MC 105

project
mercury

GODDARD PROCESSING PROGRAMS

prepared for
National Aeronautics and Space Administration
Contract No. NAS 1-430

1 august 1961

International Business Machines Corporation

in association with

WESTERN ELECTRIC COMPANY, INC.

LIST OF EFFECTIVE PAGES

* The asterisk indicates pages changed, added, or deleted by the current change.

CASE FILE COPY

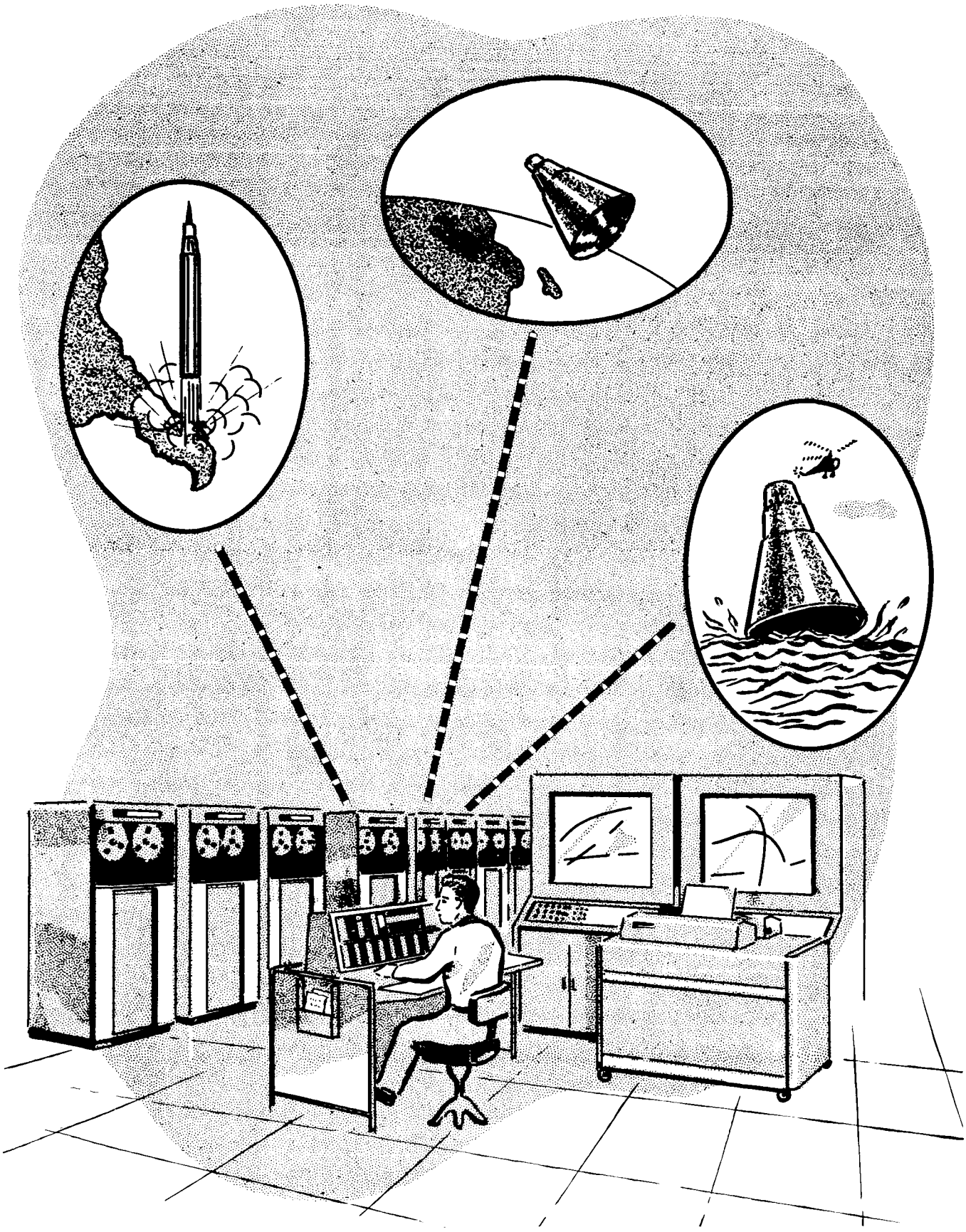


TABLE OF CONTENTS

Page

Section 1. INTRODUCTION

1.1 Goddard Processing 1-1

1.2 Scope of The Manual 1-6

Section 2. INPUT PROCESSORS

2.1 Low-Speed Teletype Input Program (I0TTIN) 2-1

2.2 Manual Insertion Program (I0MANI) 2-23

2.3 B-GE High-Speed Input Program (I0HSGB) 2-35

2.4 IP-7090 High-Speed Input Program (I0HS09) 2-47

2.5 Summary 2-69

Section 3. LAUNCH/ABORT PROCESSORS

3.1 Program System Operation During Launch 3-1

3.1.1 DCC Line 1 - Selected 3-5

3.1.2 DCC Line 2 - Selected 3-5

3.2 Program System Operation During Hold 3-6

3.3 Program System Operation During Abort 3-6

3.4 B-GE Data Conversion Program (CCGEB1) 3-9

3.5 IP 7090 Coordinate Transformation Program (CC7091) 3-13

3.6 Raw Radar Program (CCRAWR) 3-17

TABLE OF CONTENTS (Cont'd)

Page

3.7 Strip Chart Calculation Programs (CCSTIP, CCSTGE, and CCST16) 3-31

3.8 Launch Phase Main Calculation Program (CCMAIN) 3-41

3.9 Capsule Impact Prediction Calculation (EMFIT) 3-127

3.10 Computation of Elliptic Orbit Program (C9RVTH) 3-139

3.11 Computation of Circular Orbit Program (CHNOND) 3-149

3.12 Retrofire Time Correction Program (C9DTRF) 3-157

3.13 Position and Velocity at End of Retrofire Program (R6BOTH). 3-163

3.14 Compute Latitude and Longitude Program (A3MSCP) 3-167

3.15 Inertial Position Vector Conversion Program (MMAQA) 3-175

3.16 Print On-Line Message Routine (PSIP) 3-183

3.17 Hold Phase Main Calculation Program (CCHOLD) 3-187

3.18 Missing Data Computation Program (CCMISS) 3-193

3.19 Missing Data During Hold Program (CCHOMI) 3-201

3.20 Low Level Abort Program (CCRTYL) 3-203

3.21 Medium Level Abort Program (CCMEAB) 3-209

3.22 High Abort Computation Program (CCABRT) 3-217

Section 4. ORBIT AND RE-ENTRY PROCESSORS

4.1 Editing Program (E0LED1) 4-1

4.2 Numerical Integration Control Program (N0CPNI) 4-19

4.3 Set Up Function Table Program (N1SUFT) 4-33

4.4 Extrapolation and Correction Equation Derivation Program (N2EXCR) 4-37

TABLE OF CONTENTS (Cont'd)

		Page
4.5	Second Derivative Calculation Control Program (N3SDCL)	4-43
4.6	Capsule Drag Formulation Program (N4DRAG)	4-47
4.7	Atmospheric Density Computation Program (N4PDEN)	4-53
4.8	Coefficient of Drag Calculation (N4COEF)	4-57
4.9	Numerical Integration Table Compiler (N5OTBL)	4-63
4.10	Earth's Effect on Capsule Calculation (N6OPEP)	4-67
4.11	Variable Step Integration Equations Program (N7VARS)	4-71
4.12	Differential Correction	4-77
4.13	Differential Correction Control Program (D0DIFC)	4-79
4.14	Calculate Functions of $\bar{r}$ and $\bar{v}$ Program (D4CFRV)	4-99
4.15	Partial Coefficient Calculation Program (D1PCDC)	4-103
4.16	Development of Six Numerical Integration Output Tables Program (R2AINT)	4-109
4.17	Numerical Differential Correction Program (R1NDCP)	4-115
4.18	Weight Lookup Table Program (D3WTLU)	4-121
4.19	Weighting The Equations of Condition Program (D2WEQT)	4-127
4.20	Choose Desired Radar Message Program (D3CDRM)	4-135
4.21	Set Up Normal Equations Program (D2SUNE)	4-141
4.22	Solve Normal Equations Program (D2SLNE)	4-147
4.23	Calculate Standard Error Fit Program (D2CSFE)	4-153
4.24	Station Sum of Squares Over M Program (D2SSMT)	4-157
4.25	Retrofire and Re-entry Calculations Program (R5RARF)	4-161

TABLE OF CONTENTS (Cont'd)

Page

4.26 Delta Time For Retrofire Program (C9DTRF) 4-171

4.27 Compute $\overline{r}$ and $\overline{v}$ at Burnout Program (R6BOTH) 4-177

4.28 Elapsed and Incremented Capsule Time Program (O5RARF) 4-181

4.29 Acquisition Data Generation Program (A0STAD) 4-187

4.30 Calculation of High Speed Output Data Program (O5ORMC) . 4-203

Section 5. OUTPUT PROCESSORS

5.1 Low-Speed Output Program (O0LSTY) 5-1

5.2 High-Speed Output Programs (O0LANA and O0ORRE) 5-9

5.3 Binary Coded Decimal Conversion Program (BCD) 5-53

5.4 Packing Routine (PACK) 5-57

5.5 Time Measure to BCD Degrees Conversion Program (DGHR) 5-61

5.6 BCD to Modified BCD Conversion Program (MBCD) 5-65

5.7 Data Preparation for Goddard Plotboards (PLOT) 5-67

LIST OF ILLUSTRATIONS

Figure		Page
2-1	I0TTIN Program Flow Chart	2-8/2-21
2-2	Raw Radar Data From Remote Radar Sites to Goddard (TTY) Message Format	2-22
2-3	I0MANI Program Flow Chart	2-28/2-33
2-4	I0HSGB Program Flow Chart	2-41/2-43
2-5	B-GE Data, Cape Canaveral to Goddard Message Format	2-44/2-45
2-6	I0HS09 Program Flow Chart	2-57/2-66
2-7	IP 7090 Data, Cape Canaveral to Goddard, Message Format	2-67/2-68
3-1	Launch Program Organization (From Lift-off to Hold)	3-3
3-2	Launch Program Organization (During Hold)	3-4
3-3	Abort During Launch Phase Organization	3-7
3-4	CCGEB1 Program Flow Chart	3-12
3-5	CC7091 Program Flow Chart	3-16
3-6	CCRAWR Program Flow Chart	3-26/3-29
3-7	Strip Chart Processors Flow Chart	3-38/3-39
3-8	CCMAIN Program Flow Chart	3-125/3-126
3-9	CCMAIN Block	3-44
3-10	CCMLB Block	3-45
3-11	CCMLC Block	3-47

LIST OF ILLUSTRATIONS (Cont'd)

Figure		Page
3-12	PREVV Block	3-48
3-13	CCMLD Block	3-49
3-14	CCMLF Block	3-52
3-15	CCMLG Block	3-53
3-16	CMMLG Block	3-65/3-71
3-17	CCMLH Block	3-72
3-18	CCMLJ Block	3-74
3-19	CMMLJ Block	3-77
3-20	CCMLK Block	3-79
3-21	CCMLL Block	3-80
3-22	CCMLN Block	3-81/3-82
3-23	CCMLO Block	3-85
3-24	CMMLO Block	3-87
3-25	CCMLP Block	3-88
3-26	CCMLR Block	3-90
3-27	CCLOK Block	3-94/3-95
3-28	CCMLS Block	3-97
3-29	CCMLT Block	3-98
3-30	CCMLU Block	3-100
3-31	CCMLV Block	3-101
3-32	CCMLY Block	3-102
3-33	CVEXT Flow Chart	3-104

LIST OF ILLUSTRATIONS (Cont'd)

