.EQ.0)10,12
   10 IF(IMAX3.EQ.0)11,12
C     INITIALIZATION IF INDEX AND IMAX3 ARE ZERO
   11 XEE1=XEE2=XRRDOT=XRDD=EE2=0.
      XRR= RANGE*800.
C
   12 IF(IABS(I3).LE.510)1,2
    1 EE1=I3
      GO TO 5
    2 IF(IABS(I2).LE.510)3,4
    3 EE1=I2*32
      GO TO 5
    4 EE1=I1*1024
    5 RRDOT=(EE1*30.+EE2)*12.
      EE2=XEE2+EE1*3.-XEE1
      XEE1=EE1
      XEE2=EE2
      RGNRATE=RRDOT*0.000104166
      RDD=(RRDOT-XRRDOT)*2.6
      X=(RDD+XRDD)*.5
      RR=XRR+(RRDOT*3.-XRRDOT)*0.000166666
      RANGE=RR*.00125 +X*0.00002547
C
      XRRDOT=RRDOT
      XRDD=RDD
      ICHNG=(RR-XRR)
      XRR=XRR+ICHNG
      J1=ICHNG*0.009765625
      J2=ICHNG*0.031250
      J3=ICHNG
      RETURN
      END

       3200 FORTRAN DIAGNOSTIC RESULTS - FOR   LT05

NO ERRORS
```

POLFITW

Program to compute the weighted least-squares fit of a polynomial to data in arrays AX and AY. The fit starts at IMIN, includes every (IS)th point until IMAX.

INPUTS

1. AX--Array of values of independent variable.
2. AY--Array of values of dependent variable.
3. IMIN--Location in arrays AX and AY at which fit is to begin.
4. IMAX--Location in arrays AX and AY at which fit is to end.
5. IS--Interval between those points in arrays AX and AY which are to be included in fitting.
6. NDEG1, NDEG2, NS--Every (NS)th polynomial from degree NDEG1 to degree NDEG2 is fitted to the data.
7. NPRINT = -1--No printout

   = 0--Print coefficients and bounds on error at fitting points

   = +1--Print coefficients, error bounds, and table with AX and AY arrays, values for dependent variable computed from polynomial, and the error at each fitting point.

Uses subroutines CHOLSQR, MATMP, and GINV, and function R.

```
                                     31/32/3300 FORTRAN (2.2)/MSOS   04/25/68
      SUBROUTINE POLFITW(AX,AY,IMIN,IMAX,IS,NDEG1,NDEG2,NS,NPRINT,C)
      DIMENSION AX(1),AY(1),A(200) ,U(100),AFLAG(10),ATEMP(10),C(10),X(2
     10),Y(20),YE(8,20),YA(20),SINV(400)
      COMMON STEMP(200)
      EXTERNAL R
      IF(IMIN) 21,20,21
   20 IMIN=IMIN+1
   21 NR=(IMAX-IMIN)/IS +1
      DO 1 I=1,NR
      IXY = IMIN+(I-1)*IS
      X(I)=AX(IXY)
    1 Y(I)=AY(IXY)
      DELX=X(2)-X(1)
      NC1=NDEG1+1
      NC2=NDEG2+1
      DO 30 NC=NC1,NC2,NS
      YMIN=0.
      YMAX=0.
      DO 2 I=1,NR
      BX=1.
      DO 2 J=1,NC
      IND=I+(J-1)*NR
      A(IND)=BX
    2 BX=BX*X(I)
      CALL CHOLSQR(R,SINV,NR,DELX)
      CALL MATMP(SINV,NR,NR,A,NC,STEMP)
      CALL GINV(STEMP,U,AFLAG,ATEMP,NR,NC)
      CALL MATMP(SINV,NR,NR,Y,1,YA)
      CALL MATMP(YA,1,NR,STEMP,NC,C)
      IF(NPRINT) 50,51,51
   51 NCM1=NC-1
      PRINT 3,NCM1
    3 FORMAT(/1X,16HCOEFFICIENTS FOR,I3,63HTH DEGREE POLYNOMIAL FIT IN F
     1ORM Y=C(0) + C(1)X + C(2)X**2,ETC.)
      PRINT 4,(C(I),I=1,NC)
    4 FORMAT(6(E20.9))
   50 CONTINUE
      DO 5 I=1,NR
      YA(I)=C(NC)
      DO 5 J=2,NC
      IND=NC+1-J
    5 YA(I)=C(IND)+YA(I)*X(I)
      IP=NC-NC1+1
      DO 6 I=1,NR
    6 YE(IP,I)=Y(I)-YA(I)
      DO 7 I=1,NR
      IF(YMIN-YE(IP,I)) 9,9,8
    8 YMIN=YE(IP,I)
      GO TO 7
    9 IF(YE(IP,I)-YMAX) 7,7,11
   11 YMAX=YE(IP,I)
    7 CONTINUE
      IF(NPRINT) 30,45,40
   40 PRINT 10
   10 FORMAT(10X,5H X(I),15X,5H Y(I),15X,12H APPROX Y(I),8X,30H ERROR IN
     1 FIT=Y(I)-APPROX Y(I))
      DO 13 I=1,NR
      PRINT 12,X(I),Y(I),YA(I),YE(IP,I)
   12 FORMAT(4(E20.9,3X))
   13 CONTINUE
```

```
      45 PRINT 46,YMIN,YMAX
      46 FORMAT(/1X,39HERROR IN FIT CONFINED TO THE INTERVAL (,1X,E17.9,1H,
        1,1X,E17.9,1X,1H)/)
      30 CONTINUE
         RETURN
         END

            3200 FORTRAN DIAGNOSTIC RESULTS - FOR   POLFITW

NO ERRORS
```

GAIN

This program computes the gain arrays to be used in SUBROUTINE LT03.

INPUTS

1. NSAM--For the current version of LT03, NSAM = 50. It is the total number of columns of the gain array, or the total number of measurements actually processed.

2. NBATCH--Presently set at 20. It is the number of measurements processed in the batch processing phase.

3. SIGMA, TAU--The measurement noise is assumed modeled as (SIGMA**2) *EXP(-TAU*ABS(T)). For batch processing, set T = 3/64. For sequential processing, set T = 1/64.

4. NPRINT--Equals 0 implies no printout. Equals +1 results in printout of the gain array.

OUTPUT

1. AK--Dummy variable for external gain array.

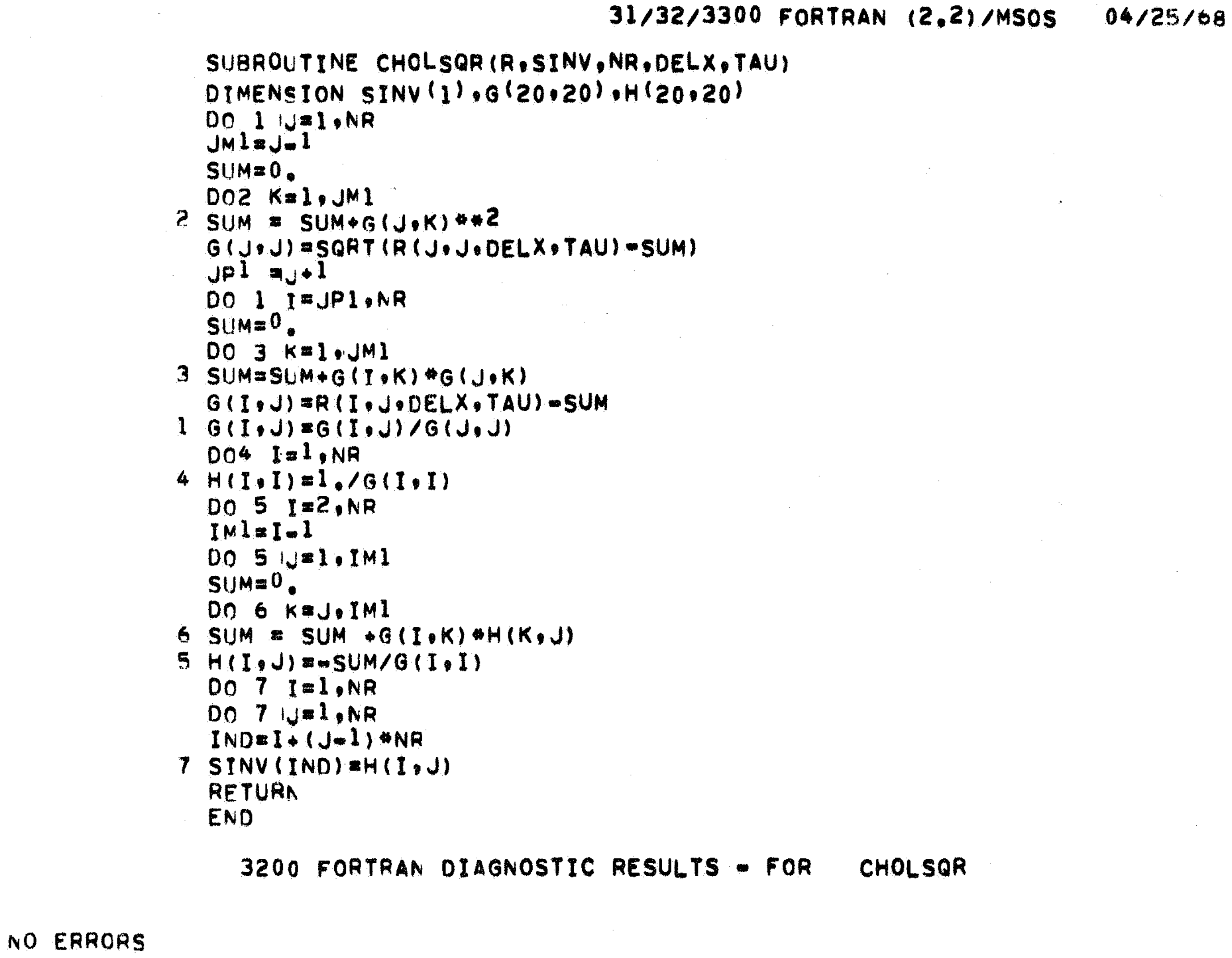
This program calls CHOLSQR, R, MATMP, and GINV.

```
                                          31/32/3300 FORTRAN (2.2)/MSOS   04/25/68
      SUBROUTINE GAIN(AK,NSAM,NBATCH,SIGMA,TAU,NPRINT)
      DIMENSION P(16),U(16),AFLAG(4),ATEMP(4),H(4),AK(5,150),AKW(4),B(4)  **
     1,F(4),D(16),ALPH(4),HH(240),SINV(400),STEMP(80),TEMP(80)
      EXTERNAL R
      T=3./64.
      DO 1 I=1,NBATCH
      BX=1.
      XT=FLOAT(I)*T
      DO 1 J=1,4
      IND=I+(J-1)*NBATCH
      HH(IND)=BX
    1 BX=BX*XT
      CALL CHOLSQR(R,SINV,NBATCH,T,TAU)
      CALL MATMP(SINV,NBATCH,NBATCH,HH,4,STEMP)
      DO 20 I=1,NBATCH
      DO 20 J=1,4
      IND=I+(J-1)*NBATCH
      INDT=J+(I-1)*4
   20 TEMP(INDT)=STEMP(IND)
      CALL MATMP(TEMP,4,NBATCH,STEMP,4,P)
      CALL GINV(P,U,AFLAG,ATEMP,4,4)
      CALL GINV(STEMP,U,AFLAG,ATEMP,NBATCH,4)
      DO 21 I=1,NBATCH
      DO 21 J=1,4
      IND=I+(J-1)*NBATCH
      INDT=J+(I-1)*4
   21 TEMP(INDT)=STEMP(IND)
      CALL MATMP(TEMP,4,NBATCH,SINV,NBATCH,STEMP)
      DO 3 I=1,4
      DO 3 J=1,NBATCH
      IND=I+(J-1)*4
    3 AK(I,J)=STEMP(IND)
      DO 6 I=1,16
    6 P(I)=(SIGMA**2)*P(I)
      H(1)=1.
      T=1./64.
      NBATCH1=NBATCH+1
      DO 210 K=NBATCH1,NSAM
      KT=K+40
      DO 211 I=2,4                                                        **
  211 H(I)=(FLOAT(KT)*T)**(I-1)
      CALL MATMP(P,4,4,H,1,B)                                             **
      CALL MATMP(H,1,4,B,1,F)                                             **
      DO 212 I=1,4                                                        **
  212 AKW(I)=B(I)/(F(1)+SIGMA**2)
      DO 213 J=1,4                                                        **
  213 AK (J,K)=AKW(J)
      CALL MATMP(AKW,4,1,H,4,U)                                           **
      DO 216 I=1,16
  216 U(I)=-U(I)
      DO 214 I=1,4                                                        **
      IND=I+(I-1)*4                                                       **
  214 U(IND)=1.+U(IND)
      CALL MATMP(U,4,4,P,4,D)                                             **
      DO 215 I=1,16                                                       **
  215 P(I)=D(I)                                                           **
  210 CONTINUE                                                            **
      IF(NPRINT.EQ.0) 8,9
    9 PRINT 4,((AK(I,J),I=1,4),J=1,NSAM)
    4 FORMAT(1X,2HAK,4E19.8)
```

```
          8 RETURN
            END
              3200 FORTRAN DIAGNOSTIC RESULTS - FOR   GAIN

NO ERRORS
```

CHOLSQR

This program computes SINV, the inverse of the lower triangular square root of the positive-definite matrix R. NR is the size of R. It is assumed that R is generated by the function subroutine R (with DELX ≡ D).



```
                                    31/32/3300 FORTRAN (2.2)/MSOS   04/25/68
          SUBROUTINE CHOLSQR(R,SINV,NR,DELX,TAU)
          DIMENSION SINV(1),G(20,20),H(20,20)
          DO 1 J=1,NR
          JM1=J-1
          SUM=0.
          DO2 K=1,JM1
        2 SUM = SUM+G(J,K)**2
          G(J,J)=SQRT(R(J,J,DELX,TAU)-SUM)
          JP1 =J+1
          DO 1 I=JP1,NR
          SUM=0.
          DO 3 K=1,JM1
        3 SUM=SUM+G(I,K)*G(J,K)
          G(I,J)=R(I,J,DELX,TAU)-SUM
        1 G(I,J)=G(I,J)/G(J,J)
          DO4 I=1,NR
        4 H(I,I)=1./G(I,I)
          DO 5 I=2,NR
          IM1=I-1
          DO 5 J=1,IM1
          SUM=0.
          DO 6 K=J,IM1
        6 SUM = SUM +G(I,K)*H(K,J)
        5 H(I,J)=-SUM/G(I,I)
          DO 7 I=1,NR
          DO 7 J=1,NR
          IND=I+(J-1)*NR
        7 SINV(IND)=H(I,J)
          RETURN
          END
            3200 FORTRAN DIAGNOSTIC RESULTS - FOR   CHOLSQR

NO ERRORS
```

R

This function subroutine generates the correlation matrix R for samples from a noise process with autocorrelation function exp(-TAU|T|). The element R(I,J) of R is returned for a sample interval of D seconds.

```
                                   31/32/3300 FORTRAN (2.2)/MSOS   04/25/68
          FUNCTION R(I,J,D,TAU)
          A=EXP(-TAU*D)
          NEXP=IABS(I-J)
          R=A**NEXP
          RETURN
          END
            3200 FORTRAN DIAGNOSTIC RESULTS - FOR   R

NO ERRORS
```

GINV

This program forms the generalized inverse of a matrix.

INPUTS

1. A, NR, NC--A is the matrix to be inverted, with NR rows and NC columns. The external array must be single subscripted and stored by columns.

2. U--Bookkeeping array with NC**2 locations.

3. AFLAG, ATEMP--Temporary storage arrays, each with NC locations

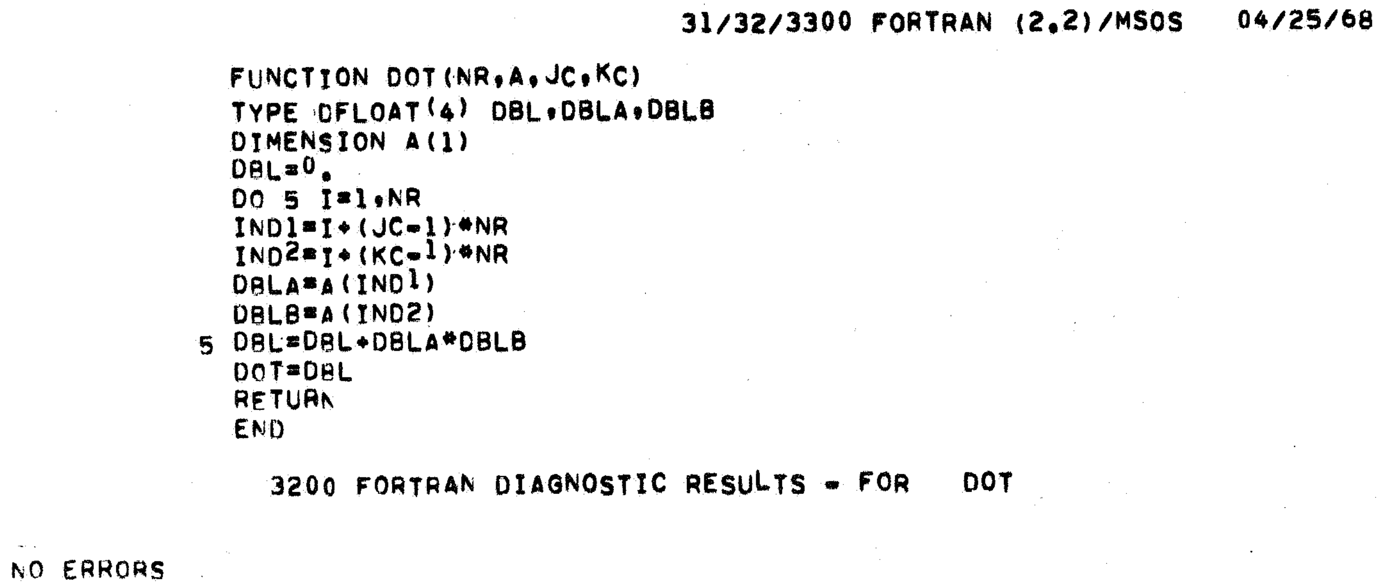
The transpose of the generalized inverse of A is stored back in A upon return. This program calls DOT.

```
                                   31/32/3300 FORTRAN (2.2)/MSOS   04/25/68
      SUBROUTINE GINV(A,U,AFLAG,ATEMP,NR,NC)
      DIMENSION A(1),U(1),AFLAG(1),ATEMP(1)
      DO 10 I=1,NC
      DO 5 J=1,NC
      IND1=I+(J-1)*NC
      IND2=I+(I-1)*NC
    5 U(IND1)=0.
   10 U(IND2)=1.
      FAC=DOT(NR,A,1,1)
      FAC=1.0/SQRTF(FAC)
      DO 15 I=1,NR
   15 A(I)=A(I)*FAC
      DO 20 I=1,NC
   20 U(I)=U(I)*FAC
      AFLAG(1)=1.0
      TOL1=14551915228.E-20
      TOL=TOL1*TOL1
      DO 100 J=2,NC
      DOT1=DOT(NR,A,J,J)
      JM1=J-1
      DO 50 L=1,2
      DO 30 K=1,JM1
   30 ATEMP(K)=DOT(NR,A,J,K)
      DO 45 K=1,JM1
      DO 35 I=1,NR
      IND1=I+(J-1)*NR
      IND2=I+(K-1)*NR
   35 A(IND1)=A(IND1)-ATEMP(K)*A(IND2)*AFLAG(K)
      DO 40 I=1,NC
      IND1=I+(J-1)*NC
      IND2=I+(K-1)*NC
   40 U(IND1)=U(IND1)-ATEMP(K)*U(IND2)
   45 CONTINUE
   50 CONTINUE
      DOT2=DOT(NR,A,J,J)
      IF(DOT2/DOT1-TOL) 55,55,70
   55 DO 60 I=1,JM1
      ATEMP(I)=0.
      DO 60 K=1,I
      IND1=K+(I-1)*NC
      IND2=K+(J-1)*NC
   60 ATEMP(I)=ATEMP(I)+U(IND1)*U(IND2)
      DO 65 I=1,NR
      IND1=I+(J-1)*NR
      A(IND1)=0.
      DO 65 K=1,JM1
      IND2=I+(K-1)*NR
   65 A(IND1)=A(IND1)-A(IND2)*ATEMP(K)*AFLAG(K)
      AFLAG(J)=0.
      FAC=DOT(NC,U,J,J)
      FAC=1./SQRTF(FAC)
      GO TO 75
   70 AFLAG(J)=1.
      FAC=1./SQRTF(DOT2)
   75 DO 80 I=1,NR
      IND=I+(J-1)*NR
   80 A(IND)=A(IND)*FAC
      DO 85 I=1,NC
      IND=I+(J-1)*NC
   85 U(IND)=U(IND)*FAC
```

```
100 CONTINUE
    DO 130 J=1,NC
    DO 130 I=1,NR
    IND1=I+(J-1)*NR
    FAC=0.
    DO 120 K=J,NC
    IND2=I+(K-1)*NR
    IND3=J+(K-1)*NC
120 FAC=FAC+A(IND2)*U(IND3)
130 A(IND1)=FAC
    RETURN
    END

      3200 FORTRAN DIAGNOSTIC RESULTS - FOR    GINV

NO ERRORS
```

DOT

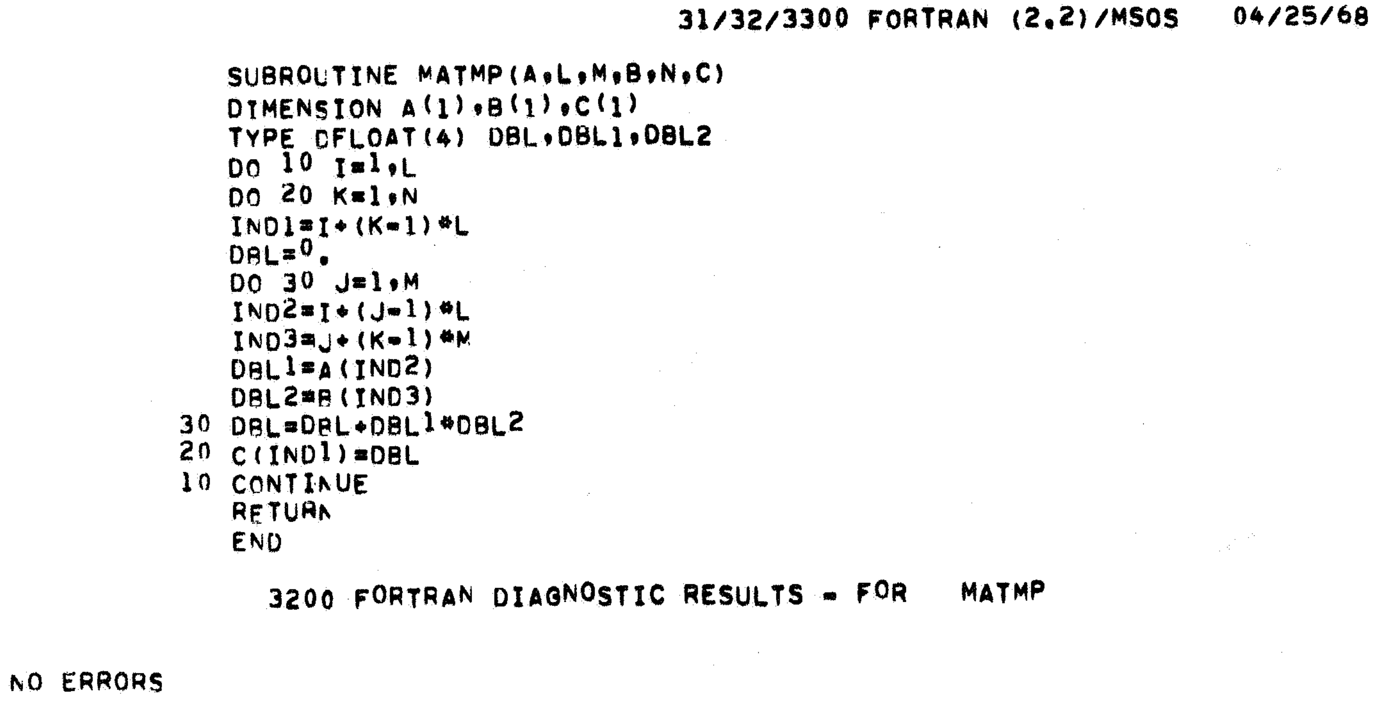
Called by GINV, this function subprogram forms the dot product with double precision accumulation.