Figure		Page
3-34	EMFIT Program Flow Chart	3-134/3-137
3-35	EMFIT Diagram	3-138
3-36	C9RVTH Program Flow Chart	3-145/3-148
3-37	CHNOND Program Flow Chart	3-155/3-156
3-38	C9DTRF Program Flow Chart	3-160/3-161
3-39	R6BOTH Program Flow Chart	3-166
3-40	A3MSCP Program Flow Chart	3-172
3-41	MMAQA Program Flow Chart	3-181/3-182
3-42	PSIP Program Flow Chart	3-185
3-43	CCHOLD Program Flow Chart	3-189/3-191
3-44	CCMISS Program Flow Chart	3-194/3-199
3-45	CCHOMI Program Flow Chart	3-200
3-46	CCRTYL-CCTRMI Program Flow Chart	3-207/3-208
3-47	CCMEAB Program Flow Chart	3-214/3-216
3-48	CCABRT Program Flow Chart	3-221/3-223
4-1	Orbit and Re-Entry Phase General Block Diagram	4-2
4-2	E0LED1 Program Flow Chart	4-10/4-18
4-3	N0CPNI Program Flow Chart	4-31
4-4	N1SUFT Program Flow Chart	4-36
4-5	N2EXCR Program Flow Chart	4-41
4-6	N3SDCL Program Flow Chart	4-46
4-7	N4DRAG Program Flow Chart	4-52

LIST OF ILLUSTRATIONS (Cont'd)

Figure		Page
4-8	N4PDEN Program Flow Chart	4-56
4-9	N4COEF Program Flow Chart	4-60/4-61
4-10	N5OTBL Program Flow Chart	4-66
4-11	N6OPEP Program Flow Chart	4-70
4-12	N7VARS Program Flow Chart	4-75
4-13	D0DIFC Program Flow Chart	4-95/4-98
4-14	D4CFRV Program Flow Chart	4-102
4-15	D1PCDC Program Flow Chart	4-108
4-16	R2AINT Program Flow Chart	4-114
4-17	R1NDCP Program Flow Chart	4-119/4-120
4-18	D3WTLU Program Flow Chart.	4-124/4-125
4-19	D2WEQT Program Flow Chart.	4-132/4-133
4-20	D3CDRM Program Flow Chart.	4-140
4-21	D2SUNE Program Flow Chart	4-146
4-22	D2SLNE Program General Flow Chart	4-149
4-23	D2SLNE Program Detailed Flow Chart	4-150/4-152
4-24	D2CSFE Program Flow Chart	4-155
4-25	D2SSMT Program Flow Chart	4-159
4-26	R5RARF Program Flow Chart	4-169
4-27	C9DTRF Program Flow Chart	4-174/4-175
4-28	R6BOTH Program Flow Chart	4-179
4-29	O5RARF Program Flow Chart	4-185

LIST OF ILLUSTRATIONS (Cont'd)

Figure		Page
4-30	A0STAD Program Flow Chart	4-196/4-201
4-31	O50RMC Program Flow Chart.	4-209/4-211
5-1	O0LSTY Program Flow Chart	5-6/5-7
5-2	O0LANA Program Flow Chart	5-26/5-38
5-3	O0ORRE Program Flow Chart	5-39/5-51
5-4	Display Format of TMOLAB Contents	5-15/5-19
5-5	BCD Routine Flow Chart.	5-55/5-56
5-6	PACK Routine Flow Chart	5-59/5-60
5-7	DGHR Routine Flow Chart	5-63
5-8	PLOT Routine Flow Chart	5-69

LIST OF TABLES

Table		Page
1-1	Input Data to Goddard (via the IP 7090)	1-2
1-2	Main Goddard Processors	1-8
2-1	I0MANI Input Messages Format	2-25
2-2	Mercury Control Center (Telemetry Event Transmitting Buffer) to High-Speed Retransmitters, Message Format	2-51/2-52
3-1	Launch/Abort Main Processors	3-2
3-2	TCCOUT Table Format	3-105/3-106
3-3	TCCONS Table Format	3-107/3-112
3-4	TCMANR Table Format	3-113/3-121
3-5	C9RVTH Input and Output Data Format	3-140
4-1	TNFUNK Table Format	4-25/4-26
4-2	TEMNO Table Format	4-27/4-30
4-3	Station Characteristics Block Contents	4-81
4-4	TDRANV Table Format	4-84
4-5	TDPRNT Table Format	4-85/4-86
4-6	Times of Input Messages In Testing D3CDRM	4-139
4-7	TMORMC Table Format	4-183
4-8	A0STAD 16-Word Output Block Format	4-188/4-189
4-9	TMORMC Output Table Format	4-204/4-205
5-1	Station Identification Tables	5-4

LIST OF TABLES (Cont'd)

Table		Page
5-2	TMLANA Table Format	5-10/5-11
5-3	TMORRE Table Format	5-12/5-13
5-4	Characteristics of Display Media	5-20/5-21
5-5	Displayed Quantities and Their Format	5-22

Section 1
INTRODUCTION

The computer complex at Goddard Space Flight Center at Greenbelt, Maryland, has been established to provide reliable, high-speed computation and communications for space explorations directed by the National Aeronautics and Space Administration (NASA). This complex serves as the coordinating center for space flight communications.

Particular attention has been given the man-in-space program—Project Mercury—in designing the computing system complex now in operation at Goddard.

Two IBM 7090 computing systems, the heart of this complex, operate in parallel during a Project Mercury mission. Outputs from only one of these computers is transmitted at any one time to the Mercury Control Center at Cape Canaveral and to the worldwide tracking sites during a mission; however, the duplexed operation ensures computational reliability and accuracy.

The Goddard computers accept:

a) C-band and S-band radar data from the tracking sites

b) Telemetry data

c) Outputs from the Burroughs-General Electric (B-GE) complex

d) Outputs from the IP 7090 Impact Predictor computer

After such data has been accepted, the computers perform the calculations necessary to convert these inputs into the orbital and trajectory parameters used for monitoring the launch/abort, orbital and re-entry phases of the Mercury mission.

Outputs from the Goddard computers include:

a) Display data to the Mercury Control Center at Cape Canaveral

b) Acquisition data to the radar tracking sites

c) Special data to the Bermuda Control Center

d) Pertinent information to displays and personnel at Goddard.

TABLE 1-1. INPUT DATA TO GODDARD (VIA THE IP 7090)

Type of Information	Source
Two-inch lift-off	Launch Pad
Booster engine cut-off command	B-GE
Staging	Vehicle Telemetry
Tower release from capsule	Telemetry
Tower rockets fired from capsule	Telemetry
Sustainer engine cut-off command	B-GE
Posigrades 1, 2 and 3 fired	Capsule Telemetry
Capsule separation	Capsule Telemetry
Abort command (from ground)	Mercury Control Center
Abort sequence initiated	Capsule Telemetry
Retrofire command	Mercury Control Center
Data source displayed at the Mercury Control Center	Mercury Control Center
Elapsed time as read from the capsule clock	Capsule Telemetry
Retrofire capsule clock reading	Capsule Telemetry
Position and velocity data	B-GE/IP-7090
GMT of lift-off	Mercury Control Center
Capsule attitude at time of retrofire	Capsule Telemetry

1.1 GODDARD PROCESSING

Goddard input and output processing functions relative to the launch, abort, orbit and re-entry phases of a Mercury mission are discussed in the following paragraphs.

1.1.1 Launch

The main function of the Goddard computers during the launch phase is to make the GO, NO-GO recommendation to the Mercury Control Center. Other functions of Goddard are to transmit to the Control Center predicted impact points for impending abort situations and provide acquisition to Bermuda. In the event of an abort, the Goddard computers determine times to fire the retrorockets for the capsule to land in one of several designated recovery areas. When retrorockets are fired, the computers calculate the return trajectory and the resulting impact point.

Launch Inputs: During the launch phase most of the data entering the Goddard computers is transmitted from the Mercury Control Center (MCC) via high-speed lines. This input data to Goddard (sent via the IP 7090 at the MCC) and the sources at which it originates are presented in Table 1-1. The only other data which enters the Goddard computers during launch is radar data from the Bermuda site.

Launch Outputs: Data received at Goddard during the launch phase from either the B-GE system or the AN/FPS-16 system is processed and sent to the Mercury Control Center for display on plotboards 1, 2, 3 and 4. Launch phase displays on plotboards 1, 2, 3 and 4 are discussed below.

a) Plotboard 1—The flight path angle versus the ratio of inertial velocity to required velocity (γ vs V/V_r) is plotted on an appropriate overlay.

b) Plotboard 2—This plotboard carries two plots during the launch period. On the lower portion of the board the capsule altitude (H) versus the downrange distance from the pad (S) is displayed. On the upper part of the overlay, the crossrange deviation (Y) versus the downrange distance from the launch pad is displayed. All quantities are measured in nautical miles.

c) Plotboard 3—A variety of parameters is displayed on plotboard 3 during launch. However, during this phase these quantities are all furnished to the plotboard directly from the Burroughs special-purpose computer.

d) Plotboard 4—Certain impact point computations made during launch, and in the event of abort situations, are displayed on this plotboard. The overlay is a map of the Atlantic Ocean area, showing land masses and recovery areas.

1.1.2 Orbit

The Goddard IBM 7090 computers determine the orbital characteristics and predict the instantaneous position of the Mercury capsule at any time. Using refined orbital parameters the computers continually update the orbit lifetime. In addition, the computers provide accurate acquisition data to all tracking sites during this period.

Orbit Inputs: During orbital flight, radar and timing information is entered into the Goddard computers.

a) Radar Data from Sites—During each pass over a radar site, capsule position information is sent to Goddard every six seconds (± two milliseconds). This information is encoded at the site by special equipment and transmitted to Goddard.

   At those sites having both Verlort and the AN/FPS-16 radars, only the AN/FPS-16 radar is processed. If the data AN/FPS-16 is not available, Verlort data is processed.

b) Capsule Time Data—A reading of the capsule elapsed time clock and of the retrofire-time clock is transmitted to Goddard via teletype summary after each pass over a tracking station. The message reads: "At time (hr min sec) GMT, capsule elapsed time clock read (hr min sec) and retrofire setting as (hr min sec)."

Orbit Outputs: During the orbital phase, capsule flight data is transmitted from the radar sites to the Goddard computers, where it is processed and sent to the Mercury Control Center for display on plotboards 1, 2, 3 and 4. Each quantity is sent at a frequency of five messages per minute and consists of ten bits for each coordinate. The plot and overlays for the four plotboards during orbit are:

a) Plotboard 1—The difference between the radius of the capsule from the center of the earth and the average radius of the earth versus the inertial velocity ($r-\bar{R}$ vs V) is displayed on this board.

b) Plotboard 2—During orbital flight, two plots appear on this plotboard. On the lower part of the plotboard the altitude of the capsule above an oblate earth as a function of time (h vs. t) is displayed. On the upper part of the plotboard the difference between the semimajor axis of the

orbit and the average radius of the earth as a function of time ($a-\bar{R}$ vs. t) is displayed.

c) Plotboard 3—Two plots are displayed on plotboard 3. On the lower portion of the board the longitude of perigee as a function of time (λ_p vs. t) is displayed; on the upper portion the eccentricity as a function of time (ϵ vs. t) is displayed.

d) Plotboard 4—Only two items are displayed on plotboard 4 during orbit. These items are the present position of the capsule and an impact point for retrofire in 30 seconds.

1.1.3 Re-Entry

Re-entry begins at the time of manual insertion to the Goddard computers of retrofiring time and the number of retros fired. Refined impact points are computed and furnished for the recovery effort. During re-entry, the Goddard computers continuously furnish the site with acquisition data.

Re-entry Inputs: Since the fact that retrorockets have fired may be noted by any station (including a telemetry site), the time of firing of each rocket is transmitted to Goddard via teletype summary message. These firing times are sent to the nearest second in GMT; if possible, attitude angles at the moment of each firing are also sent. This information is manually inserted into the Goddard computers.