```
                                          31/32/3300 FORTRAN (2.2)/MSOS   04/25/68
          FUNCTION DOT(NR,A,JC,KC)
          TYPE DFLOAT(4) DBL,DBLA,DBLB
          DIMENSION A(1)
          DBL=0.
          DO 5 I=1,NR
          IND1=I+(JC-1)*NR
          IND2=I+(KC-1)*NR
          DBLA=A(IND1)
          DBLB=A(IND2)
        5 DBL=DBL+DBLA*DBLB
          DOT=DBL
          RETURN
          END
            3200 FORTRAN DIAGNOSTIC RESULTS - FOR   DOT

NO ERRORS
```

MATMP

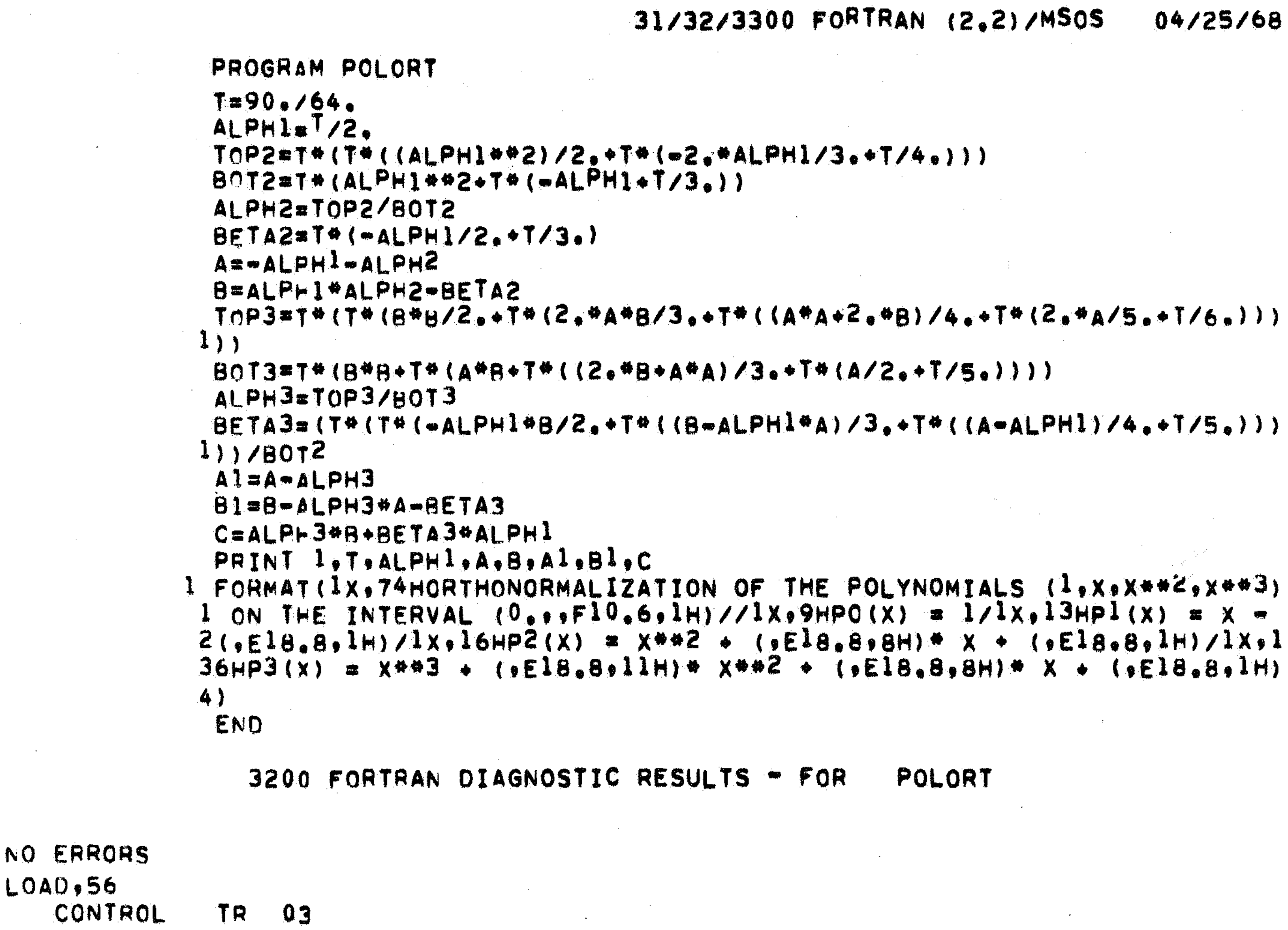
This program forms the matrix product C = AB where A is L x M and B is M x N. In the calling program the actual arrays corresponding to the dummy arrays A, B, and C must be single subscripted and stored by columns. Double precision accumulation of all inner products is used.



```
                                       31/32/3300 FORTRAN (2.2)/MSOS   04/25/68

            SUBROUTINE MATMP(A,L,M,B,N,C)
            DIMENSION A(1),B(1),C(1)
            TYPE DFLOAT(4) DBL,DBL1,DBL2
            DO 10 I=1,L
            DO 20 K=1,N
            IND1=I+(K-1)*L
            DBL=0.
            DO 30 J=1,M
            IND2=I+(J-1)*L
            IND3=J+(K-1)*M
            DBL1=A(IND2)
            DBL2=B(IND3)
         30 DBL=DBL+DBL1*DBL2
         20 C(IND1)=DBL
         10 CONTINUE
            RETURN
            END

              3200 FORTRAN DIAGNOSTIC RESULTS - FOR   MATMP

NO ERRORS
```

POLORT

This subroutine orthonormalizes the polynomials 1, X, X^2, and X^3 on the interval [0,T]. The orthonormal polynomials P_0, P_1, P_2, and P_3 are printed out.



```
                              31/32/3300 FORTRAN (2.2)/MSOS   04/25/68

      PROGRAM POLORT
      T=90./64.
      ALPH1=T/2.
      TOP2=T*(T*((ALPH1**2)/2.+T*(-2.*ALPH1/3.+T/4.)))
      BOT2=T*(ALPH1**2+T*(-ALPH1+T/3.))
      ALPH2=TOP2/BOT2
      BETA2=T*(-ALPH1/2.+T/3.)
      A=-ALPH1-ALPH2
      B=ALPH1*ALPH2-BETA2
      TOP3=T*(T*(B*B/2.+T*(2.*A*B/3.+T*((A*A+2.*B)/4.+T*(2.*A/5.+T/6.)))
     1))
      BOT3=T*(B*B+T*(A*B+T*((2.*B+A*A)/3.+T*(A/2.+T/5.))))
      ALPH3=TOP3/BOT3
      BETA3=(T*(T*(-ALPH1*B/2.+T*((B-ALPH1*A)/3.+T*((A-ALPH1)/4.+T/5.)))
     1))/BOT2
      A1=A-ALPH3
      B1=B-ALPH3*A-BETA3
      C=ALPH3*B+BETA3*ALPH1
      PRINT 1,T,ALPH1,A,B,A1,B1,C
    1 FORMAT(1X,74HORTHONORMALIZATION OF THE POLYNOMIALS (1,X,X**2,X**3)
     1 ON THE INTERVAL (0.,,F10.6,1H)//1X,9HP0(X) = 1/1X,13HP1(X) = X -
     2(,E18.8,1H)/1X,16HP2(X) = X**2 + (,E18.8,8H)* X + (,E18.8,1H)/1X,1
     36HP3(X) = X**3 + (,E18.8,11H)* X**2 + (,E18.8,8H)* X + (,E18.8,1H)
     4)
      END

        3200 FORTRAN DIAGNOSTIC RESULTS - FOR   POLORT

NO ERRORS
LOAD,56
   CONTROL   TR  03
```

APPENDIX B

COMPUTER CONTROL OF ANGLE SERVOS

RESEARCH REPORT NUMBER 541

COMPUTER CONTROL OF ANGLE SERVOS

By

T. Knutrud

December 1967

Project Nos. 609-01 & 180

APPLIED RESEARCH LABORATORY
SYLVANIA ELECTRONIC SYSTEMS
Sylvania Electric Products Inc.
40 Sylvan Road, Waltham, Massachusetts 02154

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

SECTION 1

INTRODUCTION

Conventional closed loop control of angular pointing servos for telescopes and steerable antennas is accomplished by special electronic controllers of analog or mixed analog/digital nature. A computer program may replace the electronic controller entirely and control the servos in a sampled data process. This may result in a significant cost saving when a digital computer is used as part of a telescope or antenna system, and computer time is available. It may also be possible to improve the control characteristics by computer programming.

A particular application is reported here of computer control to the pointing of a laser tracking telescope. This work has been carried out in part under NASA contract No. NAS8-21089 and applied to a cassegrain telescope and mount[2] to be delivered to NASA by Goerz Optical Co. Inc. Besides making use of available computer hardware to eliminate the need for an electronic controller the primary objective for this application was to:

a) improve tracking performance by the use of rate feed-forward signals available as a result of other data processing

b) to provide additional control for optimum performance of the servos under conditions of servo saturation.

Control equations are developed to achieve the above objectives and a final program is written. The results of an extensive digital simulation study are also reported to demonstrate the control stability under all conditions and to support the theoretical estimates of the tracking performance. Simulation results of linear and saturating step responses are given and accurate tracking performance is illustrated in tracking simulations of slowly moving targets and the tracking of a Saturn V launch.

SECTION 2

CONTROL LOOP DESCRIPTION

A simplified block diagram of the angle servo control loop is shown in Figure 1 for one of the two axis. Angular control of the telescope is accomplished by use of electric torque motors driving the telescope directly (no gear reducers) around the two axis of the equatorial mount. A solid state, high output impedance amplifier which is controlled by the output of a digital-to-analog converter provides a drive with an accurate translation of digital input to axis torque.

The motion of the telescope is measured by three sensors for each axis. The first is an angle encoder of high accuracy (22 bits) mounted on each axis. The second sensor is a tachometer also mounted directly at each axis provides an analog (voltage) signal proportional to axis rates. The third sensor is an electro-optical star tracker system providing an analog signal proportional to the difference between a telescope axis reference (or boresight) angle and the target angle during target tracking. The output of the latter two sensors are converted to provide all-digital signals for the telescope motion.

Accurate positioning of the telescope is accomplished by sample data computer control. The main routines of the computer program pertinent to a single axis angular control are indicated in the simplified flow diagram to the left of the dotted line in Fig. 1. The processes carried out in this program can be summarized in the following 5 classes:

1. Signal summation and compensation which are carried out for stable and accurate control in the linear dynamic range of the angle servos.

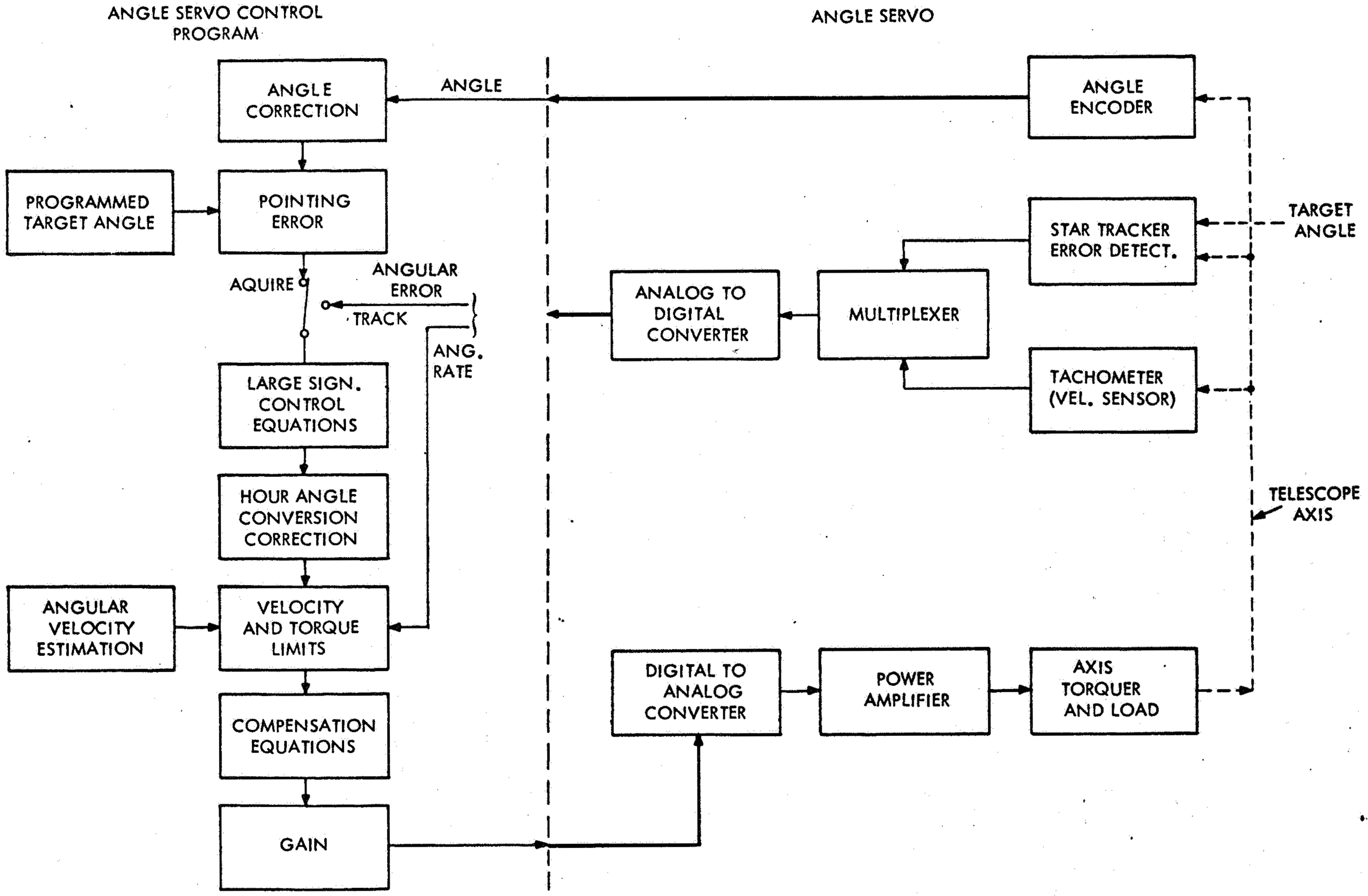


Figure 1. Block Diagram of the Hc'ir Angle Servo and Computer Control Program

2. Compensation for well-defined transducer sensitivity changes.

3. Compensation for servo saturation to improve control characteristics during large signal transients.

4. Correction of angular (encoder) measurements by means of calibration tables and interpolation techniques.

5. Angular position and rate estimation and smoothing for improved readings in the presence of noise.

The processes in the first three classes above are normally carried out by conventional analog servo controllers for antennas and telescopes. In 2) and 3), however, we are able to take significant advantage of the flexibility in computer programming due to the nonlinear characteristics of the processes involved. All of these processes will be discussed in detail in Sections 4 and 6. The process of 4) and 5) above may be considered separate and independant of the servo control loops and are considered in Reference 1.

SECTION 3

LINEAR CONTROL SYNTHESIS

A simplified block diagram of the linear part of the angle servo control loops is shown in Fig. 2. The axis drive consists of a power amplifier and a linear, direct drive torque motor. The amplifier has a high gain current loop resulting in a flat frequency response characteristic between input voltage and drive torque within the amplifier-motore bandwidth W_{ME}. This characteristic of the axis drive is described by the transfer function

$$G_D(s) = \frac{K_A}{J} \frac{W_{ME}}{\left(s + W_{ME}\right)} \tag{1}$$

The sampled data control process to be carried out by the computer introduces a phaseshift in the control loop of

$$\phi_s = -T_P\omega \tag{2}$$

where T_P is the sampling interval. The computational burden on the computer limits the value of T_P to no less than (1/64) second. With the axis drive bandwidth set conservatively at W_{ME} = 400 1/sec, a closed rate loop cross-over frequency of W_{CR} = 54 can be obtained with a phase margin of 35°.* Thus

$$K_R \simeq 54 \frac{J}{K_A} \frac{1}{K_T} \tag{3}$$

*This is probably the minimum phase margin for acceptable rate loop stability.

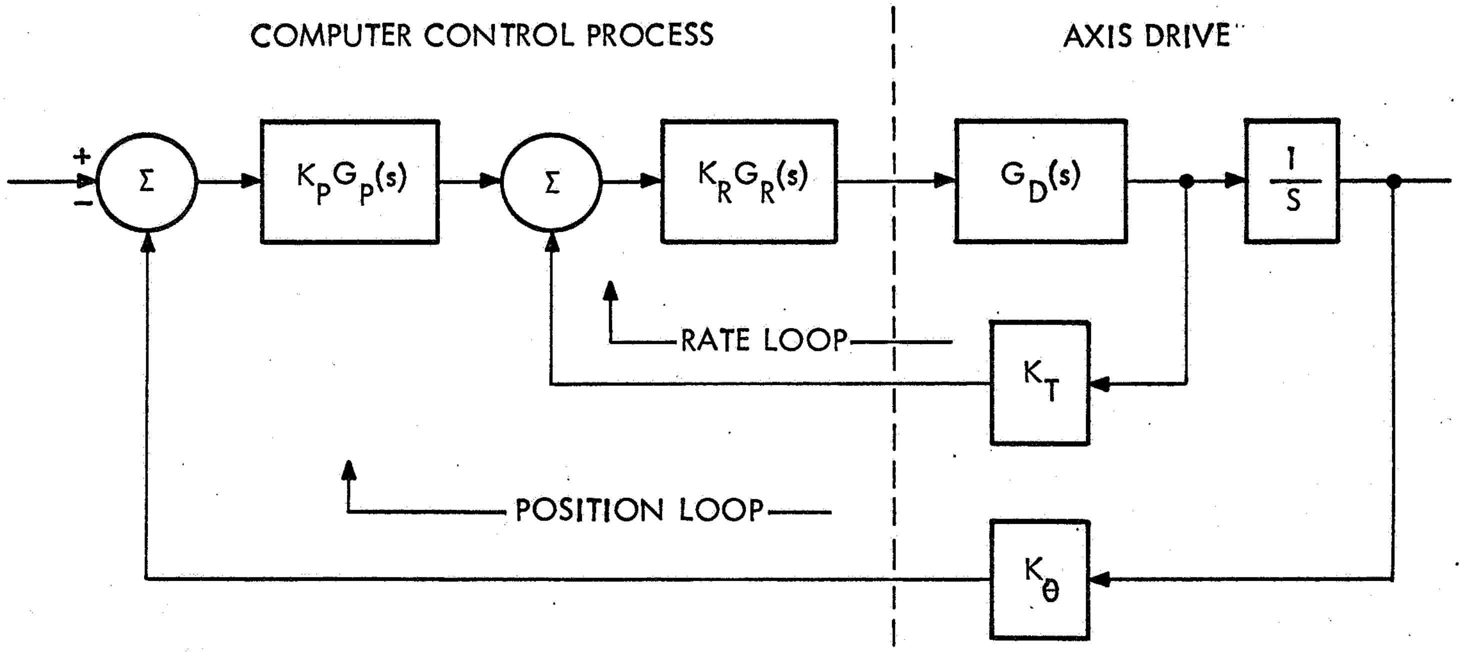


Figure 2. Simplified Block Diagram of a Single Axis Servo

Pointing errors that develop due to torque disturbances such as friction and wind loads are reduced substantially when integral compensation is used in the rate loop. When these disturbances are expected to be of relatively minor importance, such as in this case, the compensation may be of the form

$$G_R(s) = \frac{s + W_{IR}}{s + W_{IR/\alpha}} \tag{4}$$

which differ from the compensation with $\alpha = \infty$ by a finite low frequency gain and reduced additional phase lag for a given value of W_{CR}/W_{IR}. The latter ratio is usually picked at about 3 for near optimum compromise between rate loop stability and compensation bandwidth.

Compensation for the position loop is introduced to obtain a finite positional error during target acceleration. It can be readily shown that this error is

$$\theta_E = \frac{\ddot{\theta}_T}{W_{IP}W_{CP}} \tag{5}$$

where $\ddot{\theta}_T$ is a constant target acceleration. For near optimum control the following choice is usually made:

$$W_{CR} \simeq 3W_{CP} \simeq 9W_{IP} \tag{6}$$

Substituting (6) into (5) we may express the required minimum value for W_{CR} in terms of maximum target acceleration and maximum allowed error as follows

$$W_{CR} \geq \left[27. \frac{\left(\ddot{\theta}_T\right)_{max}}{\left(\theta_E\right)_{max}} \right]^{1/2} \qquad (7)$$

When rate feedforward is used in the absence of position loop compensation it can be shown in a similar way that the requirement on W_{CR} is now

$$W_{CR} \geq \left[3.0 \frac{\left(\ddot{\theta}_T\right)_{max}}{\left(\theta_E\right)_{max}} \right]^{1/2} \qquad (8)$$

where it is assumed that the rate command is an accurate derivative of θ_T.

The accuracy of the estimated target rate to be used for rate feedforward is not known. Also, slowly varying load torques and offset disturbances are present in the loops. For those reasons it is advisable to include position loop integral conpensation in addition to rate feedforward. Using the expected maxima of $\left(\ddot{\theta}_T\right)_{max} = .13^o/sec^2$ and $\left(\theta_E\right)_{max} = 1/3600$ degrees in (7) and (8) we get the following range for W_{CR}:

$$37 \leq W_{CR} < 112 \qquad (9)$$

where the lower value is to be applied when a perfect rate command is available. It is clear that the value of $W_{CP} = 54$ rad/sec selected for stability reasons puts a high requirement on the rate feedforward signal if the 1 sec tracking accuracy is to be met. It should be noted, however, that the above calculations are very conservative since

a) both rate feedforward and integral compensation can be expected to combine to improve the tracking accuracy

b) the .13°/sec^2 acceleration is the peak expected value and does not last long enough for the servo error to reach its corresponding final value.

These conclusions are born out by the simulations reported in Section 7.

Using integral-plus-proportional compensation for the position loop and the relation of (6), we get with $W_{CR} = 54$ the following:

$$K_p\, G_p(s) = 18\, \frac{K_T}{K_\theta}\, \frac{s+6}{s} \tag{10}$$

and with $\alpha = 10$ for the rate loop;

$$K_R\, G_R(s) = 54\, \frac{J}{K_A}\, \frac{s+18}{s+1.8} \tag{11}$$

SECTION 4

LINEAR SERVO MODEL

A detailed model of the hour angle servo is shown in Fig. 3. An H-subscript is used to denote hour angle constants and variables. With these changed to D-subscripts, Fig. 3 is also the model of the declination servo. The only significant difference between the two servos, other than differences in the constant values, is the "constant" K_{H3} which is unity for the declination angle. For the hour angle servo this is a gain which corrects for the variation of the star tracker sensitivity with the declination angle and takes the form

$$K_{H2} = \frac{1}{\cos \theta_D} \tag{12}$$

which is exact for small :racking errors. All other constants and variables for the two angle servos are listed in Tables 1 and 2 respectively.

The servos may be conveniently separated in 3 parts:

a) the axis power amplifier and motor (or "torquer"),

b) the angle sensors and A-to-D converting equipment,

c) the computer control program.

The control program shown in Fig. 3 is an exact model of the linear part of the desired control equations and has been used to facilitate the design of the computer program as outlined in the Appendix. Some approximations have been made. First, the power amplifier and motor combination is given a single order electrical

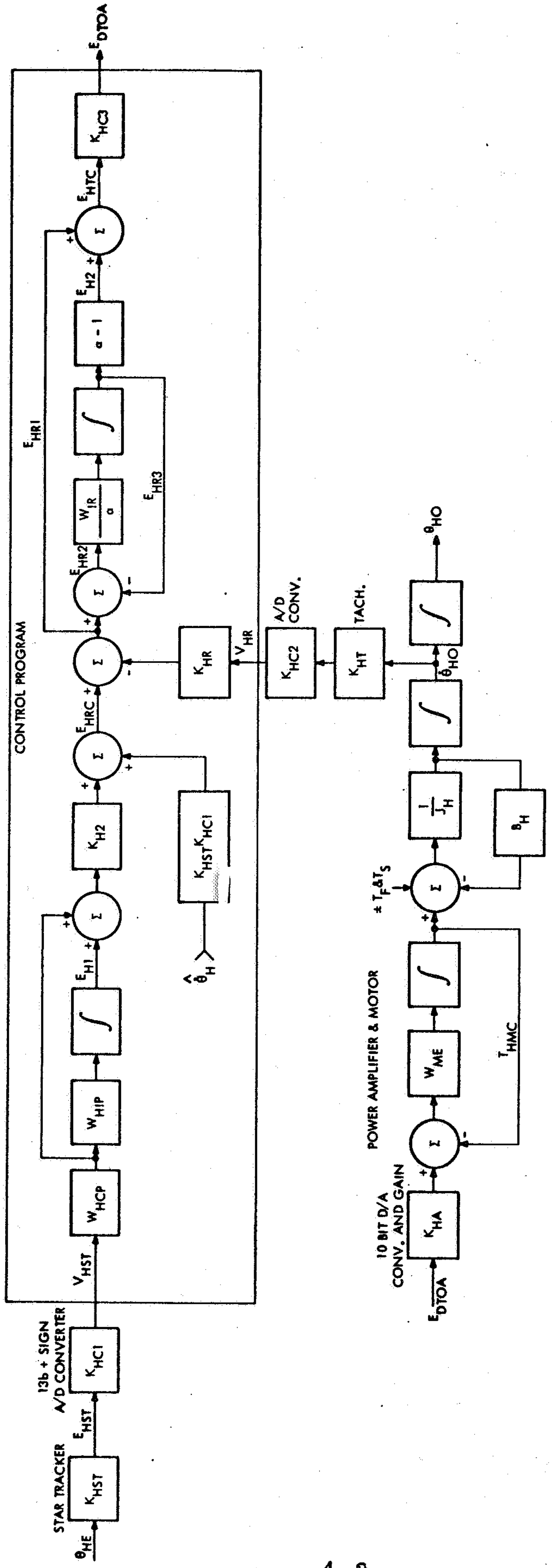


Figure 3. Linear Model of the Hour Angle Servo

lag $\left(W_{ME}\right)$ only. This is a simplification which is valid and useful when

$$W_{CR} \ll W_{ME} < W_{LP} \tag{13}$$

where W_{LP} is the lower frequency (pole) in the exact amplifier-motor transfer function and W_{CR} is the cross-over frequency (or approximate bandwidth) of the servo rate loop. Second mechanical resonances of the telescope mount are also neglected in the model. This second approximation is considered valid because the lower resonance for the type of mount to be used[2] is generally at frequencies of 500 rad/sec or higher. This value is sufficiently high so that the resonance has no significant influence on the closed rate loop stability or other performance characteristics when the rate loop bandwidth is about 50 rad/sec.

TABLE 1

ANGLE SERVO CONSTANTS

Hour Angle

B_H	Axis viscous damping	100 ft-lb-sec
E_{HRL}	Limit on Rate Command	$5.56 * 10^4$
E_{HTL}	Limit on Torque Command	5880
J_H	Axis Inertia	600 ft-lb-sec^2
K_{HA}	Motor torque per D/A bit (10 bits)	0.3 ft-lb/bit
K_{H1}	Constant for computation of W_{HCPL}	$6.4 * 10^5$
K_{H2}	Star tracker sensitivity correction	$1/\cos \theta_D$
K_{HC1}	Star tracker conversion gain	819.1 bits/volt
K_{HC2}	Tachometer output conversion gain	4700 bits/volt
K_{HC3}	Rate loop gain constant	0.17
K_{HR}	Rate feedback gain	6.8
K_{HST}	Star tracker sensitivity	782 volts/rad
K_{HT}	Tachometer sensitivity	20 volts/rad/sec
T_P	Computer sampling period	1/64 sec
T_{MCL}	Motor torque limit	300 ft-lb
W_{HCP}	Position Loop cross-over freq.	18 rad/sec
W_{HIP}	Position Loop compensation freq.	6 rad/sec
W_{HIR}	Rate loop compensation freq.	18 rad/sec
α	Rate loop integral constant	10
$\dot{\theta}_{HL}$	Rate limit	.087 rad/sec

Declination Angle

Same as for hour angle except

$B_D = 20$ lb-ft-sec

$J_D = 120$ lb-ft-sec^2

$K_{D1} = 3.2 * 10^6$

$K_{D2} = 1.$

$K_{DC3} = 0.034$

TABLE 2

ANGLE SERVO VARIABLES; BOTH AXES

E_1	Position loop compensation value
E_2	Rate loop compensation value
E_{DTOA}	Final control program output (digital)
E_{R1}	Rate loop error value
E_{R2}	Rate loop compensation variables
E_{R3}	
E_{RC}	Rate command
E_{ST}	Star tracker output in volts
E_{TC}	Torque command
V_R	Telescope rate converted to bits/rad/sec
V_{ST}	Telescope position converted to bits/sec
T_{MC}	Motor torque
W_{CPL}	Computed limit on W_{CP}
W_{CP1}	Modified W_{CP}
$\hat{\dot{\theta}}$	Estimated target rate in rad/sec
θ_o	Telescope angle in rad
$\dot{\theta}_o$	Telescope rate in rad/sec
θ_T	Known target angle in rad

SECTION 5

CONVERSION PARAMETERS AND GAIN CONSTANTS

The conversion of the star tracker and tachometer outputs introduces random errors with an average magnitude of 1/2 least significant bit. In the case of the star tracker, this noise is introduced directly at the point of angular measurement and should, therefore, be considerably smaller than the desired accuracy. If the linear range of the star tracker to be covered by the converter is $\left(\theta_E\right)_L$ arc-sec and the conversion accuracy desired is $\Delta\theta_E$ arc/sec, we have

$$\Delta\theta_E \geq \frac{\left(\theta_E\right)_L}{2^n-1} \qquad (14)$$

where n is the number of bits of conversion. Substituting 3600 arc/sec for $\left(\theta_E\right)_L$ and 1/4 arc/sec for $\Delta\theta$ in (14) and solving for 2^n we get

$$2^n \geq 1440 \qquad (15)$$

Thus, for a 1/4 arc/sec accuracy the converter must be at least 14 bits. With a full range of the star tracker output of ± 10 V the conversion gain with 14 bits is

$$K_{HC1} = \frac{16383}{20} = 819.1 \text{ bits/volt.} \qquad (16)$$

The total gain between the tracking error and the computer input is therefore

$$K_{HST}\ K_{HC1} = 782 \times 819.1 = 6.4 \times 10^5 \text{ bits/radians} \qquad (17)$$

The effect of the noise introduced by the conversion of the tachometer output is difficult to estimate accurately, but since this noise is being inserted after the position loop gain W_{CP} the error resulting should be

$$\theta_E < \frac{\Delta\dot{\theta}}{W_{CP}} \qquad (18)$$

where $\Delta\dot{\theta}$ is the conversion noise and where it is assumed that both servo loops are well damped. It has been found[1] that the maximum angular rate required for tracking purposes is well below 5°/sec. Using ± 5°/sec as the desired range for the rate measurement and a conversion accuracy of 14 bits in (18) gives

$$\theta_E < .12 \text{ arc/sec}$$

which is below the desired accuracy of 1/4 arc/sec.

With a tachometer sensitivity given at K_{HT} = 20 volts/rad/sec the 14 bits conversion yield

$$K_{HC2} = \frac{16383}{10^o/\text{sec} \times \frac{1}{57.3} \times 20 \text{ volts/rad/sec}} = 4700$$

The rate feedback signal must be inserted into the control program equations with a gain which is exactly equal to the gain applied to the angular error signal, Eq. (17). The rate feedback "equalizing" gain is therefore

$$K_{HR} = \frac{K_{HST}\ K_{HC1}}{K_{HT}\ K_{HC2}} \tag{19}$$

Substituting values in (19) we get

$$K_{HR} = 6.8$$

The least significant bit in the digital-to-analog conversion determines the smallest discrete value of torque being applied to the axis inertia load. In the sample time T_P this value of torque will result in an angular motion of

$$\Delta\theta = \frac{1}{2}\left[\frac{T_{MCL}}{2^N - 1}\right]\frac{T_P^2}{J} \tag{20}$$

where N is the bits of D/A conversion and T_{MCL} is the torque limit. The value of $\Delta\theta$ determines the accuracy with which the axis inertia can be controlled and should again be less than the desired 1/4 arc/sec. Thus, if (20) is solved for 2^N we get

$$2^{N+1} \geq \frac{T_{MCL}}{J\Delta\theta}\ T_P^2 \tag{21}$$

Substituting values of J for the declination angle servo we get the upper requirement

$$2^N \geq 250$$

which is satisfied if N = 8. A 10 bit conversion can be had with little additional cost and is used here. Thus

$$N = 10 \text{ for D/A conversion}$$

giving

$$K_{HA} \cong 0.3 \text{ ft-lb/bit}$$

The closed loop gain of the rate loop is given by the product of all gains in the loop and must equal W_{CR}, thus

$$K_{HC3} = \frac{J_H \; W_{CR}}{K_{HA} \; K_{HT} \; K_{HC2} \; K_{HR}} \tag{22}$$

or with values

$$K_{HC3} = 0.17$$

SECTION 6

COMPENSATION FOR SERVO SATURATIONS

An axis drive of this type is generally limited by two types of saturations. Of these the more important is the torque saturation characteristics of the electromagnetic torque motors. The inherent saturation of the torque motors is very gradual, but a very precise limit must be built into the motor drive circuits to prevent motor demagnetization. This gives an abrupt torque (or acceleration) limit which is very detrimental to servo stability and requires some form of compensation.

The rate of the servo drive is also limited either by the motor and the power amplifier or by the linear range of the angular velocity sensor. A set of limits, therefore must be designed into the servo control circuits (or control program) which prevent the servos from exceeding these limits without affecting loop stability. The methods designed to implement compensations for both types of saturations in a computer control program is discussed in separate paragraphs below.

6.1 TORQUE SATURATION COMPENSATION