During re-entry, the computers continue to process radar data received from the sites to refine the predicted impact point and to send new acquisition data. The format and means of entry into the computer remain the same as those during the orbital phase.

Re-entry Outputs: During re-entry, outputs from the Goddard computers consist of acquisition data to the sites and the following data is sent to the Mercury Control Center for display on plotboards 1, 2, 3, and 4.

a) Time of retrofire

b) Latitude and longitude of impact

The plots and overlays for the four plotboards during the re-entry phase are:

Plotboard 1 is the same as for the orbital phase.

Plotboard 2 is the same as for the orbital phase.

Plotboard 3 is not used during re-entry.

Plotboard 4 is the same as for the orbital phase.

1.2 SCOPE OF THE MANUAL

The Goddard Computing and Communications Center uses many processors and their subprograms in performing computations during a Project Mercury mission (see Table 1-2). The main processors are:

Input Processors

Launch/Abort Processors

Orbit Processors

Re-entry Processors

Output Processors

These processor programs and their subprograms are discussed in detail in Sections 2 through 6 of this document. Writeups for the library subroutines which are used with these processors are presented in the MC 107, External System Programs.

TABLE 1-2. MAIN GODDARD PROCESSORS

Program	Launch	Orbit	Re-entry
I0TTIN		X	X
I0MANI	X	X	X
I0HSGB	X	X	X
I0HS09	X		
CC7091	X		
CCGEB1	X		
CCFP16	X		
CCMAIN	X		
CCSTGE	X		
CCSTIP	X		
CCST16	X		
E0LED1		X	X
N0CPNI	X	X	X
D0DIFC		X	X
R5RARF		X	X
A0STAD		X	X
O50RMC		X	X
O0LSTY		X	X
O0LANA	X		
O0ORRE		X	X

Section 2
INPUT PROCESSORS

During a Mercury mission pertinent information regarding the functioning of the launch vehicle prior to capsule separation, during orbit, and during reentry is continuously sent to and received by the Goddard Computing and Communications Center. This data accepted by Goddard is transmitted from the IP 7090 computer located at the Mercury Control Center, the B-GE complex and from the worldwide tracking network.

Information concerning the mission must be processed and arranged for input into the Goddard 7090 computers according to priority. Four main input processor programs have been written for this purpose:

I0TTIN—Low-Speed Teletype Input

I0MANI—Manual Input

I0HSGB—High-Speed B-GE Input

I0H709—High-Speed IP 7090 Input

These processors and their subprograms are described in this section.

2.1 LOW-SPEED TELETYPE INPUT PROGRAM (I0TTIN)

The I0TTIN program processes capsule position radar data received by Goddard, via the 16 input teletype lines from the remote radar sites. The flow chart for the I0TTIN program is shown in Figure 2-1.

2.1.1 Input Requirements

Each time the Mercury capsule passes over a radar site capsule positional data is sent to the Goddard Computing and Communications Center every six seconds. Time is specified in Greenwich Mean Time (GMT) in hours, minutes and seconds. Negative elevation angles of observations below the horizontal plane appear in complemented form ($-10^{\circ} = 350^{\circ}$, $-8^{\circ} = 352^{\circ}$).

Two types of radars are used by the Mercury tracking sites: Verlort and AN/FPS-16. The Verlort radar has the greater long-range capabilities; the AN/FPS-16 is more accurate at close range. Every radar site except Eglin Air

Field is equipped with either one or both of these radars. Eglin Air Field has two radars, an AN/FPS-16 and an AN/MPQ-31 (instead of Verlort). The digital output from the AN/MPQ-31 is encoded into the Verlort radar message format. Because the AN/MPQ-31 radar type character is identical to that of the Verlort radar, I0TTIN assumes that the data it is receiving is from a Verlort radar.

For a site having both types of radars, the normal tracking and data transmission procedure is as follows:

a) The Verlort normally "locks on" the capsule first and readings from it are selected by a manual switch for transmission over the teletype line.

b) As the capsule comes into AN/FPS-16 range the switch is turned to "Select and Transmit the AN/FPS-16 Data;" Verlort continues tracking the capsule, meanwhile.

c) When the capsule passes out of AN/FPS-16 range the data transmitter is switched back to transmit the Verlort data until the capsule passes out of range of the site.

d) Data from both radars is recorded on paper tape at the site.

The 16 input teletype lines to Goddard present teletype characters to the Data Communications Channel (DCC) at the rate of six characters per second. The DCC accepts one five-bit character at a time and places it in the low-order positions of one IBM 7090 word (e.g., teletype bit 1 to 7090 bit 31, teletype bit 5 to 7090 bit 35) according to the subchannel address register of that line and then increments the subchannel address register by one. When six characters have been inserted into six consecutive locations of storage in the computer, the DCC resets the subchannel address register and causes a trap in the program. The Monitor program then picks up these six five-bit characters each with a leading zero and packs them into one 36-bit word and stores this word in a ten-word (60 teletype characters) block set aside for this particular line. When this ten-word block has been filled by Monitor, the particular input subchannel number (1-16) is placed in the D(AC) and control is given to I0TTIN. Location TMTTIN contains the BES address of the ten-word block and a tag of 1.

2.1.2 Method

I0TTIN processes the data from the 16 teletype lines in any sequence, but it must finish processing one 60-character block of data before starting another. The program maintains a log for each line with respect to:

a) The presence of a station's transmission on that line.

b) If a station is transmitting, its external station number and the particular radar type last transmitted; if no station is transmitting the external station number is zero.

c) Whether or not there were four teletype characters following the radar message found on the last pass. These four characters are checked for the end-of-transmission code.

The I0TTIN program accepts positional data from both radars at a two-radar site in any sequence, alternating or otherwise, within one transmission sequence, i.e., one start-of-transmission code, a series of radar messages, and one end-of-transmission code. The message format of the raw radar data from remote sites to Goddard is shown in Figure 2-2. This version of I0TTIN automatically rejects any subsequent transmission of data from any site and no request is made for retransmission of data. Once data is received on a particular line, it should continue on that line to ensure acceptance by I0TTIN. If another station starts transmission over a line abandoned by a station, transmission is automatically ended for the station formerly transmitting.

I0TTIN initially scans the ten-word input data block for the beginning of transmission characters, JJ↓<≡↓. If this code is found, the program searches the data following the start code for a radar message. If the start code is not found, the program scans the entire block for a radar message.

If a radar message is found, the computer words containing the remainder of the teletype data in the ten-word block are automatically moved to the top of the block and the decrement of TMTTIN + i for line i is set to the number of words necessary to fill the block again just prior to exiting from the program. If no data is found in the ten-word block, the last word of the block is moved to the first word and the count in the decrement of TMTTIN + i is set to 9.

The criteria used in identifying radar messages are checked in the following order:

a) The first three characters of the message must be carriage return, line feed and figure shift (<≡↑).

b) The 34th character must be a slash (/).

c) There must be more than one message in the station's transmission, i.e., more than one radar message must exist between the beginning of transmission and the end of transmission.

d) Characters 4 through 33 must be numeric.

e) Characters 5 and 6, the station identification code, must be a valid station code.

f) The radar type character must indicate that the radar is either AN/FPS-16 or Verlort.

g) The combination of the external station number and the radar type must convert to a valid internal station number. This is done to separate each radar's observational readings for computation purposes.

h) The internal station's message block must be in the "data acceptable" mode, in which the sign of the first word of the block must be minus.

i) The validity character in the message must indicate a valid message.

j) The elevation angle must lie between three and 90 degrees.

k) The internal station message block must not be full. No more than 50 radar messages can be stored in the block.

The end of transmission is recognized by the present version of I0TTIN in the following ways:

a) The end of transmission code ∬ ↑H↓ (blank, figures, H, letters) is recognized as the four characters immediately following the 34th character (/) of a radar message.

b) The last message indicator, character 4 in the radar message, is recognized as equal to seven.

c) A change in station identification code on a particular line is also recognized as the end of transmission of the station which was previously using the line.

The end of transmission must be recognized so the data may be submitted to the Edit and Computation programs; otherwise, the data is not processed.

The Monitor program has enabled I0TTIN to maintain an elapsed time check on radar since the message which initiates transmission of radar data never sends a recognizable end-of-transmission signal.

If the computer has not received an end-of-transmission signal from a particular station within eight minutes of the receipt of the first radar message, Monitor returns to I0TTIN at location I0STPR, with that station's internal station number in location MCISTN. I0TTIN then deletes this internal station number from its log tables and gives the normal indications to Monitor that this station has ended transmission (if it was present in the log tables otherwise no indication is made). This permits the computations programs to use all the data received by the computer.

The Differential Correction programs discards stations from its computations if the observations in the particular station's message blocks are 72 or more minutes old.

When the input program receives data from a radar site whose block is not initialized, i.e., put in a "data accept" mode, the station message block is checked to determine whether or not the block can be initialized.

Each block can be initialized to accept data if both the message count in the block and the pseudo-message count are zero. The pseudo-message count is kept by a section of the half-second processor, which can delete in effect, a station from the differential correction table (TMSTMS) by setting the message count in a block zero and storing the count in a pseudo-message count block (TAVRMB).

If the message count is not zero, the time of minimum approach is checked to see if it is contained in the current message count, or was left from a previous pass. Some blocks may not have a time of minimum approach determined for a particular pass. When the time of minimum approach is found within the current message count, it is checked against current machine GMT time to see if 73 minutes have elapsed. If it has, the block is reinitialized by setting the sign of the first location minus and setting the message count to zero. The 73 minutes must elapse before a block is reinitialized to allow proper operation in the differential correction check of 72 minutes.

If the time of minimum approach has not been determined or it is not included in the current message count, the time of the first message in the block is compared with the current machine GMT for initialization as mentioned above.

After processing a station, a check is made to determine if it is at a co-located radar site. If it is, the other station message block is checked in the same manner as above for initialization.

When a station or both stations at a co-located site have been checked, the first location of block corresponding to the station being processed is placed in the AC and I0TTIN continues.

At the exit to Monitor, I0TTIN indicates when a radar at a station starts or ends transmission. When I0TTIN recognizes that any valid station has started transmission, its corresponding internal station number is placed in the low-order position of the MQ. Only one internal station can start transmission on a particular line at any one time. The second internal station can start transmission on a particular line at any one time after the initial internal station has started. The second radar at a double radar site has its transmission initiated by the appearance of its radar message on the line after the first radar has transmitted at least one message. No start-of-transmission code is sent with the data from the second radar.

In the case of a double radar site, if both radars are active during a pass, both end transmission simultaneously where the end of transmission is recognized or by the lapse of 8 minutes. If no data is received from one of the two radars, its station message block remains in the "data acceptable" mode.

In the case of a double radar site, also, when one or two internal stations end transmission, the internal station numbers are placed in the low-order positions of the D(AC) and/or the A(AC) at the exit of I0TTIN. To ensure no overlap of the starting and ending of transmission, the transmission of one radar message from a station is not processed. This eliminates the possibility of three internal stations transmitting at once.

If launch occurs so time passes through 2400 hours during the mission, I0TTIN automatically adds 24 hours to each time entry differing by more than 23 hours from the computer time. The GMT maintained by the computer does not reset at midnight after launch; it is reset at the radar sites, however.

Simultaneous data transmission over duplexed lines from Guaymas and Bermuda to Goddard is not handled by I0TTIN. Also, lines switching (route changes) and terminal patching (subchannel changes) are not handled by I0TTIN.

Emergency messages which may interrupt the radar messages cause no interference with the acceptance of subsequent radar data when transmission is resumed. No provision is made at present to indicate the absence of radar data or errors found in the radar data.

The output of I0TTIN, in addition to indicating the starting or ending of transmission in the AC and MQ, consists of the building up of station message blocks for each internal station number. Each block is set to contain 203 locations for 50 radar messages and three preface locations. Output block format is shown below.

Word 1—Internal station identification in the address field is minus (-) for "data acceptable" mode, and plus (+) for "no data acceptable" mode. The sign is set by Monitor.