The design goals for the saturation compensation are not only to prevent a deterioration of servo stability when an abrupt torque limit is introduced, but also to minimize the time required to bring the servos out of the saturated condition. These goals are accomplished by modifying the control program such that the servos operate with full torque applied, first of one polarity, then switched to the opposite polarity until the servo error is brought to near zero. If the switching time is chosen properly, the servo will then enter the linear operating region at approximately the correct velocity.

The pointing servos have integral compensation in both rate and position loops. These compensations are effective only during linear servo operation and must be removed from the control equations during the saturated mode of operation. The method and conditions under which this removal is done is discussed at the end of this paragraph. The integral compensation equations are removed at the instant of time, say T = 0, when the torque command has reached the + or - limit level. A simplified block diagram of the servo in this condition is shown in Figure 4.

Let us assume that over the time interval of interest the target velocity, $\dot{\theta}_T$, is approximately constant (note that $\dot{\theta}_T$ can be either the feedforward signal or, if this signal is not used, the output of the integral compensation). To obtain a stable operation during the torque limited period of a transient, we may seek to adjust W_{CP} such that the velocity command to the rate loop does not exceed the velocity that can be obtained with the maximum torque command. Let the corresponding maximum acceleration be A and denote the adjusted value of W_{CP} by W_{CPL} then

$$\theta_e(t)W_{CPL}(t) \leq A\ t \tag{23}$$

where it is assumed that

$$\dot{\theta}_T(0) = \dot{\theta}_0(0) \tag{24}$$

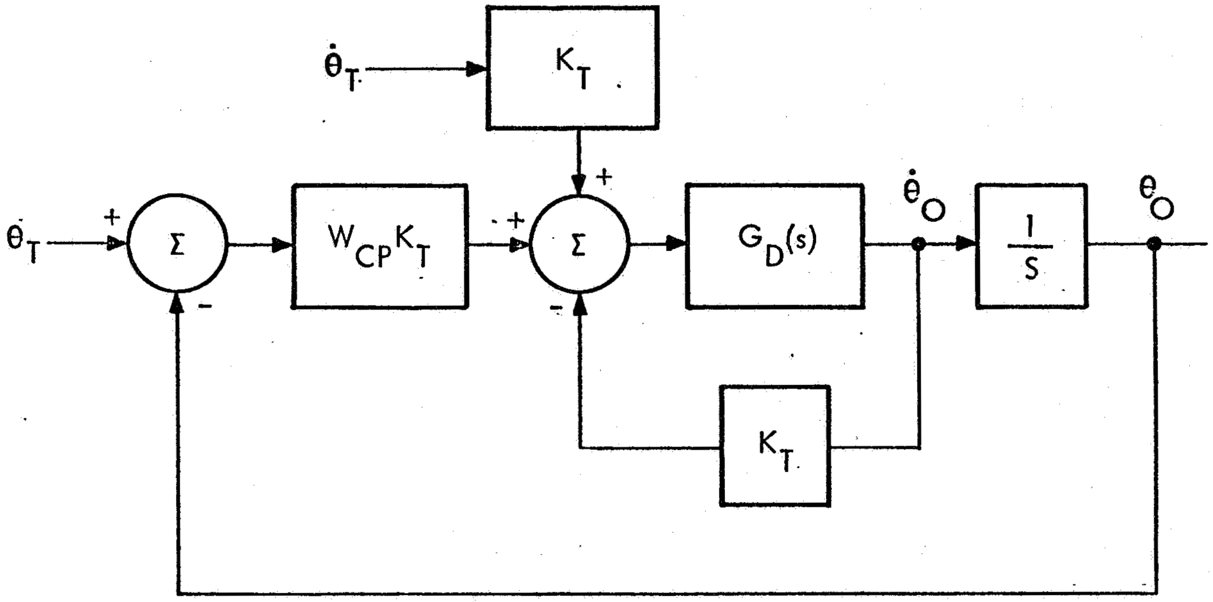


Figure 4. Simplified Block Diagram of the Angle Servo with Integral Compensations Removed

Taking the derivative of (23) we get

$$\dot{\theta}_e(t)W_{CPL}(t) + \theta_e(t)\dot{W}_{CPL}(t) \leq A \tag{25}$$

For the transient condition it is highly probably that

$$\dot{\theta}_e W_{CPL} \gg \theta_e \dot{W}_{CPL} \tag{26}$$

We shall make this assumption and attempt to show its validity in simulations (see Section 7). We then have that

$$W_{CPL}(t) \leq \frac{A}{\dot{\theta}_e(t)} \tag{27}$$

Equation (27) is valid for both + and - since A changes sign as $\dot{\theta}_e(t)$ changes sign. For values of the drive torque below the limits the system is linear and W_{CP} is determined from the linear analysis (see Section 3). During torque saturation the values of W_{CP} must be modified to W_{CP1} such that

$$W_{CP1} = \begin{cases} W_{CP} & W_{CP} \leq W_{CPL} \\ W_{CPL} & W_{CP} > W_{CPL} \end{cases} \tag{28}$$

For the absence of rate feedforward the velocity command to the rate loop is obtained from the position loop integral compensation. When rate feedforward is used the compensation equations correct for the inaccuracies in the rate feedforward signal only. In either case, however, when the drive torque reaches the limit, this compensation must be removed from the control program to prevent a build-up of rate command during the saturated mode of operation. At the end of this period the compensation should be reinstated with an initial condition equal to the condition at the time it was removed. The integral compensation in the rate loop provides a torque command during acceleration. This compensation must also be removed when the torque limit is reached and reinstated with the last computed integral value as the initial condition. During the saturation period this value is used together with the proportional component to determine the torque requirement.

6.2 RATE LIMIT IMPLEMENTATION

The servo rate may be limited simply by limiting the rate command to the servo rate loop. It is necessary under this condition also to remove the position loop integral compensation to prevent a build-up of the rate command. The compensation in the rate loop could remain in tact during this mode of operation. However, if the servo is in the saturated mode of operation when the rate limit is reached, the response of the rate loop is relatively slow and the velocity would overshoot significantly. This overshoot can be avoided by resetting (to a low level) the integral part of the torque command momentarily upon reaching the rate limit.

SECTION 7

SIMULATION STUDIES

The angle servo control performance was studied in a digital simulation which included all linear characteristics of the model of Fig. 3 and the nonlinear parts of the control program (see the Appendix). In addition to the nonlinearities built into the control program the simulation also included the sampled data process, the conversion processes and the axis friction-stiction characteristics. The purpose of the simulation study was to show that satisfactory performance can be achieved under the assumptions of low-noise star tracker output and under all possible transient conditions. Specifically, the following performance characteristics where studied:

1. Small and large signal step responses.
2. Tracking of slowly moving targets.
3. Tracking a Saturn V Vehicle during launch (worst case).
4. The significance of rate feedforward.

The response to a small step change in the angular command is shown in the plots of Fig. 5. The 0.057 degrees (1 mr) step is too small to cause torque saturation and the control program operates entirely in the linear range. The linear response is distorted, however, by the axis friction-stiction which was set at 5 and 8 ft-lb respectively. Adequate stability and satisfactory speed of response is indicated. The response to a step angular command of 0.57 degrees (10 mr) is shown in Fig. 6. This command is sufficient to cause torque command saturation which activates the part of the control program that compensates for

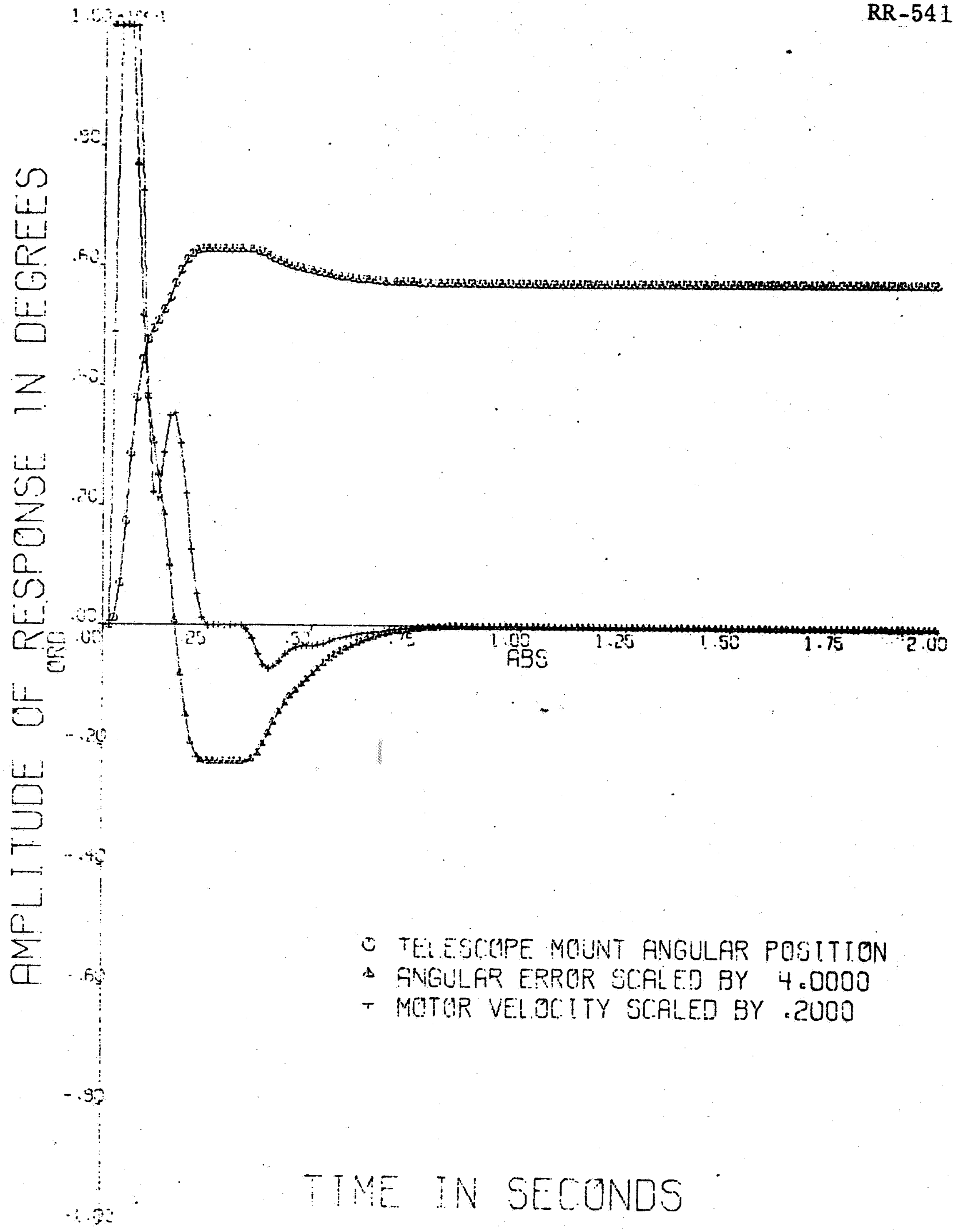


Figure 5. Small Signal Step Response

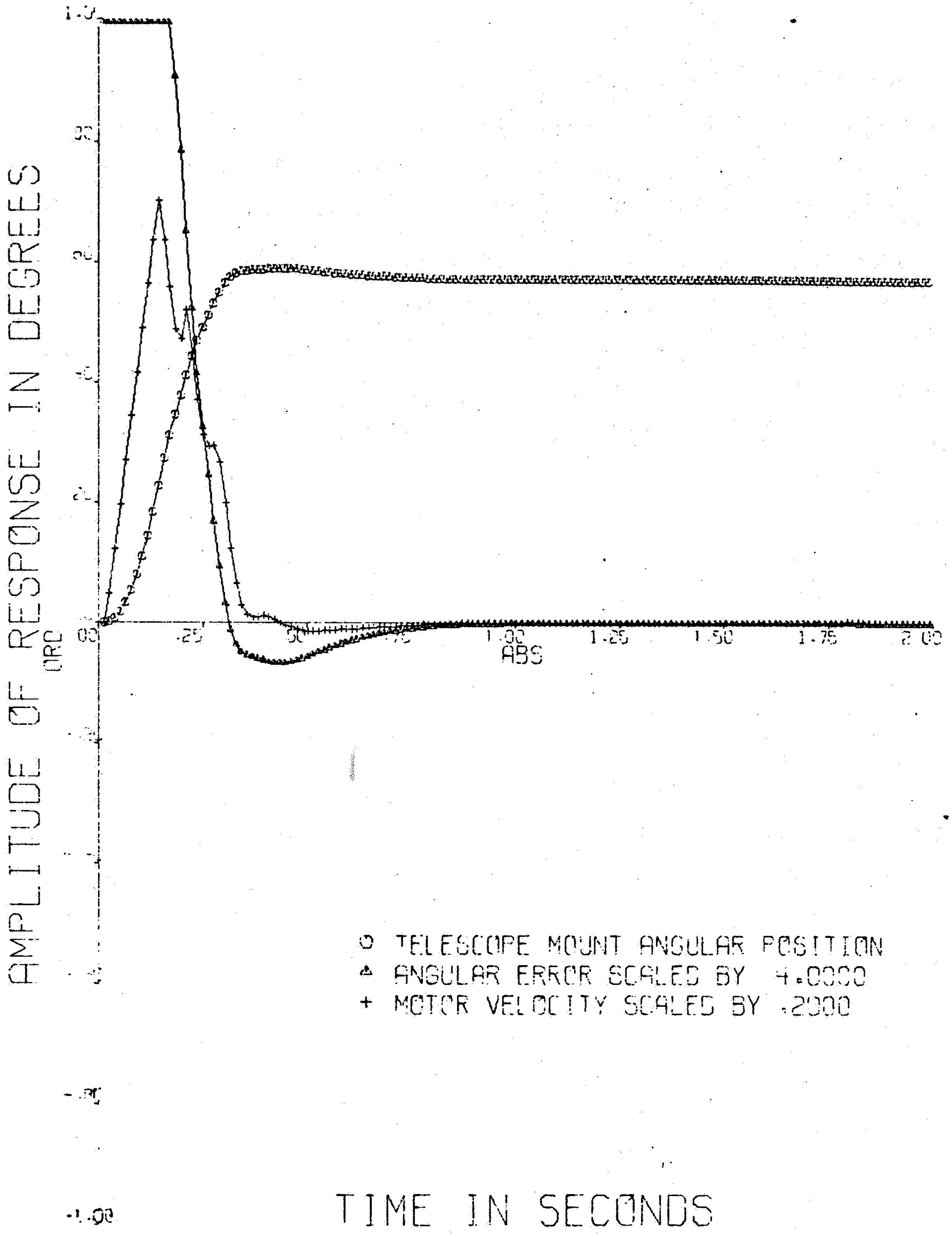


Figure 6. Response to Step Change in Angular Command which Result in Torque Command Saturations

this saturation. It is evident that the compensation yields a near optimum response for this input since the new commanded position is reached with practically no overshoot under the condition of maximum acceleration followed by near maximum deceleration.

When the step command is increased the angular rate of change reaches the rate limit. Figure 7 shows a set of plots for a step input command of 1.43° (25 mr). In this case, maximum acceleration takes place until the rate limit of 5°/sec is reached. The rate is limited to this level without overshoot and remains there until maximum deceleration is switched on. In this case there is a relatively small overshoot at the end of the transient, but no instability is indicated. Both types of saturation also occur when a large step change of velocity command is applied. This type of response is shown in Fig. 8 which again indicates fast recovery and good stability.

The performance of the servo during tracking of a slowly moving target is illustrated by the plots of Fig. 9. In this run the target velocity is varied according to the equation given below the plots. The dynamic errors in this example are negligible so that the tracking errors developed results from the combined effect of the discrete errors in the conversion processes and the friction-stiction load. The effect of the latter is quite evident in the stop-and-go characteristics at very low velocities in Fig. 9. Peak errors are found at the point where the velocity changes sign. The value of the errors are here approaching $\pm$ 1 arc/sec.

The tracking of a Saturn V launch has been simulated by approximating the expected launch trajectory with a 6-order polynomian[1] of the form

$$\theta = C_D + C_1t + C_2t^2 + \dots C_6t^6 \qquad (29)$$

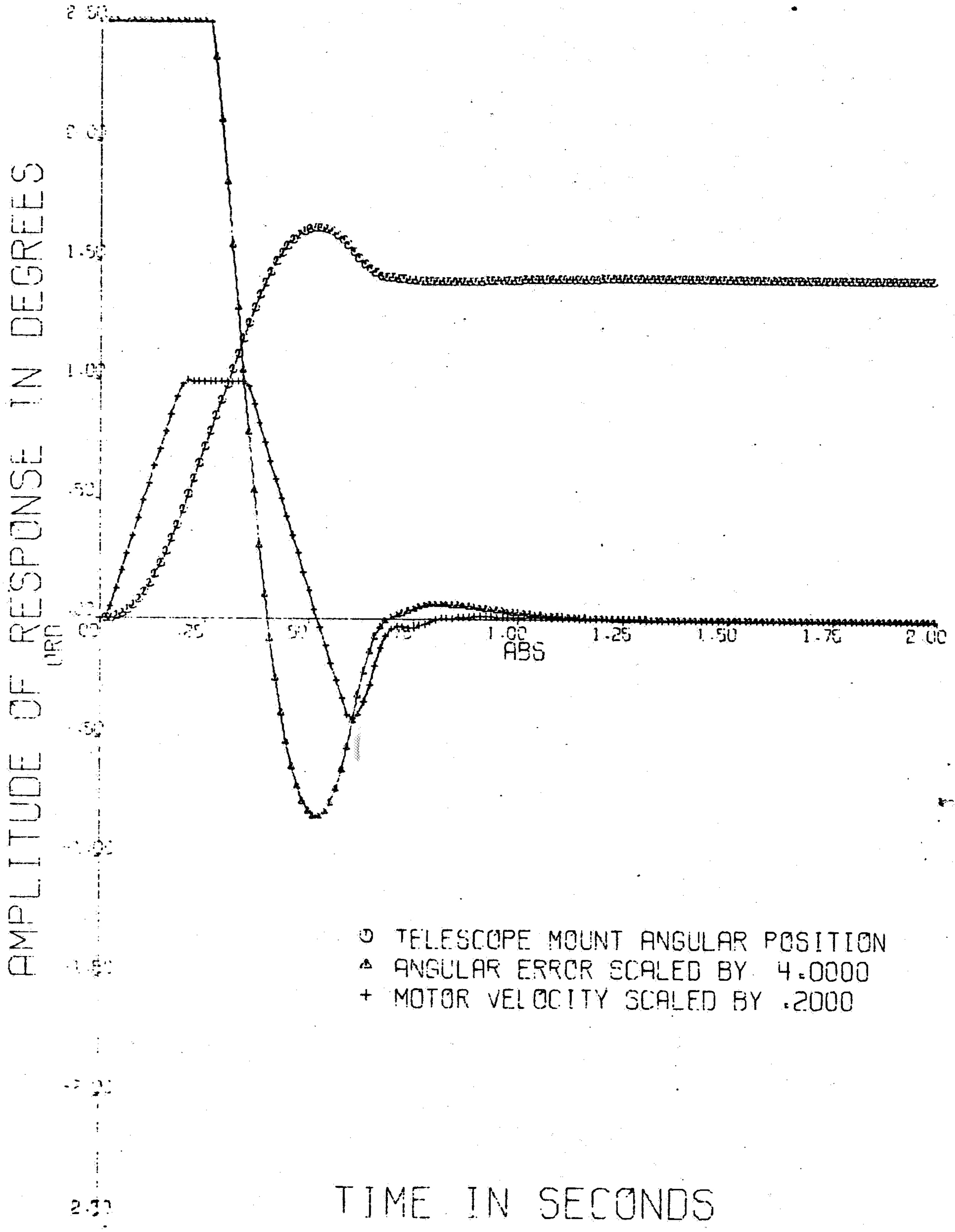


Figure 7. Response to Step Change in Angular Command which Results in Both Torque and Velocity Command Saturations

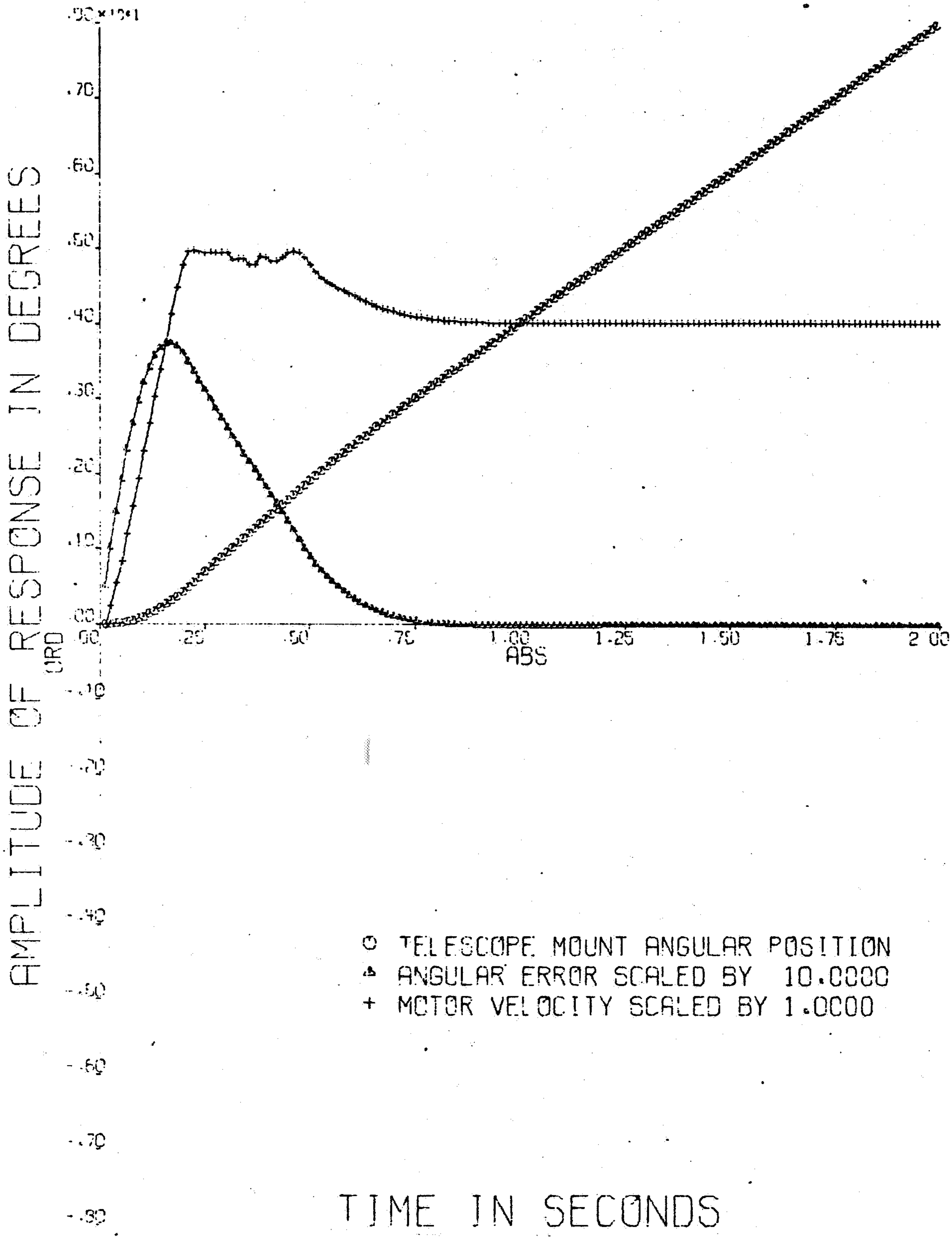


Figure 8. Response to Step Change in Angular Velocity Command Resulting in Saturations of Both Torque and Velocity Commands.

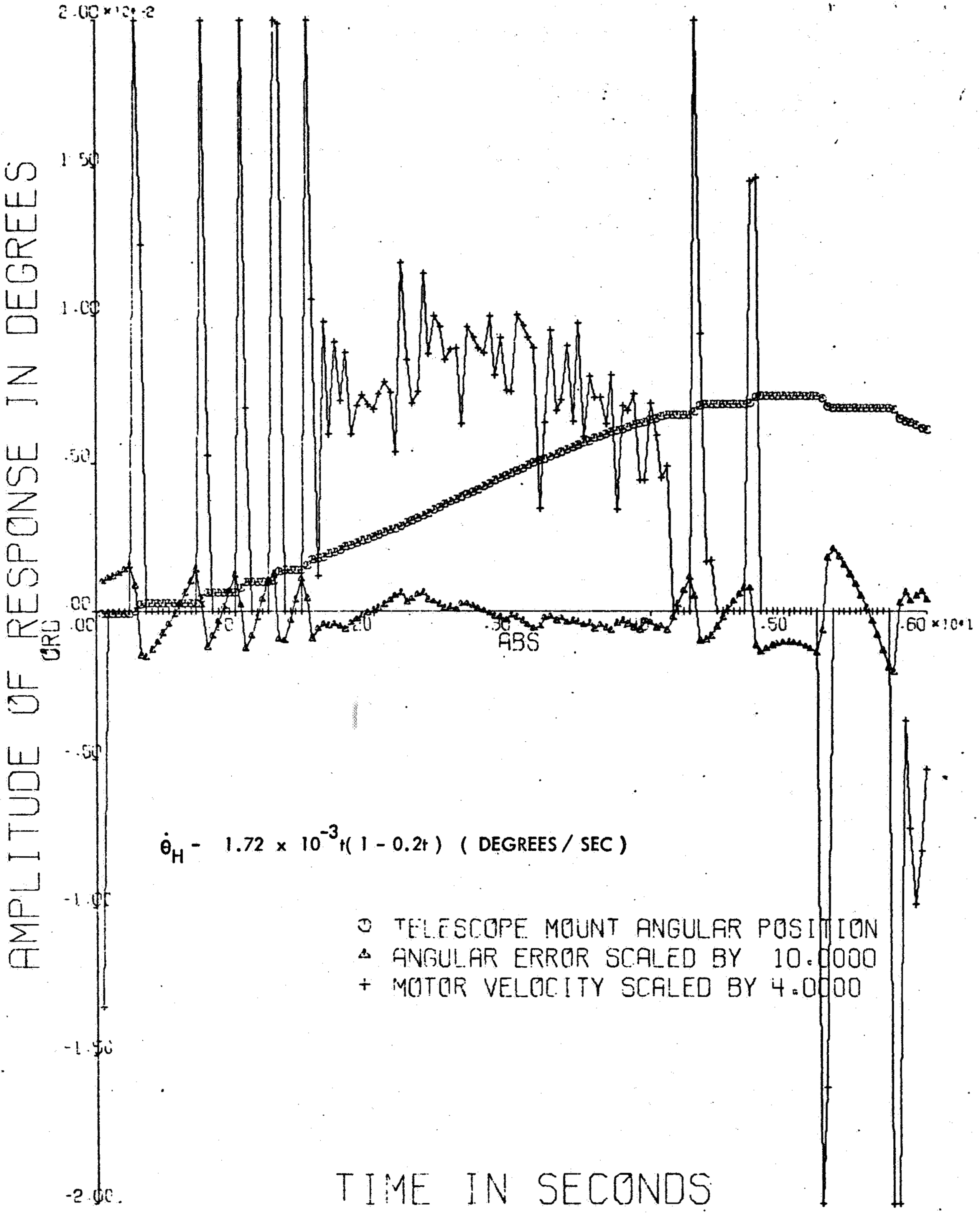


Figure 9. Tracking a Slowly Moving Target. Velocity Range of 0.002°/sec through Zero

For the hour angle the coefficients used are:

$$C_1 = -8.1293125 * 10^{-2}$$

$$C_2 = -2.0993992 * 10^{-2}$$

$$C_3 = 8.1421283 * 10^{-4}$$

$$C_4 = -8.8581258 * 10^{-5}$$

$$C_5 = 1.9413534 * 10^{-6}$$

$$C_6 = 1.2218085 * 10^{-8}$$

which give the following significant peak values:

$$\dot{\theta}_H = -3.258 \text{ deg/sec} \qquad \ddot{\theta}_H = -0.1269 \text{ deg/sec}^2$$

The above values of rate and acceleration, which are considered typical worst case, would result in a peak dynamic error of approximately 0.0013 degrees (4.17 arc/sec) without rate feedforward and less than 1/2 arc/sec when perfect rate feedforward is used. The plots of Fig. 10 show the target angular trajectory, angular rate and tracking errors when no rate feedforward is used. The peak error developed is close to the 4 arc/sec expected. The simulation is repeated in Fig. 11 but with 85 percent of the exact rate feedforward signal applied. In Fig. 12 the exact rate feedforward is applied 100 percent. The improvement of the tracking error is better than what would be expected. This unexpected improvement is due in part to the fact that the estimate of the acceleration error is based upon a servo using rate-feedforward without the simultaneous action of integral networks. A perfect rate signal is not expected, but the plots indicate that slowly varying deviations of up to 15 percent of the true target angular rate would still yield a performance improvement of 5 to 1.

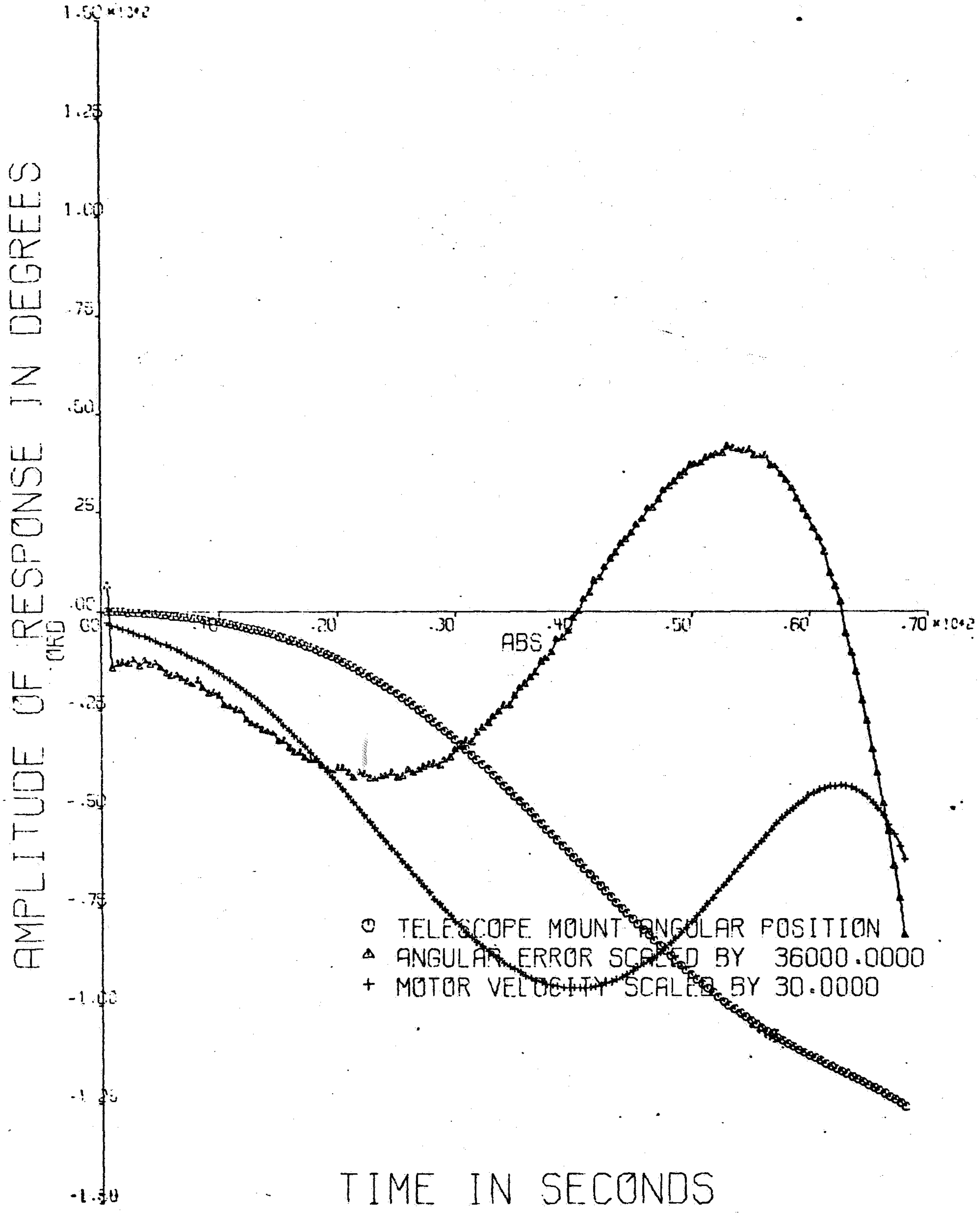


Figure 10. Simulation of the Hour Angle Servo During "Worst-case" Tracking of a Saturn V Launch no Rate Feedforward Used

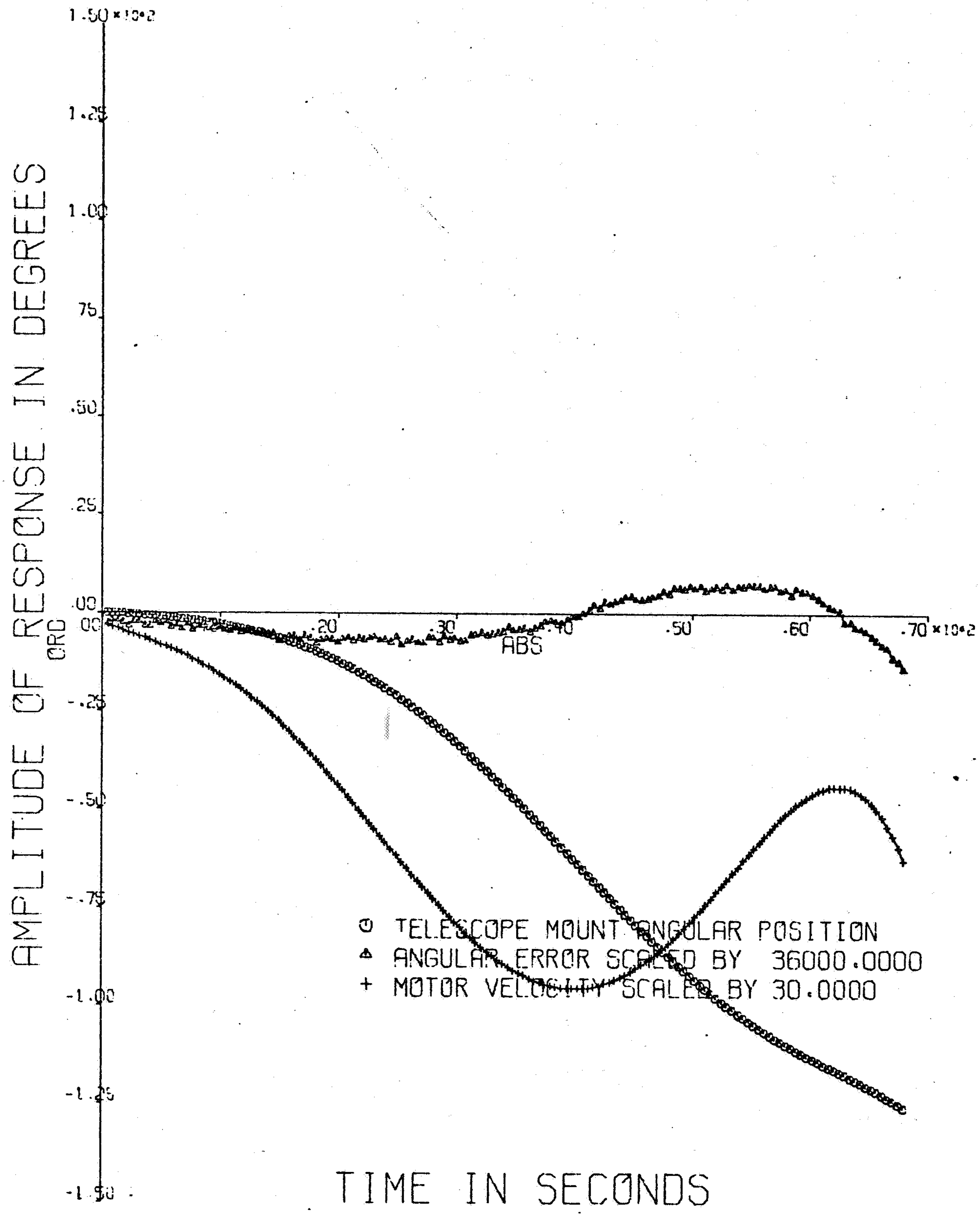


Figure 11. Same as Figure 10 with 85% of Rate Feedforward Applied

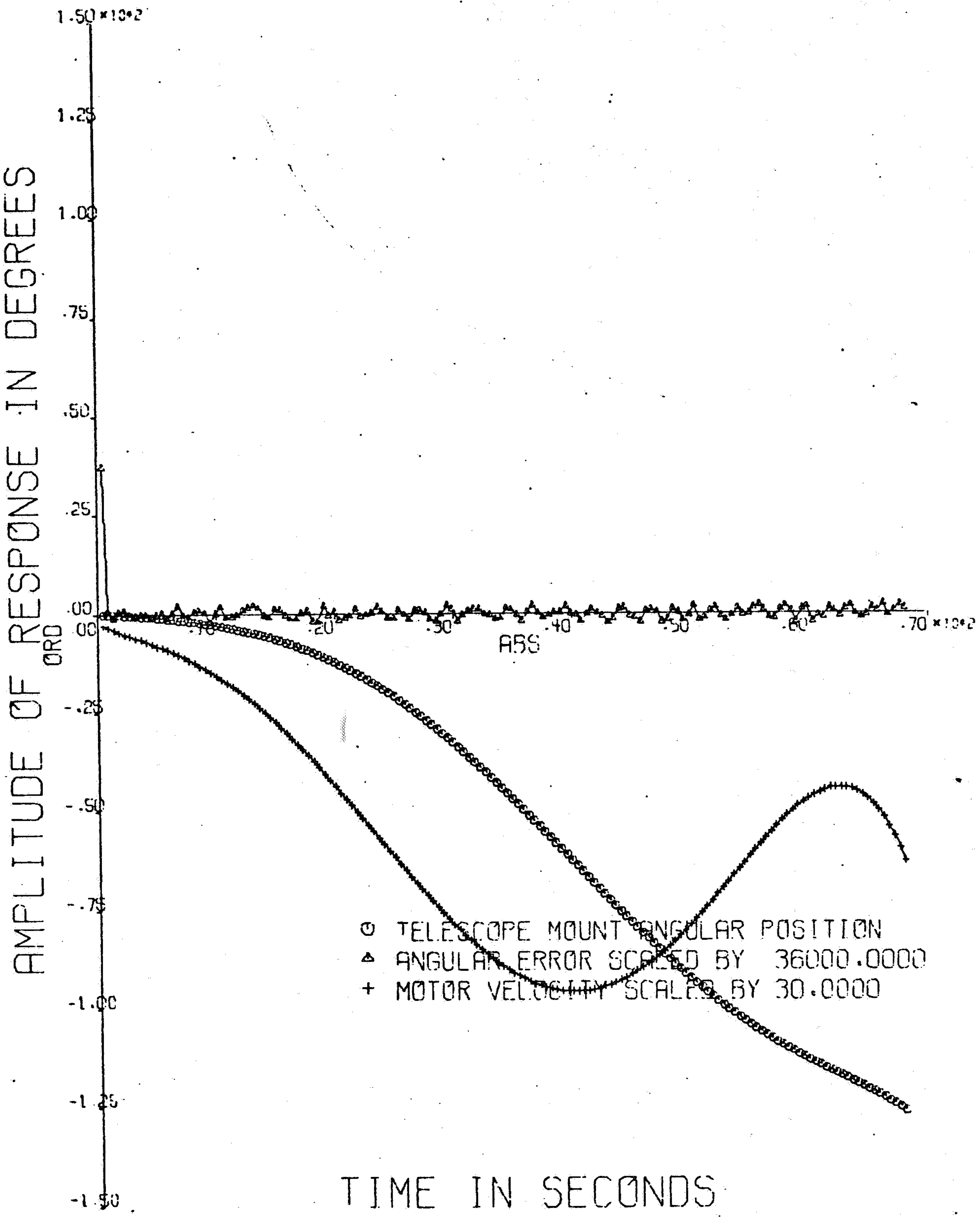


Figure 12. Same as Figure 10 with 100% of Rate Feedforward Applied

SECTION 8

SUMMARY AND CONCLUSION

The linear control equations for the telescope angle servos were derived by means of conventional control theory. These equations involve linear integration which could make the equations for numerical solutions lengthy and complex if many digits of accuracy is required. However, the control equations are part of a closed loop and inaccuracies are automatically compensated for. It has been found sufficient, therefore, to use numerical integrating techniques which involve the previous and present sample (or calculated) values only.

The flexibility of programming has been used with advantage to implement torque and rate limits and to solve the control equations for optimum performance during operations when these limits are applied. Simulations of the complete angle servo indicate unusually quick and stable responses in both linear and saturated conditions.

The accuracy of A/D and D/A conversion required to hold the discretizing noise down to a level which does not deteriorate the tracking accuracy was found to be well within the state-of-the-art. An A/D conversion of 13 bits total and a D/A of 9 bits was found adequate for an accuracy of 1/4 arc/sec in this application. The overall noise introduced by the conversion and computation processes was found by simulation to be approximately 0.5 arc/sec peak-to-peak.

When a good target angular rate signal is available a very significant improvement may be realized by so called rate feed-forward. Simulations indicate that dynamic errors may be reduced by better than 10:1 if a true rate signal is available. Some indication of how much the rate signal may deviate from the value was indicated by introducing a fixed 15 percent error in the signal used for feedforward. Under this condition a 5:1 improvement of the dynamic errors was realized.

SECTION 9

REFERENCES

1. Final Report, NASA Contract No. NAS8-21089, to be published.

2. "Proposal for a 24 Inch Cassegrain Code Telescope and Mount," No. P-487, Submitted to NASA-MSFC by Goeuz Optical Company, Inc., 1967.

APPENDIX

CONTROL PROGRAM

A flow diagram of the hour angle servo control program is given in Fig. 13. The blocks at Level (1) simulates the target and the hardware that provides the signals to the computer. Similarily, the block at Level (7) is a set of equations simulating the hour angle drive mechanism. (The latter is discussed in Section 4). The branching at Level (2) is determined by the mode of operation which have been picked as follows:

mode = 1; Automatic Tracking

mode = 0; Programmed Control

mode = -1; Reacquisition

In the case of automati: tracking the star tracker output signal V_{HST} is used directly in the control program. In the other two modes V_{HST} is calculated or estimated in programs which are described in Ref. 1.

The test at Level (3) will branch the program into a set of calculations which are designed to compensate for the limit imposed on the torque output level of the drive motors. This compensation, which will optimize the servo response under the saturated condition (see Section 6), determines the value of the position loop gain W_{HCP1}. The command signal to the servo rate loop is computed at Level (4) and tested to determine if this command exceeds the maximum rate for which the servo is designed. The gain correction K_{H2} at the beginning of Level (4) is included to compensate for the change of star tracker sensitivity to hour

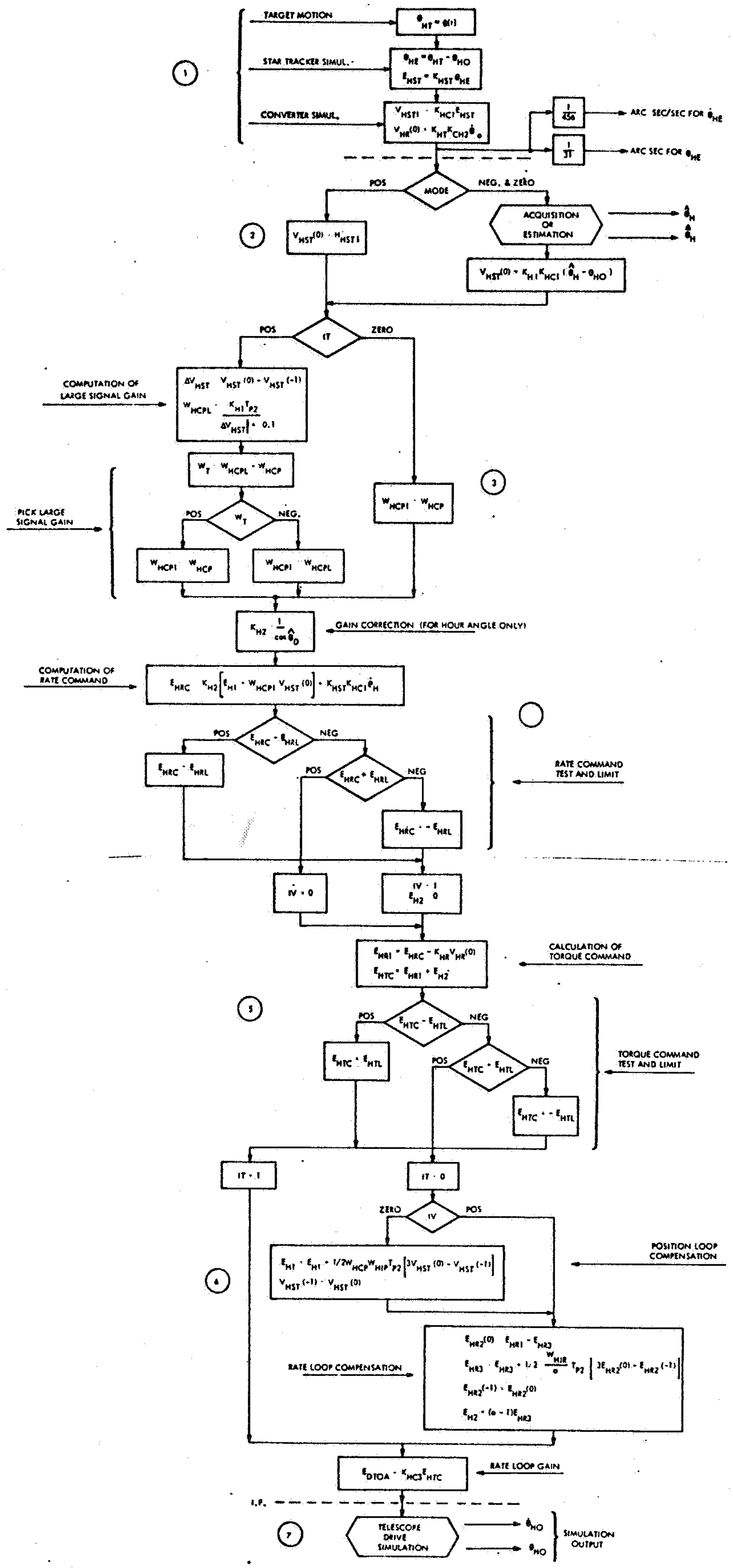


Figure 13. Angle Servo Control Program

angle errors as the declination angle changes. This cross coupling does not exist for the star tracker sensitivity to errors in the declination angle making K_{D3} unity for all hour angles.

The computation of the torque command E_{HTC} which follows at Level (5) depends to some degree on the result of the tests at Level (4). Thus, if a rate limit has been reached the part of the command, E_{H2}, caused by the rate loop integral compensation, is reset to zero. This eliminates most of the torque command in transient conditions (see Section 6.2) and stops the drive acceleration quickly. The torque command is tested next at Level (5) and both rate and position loop compensation equations are bypassed if the torque exceeds the limits. If the rate limit is reached the torque command will drop abruptly as explained above and the test at Level (5) will exclude the position loop compensation equation to prevent build-up of the rate command. The torque command is finally multiplied by a gain constant and then converted to an analog (voltage) signal for control of the power amplifier.

APPENDIX C

THE RANGE SERVO

C.1 SHORT DESCRIPTION

A simplified block diagram at the range servo loop is shown in Figure C-1. The input signal to the loop, f_{IFR}, is an IF signal, generated in the heterodyne receiver, with a phase shift proportional to the target range, R_T. A locally generated IF signal f_{IFfb}, with a phase shift proportional to the computed range, R_0, is multiplied with f_{IFR} in a phase detector circuit to provide dc signal, $E_{\phi E}$, with an amplitude proportional to the difference in the phase ϕ_E, between f_{IFR} and f_{IFfb}. The detector output is filtered and converted to a binary signal for "sampled data" processing by the computer.

The computer process pertaining to the range loop can be separated into three parts for a simplified description. The first part is a computation which provides proportional plus integral compensation for the servo loop to effectively eliminate velocity errors, minimize acceleration errors, and assure good stability. The second part calculates the range rate, $\dot{R}_0$, and range, R_0, output words. The third process provides the feedback signal which enables a sufficiently accurate control of the phase shift of the locally generated IF signal. This is accomplished by controlling, with a digital word from the computer, the number of pulses which are to be added to a train of clock pulses. The resulting pulse train is then divided down to yield the local IF signal. The later is a 3 to 4 bit binary signal which is used in a technique of gain switching to provide the desired multiplication for phase detection.

Two other channels of the range servo are indicated in Figure C-1. Each channel has a separate phase control, IF generator, phase detector, filter, and sample-hold circuits. Switching between channels is accomplished by software programming only, as elaborated on later in this appendix. The following discussions pertain to the fine channel only unless otherwise stated.

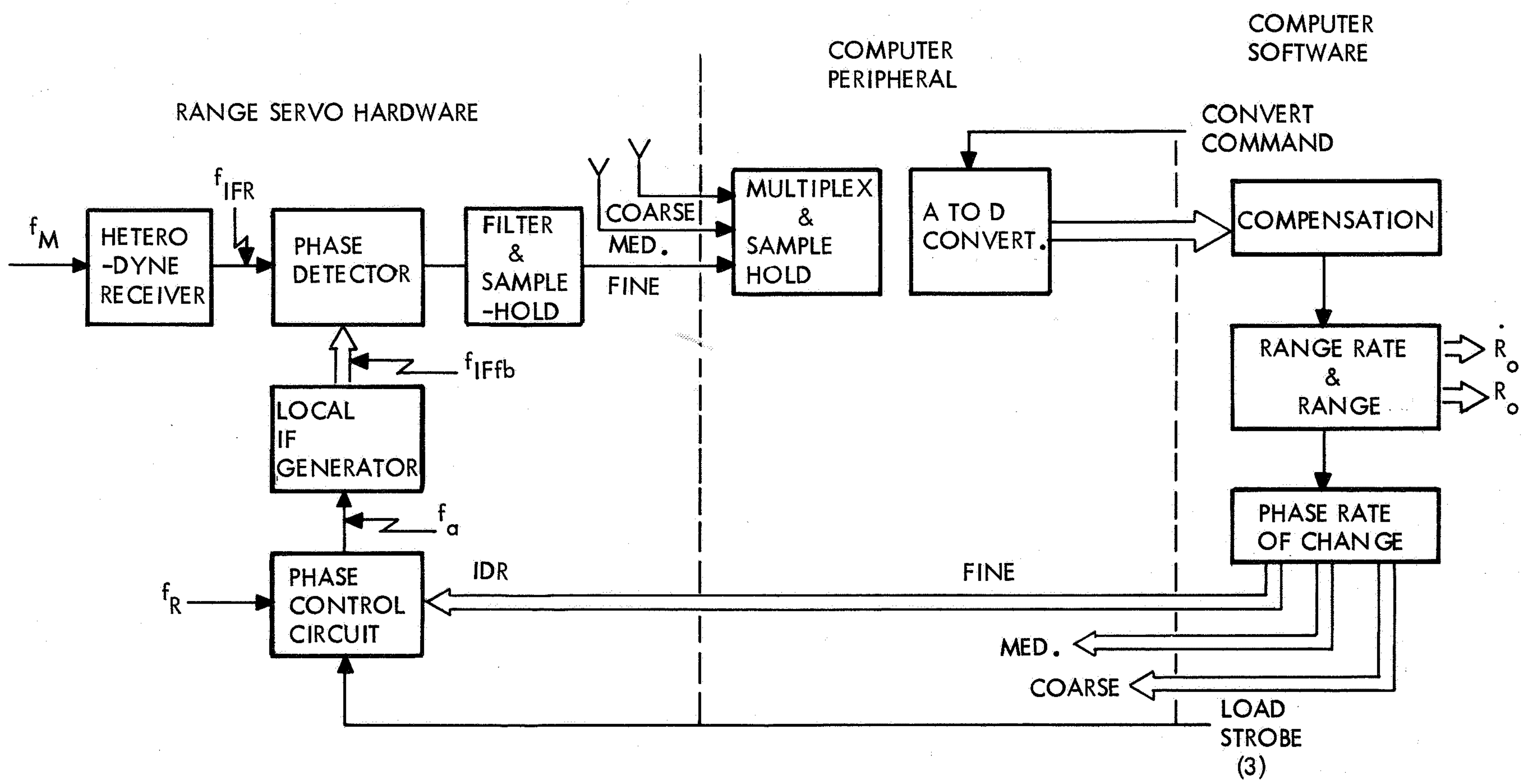


Figure C-1. Range Servo Block Diagram

C.2 PHASE CONTROL OF THE LOCAL IF SIGNAL

The local IF signal is generated by counting in a 4 bit register a pulse train of frequency $N_2 f_{IFfb}$ where N_2 is the maximum number of counts (up to 16). If the pulse train is generated in synchronism with the heterodyne generated IF signal, the two signals will have a constant relative phase shift. Control of the phase shift can be accomplished by addition of a controlled number of pulses per second to the pulse train. One method of implementing this is shown in Figure C-2 where the control pulse train, f_a, is generated by counting a preset number of pulses of the pulse train, f_R, in response to the computer feedback word. In the following is a development of the parameters of the circuit of Figure C-2. The frequency of the local IF generated by the circuit of Figure C-2 is

$$f_{IFfb} = \frac{f_R}{N_1 N_2}\left[1 + a_0 \frac{1}{2} + a_1 \frac{1}{2^2} + \dots a_{n-1} \frac{1}{2^n}\right] \qquad \text{(C-1)}$$

where a_0 is the sign bit, a_1 the most, and a_{n-1} the least significant bits of the computer feedback wo"d. To give f_{IFfb} an approximately equal range around f_{IF} we set

$$\frac{f_R}{N_1 N_2}\left[1 + \frac{1}{2}\right] = f_{IF} \qquad \text{(C-2)}$$

where $a_0 = 1$ and a_1 to $a_{n-1} = 0$. The maximum difference between f_{IF} and f_{IFfb} is therefore

$$\left[f_a\right]_{max} = \frac{f_R}{2N_1 N_2} \qquad \text{(C-3)}$$

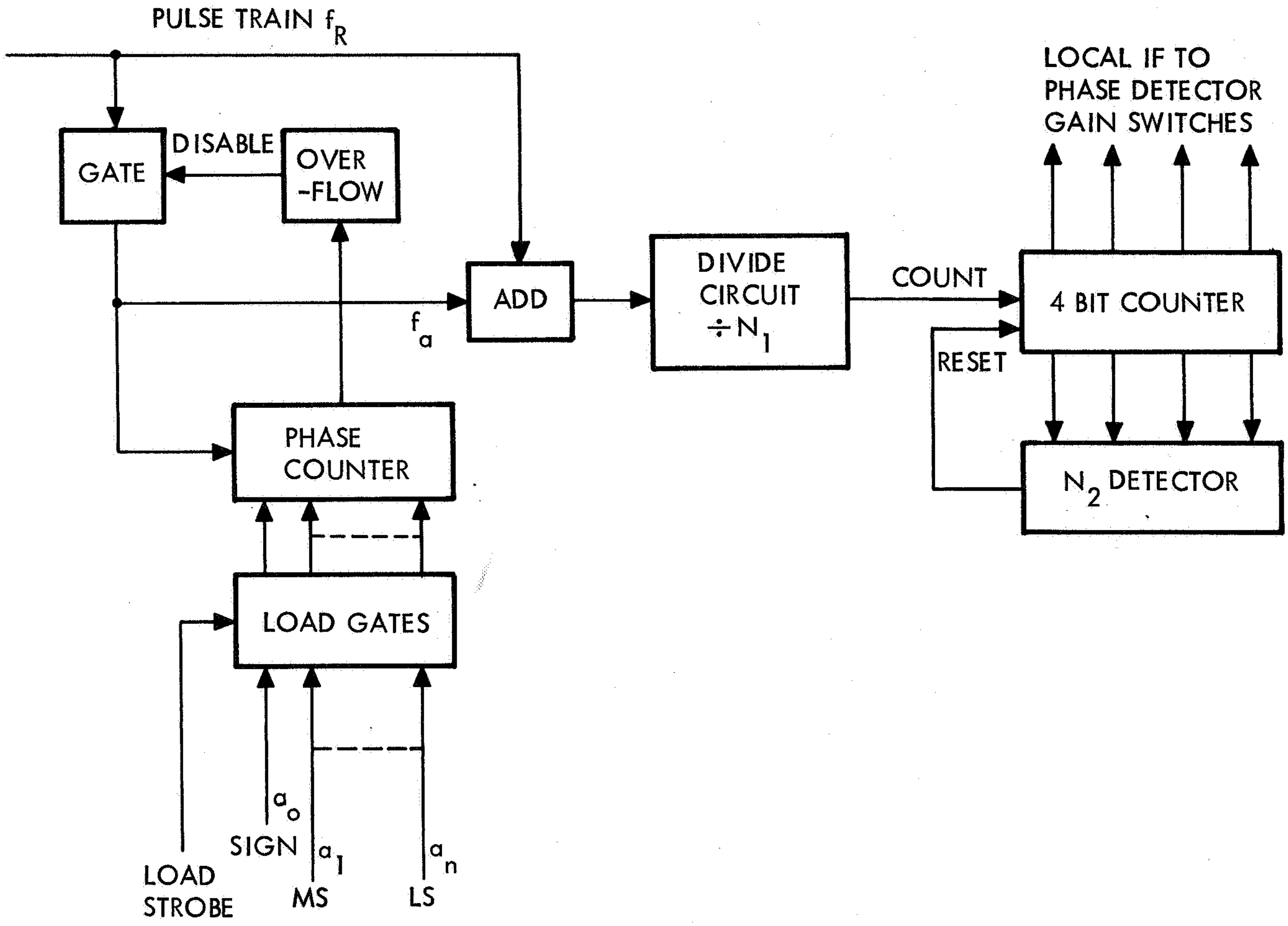


Figure C-2. Block Diagram of Phase Control and Local IF Generator Circuits (One Channel Only) which corresponds to a maximum phase rate-of-change of

$$\left(\dot{\phi}_{fb}\right)_{max} = 2\pi \left[f_a\right]_{max} \tag{C-4}$$

Each pulse of $f_R/2$ corresponds to a phase increment $\Delta\phi$ where

$$\Delta\phi = \frac{2\pi}{N_1 N_2} \tag{C-5}$$

But since

$$\dot{\phi} = 2\dot{R}\,\frac{2\pi f_M}{c} \tag{C-6}$$

where f_M is the laser beam modulation frequency and c the speed of light, we also have that

$$\Delta\phi = 2\Delta R\,\frac{2\pi f_M}{c} \tag{C-7}$$

where ΔR is the corresponding range increment. Equating Eqs. (C-5) and (C-7) we get

$$N_1 \times N_2 = \frac{c}{2\Delta R f_M} \tag{C-8}$$

Equations (C-3), (C-4), and (C-6) yield

$$\frac{f_R}{N_1 \times N_2} = 4(\dot{R})_{max}\,\frac{f_M}{c} \tag{C-9}$$

Substituting Eq. (C-9) into Eq. (C-2) and solving for $(\dot{R})_{max}$ we get

$$[\dot{R}]_{max} = \frac{f_{IF}c}{6f_M} \tag{C-10}$$

The computer range feedback word is provided every $t_S = \frac{1}{f_{IF}}$ seconds during which time interval the most significant bit of the word should provide the required $\left(\dot{\phi}_{fb}\right)_{max}$. Since

$$[\dot{\phi}_{fb}]_{max} = \frac{2^{n-1}\Delta\phi}{t_S} \qquad \text{(C-11)}$$

we find, by use of Eqs. (C-7) and (C-10), that

$$2^n = \frac{2}{3} N_1 N_2 \qquad \text{(C-12)}$$

To hold errors in the range servo down to 0.5 cm or less it is expected that ΔR must be about 0.5 cm also.* Substituting this value for ΔR and $f_M = 29.3 \times 10^6$ Hz into Eq. (C-8) we get

$$N_1 * N_2 = 1024 \text{ or } 2^{10}$$

Worst case values of $(\dot{R})_{max}$ for a Saturn V launch have been found to be about 360 m/sec. Using a value of $f_{IF} = 256$ in Eq. (C-10) yields a value for $(\dot{R})_{max}$ of

$$(\dot{R})_{max} = 437 \text{ m/sec}$$

which is sufficient to cover the peak expected velocity. The value of f_R from Eq. (C-2) is

$$f_R = 2^{19}/3 \text{ per sec}$$

Equation (C-12) then finally yields the following closest (integer) value for the number, n, of the computer feedback word:

$$n = 10$$

*Simulations of the range servo have indicated that ΔR may be significantly larger.

C.3 GENERATION OF THE COMPUTER FEEDBACK WORD

The computer feedback word must be proportional to the rate of change of the range word. The integration which generates the phase of the local IF is taking place in the 4 bit counter of the circuit of Figure C-2. The feedback word cannot be taken directly from the range rate word $\dot{R}_0$, however, since this would require an integration for the calculation of the range word R_0 which would be outside the control loop. Instead, the feedback word is generated by comparing a stored value of the range R_{01} with each computed value of R_0. A feedback word is then generated which is proportional to the difference between the two range samples when the difference exceeds ΔR. R_{01} is then updated and the process repeated. In brief the following steps are followed:

$$\left.\begin{array}{ll} 1. & R_0 = R_0 + F\left(\dot{R}_0\right)\, t_s \\ 2. & IDR = \left(R_0 - R_{01}\right)/\Delta R \\ 3. & R_{01} = R_{01} + \Delta R * IDR \end{array}\right\} \qquad (C\text{-}13)$$

In step number 1 the range word is computed from samples of the range rate. In step number 2 the fixed point computer feedback number is computed which is equal to the number of increments ΔR. In step number 3 the value of R_{01} is updated if IDR $\neq 0$ giving a new value of R_{01} which has been incremented exactly the same amount as the phase circuit. Any drift in the calculation of R_{01} due to truncation errors in IDR will be corrected in subsequent calculations of IDR. The accuracy of the feedback loop is now assured if the number of increments $\Delta\phi$ in one sample period is equal to IDR. This was accomplished by proper choice of f_R, N_1N_2, and n[Eqs. (C-2), (C-8), and (C-12)].

C.4 LOOP COMPENSATION AND DYNAMICS

Proportional plus integral compensation has been chosen for the range loop to eliminate velocity errors. The errors due to range acceleration only for the servo can be estimated with good accuracy from

$$R_E = \frac{\ddot{R}}{\omega_{ip}\,\omega_{cp}} \tag{C-14}$$

where ω_{ip} is the lead corner frequency in rad/sec of the proportional-plus integral compensation, and ω_{cp} the cross-over frequency for the servo loop. Good loop stability is obtained when

$$\omega_{ip} \leq \frac{1}{3}\,\omega_{cp} \tag{C-15}$$

Using this upper value of ω_{ip} and solving Eq. (C-14) for ω_{cp} we get

$$\omega_{cp} = \left(3\,\frac{\ddot{R}}{R_E}\right)^{1/2} \tag{C-16}$$

The peak value of $\ddot{R}$ expected is about 10 m/sec^2. For an error due to acceleration of $R_E = 1$ cm we should therefore make

$$\omega_{cp} \geq 55 \text{ rad/sec}$$

Phase lag introduced by the sampling time interval will cause significant deterioration of stability if ω_{cp} is made larger than 1/3 of the sampling frequency. This limit puts an upper bound on ω_{cp} of about 85 rad/sec. The low pass filter after the phase detection introduces further lag and should have a corner at least three times higher than ω_{cp}. Assuming an upper value of $\omega_{cp} = 70$ rad/sec will result in a filter corner frequency of 210 rad/sec which would allow little filtering action in the 4 msec sampling period. This difficulty can be avoided, however, by the use of sample-hold circuits which are synchronized to the heterodyne IF output.

C.5 MULTICHANNEL SWITCHING

Switching between three phase detectors operating with sensitivities separated by 2^5 is accomplished by multiplexing, at each sampling time, all detector outputs and converting. The coarse channel output is tested first for the presence of a minimum sample value of 2^{-5} times the peak linear detector output. If this or a greater value is detected the coarse channel is selected with appropriate gains for generation of range rate and range. If not, the medium channel is tested in a similar manner, bringing the selection to the fine channel if the fine detector output is within its linear range. (It is estimated that each detector will have a useful linear range of about $\pm$ 1/2 rad of phase difference which, for the fine detector, corresponds to about 40 cm range errors.)

C.6 SYNCHRONIZATION AND TIMING

The computer controlled range servo has been designed with particular emphasis on asynchronized operation of the servo hardware and the computer I/0 and peripheral equipment. This is accomplished by using the phase counter of Figure C-2 (with anti-coincidence circuits wherever required to avoid pulse splitting), the multiplexer, and the sample and hold computer input equipment. The use of the phase counter assures that the required phase shift is provided for with sufficient smoothness within each sample period. The computer multiplexer provides the channel switch capability and the sample and hold provides an input independent of the sample and hold circuit in the range servo hardware.

The timing of the computer input and subsequent output is not critical so long as the interval between input samples and output feedback words remain fixed. If the delay between computer input and output approaches one full sample time (~4 msec) however, the stability of the loop will deteriorate. This delay should therefore be held to a minimum.

APPENDIX D

COORDINATE CONVERSIONS

In giving a unified treatment of all coordinate conversions used, both in the real-time program and in the program to obtain tracking data, all coordinate systems will be referred to an equatorial inertial coordinate system. In particular this coordinate system will have its 1-axis directed to the North Pole, its 2- and 3-axes lying in the equatorial plane, and its 2-axis passing through the point of intersection between the launch site meridian and the equator. The system is depicted in Figure D-1 where

$\delta_\ell \triangleq$ geocentric declination of launch site,

$\delta_t \triangleq$ geocentric declination of tracker site,

$\Delta\phi \triangleq$ difference in geodetic latitude between launch and tracking sites,

and

$\Delta\lambda \triangleq$ difference in longitude between launch and tracker sites.

Note that $\Delta\lambda$ is measured positive in an easterly direction.

A topocentric coordinate system is one centered at a point on the geoid with its 1-axis in the direction of the local vertical, its 2-axis directed south along the local meridian, and its 3-axis completing a right-handed orthogonal system. The basis vectors for any topocentric system may be expressed in the reference inertial system. To do this let $\underline{T}_\ell$ and $\underline{T}_t$ denote the matrices of basis vectors for topocentric coordinate systems at the launch and tracker sites, respectively. The results are

$$\underline{T}_\ell = \begin{bmatrix} \sin\phi_\ell & -\cos\phi_\ell & 0 \\ \cos\phi_\ell & \sin\phi_\ell & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad \text{(D-1)}$$

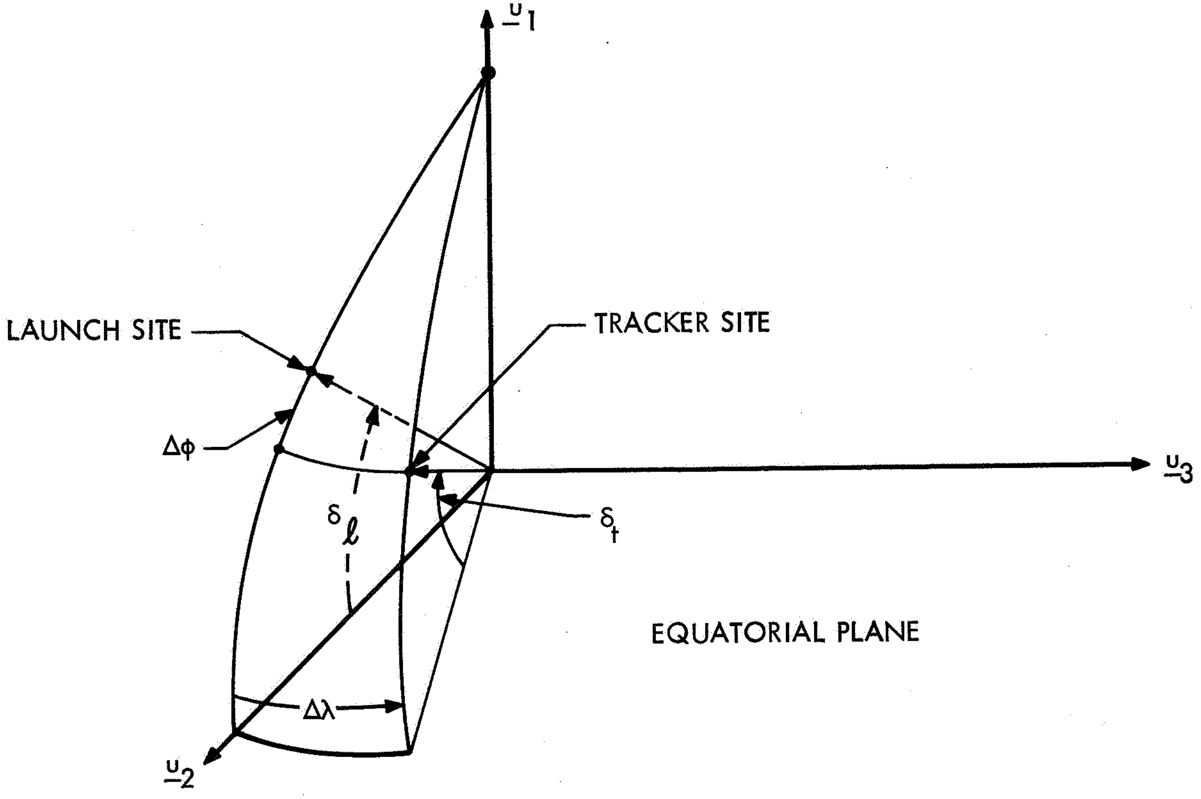


Figure D-1. Reference Geocentric, Equatorial, Inertial Coordinate System and

$$\underline{T}_t = \begin{bmatrix} \sin \phi_t & -\cos \phi_t & 0 \\ \cos \phi_t \cos \Delta\lambda & \sin \phi_t \cos \Delta\lambda & -\sin \Delta\lambda \\ \cos \phi_t \sin \Delta\lambda & \sin \phi_t \sin \Delta\lambda & \cos \Delta\lambda \end{bmatrix} \tag{D-2}$$

where

$\phi_\ell \overset{\Delta}{=}$ geodetic latitude of launch site,

$\phi_t \overset{\Delta}{=}$ geodetic latitude of tracker site.

The geocentric radius vectors of the two topocenters are

$$\underline{r}_\ell = r_{t\ell} \begin{bmatrix} \sin \delta_\ell \\ \cos \delta_\ell \\ 0 \end{bmatrix} \tag{D-3}$$

and

$$\underline{r}_t = r_{tt} \begin{bmatrix} \sin \delta_t \\ \cos \delta_t \cos \Delta\lambda \\ \cos \delta_t \sin \Delta\lambda \end{bmatrix} \tag{D-4}$$

Let $\underline{X}$ be the coordinates of any point (actually, the target position) as measured in the reference geocentric coordinate system, $\underline{X}_{t\ell}$ the coordinates of the same point as measured in the launch-site topocentric system, and $\underline{X}_{tt}$ the coordinates of the same point as measured in the tracker-site topocentric system. Then there obtains both

$$\left.\begin{aligned} &\underline{X} = \underline{r}_\ell + \underline{T}_\ell \underline{X}_{t\ell} \\ \text{and}\quad &\\ &\underline{X} = \underline{r}_t + \underline{T}_t \underline{X}_{tt} \end{aligned}\right\} \tag{D-5}$$

Equations (D-5) may be solved to yield either

$$\underline{X}_{t\ell} = \underline{T}_{\ell}^{T} \left(\underline{r}_{t} - \underline{r}_{\ell} + \underline{T}_{t}\underline{X}_{tt}\right) \tag{D-6}$$

which takes tracker topocentric to launch topocentric, or

$$\underline{X}_{tt} = \underline{T}_{t}^{T} \left(\underline{r}_{\ell} - \underline{r}_{t} + \underline{T}_{\ell}\underline{X}_{t\ell}\right) \tag{D-7}$$

which does the reverse. By introducing

$$\Delta\underline{r} \triangleq \underline{T}_{\ell}^{T}\left(\underline{r}_{t} - \underline{r}_{\ell}\right) = \begin{bmatrix} \sin\phi_{t}\sin\phi_{\ell} + \cos\phi_{t}\cos\phi_{\ell}\cos\Delta\lambda - r_{t\ell}/r_{tt} \\ -\sin\phi_{t}\cos\phi_{\ell} + \cos\phi_{t}\sin\phi_{\ell}\cos\Delta\lambda \\ \cos\phi_{t}\sin\Delta\lambda \end{bmatrix} \tag{D-8}$$

and

$$\underline{T}_{\ell t} \triangleq \underline{T}_{\ell}^{T}\,\underline{T}_{t} \tag{D-9}$$

Eqs. (D-6) and (D-7) are simplified to

$$\underline{X}_{t\ell} = \Delta\underline{r} + \underline{T}_{\ell t}\underline{X}_{tt} \tag{D-10}$$

and

$$\underline{X}_{tt} = \underline{T}_{\ell t}^{T}\left(\underline{X}_{t\ell} - \Delta\underline{r}\right) \tag{D-11}$$

In order to complete the coordinate conversion required of the real-time system program, it is necessary to detail two more coordinate conversions: from equatorial to topocentric at the tracker, and from rectangular to spherical at the launch site. The later conversion follows readily from consideration of Figure D-2; in detail it is

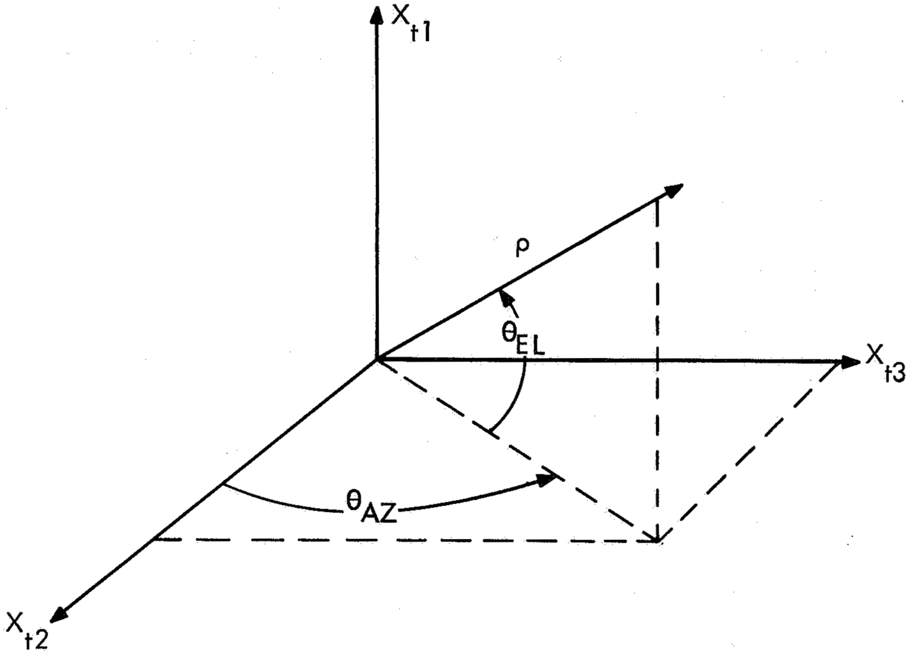


Figure D-2. Rectangular to Spherical Coordinate

$$\rho = \sqrt{X_{t\ell_1}^2 + X_{t\ell_2}^2 + X_{t\ell_3}^2}$$

$$\theta_{EL} = \sin^{-1}\left(\frac{X_{t\ell_1}}{\rho}\right) \tag{D-12}$$

$$\theta_{AZ} = \sin^{-1}\left(\frac{X_{t\ell_3}}{\sqrt{X_{t\ell_3}^2 + X_{t\ell_2}^2}}\right)$$

Conversion from equatorial to topocentric occurs in two steps. The first, from Figure D-3(a), (b), is

$$\underline{X}_e = r \begin{bmatrix} \sin \theta_{DEC} \\ \cos \theta_{DEC} \cos \theta_{HA} \\ -\cos \theta_{DEC} \sin \theta_{HA} \end{bmatrix} \tag{D-13}$$

which gives rectangular equatorial coordinates. Then, from Figure D-3(c),

$$\underline{X}_{tt} = \begin{bmatrix} \sin \phi_t & \cos \phi_t & 0 \\ -\cos \phi_t & \sin \phi_t & 0 \\ 0 & 0 & 1 \end{bmatrix} \underline{X}_e \tag{D-14}$$

The sequence of operations to convert from equatorial tracking data to launch-centered spherical coordinates is given, in order of performance, by the following equations: (D-13), (D-14), (D-10), and (D-12).

To complete the conversion required to render MSFC-supplied trajectory data into tracking data, it is necessary to consider one more coordinate conversion. In Figure D-4 a launch-azimuth aligned coordinate system is shown, relative to the launch-site topocentric system, where

$\Lambda_{AZ} \triangleq$ launch azimuth measured from north.

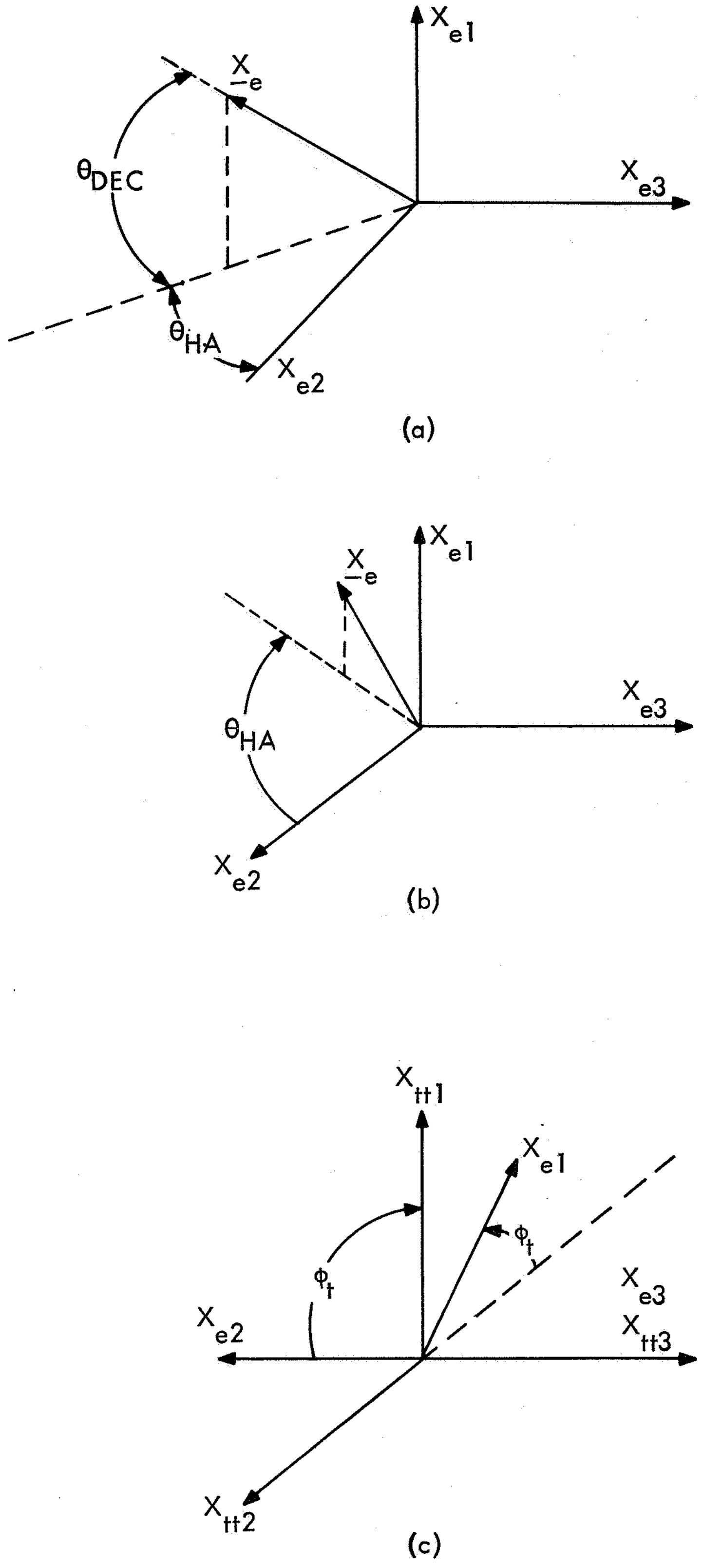


Figure D-3. Equatorial Coordinate System

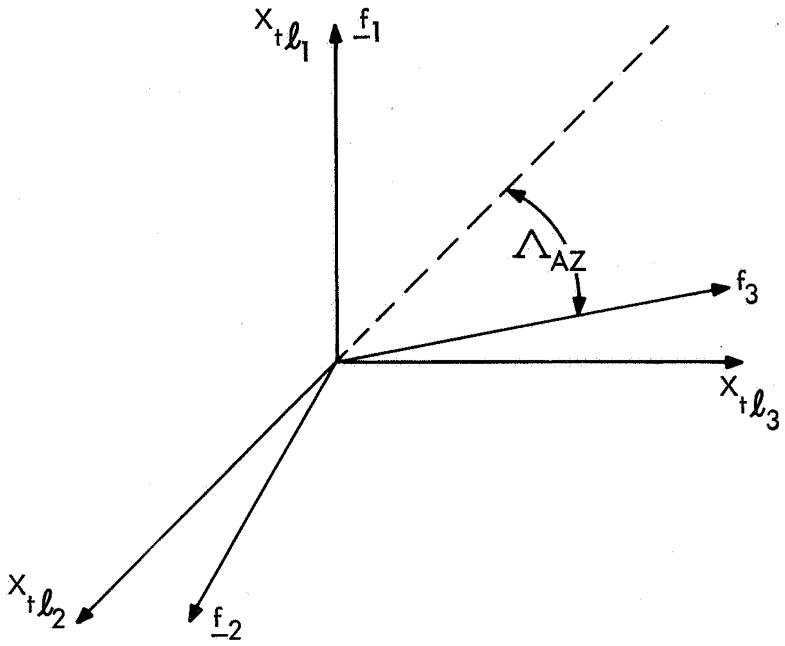


Figure D-4. Launch Azimuth - Aligned Coordinate System

The basis vectors, $\underline{f}_i$, can be expressed in the launch topocentric system, resulting in the matrix

$$\underline{F}_\ell = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \sin \Lambda_{AZ} & -\cos \Lambda_{AZ} \\ 0 & \cos \Lambda_{AZ} & \sin \Lambda_{AZ} \end{bmatrix} \qquad \text{(D-15)}$$

Now any point in a launch-azimuth-aligned system with coordinates $\underline{X}_f$ is expressible in the launch topocentric system as $\underline{X}_{t\ell} = \underline{F}_\ell \underline{X}_f$. For the trajectory data this launch-azimuth-aligned coordinate system is translated to the geocenter by using $\underline{r}_{t\ell}$, the geocentric radius of the launch topocenter. Hence, the basis vectors of the geocentric, launch-derived coordinate system in the reference inertial system are given by

$$\underline{G} \overset{\Delta}{=} \underline{T}_\ell \, \underline{F}_\ell \qquad \text{(D-16)}$$

since the axes of the geocentric, launch-derived system are parallel to those of the launch-azimuth aligned system. Let $\underline{X}_g$ denote the vehicle position as given in the MSFC-supplied trajectory data. Then the vehicle position in the reference inertial system is

$$\underline{X} = \underline{G} \, \underline{X}_g = \underline{T}_\ell \underline{F}_\ell \, \underline{X}_g \qquad \text{(D-17)}$$

Use of Eq. (D-5) gives

$$\underline{r}_t + \underline{T}_t \underline{X}_{tt} = \underline{X} = \underline{T}_\ell \underline{F}_\ell \, \underline{X}_g \qquad \text{(D-18)}$$

Solving for $\underline{X}_{tt}$ yields

$$\underline{X}_{tt} = \underline{T}_t^T \left(\underline{T}_\ell \underline{F}_\ell \, \underline{X}_g - \underline{r}_t \right) \qquad \text{(D-19)}$$

Conversion of MSFC-supplied trajectory data to tracking data is accomplished by use of Eq. (D-19) followed by the inverse of Eq. (D-14),

$$\underline{X}_e = \begin{bmatrix} \sin \phi_t & -\cos \phi_t & 0 \\ \cos \phi_t & \sin \phi_t & 0 \\ 0 & 0 & 1 \end{bmatrix} \underline{X}_{tt} \tag{D-20}$$

and the conversions:

$$r = \sqrt{X_{e1}^2 + X_{e2}^2 + X_{e3}^2} \tag{D-21}$$

$$\theta_{DEC} = \sin^{-1} \left(\frac{X_{e1}}{r}\right) \tag{D-22}$$

and

$$\theta_{HA} = \begin{cases} -\sin^{-1}\left(\dfrac{X_{e3}}{\sqrt{X_{e2}^2 + X_{e3}^2}}\right), & X_{e2} > 0 \\ -\mathrm{sgn}\,(X_{e3}) \cdot 90^\circ, & X_{e2} = 0 \\ -\mathrm{sgn}\,(X_{e3}) \cdot \left(90^\circ + \cos^{-1} \dfrac{|X_{e3}|}{\sqrt{X_{e2}^2 + X_{e3}^2}}\right), & X_{e2} < 0 \end{cases} \tag{D-23}$$

Equations (D-21), (D-22), and (D-23) follow from Figure D-3(a) and (b).

Expressions for the conversion of the time derivatives of all quantities may be easily obtained, since all of the transformation matrices are constant.

In the coordinate conversion expressions there appear certain constants associated with launch site and tracking site locations. The launch site constants were obtained, or computed, from MSFC-supplied data for the S-504 trajectory, and are

$$r_{t\ell} = 6{,}373{,}343.9 \text{ meters}$$

$$\phi_{\ell} = 28.608420 \text{ degrees}$$

and

$$\delta_{\ell} = 28.446962 \text{ degrees}$$

The tracker site used to develop the nominal tracking data was chosen to nearly coincide with that possible site which was closest to the launch pad. The constants for this specific case are in the heading of Figure 3-2. When an actual tracking site location is chosen and surveyed, the constants r_{tt}, ϕ_t, δ_t, and $\Delta\lambda$ will have to be entered into the coordinate conversion subroutine. The calculation of r_{tt} and δ_t should be made via the standard conversion from geodetic parameters to geocentric, using as a basis the Apollo reference geoid.

APPENDIX E

TARGET DYNAMICS ESTIMATOR

As discussed in paragraphs 2.5 and 3.6 the estimation procedure consists of the least-squares fitting of three overlapping cubic polynomials to successive data spans of 90 measurements (at a 64/second rate). In this appendix only the principals of operation in fitting one cubic are discussed, since the implementation of the overlapping feature is a programming problem.