Word 2—Number of radar messages in this block

Word 3—Location of the time of minimum approach (set by edit)

4(n-1)4—GMT of observation	Radar Messages 1 - 50
4(n-1)5—Range	
4(n-1)6—Azimuth	
4(n-1)7—Elevation	

The output block for the jth internal station is found through the reference table TMLSDB. The locations TMLSDB j - 1 contain the location of the first word in the jth output block.

2.1.3 Usage

a) Calling Sequence—Entrance to IOTTIN requires the input subchannel number to be processed to appear in the decrement of the accumulator. Exit from IOTTIN leaves any internal station numbers which started transmission in the MQ, and leaves those which ended transmission in the accumulator. Accepted radar data is placed in respective station message blocks.

b) Space Required:

Program	413_{10}
Program Constants	250
Common Storage	4052

c) Constants:

K00000	DEC	0
K00001	DEC	1
K00002	DEC	2
K00003	DEC	3
K00009	DEC	9
K00010	DEC	10
K00019	DEC	19
K00060	DEC	60

d) Common Locations:

MCHFSC—half-second clock
TMTTIN—16-location table
AAS01—AAS16160-location table
TMLSDB—19-location table
TTBCD—teletype-to-BCD conversion table

e) Time Required—six to eight milliseconds (average)

2.1.4 Checkout

The IOTTIN program has been operational in the Mercury Program System since May, 1960.