Let the measurements be Z_i, i = 1,..., 90. Denote by Z_B the vector of measurements

$$\underline{Z}_B^T = \left| Z_1, \ldots, Z_{60} \right| \tag{E-1}$$

which are to be batch processed. In fitting the cubic $\alpha_0 + \alpha_1 t + \alpha_2 t^2 + \alpha_3 t^3$ to the measurements $\underline{Z}_B$, the mathematical problem is that of finding a parameter vector $\underline{\alpha}_B$, where

$$\underline{\alpha}_B^T = \left| \alpha_o \ \alpha_1 \ \alpha_2 \ \alpha_3 \right| \tag{E-2}$$

so that the set of equations

$$\begin{bmatrix} 1 & \left(\frac{1}{64}\right) & \left(\frac{1}{64}\right)^2 & \left(\frac{1}{64}\right)^3 \\ \vdots & & & \\ \vdots & & & \\ 1 & \left(\frac{60}{64}\right) & \left(\frac{60}{64}\right)^2 & \left(\frac{60}{64}\right)^3 \end{bmatrix} \underline{\alpha}_B = \underline{Z}_B \tag{E-3}$$

or

$$\underline{H}_B \ \underline{\alpha}_B = \underline{Z}_B \tag{E-4}$$

has a solution, in some sense. In particular it is desired to find that solution of Eq. (E-4), $\hat{\underline{\alpha}}_B$, which minimizes the weighted error function

$$\epsilon_B = \left(\underline{H}_B \hat{\underline{\alpha}}_B - \underline{Z}_B\right)^T \underline{R}^{-1}\left(\underline{H}\, \hat{\underline{\alpha}}_B - \underline{Z}_B\right) \tag{E-5}$$

where $\underline{R}$ is the correlation matrix for the measurement noise. It has been assumed that the measurements are corrupted by additive noise which is stationary and has correlation function

$$\rho(t) = \sigma^2 e^{-\tau|t|} \tag{E-6}$$

Consequently, the elements of $\underline{R}$ are

$$\left.\begin{array}{ll} r_{ii} = 1 & i = 1, \ldots, 60 \\ r_{ji} = r_{ij} = \exp\left(-\tau \dfrac{|i - j|}{64}\right) & \end{array}\right\} \tag{E-7}$$

and are computed by the function subroutine R (see Appendix A). The choice for $\hat{\underline{\alpha}}_B$ is given by the standard Gauss-Markov formula

$$\hat{\underline{\alpha}}_B = \left(\underline{H}_B^T \underline{R}^{-1} \underline{H}_B\right)^{-1} \underline{H}_B^T \underline{R}^{-1} \underline{Z}_B \tag{E-8}$$

Basically the batch-processing procedure is to compute the gain matrix

$$\underline{K}_B \triangleq \left(\underline{H}_B^T \underline{R}^{-1} \underline{H}_B\right)^{-1} \underline{H}_B^T \underline{R}^{-1} \tag{E-9}$$

before the measurements are taken, and, while the measurements are being taken, to compute the matrix-vector product in Eq. (E-8) one column at a time. Note that any intermediate value for $\hat{\underline{\alpha}}_B$ based on a partial processing of the vector $\underline{Z}_B$ has no importance or interpretation.

In the subroutine GAIN, $\underline{K}_B$ is not computed by means of the direct formula, Eq. (E-9), because of notorious numerical difficulties with taking the indicated inverse. Rather, the generalized inverse is used, which yields better numerical results. The generalized inverse can be directly applied to the solution of Eq. (E-4) only when the weighting matrix in ϵ_B, Eq. (E-5), is the identity matrix. This can be accomplished by factoring the positive-definite matrix $\underline{R}$ as $\underline{R} = \underline{S}\ \underline{S}^T$ where $\underline{S}$ is lower triangular; the factorization is by means of Cholesky decomposition. Then $\underline{R}^{-1} = \left(\underline{S}^{-1}\right)^T \underline{S}^{-1}$ where $\underline{S}^{-1}$ is calculated directly from $\underline{S}$ by a back substitution procedure. Cholesky decomposition and back substitution are performed in the subroutine CHOLSQR (see Appendix A) which yields $\underline{S}^{-1}$ given $\underline{R}$. Now Eq. (E-5) can be written as

$$\epsilon_B = \left[\underline{S}^{-1}\left(\underline{H}_B\ \hat{\underline{\alpha}}_B - \underline{Z}_B\right)\right]^T \left[\underline{S}^{-1}\left(\underline{H}_B\ \hat{\underline{\alpha}}_B - \underline{Z}_B\right)\right] \tag{E-10}$$

and Eq. (E-8) as

$$\hat{\underline{\alpha}}_B = \left(\underline{S}^{-1}\ \underline{H}_B\right)^{\dagger}\ \underline{S}^{-1}\ \underline{Z}_B \tag{E-11}$$

giving, as an equivalent expression for Eq. (E-9),

$$\underline{K}_B = \left(\underline{S}^{-1}\ \underline{H}_B\right)^{\dagger}\ \underline{S}^{-1} \tag{E-12}$$

The operation $(\cdot)^{\dagger}$, i.e., forming the generalized inverse, is performed by the subroutine GINV.

As mentioned in paragraph 4.2 the subroutines GAIN, CHOLSQR, GINV, and R may be used for an off-line batch processing of data. To facilitate this off-line processing, in the case of performing polynomial fits, the subroutine POLFITW is made available. This subroutine performs all of the bookkeeping associated with a weighted-least-squares fit and has selectable options for printout of results and errors. However, POLFITW fits only combinations of

1, t, t^2, t^3, etc., up to t^9. For higher degree polynomials or for longer time intervals it is sometimes advisable to work with a set of polynomials which are orthonormalized on the fitting interval. The subroutine POLORT is supplied to orthonormalize 1, t, t^2, and t^3 on any interval beginning at zero. Hence, in the search for better data smoothing results in off-line processing, orthonormalized polynomials may be used to minimize error contributions from purely numerical sources.

In the sequential estimation phase of each estimator subunit, the parameter vector estimate $\hat{\underline{\alpha}}_B$ is modified as each new measurement is made available. In order to sequentialize the Gauss-Markov formula, Eq. (E-8), it is necessary to assume the noise on the measurements to be white. In effect, $\underline{R}$ is assumed to be the identity matrix. The sequential Gauss-Markov formula is derived in many places in the literature; only the results are given here. Using the subscript k to index the measurements and parameter estimates within the sequential estimation phase, the updating formula is

$$\hat{\underline{\alpha}}_k = \hat{\underline{\alpha}}_{k-1} + \underline{K}_k\left(Z_k - \underline{H}_k \hat{\underline{\alpha}}_{k-1}\right) \tag{E-13}$$

where

$$\underline{K}_k = \underline{P}_{k-1} \underline{H}_k^T \left(\underline{H}_k \underline{P}_{k-1} \underline{H}_k^T + r\right)^{-1} \tag{E-14}$$

$$\underline{P}_k = \left(\underline{I} - \underline{K}_k \underline{H}_k\right)\underline{P}_{k-1} \tag{E-15}$$

$$\underline{H}_k = \left[1 \ t_k \ t_k^2 \ t_k^3\right]$$

and

r = mean-square value of measurement noise

The times t_k start at t_1 = 61/64 and go to t_{30} = 90/64. To start requires only

$$\underline{\hat{\alpha}}_0 = \underline{\hat{\alpha}}_B$$

and

$$\underline{P}_0 = \left(\underline{H}_B^T \underline{R}^{-1} \underline{H}_B\right)^{-1}$$

from the batch-processing phase. The gain vectors, $\underline{K}_k$, are also computed within the subroutine GAIN via Eqs. (E-14) and (E-15). The gain values are outputted by GAIN in a single array (having the dummy variable designation AK(I,J) within the subroutine) with the first 20 columns giving $\underline{K}_B$ and the remaining 30 columns giving $\underline{K}_k$ for k = 1,..., 30. Note that in the batch processing phase not every measurement made available at the 64 $\sec^{-1}$ rate is actually used. The vector $\underline{Z}_B$ is constructed by using every third measurement beginning at 3/64 second and ending at 60/64 second. This is not done to reduce computation (since the time must be available anyway every third measurement) but to reduce the estimation error due to correlated noise by spacing samples further apart. In the sequential estimation phase the requirement for output every 1/64 second does not allow a similar practice.

APPENDIX F

THE EFFECTS OF RANDOM ATMOSPHERIC TURBULENCE ON LASER BEAM PROPAGATION

RESEARCH REPORT NUMBER 542

THE EFFECTS OF RANDOM ATMOSPHERIC TURBULENCE ON LASER BEAM PROPAGATION

by

S. C. Bigelow

December 1967

Project No. 180

APPLIED RESEARCH LABORATORY
SYLVANIA ELECTRONIC SYSTEMS
Sylvania Electric Products Inc.
40 SYLVAN ROAD, WALTHAM, MASSACHUSETTS 02154

TABLE OF CONTENTS

TABLES

1. INTRODUCTION

The work reported here was undertaken in support of the design of a digital computer program for the real-time control of a laser tracker. The tracker automatically follows a moving target and provides measurements of target position. In addition to controlling the tracker the computer is to be programmed to smooth the raw position data, to estimate target velocity and acceleration, and to perform certain coordinate transformations. Random turbulence of the atmosphere affects laser beam propagation in various ways. In designing the control, smoothing and estimation programs, a quantitative statistical description of the atmospheric turbulence noise in the tracker target measurements would be most valuable. The study reported here had as its purpose ascertaining whether such a description was available or could be derived from available data.

Theoretical predictions of turbulence effects are discussed in Section 2 of this report. The theory treats a number of effects as separate and distinct. In practice, however, it is not always possible to measure one such effect in isolation. Some of the published experimental measurements of laser beam disturbances caused by the atmosphere are summarized in Section 3 and compared both with the theory and with one another. The conclusions drawn from this work are presented in Section 4.

2. THEORETICAL DISCUSSION

2.1 BACKGROUND

The effects of random atmospheric turbulence on laser beam propagation have been treated from a theoretical viewpoint by

Beckmann[1] and Hodara[2]. Both assume the validity of Tatarski's[3] blob theory, and they use methods of analysis based on geometric optics.

The atmosphere is characterized as having a refractive index, n, given by

$$n = 1 + N_0 \times 10^{-6} + \Delta n$$

where

N_0 = the average refractivity

and

Δn = the random variation in n.

The average refractivity, N_0, is a function of atmospheric pressure, temperature and composition. Its consequent spatial variation gives rise to a steady (dc) refraction over a given path. The random variable, $\Delta n(r,t)$ is a function of both space and time. Its effects must be evaluated statistically and it is these effects which are of particular interest here. The spatial and temporal correlation functions of Δn are defined as

$$C_{12}(r_1, r_2) = \langle \Delta n(r_1)\ \Delta n(r_2) \rangle$$

$$R_{12}(t_1, t_2) = \overline{\Delta n(t_1)\ \Delta n(t_2)}$$

where

$\langle\cdot\rangle$ denotes ensemble average

and

$\bar{\cdot}$ denotes time average.

It is usual to assume that the statistical distribution of Δn is the same at all points in the medium (homogeneity) and that C is therefore a function only of $\rho = r_2 - r_1$. I.e.

$$C_{12}(\rho) = \langle\Delta n(r)\ \Delta n(r+\rho)\rangle$$

This correlation function may be characterized by two parameters, $C_{11}(0)$ and L_c. The first is seen to be the variance of Δn

$$C_{11}(0) = \langle\Delta n(r)^2\rangle$$

The second called the correlation distance is the value of ρ at which

$$C_{12}\left(L_c\right) = \alpha C_{11}(0)$$

where α is a predetermined constant. If the medium is isotropic, i.e., if L_c is the same for all directions in space, then it may be assumed that $C_{12}(\rho)$ has the functional form

$$C_{12}(\rho) = \langle\Delta n^2\rangle\, e^{-\left(\rho/L_c\right)^2}$$

Assuming that $\Delta n(t)$ is a stationary process, it follows that

$$R_{12}(t_1, t_2) = R_{12}(\tau)$$

$$= \overline{\Delta n(t) \ \Delta n(t+\tau)}$$

i.e. the temporal correlation function of Δn depends only on τ. Like the spatial correlation function R may be characterized by $R_{11}(0)$ and a correlation time, T_c. If Δn is ergodic then

$$R_{11}(0) = C_{11}(0) = \langle \Delta n^2 \rangle = \overline{\Delta n^2}$$

There is reason to believe that the conception of the atmosphere as consisting of frozen inhomogeneities (blobs), whose linear dimensions are proportional to L_c, is a meaningful one. This concept permits establishment of a simple relation between L_c and T_c. If the component of relative velocity between the atmosphere and the laser beam transverse to the beam is v_T, then

$$T_c = L_c / v_T$$

This transverse velocity component may be due to wind, beam motion or both.

Consider now the effects on the beam, as observed at the receiver of a laser communications or tracking system, induced by a random turbulent atmosphere as described above. These effects will include the following:

a) The angle of arrival of the beam at the receiver will fluctuate (quivering).

b) The location of the beam center will vary (beam wander, spot dancing).

c) The cross-section of the beam will vary in size and shape (breathing).

d) The transit time will vary causing phase fluctuations.

All these effects result from refraction of the beam as a whole. The remaining effects are due to differential refraction of the rays making up the beam which causes wavefront distortion.

e) The image formed by the receiving optics will be blurred.

f) The intensity distribution across the beam will fluctuate due to reinforcement and cancellation (boiling).

g) The phase distribution across the beam will fluctuate due to transit time variations.

From the point of view of a particular laser system several of these effects may act in combination and some may be of little consequence so far as system performance is concerned. The particular system of interest here is a laser tracking and ranging system. Its function is to measure the angular position and range of a target. The system consists of a cw-laser transmitter and a receiver telescope mounted coaxially in servo controlled gimbals. A retroreflector on the target returns the transmitted beam to the receiver aperture. The receiver optics focus the reflected light on a star tracker tube whose two output signals are proportional to the target image displacement from the receiver boresight. These signals are used by the gimbal servos to track the apparent direction of arrival of the received light. Low frequency intensity fluctuations (fading) induced by the atmosphere are minimized by means of automatic gain control in the receiver. The transmitted beam is modulated at one or more frequencies and this modulation is detected in the receiver. The phase shift of the received modulation relative to the transmitted modulation is measured in the receiver and provides a measure of the target range.

For this particular system the atmospheric turbulence effects of primary interest are fluctuations in angle of arrival and received phase, since these affect directly the system measurement data. Of secondary interest in designing the AGC system are random fluctuations in received intensity. The ability to predict these effects would materially aid the design of i) the tracking system servo loops, ii) the AGC system, iii) the ranging system phase comparator and iv) the algorithms for smoothing the measurement data. In the following section the theoretical analyses of these particular effects as derived by Beckmann[1] and Hodara[2] are presented and compared.

2.2 THEORETICAL RESULTS

Both Beckmann and Hodara use the methods of geometric optics in their analyses. The condition under which these methods are applicable is

$$L_c \gg \sqrt{L\lambda}$$

where L_c is the correlation distance of atmospheric inhomogeneities

L is the pathlength

λ is the wavelength.

This condition implies that diffraction effects of turbulence blobs of width L_c are negligible, i.e., that all points on the optical path are in the near field of the farthest turbulence blob. For optical wavelengths and practical atmospheric pathlengths this condition is usually met.

2.2.1 Quivering

The angle of arrival at a receiver may be resolved into two orthogonal components $\Delta\theta_x$ and $\Delta\theta_y$ transverse to the direction of beam propagation (z-direction). The mean square values of these components are according to Beckmann

$$\langle\Delta\theta_x^2\rangle = \mu \langle\Delta n^2\rangle L\, L_{cz}/L_{cx}^2$$

$$\langle\Delta\theta_y^2\rangle = \mu \langle\Delta n^2\rangle L\, L_{cz}/L_{cy}^2$$

For an isotropic medium $L_{cx} = L_{cy} = L_{cz} = L_c$ and

$$\langle\Delta\theta_x^2\rangle = \langle\Delta\theta_y^2\rangle = \mu \langle\Delta n^2\rangle L/L_c$$

where

$$\mu = 2\sqrt{\pi}$$

The analogous results derived by Hodara are identical to the above except that the constant, μ, is 1 instead of $2\sqrt{\pi}$. The variance of the total angular fluctuation may be taken to be the sum of variances of the components.

$$<\Delta\theta^2> = <\Delta\theta_x^2> + <\Delta\theta_y^2>$$

These relations give the variance of the angle of arrival of a laser beam at the receiving end of the path. The apparent direction to the source will fluctuate with the angle of arrival. The effect of this angular variation on the target direction detector and the pointing servos must be considered in the system design.

2.2.2 Beam Wander

Just as random variation in the index of refraction causes quivering of the apparent angle of arrival of the received beam, it also causes the entire beam to wander about an average position in the receiver plane. The theoretical variance of this translational wander as derived by Beckmann is $L^2/2$ times the variance of the angular fluctuation, e.g.

$$<\Delta x^2> = <\Delta\theta_x^2> L^2/2$$

$$= \mu <\Delta n^2> L^3 L_{cz}/2L_{cx}^2$$

Hodara assumes without proof time the rms wander is L times the rms quivering. I.e.,

$$\Delta x_{rms} = L \ \Delta\theta_{xrms}$$

$$= L\sqrt{\langle\Delta\theta_x^2\rangle}$$

This implies that the variance of Δx will be L^2 times the variance of $\Delta\theta_x$, which agrees with Bechmann's result to within a factor of $\sqrt{\pi}$.