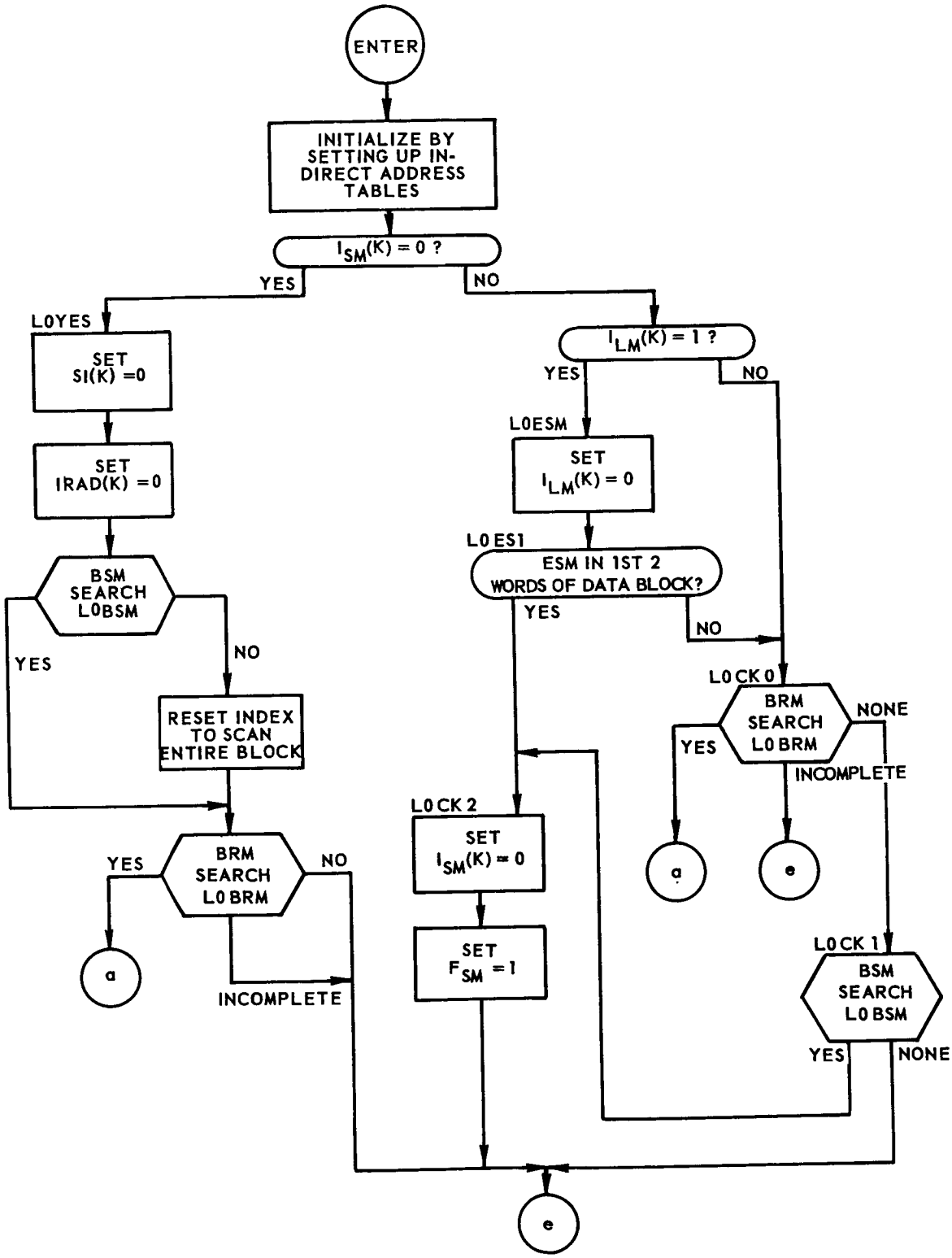


FIGURE 2-1. 10TTIN PROGRAM FLOW CHART *(Sheet 1 of 14)*

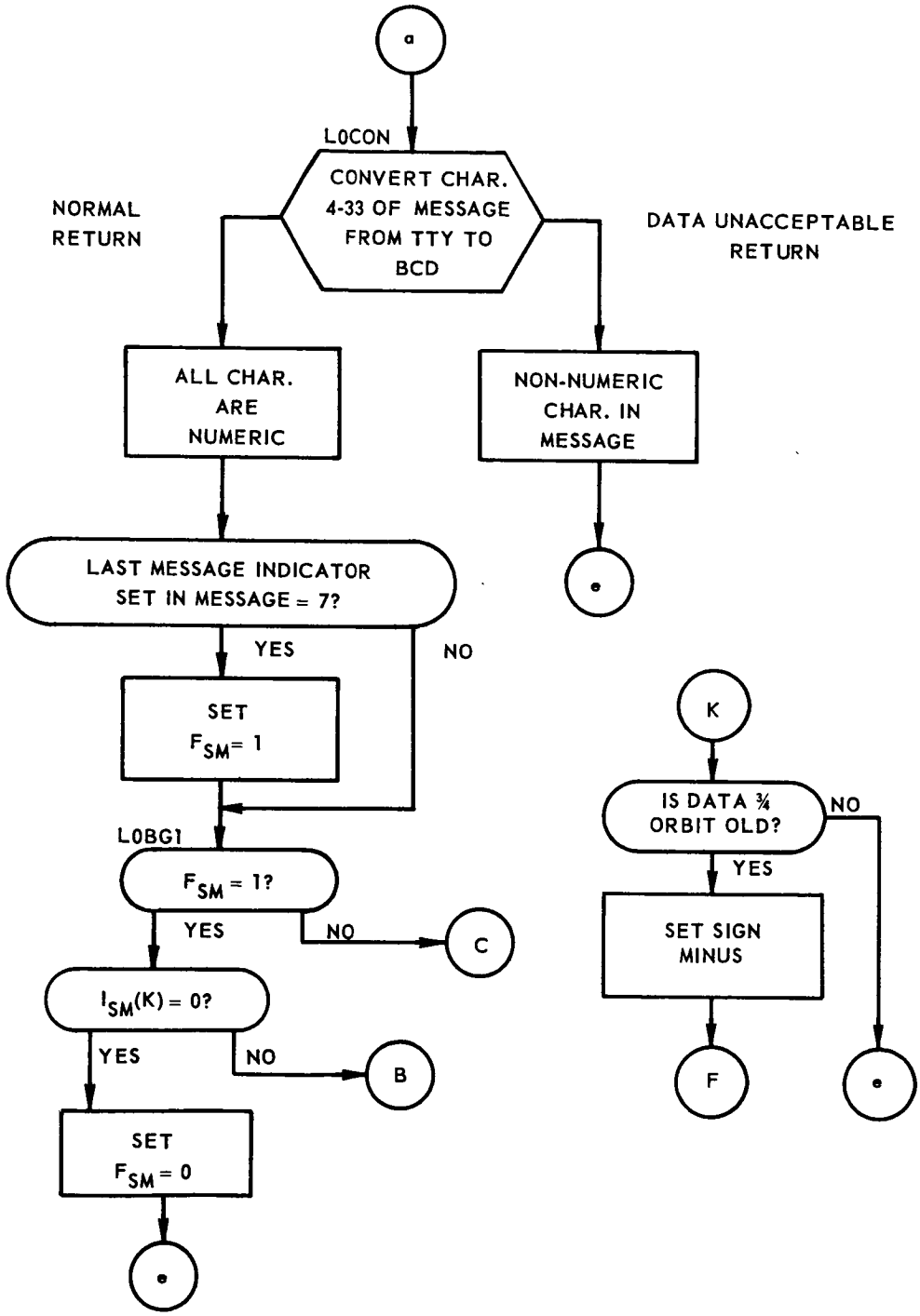


FIGURE 2-1. IOTTIN PROGRAM FLOW CHART *(Sheet 2 of 14)*

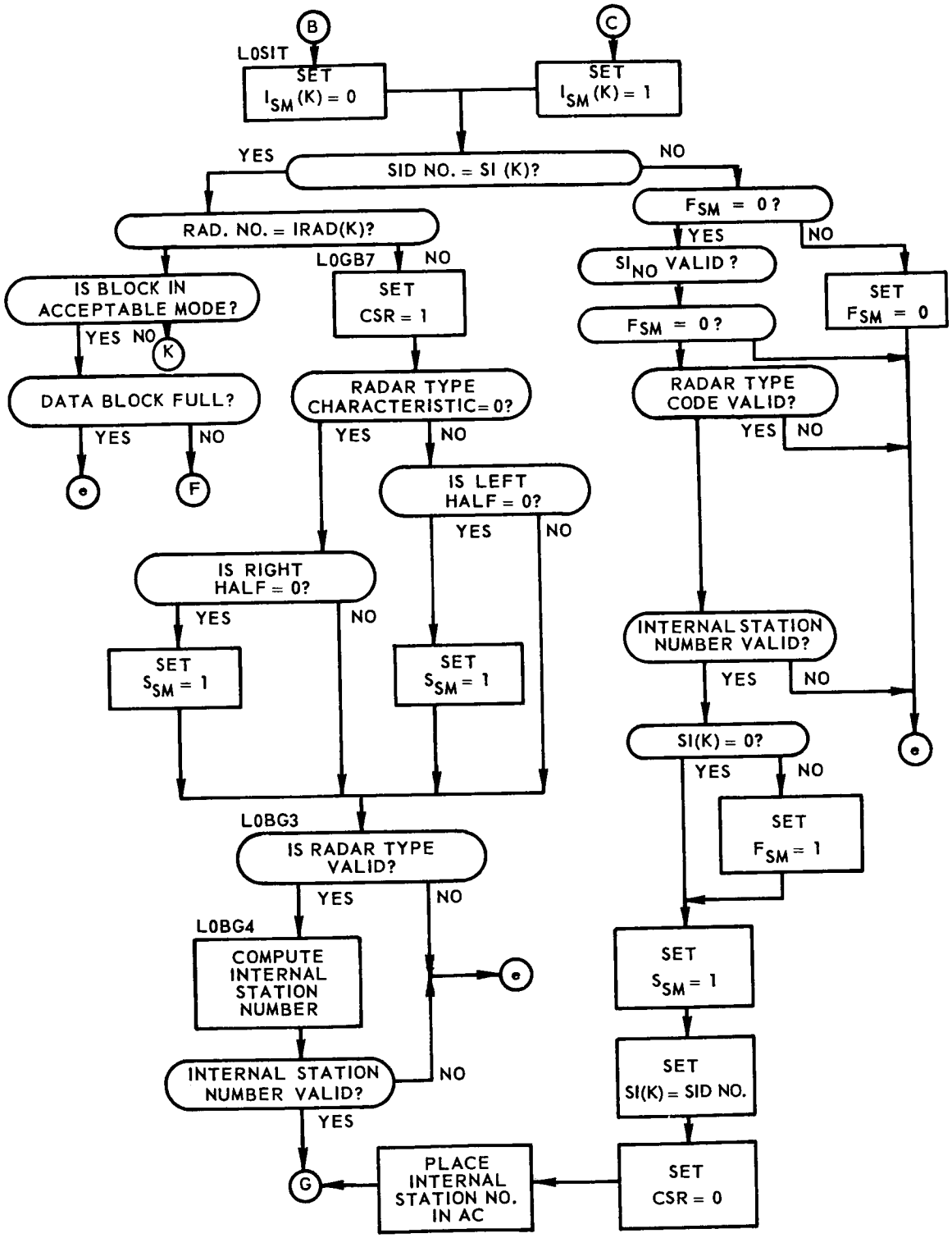


FIGURE 2-1. IOTTIN PROGRAM FLOW CHART *(Sheet 3 of 14)*

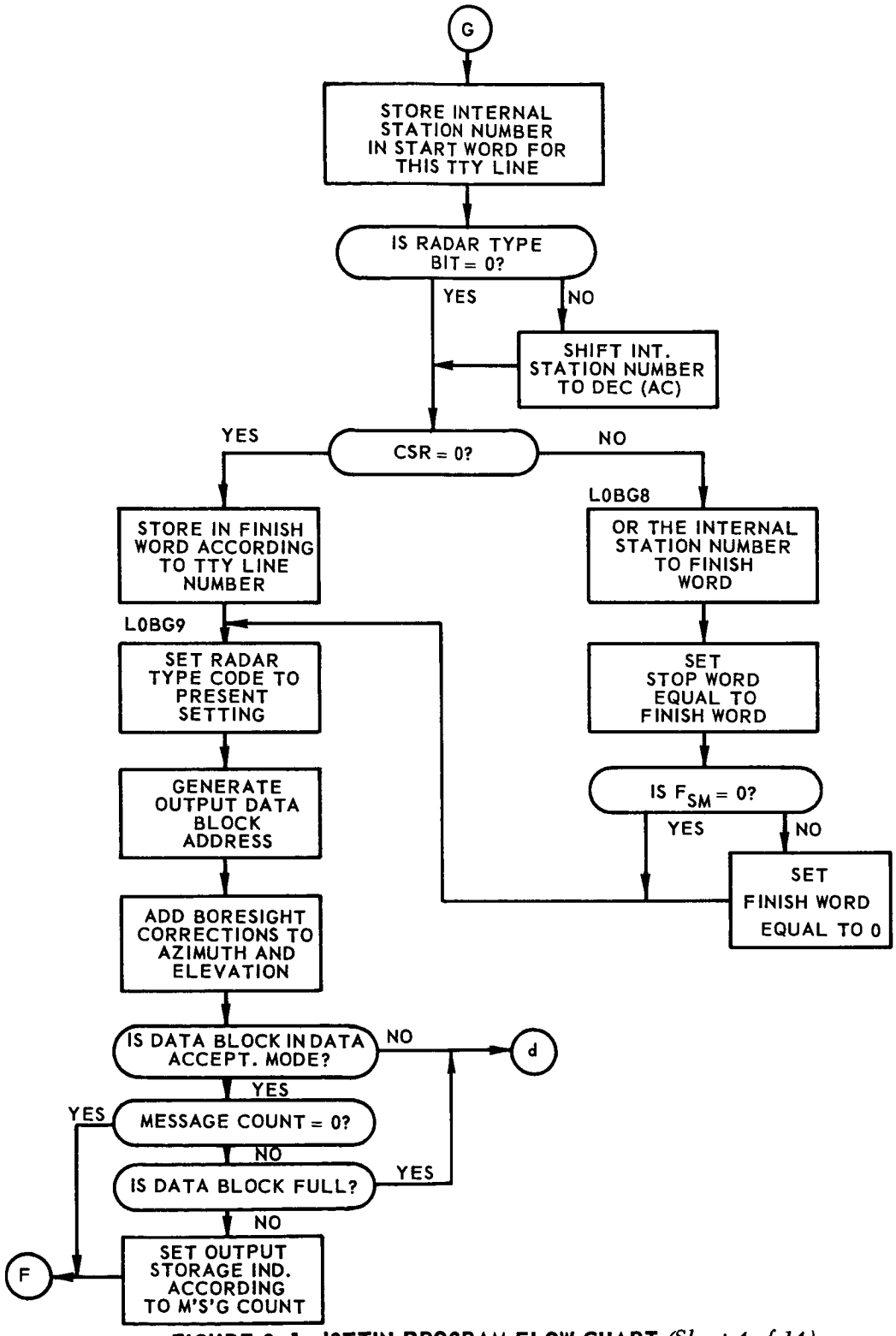


FIGURE 2-1. IOTTIN PROGRAM FLOW CHART *(Sheet 4 of 14)*

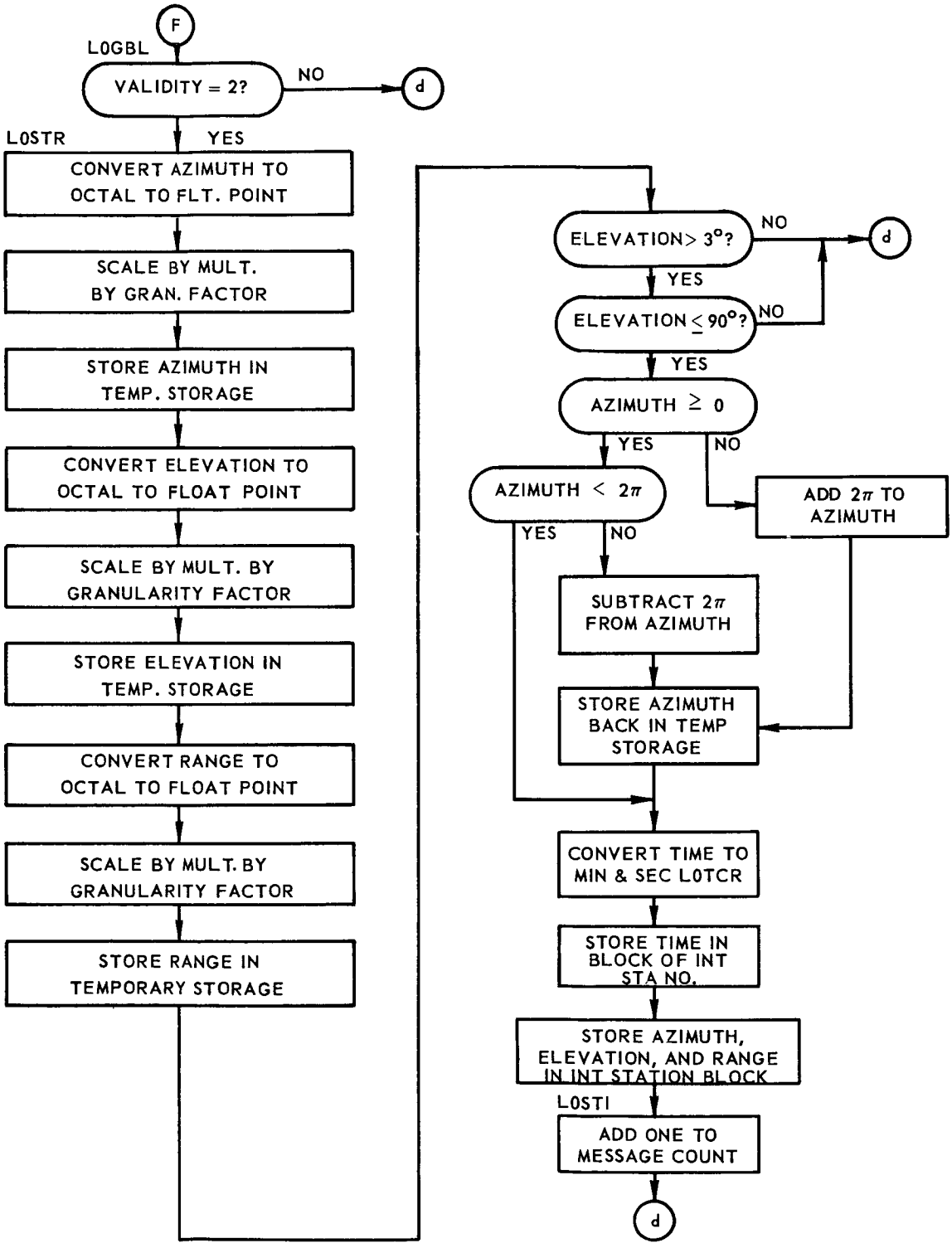


FIGURE 2-1. IOTTIN PROGRAM FLOW CHART *(Sheet 5 of 14)*

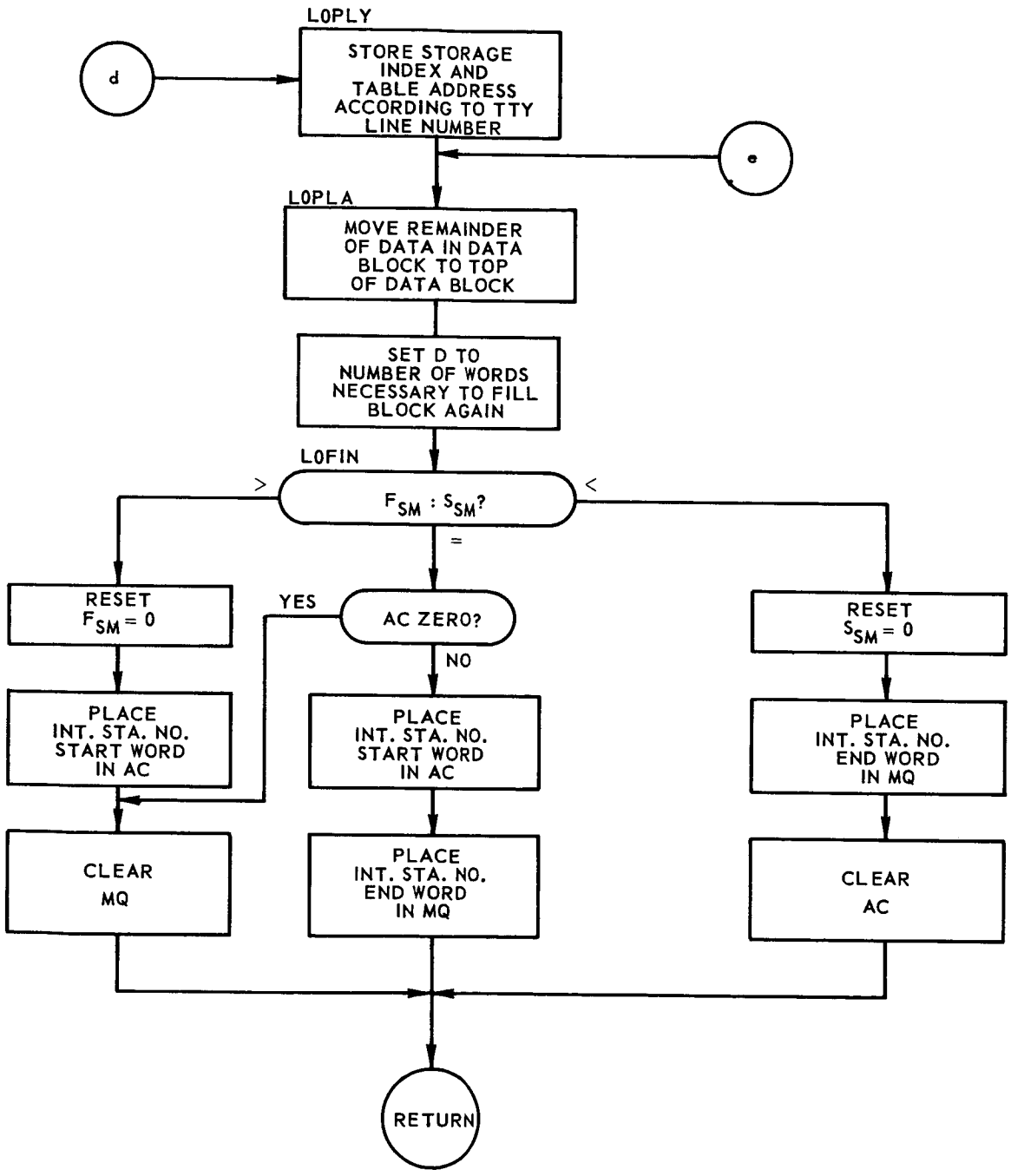


FIGURE 2-1. IOTTIN PROGRAM FLOW CHART *(Sheet 6 of 14)*

LEGEND, IOTTIN FLOW CHART

BSM	=	Start of transmission code (six characters)	
		J J Ltrs Cr Lf Ltrs	
BRM	=	Three-character start of radar message	
		Cr Lf Figs	
ERM	=	One-character end of radar message (/ - slash)	
ESM	=	Four-character end of transmission	
		Blank Figs H Ltrs	
K	=	TTY Input Line No. (1 through 16) from Monitor (DCC subchannel number minus 13)	
$I_{SM}(K)$	=	1	When a radar message is found