The effect of beam wander on a tracking system will depend on the relative magnitudes of the beam diameter at the receiver, the receiver aperture diameter and the rms beam wander. If the beam diameter plus the beam wander is small compared to the aperture, then all of the beam will be collected all of the time. In this case the usual type of angle of arrival sensor (star tracker tube or image dissector) will sense the beam wander and its effect will be indistinguishable from fluctuations in angle of arrival. If the receiver aperture is small enough compared with the beam diameter that it is always in the beam, then the beam wander will cause apparent intensity modulation. However, assuming the angle sensor to be insensitive to this modulation, the beam wander will not affect angle of arrival measurements. If the beam diameter, the aperture diameter and the rms beam wander are comparable in magnitude, then the beam wander may cause the beam to entirely miss the receiver part of the time. It must be supposed that this situation would degrade system performance to an intolerable extent and it should, therefore, be avoided by suitable system design.

2.2.3 Breathing

Random fluctuations in beam cross-sectional area, called breathing, cause intensity modulation effects to be observed if only part of the beam is collected by the receiving aperture. Both Beckmann and Hodara give conditions on the desired signal modulation index, m, such that the signal modulation power exceeds the breathing modulation power. The derivations of these conditions involve many assumptions and approximations. Furthermore, they are not readily verifiable via experiment because those modulation effects caused solely by breathing cannot be separated from those due to other causes such as beam wander and boiling. In any case the results are as follows.

According to Beckmann the condition which must be satisfied is (for the isotropic case)

$$\sqrt{2\left[\exp\left(\frac{8\sqrt{\pi}}{3}\langle\Delta n^2\rangle\frac{L^3}{L_c^3}\right)-1\right]}\leq m\leq 1 \qquad \text{(B)}$$

Hodara gives two relations for L/L_c in different ranges. They are

$$2\left(\langle\Delta n^2\rangle\frac{L^3}{L_c^3}\right)^{1/4}\leq m\leq 1 \text{ for } L/L_c \ll 10^4 \qquad \text{(H1)}$$

and

$$\sqrt{2}\left(\langle\Delta n^2\rangle \frac{L^3}{L_c^3}\right)^{-1/2} \leq m \leq 1 \text{ for } L/L_c \gg 10^4 \qquad \text{(H2)}$$

Numerical evaluation of these three expressions for $1 \leq L/L_c \leq 10^6$ and $\langle\Delta n^2\rangle = 10^{-12}$, a value which agrees well with experimental measurements over paths near the earth's surface, shows that expressions B and H1 give values of $m < 1$ for $L/L_c \leq 2 \times 10^3$ and that these values agree within a factor of ten for $2 \times 10^2 \leq L/L_c \leq 2 \times 10^3$. Expression H2 gives values of $m < 1$ for $L/L_c \geq 2 \times 10^4$. This particular expression is difficult to reconcile with physical intuition, since it implies that modulation effects due to breathing decrease with increasing pathlength when the path is very long. Moreover, for such large values of L/L_c the near field condition, $L_c \gg \sqrt{L\lambda}$ is easily violated.

If nothing else, these theoretical results indicate that breathing effects alone can produce deep fading in the received signal. Therefore, the laser receiver system should include an AGC which is wideband enough to minimize the relatively low frequency AM induced by the atmosphere.

2.2.4 Phase Fluctuations

Variations in the index of refraction along the transmission path produce random fluctuations in the velocity of propagation of the laser wavefront. If a coherent plane wavefront is assumed at the transmitter aperture, then the phase at any point in the receiver aperture will vary randomly about a value which increases linearly with time. This may be termed temporal incoherence. In addition, the phase across the receiver aperture will be non-uniform at any instant of time. This will be referred to as spatial incoherence. In what follows only the isotropic case is considered.

The first of these effects is found by Beckmann to produce a phase variance given by

$$<\Delta\phi^2> = \sqrt{\pi}\ k^2\ <\Delta n^2>\ L\ L_c$$

where

$k = \omega/c = 2\pi/\lambda$

ω = laser frequency

λ = laser wavelength

This expression also yields the phase variance of a modulation signal at ω_m if k is replaced by k_m

where

$k_m = \omega_m/c = 2\pi/\lambda_m$

Hodara's result is identical to Beckmann's except it does not contain the factor, $\sqrt{\pi}$. The second effect may be characterized in terms of the cross-correlation function of the instantaneous phase at two points in the aperture plane separated by a distance, X. Bechmann's result in this case is

$$<\Delta\phi_1\ \Delta\phi_2> = \sqrt{\pi}\ k^2\ <\Delta n^2>\ L\ L_c e^{-\left(X/L_c\right)^2}$$

and Hodara's result differs from this only in the omission of the factor, $\sqrt{\pi}$.

Random phase variations in the received signal cause spurious frequency modulation. If we define

$$\Delta\omega = \frac{d\ \Delta\phi}{dt}$$

then the variance of this frequency modulation may be found. The derivation involves the introduction of the inhomogenity correlation time, T_c, and the results are given in terms of L_c and v_T where $v_T = L_c/T_c$. Beckmann finds

$$\langle\Delta\omega^2\rangle = 2\sqrt{\pi}\ k^2\ \langle\Delta n^2\rangle\ v_T\ L/L_c$$

Hodara's result is the same except the factor $2\sqrt{\pi}$ is omitted.

3. EXPERIMENTAL RESULTS

The theoretical results presented in the previous section express the mean square values of the various effects of interest in terms of pathlength, L, wavelength λ, and the parameters of the correlation functions of the random component of the index of refraction, $\langle\Delta n^2\rangle$, L_c and T_c. A number of investigators have performed experiments designed to measure atmospheric effects on a laser beam. In this section those results which are most relevant to laser tracking and ranging systems will be presented and discussed.

3.1 FUNDAMENTAL ATMOSPHERIC PARAMETERS

As pointed out in Section 2.1 atmospheric turbulence effects on laser beam propagation can be characterized in terms of the variance of the index of refraction, $\langle \Delta n^2 \rangle$, and a correlation distance L_c. Hodara[2] gives empirical formulas for these parameters as functions of altitude, h in meters. These are presented below.

$$\langle \Delta n^2 \rangle = 10^{-12} \exp(-h/1600)$$

$$L_c = 0.4h/(1 + 10^{-2}h)$$

The first of these indicates that $\langle \Delta n^2 \rangle$ is maximum at the surface (h=0) and decreases rapidly with altitude. L_c on the other hand is minimum at the surface and increases toward an assymptotic value of 40 m with increasing altitude. Hodara does not state how these relations were obtained nor the conditions under which they are valid. However, they will be assumed for purposes of the present discussion to give results which are at least approximately correct.

3.2 CORRELATION FUNCTIONS AND SPECTRAL DENSITY FUNCTIONS

The design of computer programs for smoothing and estimation of laser tracker data would be materially benefitted if we could develop a statistical model of the measurement noise. The theoretical results of Beckmann and Hodara can be used to predict the noise variance, but the variance alone is not a complete description of the noise process. What is required is a correlation function or a spectral density function. There is some theoretical and much empirical[4] evidence indicating that an exponential cosine correlation function is a suitable choice for the modelling of atmospheric turbulence effects. Such a correlation function has the form

$$R(\tau) = A\ e^{-k|\tau|} \cos c\ \tau$$

The corresponding spectral density function is

$$W(\omega) = \frac{2\ A\ K}{\pi} \left[\frac{\omega^2 + \left(k^2 + c^2\right)}{\omega^4 + 2\left(k^2 - c^2\right)\omega^2 + \left(k^2 + c^2\right)^2} \right]$$

If $3c^2 > k^2$, $W(\omega)$ has a maximum at

$$\omega_m = \left(k^2 + c^2\right)^{1/4} \left[2c - \left(k^2 + c^2\right)^{1/2} \right]^{1/2}$$

In cases where the spectral density decreases monotonically with frequency from a maximum value at $\omega = 0$, these models may be simplified by taking $c = 0$. In this case

$$R(\tau) = A\ e^{-k|\tau|}$$

and

$$W(\omega) = \frac{2\ A\ k}{\pi} \frac{1}{\omega^2 + k^2}$$

The parameters A and k of these functions have the following interpretation. Let the random process they describe be $\{x(t)\}$. Then by definition

$$R(\tau) = \overline{x(t)\ x(t+\tau)}$$

from which it follows that

$$R(0) = \overline{x(t)^2} = \text{var } x = A$$

Thus, the parameter A is the mean square value of the random process. The parameter k may be viewed as the inverse of the correlation time constant. The parameter c is the angular frequency of the periodic component, if any, of $R(\tau)$.

3.3 QUIVERING

Kurtz and Hayes[5] have performed a series of experiments intended to measure random fluctuations in the angle of arrival of a laser beam. Two paths were used, one 3200 m long; the other 165 m long. The receiver employed a Questar telescope with 9 cm aperture and a star tracker tube. The transmitter was a He - Ne laser operating at 6328Å. The receiver was mounted 2 m above ground. The elevation angle of the receiver was about 4° for both paths. The star tracker tube produced two outputs proportional to the focused spot horizontal and vertical displacement from a central reference position. These outputs were recorded and then analyzed to obtain amplitude spectra in the range 2 - 150 Hz with a 1 Hz filter bandwidth. After filtering to a 15 Hz bandwidth these outputs were also processed by a probability density analyzer.

The results of 55 measurements over the 3200 m path and 52 measurements over the 165 m path were processed. No significant difference was found between measurements taken in the vertical and horizontal directions. The computed characteristics of the two sets of data are presented in Table 3-1.

TABLE 3-1

ANGLE OF ARRIVAL MEASUREMENTS

	Path Length	
	3200 m	165 m
Mean rms 0-15 Hz (μrad)	15.7	9.2
Limiting rms 0-15 Hz (μrad)	26.0	17.0
Limiting peak 0-15 Hz (μrad)	48.3	28.6
Mean rms 0-15 Hz (μrad)	31.4	18.4

Typical frequency spectra derived from the recorded measurements are shown graphically by Kurtz and Hayes. While these vary somewhat in shape, they are all characterized by a maximum spectral density at low frequency (below 20 Hz) and decreasing density with frequency above 20 Hz. At 150 Hz (the upper limit of the data presented) all of the spectral densities are at least 10% of their maximum values, which indicates a significant amount of noise power above this frequency. No attempt is made by the authors to compare their experimental spectral densities with theoretical results nor to fit them with analytic expressions. It has been found, however, that many of their measured densities can be approximated quite closely by a spectral density function of the form (see Section 3.2).

$$W(\omega) = \frac{2\,A\,k}{\pi}\left[\frac{1}{\omega^2 + k^2}\right]$$

Typical parameter values for the 3200 m case are

$$A \cong 10^4\ (\mu\text{rad})^2$$

$$k = 200\ \text{rad/sec}$$

Values which give a good fit to data for the 165 m case are

$$A = 5 \times 10^2\ (\mu\text{rad})^2$$

$$k = 24\ \text{rad/sec}$$

In Section 2.2.1 the theoretical variance for angular quivering in the x- and y- directions was given as

$$\langle\Delta\theta_x^2\rangle = \langle\Delta\theta_y^2\rangle = \mu\ \langle\Delta n^2\rangle\ L/L_c$$

where

$$\mu = \begin{cases} 2\sqrt{\pi} & \text{(Beckmann)} \\ 1 & \text{(Hodara)} \end{cases}$$

The variance of the beam wander is (see Section 2.2.2)

$$\langle \Delta x^2 \rangle = \alpha \langle \Delta n^2 \rangle L^3 / L_c$$

where

$$\alpha = \begin{matrix} \sqrt{\pi} & \text{(Beckmann)} \\ 1 & \text{(Hodara)} \end{matrix}$$

The theoretical formula for $\langle \Delta\theta_x^2 \rangle = A$ predicts a linear dependence on L. This is borne out by the experimental results, since A(3200) $\cong$ 20 A(165). If we assume a value for $\langle \Delta n^2 \rangle$ of 10^{-12}, we may compute L_c using the experimental values of A. Using Beckmann's formula this yields

$$L_c \cong 1 \text{ m}$$

for both experimental cases. It should be noted here that this empirical value of L_c does not agree well with Hodara's predictions (see Section 3.1 and Table 3-2) for the two cases which are considerably larger. Of course, this discrepancy may be apparent rather than real, since there is no reason to believe the assumed value of $\langle \Delta n^2 \rangle$ is either accurate or constant independent of the particular path.

It is clear that the experimental data of Kurtz and Hayes can be fitted very closely with spectral density functions which correspond to exponential or exponential cosine correlation functions and the predicted linear dependence of A on pathlength, L, is corroborated by the experimental evidence. However, too little is known about the dependence of $\langle \Delta n^2 \rangle$, L_c and k on atmospheric and equipment parameters to model the quivering correlation or spectral density functions quantitatively under general conditions.

The best fit values of A and k were used to compute rms quivering in the frequency range 0-15 Hz by numerical integration of $W(\omega)$. The result for the 3200 m case was approximately 20 μrad and for the 165 m case 8 μrad. These values agree well with the experimental results which were 15.7 and 9.2 μrad respectively (see Table 3-1). The theoretical rms quivering and beam wander for the two experimental pathlengths have been computed using the formulas derived by Beckmann and Hodara. The results are given in Table 3-2.

TABLE 3-2

THEORETICAL QUIVERING AND WANDER

	L 3200 m	165 m
Average path height, $\bar{h} = 2 + L \sin 4^{o}/2$ (m)	115.	8.75
Correlation distance $L_c(\bar{h}) = 0.4\bar{h}/(1+0.01\bar{h})$ (m)	20.4	3.2
Index variance $\langle \Delta n^2 \rangle = 10^{-12} \exp(-\bar{h}/1600)$	0.93×10^{-12}	1.00×10^{-12}
Rms quivering, $\Delta\theta_x$ (μrad)		
(Beckmann)	22.8	13.5
(Hodara)	12.1	7.16
Rms beam wander, Δx (cm)		
(Beckmann)	5.16	0.16
(Hodara)	3.87	0.12

Comparing Tables 3-1 and 3-2 we see that for each pathlength the two theoretical values for rms quivering bracket the measured value in the band 0-15 Hz. Thus, predictions based on theory are of the same order of magnitude as the measurements. However, the theoretical values include contributions at all frequencies and the measured values even in a band limited to 150 Hz are greater than predicted. This discrepancy is not surprising, since the theory predicts quivering due only to random refraction of the entire beam, whereas the experimental receiver is sensitive to other effects and it introduces some noise itself. The star tracker tube measures the position of the beam image in the focal plane of the receiver optics. This position will change not only as a result of deviations in the angle of arrival of the entire beam but also as a result of phase front disturbances within the beam (boiling). Furthermore, if the entire beam is collected by the receiver aperture, then the star tracker tube will be sensitive to beam wander too. It is not possible to say whether this last effect is significant in the experiments described here, because the characteristics of the laser transmitter are not given by Kurtz and Hayes in sufficient detail. What we need to know is the transmitter aperture or beam divergence. Assuming the laser to have a beam divergence of approximately 0.7 mrad (typical of many commerical units) then the half-power beamwidth at the receiver aperture is given by

$$D_r = 0.7 \times 10^{-3} L$$

$$= 11.5 \text{ cm for } L = 165 \text{ m}$$

$$= 224 \text{ cm for } L = 3200 \text{ m} .$$

It may be seen that under this assumption the received beam over the 165 m path is about equal in size to the receiver aperture (9 cm). It is possible, therefore, that for this pathlength beam wander may contribute to the measured angle of arrival. Over the long path, however, the postulated received beamwidth is so large compared to the predicted beam wander and the receiver aperture that sensitivity to beam wander should be very small.

To summarize this discussion it may be said that the experimental results obtained by Kurtz and Hayes show a measured rms quivering of the same order of magnitude as predicted by theory. However, in practice other random effects will contribute to the noise in angle of arrival measurements so that theoretical values based only on quivering effects may be expected to be somewhat low. No attempt was made in these angle of arrival experiments to correlate the measured spectral densities with atmospheric conditions such as pressure, temperature and crosswind velocity although some information on weather conditions during each experimental run is reported. More experimental work is needed to determine the dependence of the noise model parameters, A and k, on atmospheric conditions, pathlength, path elevation, motion of the path relative to the atmosphere and equipment aperture sizes.

3.4 PHASE FLUCTUATIONS

The results of a phase fluctuation measurement experiment have been reported by Buck[6]. The apparatus used was an equal-arm Michelson interferometer with round-trip arm lengths of 48.8 m. The reference arm was enclosed in a tube. The mixed measurement and reference beams were detected at two points separated by a distance $\lambda/4$ to obtain quadrature measurements of the random phase fluctuation at the laser source frequency ($\lambda = 0.6328\ \mu$). Data were taken at night under calm conditions over a horizontal path 1.4 m above ground. The rms phase fluctuation was found for about 16 samples each 1.7 sec long. The average measured fluctuation was 2.5 rad.

The phase fluctuation predicted by theory has been calculated using Beckmann's formula

$$\Delta\phi_{rms} = \left(\sqrt{\pi}\ k^2\ \langle\Delta n^2\rangle\ L\ L_c\right)^{1/2}$$

with

$$k = 2\pi/\lambda = 0.99 \times 10^7 \text{ rad/m}$$