		0	When last message, ESM or BSM is found
S_{SM}	=	1	When 1st radar message is found
		0	When 1st radar message is processed
F_{SM}	=	1	When last radar message or ESM code is found
		0	When last radar message of station message is processed
SI (K)	=	Octal external station number, line K, previous message	
$I_{IM}(K)$	=	1	When there is no room for ESM code in the input block after the message was found
		0	When ESM code has been searched for in the first 2 words of the input block
IRAD (K)	=	Octal radar type of line K of previous message	
CSR	=	Internal indication for setting finish transmission words	
		1	When a change in internal station number is found
		0	When a change in external station number is found

FIGURE 2-1. IOTTIN PROGRAM FLOW CHART *(Sheet 7 of 14)*

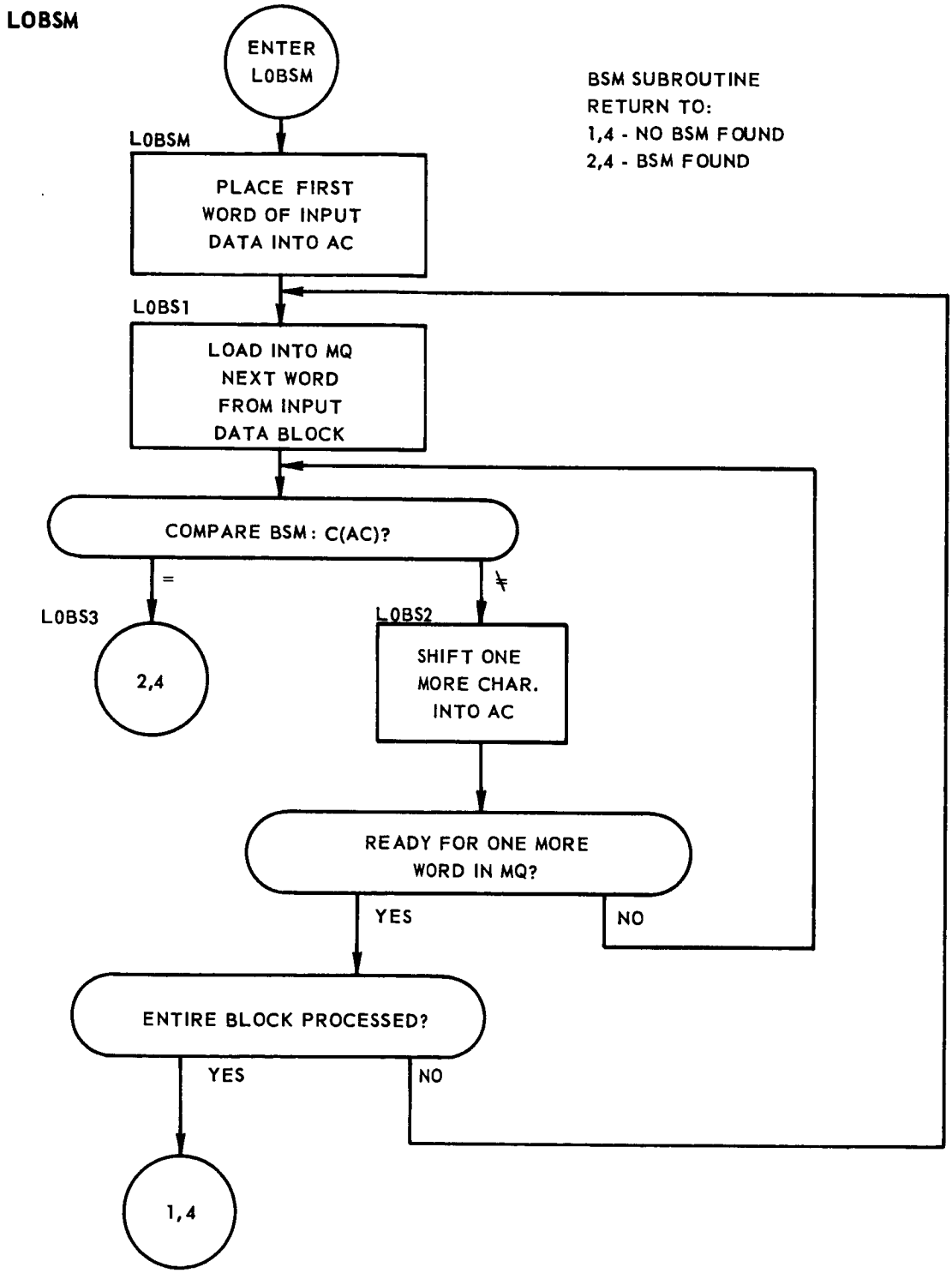


FIGURE 2-1. IOTTIN PROGRAM FLOW CHART *(Sheet 8 of 14)*

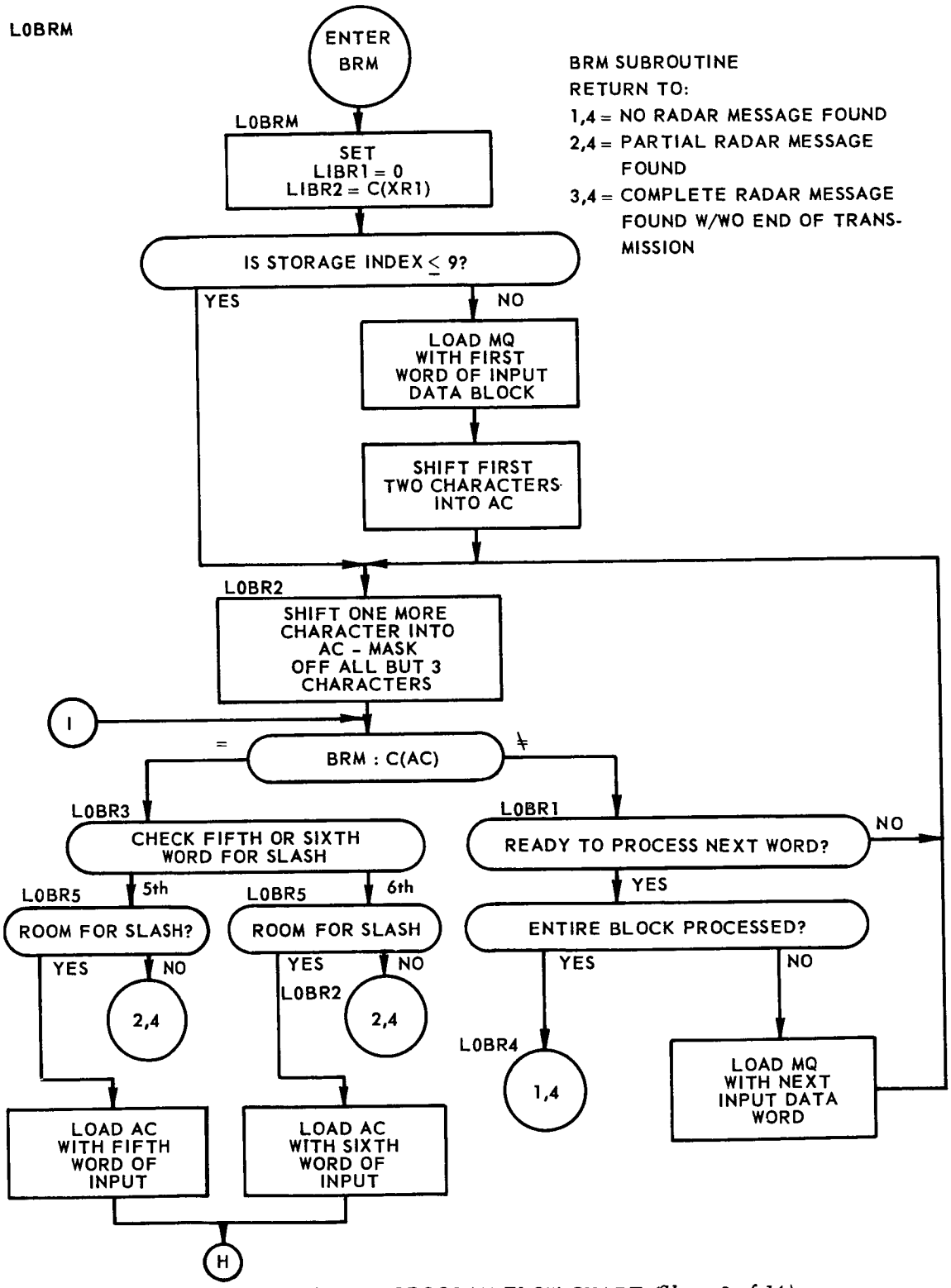


FIGURE 2-1. IOTTIN PROGRAM FLOW CHART *(Sheet 9 of 14)*

LOBRM (Cont'd)

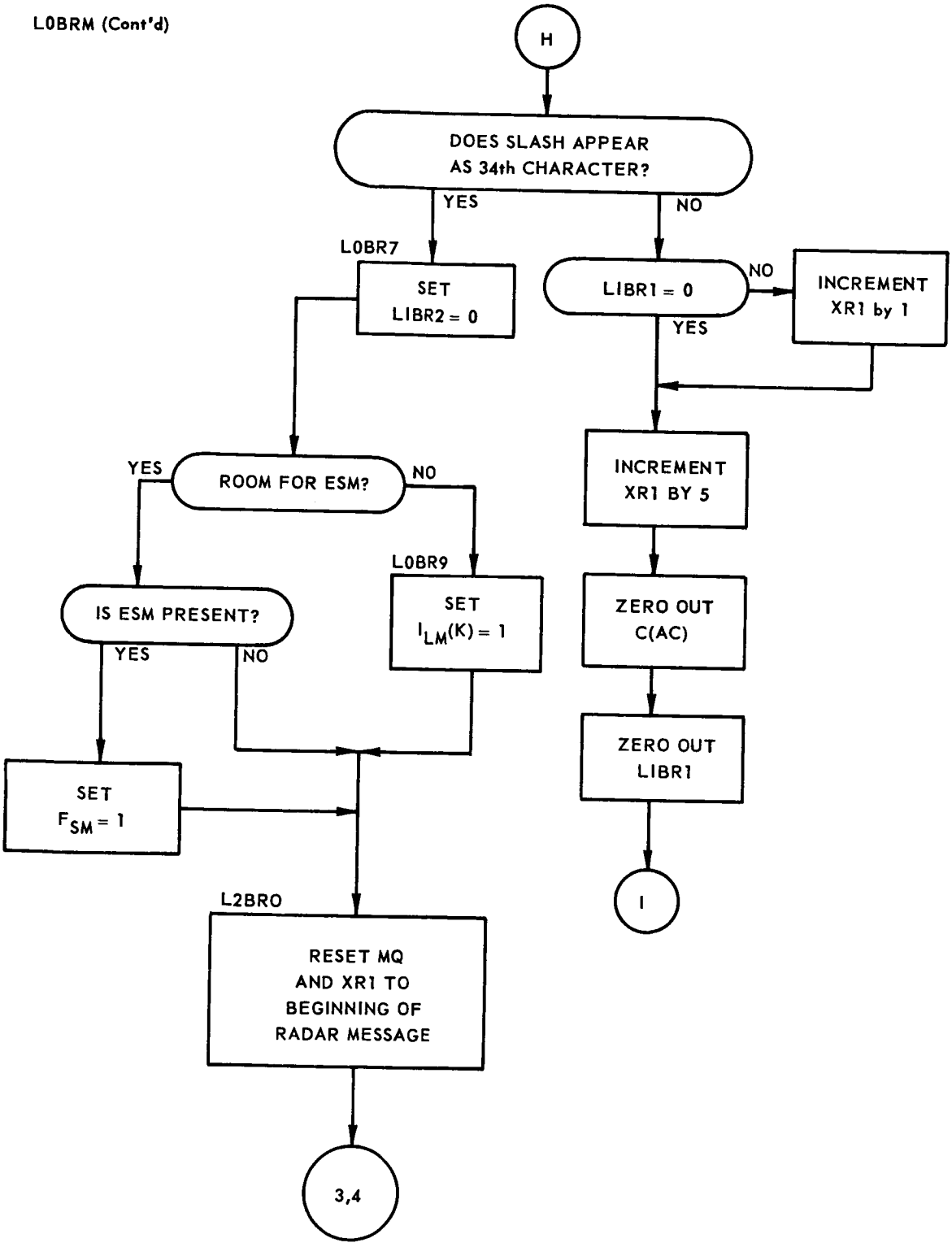


FIGURE 2-1. IOTTIN PROGRAM FLOW CHART *(Sheet 10 of 14)*

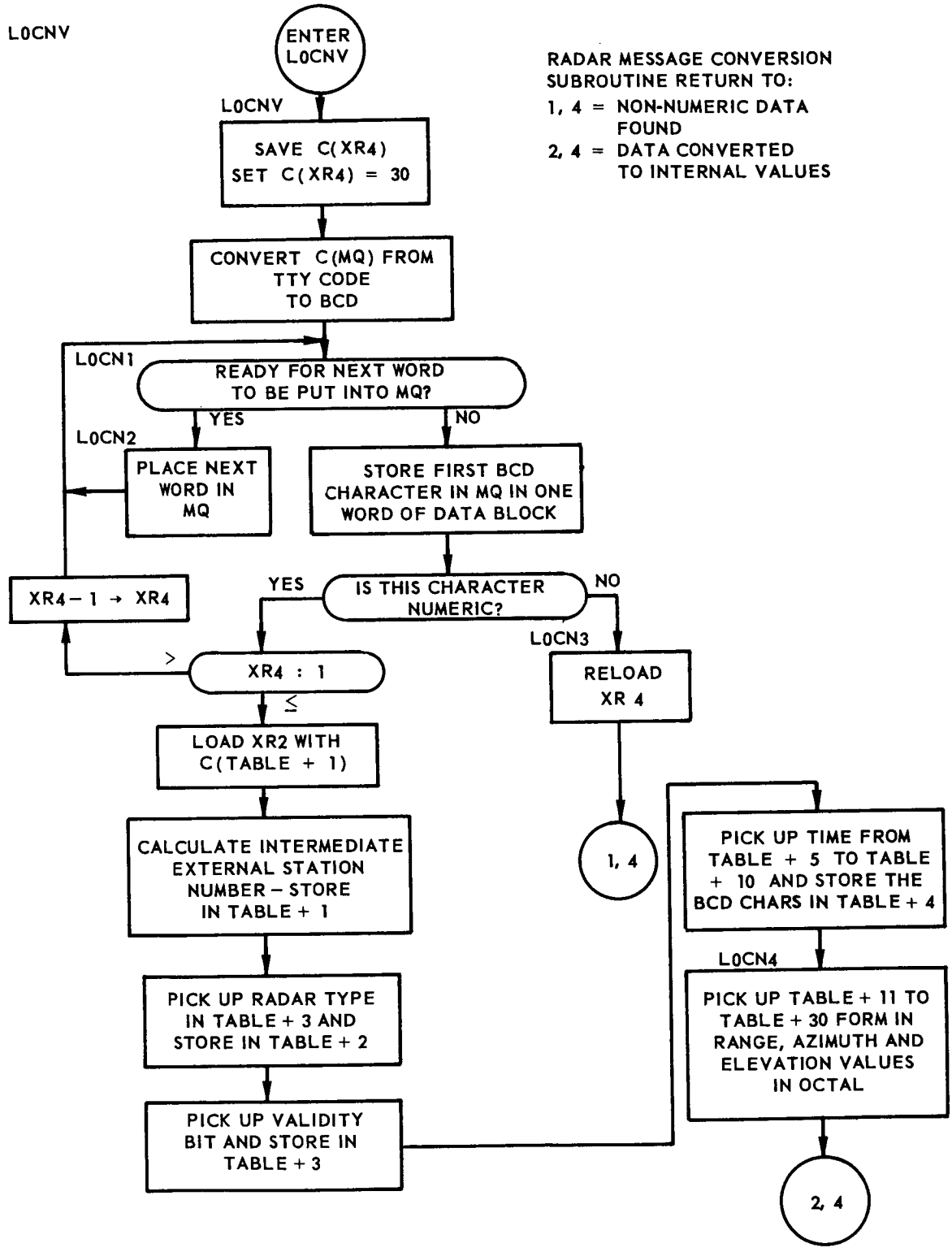


FIGURE 2-1. 10TTIN PROGRAM FLOW CHART *(Sheet 11 of 14)*

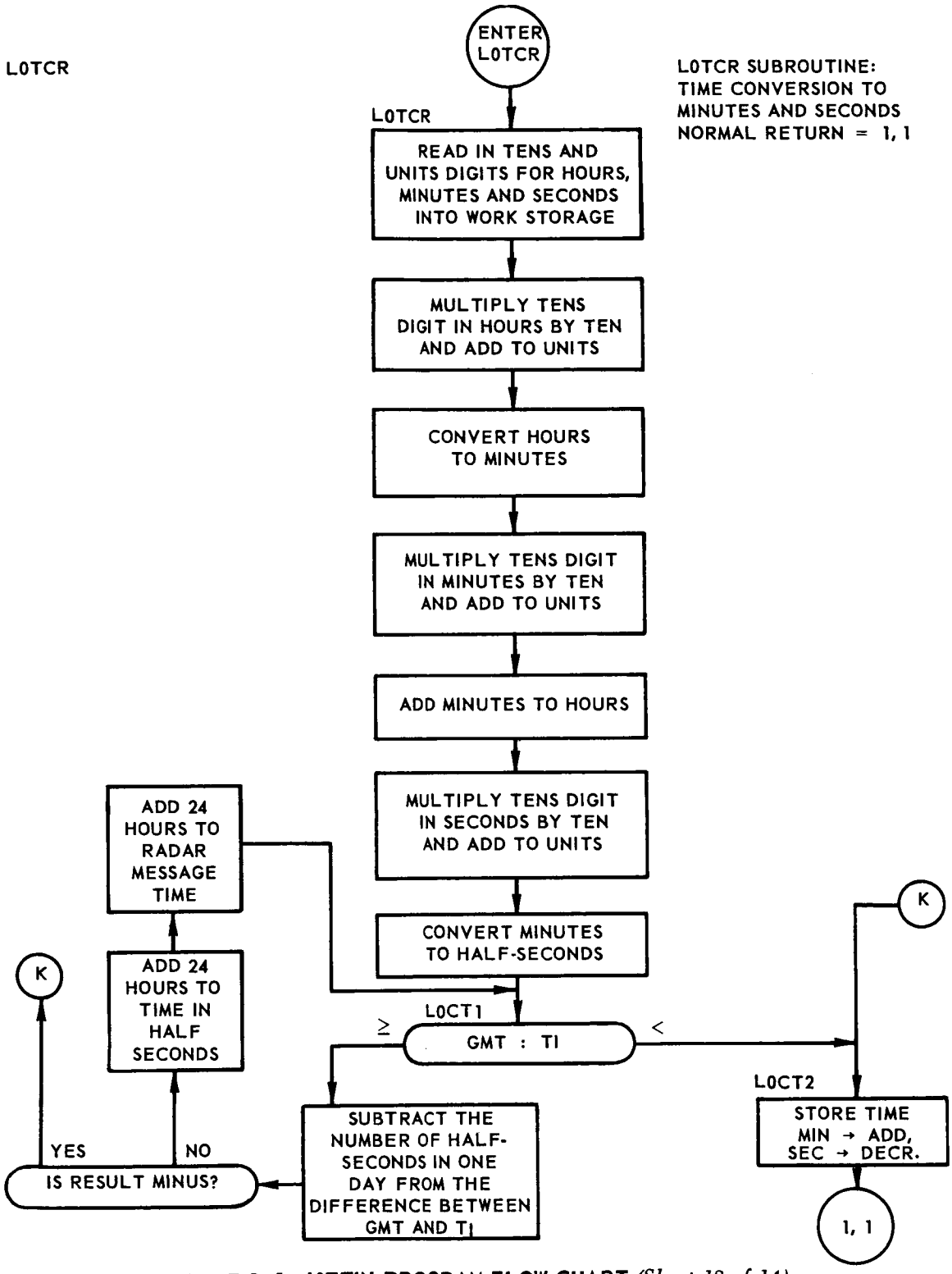


FIGURE 2-1. 10TTIN PROGRAM FLOW CHART *(Sheet 12 of 14)*

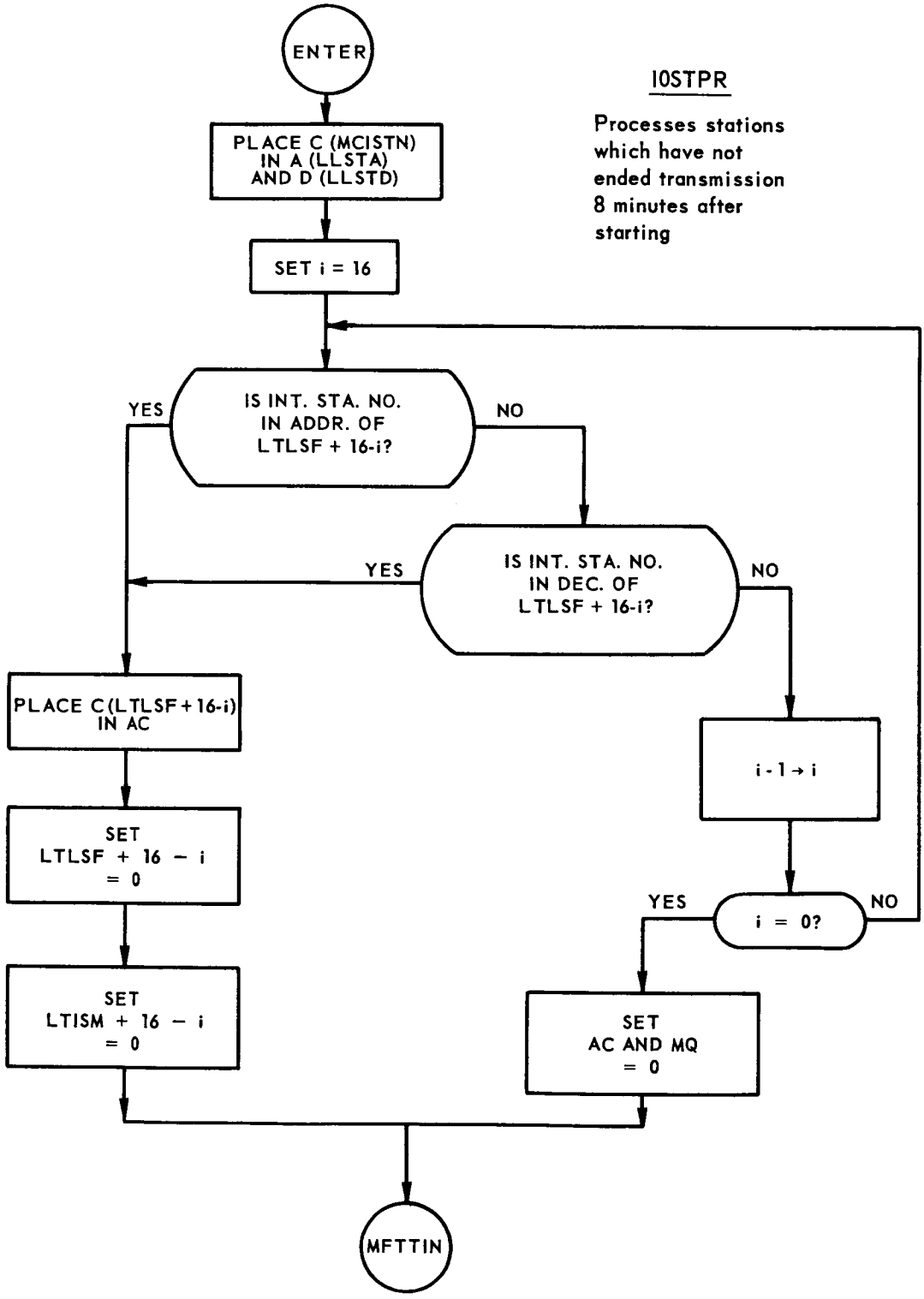


FIGURE 2-1. IOTTIN PROGRAM FLOW CHART *(Sheet 13 of 14)*

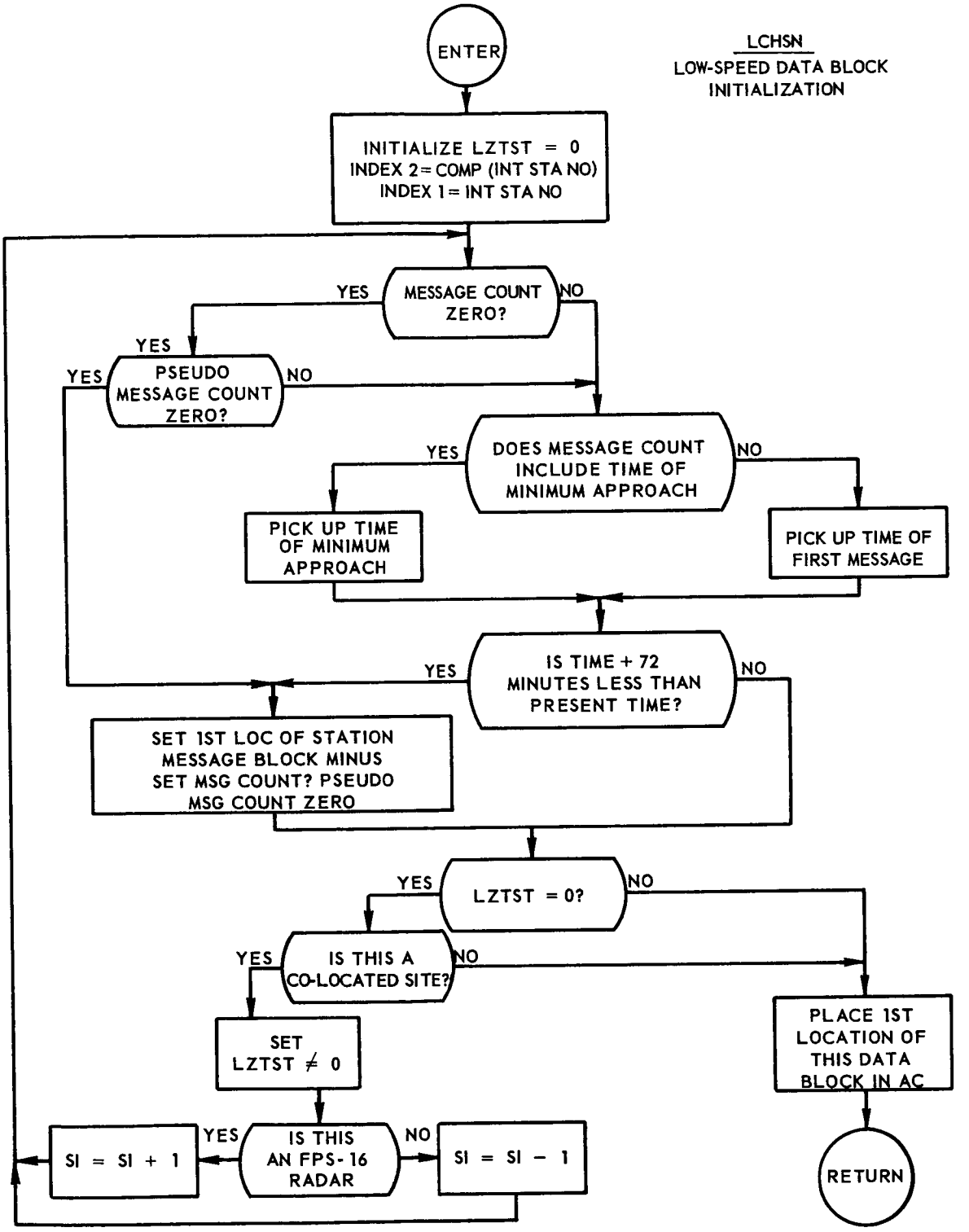


FIGURE 2-1. IOTTIN PROGRAM FLOW CHART *(Sheet 14 of 14)*

Radar range, azimuth, and elevation data is transmitted by teletype from radar tracking stations in a 34-character message as follows:

	Character		
Start of transmissions, sent once per transmission		J	
		J	
		Letters	
		Carriage return	
		Line feed	
		Letters	
Individual data frame	1	Carriage return	
	2	Line feed	
	3	Figures	
	4	Kind of data	2 = Real radar, 3 = Simulated, 7 = Last message
	5	Station	See Note A
	6	Identity	
	7	Radar type	0 = FPS-16, 2 = Verlort, 3-9 = Other
	8	Data validity	0 = Invalid, 2 = Valid
	9	Hours-Tens	
	10	Hours-Units	
	11	Time Mins-Tens	
	12	Mins-Units	
	13	Secs-Tens	
	14	Secs-Units	
	15		
	16		
	17	Azimuth	
	18		
	19		
	20		
	21		
	22		
	23		
	24	Elevation	
	25		
	26		
	27		
	28		
	29		
	30	Range	
	31		
	32		
	33		
	34	Oblique stroke	
End of each transmission		Blank	
		Figures	
		H	
		Letters	

Note A

Station Identification Codes

Station	Char. 5	Char. 6
Cape Canaveral	3	1
Grand Bahama Island	4	1
Grand Turk Island	5	1
San Salvador	6	1
Bermuda	0	2
Atlantic Ship	0	3
Grand Canary Island	0	4
Kano, Nigeria	0	5
Zanzibar	0	6
Indian Ocean Ship	0	7
Muchea	0	8
Woomera	0	9
Canton Island	1	1
Kauai, Hawaii	1	2
Point Arguello	1	3
Guaymas, Mexico	1	4
White Sands	1	5
Corpus Christi	1	6
Eglin	1	7
Goddard	1	8
Mercury Control Cen	1	9

Most significant character of azimuth, elevation, and range is transmitted first.

FIGURE 2-2. RAW RADAR DATA FROM REMOTE RADAR SITES TO GODDARD (TTY), MESSAGE FORMAT

2.2 MANUAL INSERTION PROGRAM (I0MANI)

The I0MANI program converts paper tape input from teletype code to the various formats which are required by the applicable processing programs. The flow chart for the I0MANI program is shown in Figure 2-3.

2.2.1 Input Requirements

Only the following 20 legal characters are accepted by I0MANI. The appearance of any other character causes a rejection of the message if I0MANI is unable to ignore it.

a) FIGURE SHIFT (↑)

b) LETTER SHIFT (↓)

c) SLASH (/)

d) PERIOD (.)

e) LETTER A (A)

f) LETTER B (B)

g) LETTER I (I)

h) LETTER O (O)

i) LETTER R (R)

j) LETTER T (T)

k) TEN DIGITS

Every message must be preceded by a beginning of message indication consisting of at least 6 consecutive figure shifts. An end of message must be indicated by at least 15 consecutive letter shifts. In addition, the beginning of a message indicator should be followed by a slash as should every item in the body of the message itself. I0MANI can ignore certain illegal characters, and also certain violations of the above format. Adherence to the format, however, is recommended as this decreases the chances of I0MANI rejecting a message.