$$\langle\Delta n^2\rangle = 10^{-12}$$

$$L_c(h) = L_c(1.4) = 0.55 \text{ m}$$

$$L = 48.8 \text{ m}$$

The result is

$$\Delta\phi_{rms} = 68.8 \text{ rad}$$

The theoretical and experimental results do not agree well. The reason for this large discrepancy is not know. Buck points out that his experimental data are quite limited in number. Also, the value of $\langle n^2\rangle$ used to predict $\Delta\phi_{rms}$ in the above calculation was based on Hodara's empirical formula. If the experimental conditions were so calm that $<\Delta n^2>$ was as small as 10^{-15}, then the theoretical prediction of $\Delta\phi_{rms}$ would be 2.17 rad, which agrees well with the measured value.

The only conclusion that may be drawn from this discussion is that insufficient experimental evidence is available to either confirm the theory or discredit it. In any case, since in the application of interest the target range is measured in terms of phase shift in several modulating frequencies, it is the atmospheric effects on the modulation phase rather than the laser carrier phase that are of interest. If one considers the following worst case conditions

$$f_{m\ max} = 30 \text{ mHz}$$

$$\lambda_{m\ min} = 10 \text{ m}$$

$$k_{m\ max} = 2\pi/\lambda_{m\ min} = 0.628 \text{ rad/m}$$

$$<\Delta n^2>_{max} = 10^{-12}$$

$$L_{\ max} = 20 \text{ km (round trip)}$$

$$L_{c}\ max = 40 \text{ m}$$

then the theoretical phase fluctuation is

$$\Delta\phi_{rms} = 0.75 \text{ mrad} .$$

In this example a change in phase of 2π rad indicates a change in target range of 5 m. The phase-to-range scale factor is thus $5/2\pi$ m/rad. The theoretical worst case phase fluctuation corresponds, therefore, to an rms range error of only 0.06 cm. If the phase noise due to atmospheric turbulence actually encountered in practice is no worse than this prediction, it may be concluded that atmospheric effects on the laser range measuring system will be small compared to equipment induced noise.

3.5 INTENSITY FLUCTUATIONS

In a laser ranging and tracking system the fundamental variables which are measured are angle of arrival and modulation phase shift. To a large extent the measurement instruments are inherently or by design (through automatic gain control) insensitive to intensity fluctuations. However, such fluctuations are of interest in the design of the AGC system and in evaluating the second order effects which they will have on the primary measurements. Some of the experimental observations of intensity variations will, therefore, be discussed in this section.

3.5.1 Atmospheric Modulation

The extent to which the atmosphere causes modulation of the intensity of a laser beam has been investigated by Buck[6] and by Subramanian and Collinson[7]. Buck defines a fluctuation index, σ_n as the standard deviation of the received intensity signal normalized to its mean.

$$\sigma_n = \sigma/\mu$$

The fluctuation index was measured as a function of receiver aperture for pathlengths in the range 0.55 - 145 km. The index was found to decrease from a value of about 1 at an aperture of 1 mm to values less than 0.02 for apertures which were large compared to received beamwidth. No systematic dependence on pathlength was observed in these experiments. Buck attributes this modulation to boiling within the beam. However, even if the intensity distribution across the beam were smooth, some fluctuation would be expected due to beam wander.

The results reported by Subramanian and Collinson are in terms of percent modulation as a function of receiver aperture. They also found a decrease in modulation with increasing aperture to the point where essentially all of the beam was collected. However, they observed a relatively large (around 1%) and constant residual modulation for apertures larger than the received beam.

These experiments indicate that beam boiling and wander cause large spatial and temporal fluctuations in the intensity of the received beam. If the receiving aperture is small compared to the extent of the beam, severe intensity modulation of the received signal results. Enlarging the receiving aperture reduces this effect, but only to the point where the aperture collects substantially all of the beam power. Decreasing the size of the transmitted beam by increasing the collimator aperture can reduce the atmospheric modulation of the received beam just as increasing the receiver aperture does. However, there is a limitation to this approach. Buck has found experimentally that a transmitting aperture of 11 cm yields a minimum beamwidth over paths a few kilometers in length. The minimum diameter of the beam is about four times the theoretical Airy disc size for this transmitting aperture. This effect is attributed by Buck to spherical aberration in the transmitter optics. Since the measurement technique involved the direct exposure of photographic plates for 30 seconds, the measured beamwidths include the integrated beam wander. This probably accounts for most of the discrepancy between experimental and theoretical beamwidth.

3.5.2 Spectral Density of Modulation

Reports of experimental measurements of the spectral density functions of atmospheric laser intensity modulation have been given both by Buck[6] and by Gilmartin and Horning[8]. However, neither of these experiments appears to have been well controlled, and the results are presented in sufficiently different forms as to make comparison difficult. Generally, both investigations yielded power spectral densities which decreased with increasing frequency. The frequency range from 0.2 to 500 Hz was covered.

Buck's measurements were made in Colorado at an average elevation of 2 km above sea level. The transmitter was on a mesa 60 m above the general terrain. Measurements were made over distances of from 0.55 to 145 km. The average spectral slopes in the range 1 - 10 Hz and 10 - 20 Hz were compared as functions of path length and receiver aperture with a fixed transmitting aperture of 15 cm. No obvious correlation was found between these slopes and pathlength. Both slopes did, however, become more steeply negative with increasing receiver aperture. Apparently, no crosswind velocity data were taken.

The experiments performed by Gilmartin and Horning used a folded optical path at a desert location. The transmitter aperture was 5 cm and the receiver aperture was 32 cm. In one setup both receiver and transmitter were on a ridge 247 m above the desert floor and an array of from 12 to 25 corner cube reflectors each 5 cm in diameter on the desert was used to return the beam. One way path lengths of from 3.2 to 25.6 km were used. In the other setup both ends of the range were on the desert floor separated by distances of from 0.4 to 4.15 km. Measured power spectral densities W(f) were plotted in the form fW(f) vs ln f. The function, fW(f), has a maximum value at some frequency, f_p. The formula for this frequency as derived by Tatarski[3] is

$$f_p = K\, v_T/\sqrt{L\lambda}$$

where

v_T = crosswind component of velocity

L = path length

λ = wave length

Gilmartin and Horning found good correlation between their measurements and this predicted value of f_p. Their experimental value for K was 2.5 ± 32% for the ridge-floor paths and 2.8 ± 34% for the floor-floor paths. They found further that all of their measured spectral densities could be closely fitted with a log-Maxwellian curve, viz

$$fW(f) = \frac{4}{\sqrt{\pi}} \frac{\left(\ln(\beta f_p/f)\right)^2}{(\ln\beta)^3} \exp - \frac{\left(\ln(\beta f_p/f)\right)^2}{(\ln\beta)^2}$$

The parameter β was found experimentally to have a value of about 12.

Buck has not taken crosswind velocity into account and he reports no significant variation with path length in measured spectral densities. This finding can be reconciled with Gilmartin and Horning's only if one supposes that in Buck's experiments the crosswind velocity increased fortuitously as the square-root of path length. Buck did find, however, that for a given transmitter aperture and path length the receiver aperture size affected the spectral density shape. Gilmartin and Horning on the other hand used fixed transmitting and receiving apertures and a variable (but not precisely specified) retroreflector aperture. Thus, it would appear that their results may depend on the aperture effect found by Buck as well as the crosswind effect to which they attribute them. It must be concluded that neither group of investigators has controlled or reported all of the parameters which a comparison of their experiments shows may be of significance. Clearly more carefully designed experiments need to be performed, if these ambiguities are to be resolved.

4. CONCLUSIONS

The purpose of this section will be to draw from the preceeding discussion of theoretical and experimental results those conclusions which are of relevance to the problem of designing computer programs for use in conjunction with the planned laser tracking and ranging system. The raw data input to the computer from the tracker consists of two orthogonal angular pointing error measurements and a range measurement. What is desired for computer program design purposes is a model of the statistical characteristics of the noise in these measurements caused by random turbulence in the atmosphere.

The primary causes of noise in the angle of arrival of the laser beam at the receiver are quivering and boiling. As was pointed out earlier, beam wander and intensity modulation can also contribute to this noise depending on the size of the receiver aperture in relation to received beamwidth and on the characteristics of the error sensing device and associated AGC.

Theoretical predictions and experimental measurements of angular quivering have been compared in Section 3.3. It was found that the experimentally measured spectral density functions for 3200 and 165 m fixed pathlengths could be fitted in many cases with a model having an exponential correlation function of the form

$$R(\tau) = A\ e^{-k|\tau|}$$

where

$$A \cong 3L\ (\mu\text{rad})^2$$

$$L = \text{pathlength (m)}$$

$$k \cong 200\ (\text{rad/sec})\ \text{ for } L = 3200$$

$$\cong 24\ (\text{rad/sec})\ \text{ for } L = 165$$

Attempts to correlate this empirical result with theoretical predictions were not very successful. The parameter A is the variance of the random quivering which is given theoretically by

$$A = \langle \Delta\theta_x^2 \rangle = \langle \Delta\theta_y^2 \rangle$$

$$= \mu \langle \Delta n^2 \rangle L/L_c$$

where

$$\mu = 2\sqrt{\pi} \text{ (Beckmann)}$$

$$= 1 \quad \text{(Hodara)}$$

$\langle \Delta n^2 \rangle$ = variance of index of refraction

L_c = correlation distance of inhomogeneities

According to Hodara both $\langle \Delta n^2 \rangle$ and L_c are functions of altitude, h. Thus, over a constant altitude path the thoery predicts a linear dependence of A on L. However, for the two paths in question h was not constant, since they were both elevated about 4° above horizontal. If $\langle \Delta n^2 \rangle$ and L_c are evaluated using Hodara's formulas and an average value of h over the path, values of A are obtained which are much smaller than the best fit experimental values. This difference is due in part to the fact that the experimental measurements include noise from other sources in addition to quivering. It seems quite probable, too, that both $\langle \Delta n^2 \rangle$ and L_c are strongly dependent on atmospheric conditions which vary with both time and space. Representing them as functions of altitude only must be a simplification based on some average condition of the atmosphere.

If one considers the spectral density function

$$W(\omega) = \frac{2\ A\ k}{\pi}\left[\frac{1}{\omega^2 + k^2}\right]$$

the parameter A is the total area under the $W(\omega)$ - curve i.e.,

$$\int_0^\infty W(\omega)\,d\omega = A$$

The parameter, k, determines the shape of the $W(\omega)$ - curve. In particular

$$W(0) = 2A/\pi k$$

and

$$W(k) = A/\pi k = W(0)/2$$

Thus, at $\omega = k$ the spectral density function is one half of its zero-frequency value. In view of the theoretical model of atmospheric turbulence discussed in Section 2, the parameter k must depend on correlation distance, L_c, wind velocity, v_T, and temperature and pressure gradients along the path. However, the nature of this dependence is unknown. More experimental work must be done, before it will be possible to predict k quantitatively. Qualitatively Kurtz and Hayes' experiments show that k increases with pathlength and is widely variable with time for a fixed path. (Values in the range 125 - 375 rad/sec were obtained in fitting $W(\omega)$ to different experimental spectra for the 3200 m path.)

The theoretical formulas for angular quivering may well be correct, but because of the large discrepancies between what they predict and the experimental results available at present, they must be considered unproven. The difficulty may simply be due to lack of precise knowledge of the variance and correlation distance of the random fluctuations in the atmospheric index of refraction. It is clear, however, that angular quivering effects are of sufficient magnitude to significantly affect the operation of the tracker pointing system and data processing algorithms. For purposes of designing computer programs for use with the tracker all that can be done pending more thorough experimentation is to assume the angle noise may be represented by an exponential correlation function with fixed and hopefully conservative parameters.

As pointed out in Section 3.4 the theoretical phase angle noise at the modulation frequencies used in the laser tracker ranging system is very small. Because the phase comparison circuits used to measure phaseshift over the laser path are sensitive to amplitude noise, atmospheric intensity modulation effects can produce noise in the range measurements. Such intensity modulation noise may be minimized by careful design of the optical and electronic components of the tracker. The choice of transmitter, retroreflector and receiver aperture sizes is important (see Section 3.5.1). Probably even more important is the design of the AGC circuitry. This should be capable of minimizing the effect of fading which has been found to have a frequency spectrum from dc to several hundred Hertz. The capabilities of the AGC system also have bearing on the problem of distinguishing deep fades from loss-of-track. The automatic reacquisition system must be able to discriminate between these two conditions.

To summarize this discussion the following observations are made:

1) With respect to angular position measurements, atmospheric quivering will introduce significant noise. While this effect cannot as yet be predicted quantitatively with precision, a method for approximating its correlation function has been presented based on the available theoretical and experimental results.

2) So far as can be determined from the literature on the subject, range noise due to phase shift in the atmosphere will be negligibly small at the planned modulation frequencies.

3) Range noise is more likely to result from sensitivity of the phase measuring circuitry to intensity modulation of the laser beam by the atmosphere. The extent of this effect depends on the parameters of many systems such as the AGC circuit and the various pre- and post-detection filters used in generating phase error signals. The fact that the range measurement is derived from three different phase measurements each of which is subject to noise further complicates any noise analysis. To develop a quantitative model of this noise would require detailed information on the characteristics of the range measuring circuitry. This information is not available for use in connection with this program. The best that can be done is to design the range data processing algorithms on the basis of reasonable assumptions concerning and approximations to the range noise model.

4) As a final general comment, it is evident that a great deal more experimental work of a carefully controlled nature needs to be done, before the dependence of turbulence noise on the measurable state of the atmosphere and on laser system parameters can be established in an accurate, quantitative fashion.

For purposes of the present project the information contained in this report will be utilized as fully as possible in designing the computer programs for control, smoothing and estimation. Actual field testing of the resulting laser tracking system may well reveal a need for modification or refinement of certain system parameters to optimize noise rejection. Since the data processing is being done digitally such changes in parameters can be made very easily at the software level.

REFERENCES

1. Beckmann, P., "Signal Degradation in Laser Beams Propagated Through a Turbulent Atmosphere," Radio Science Journal of Research NBS/USNC-URSI, vol. 69D, pp. 629-640; 1965.

2. Hodara, H., "Laser Wave Propagation Through the Atmosphere," Proc. IEEE, vol. 54, pp. 368-375; 1966.

3. Tatarski, V., Wave Propagation in a Turbulent Medium, McGraw-Hill; 1961.

4. Bendat, J., Principles and Applications of Random Noise Theory, Wiley, p. 201; 1958.

5. Kurtz, R. and J. Hayes, "Experimental Measurement of Optical Angular Deviation Caused by Atmospheric Turbulence and Refraction," TN-D-3439, NASA, Washington, D.C.; May 1966.

6. Buck, A., "Effects of the Atmosphere on Laser Beam Propagation," Applied Optics, vol. 6, pp. 703-708; 1967.

7. Subramanian, M. and J. Collinson, "Comments on the Nature of Atmospheric Modulation by Laser Beams," IEEE Conference on Laser Engineering and Applications, Washington, D.C.; June 1967.

8. Gilmartin, T. and R. Horning, "Spectral Characteristics of Intensity Fluctuations on a Laser Beam Propagating in a Desert Atmosphere," IEEE Conference on Laser Engineering and Applications, Washington, D.C.; June 1967