At the present time there are only five types of input messages which are acceptable to I0MANI. Their individual formats are discussed later.

Each time Monitor enters I0MANI the following information must be contained in the accumulator:

A(AC) The first location of the block where the paper tape input characters have been stored (6 bits character).

D(AC) The number of words of information in the above mentioned input block (6 characters word).

2.2.2 Output Requirements

When a complete message is received IOMANI converts each item to a specific format and stores it in the proper location. In the cases of messages 5 and 7 this action takes place after the message has been inserted 3 times consecutively. When this is completed, IOMANI returns to a certain Monitor location which is dependent on which message was successfully processed. Pertinent information concerning each of the five types of messages is shown in Table 2-1.

Before IOMANI processes an incoming message, the program demands that it have a complete message to work with. Until the complete message is received, IOMANI returns to location MFMANI with a positive accumulator. This same return with a negative accumulator indicates that IOMANI has rejected the message and that it should be inserted again. In the case of a rejection of message 5 or 7, the insertion process must start again from the beginning. That is, message 5 must be inserted 3 times followed by message 7 three times.

The following is a list of the various situations which will cause a message to be rejected by IOMANI:

IOMANI will reject if:

a) Eleven words, none of which consists entirely of figure shifts, have been inserted and an end of message indication is not found. (Eleven is large enough to accommodate the longest message at this time.)

b) There isn't a 2 out of 3 agreement between the 3 characters making up the message code. The first character of the message code is assumed to be the first none figure shift or slash following the beginning of message indication.

c) The message code isn't 2, 3, 4, 5, 7, or 8.

d) The program fails to find sufficient information for successful processing after receiving the end of message indicator.

e) The word ABORT or ORBIT is not found at least once in messages 2 and 3 respectively.

f) The times in messages 4, 5, 7, and 8 are not found duplicated from the three times ferreted out of the message (in the cases of messages 5 and 7, from the three insertions).

TABLE 2-1. IOMANI INPUT MESSAGES FORMAT

TIME	Format, Action Taken, Monitor Return	Notes
ABORT SW	/222/↓ABORT↑/↓ABORT↑/↓ABORT↑/	IOMANI scans left to right one character at a time in bytes of 5. Accepts message if "abort" is found once.
	Will or A "2" into location MCMAOS	
	MFMAOS	
ORBIT SW	/333/↓ORBIT↑/↓ORBIT↑/↓ORBIT↑/	Same as above except the word "orbit" is searched for. ↑ = Figure Shift ↓ = Letter Shift
	Will or A "1" into location MCMAOS	
	MFMAOS	
GMTLO	/444/hh mm ss/hh mm ss/hh mm ss/	IOMANI looks for a duplication of time (2 out of 3), compared in groups of six characters. hh mm ss = hours, mins, secs
	GMTLO into MCGTLO – Fx. Pt. secs B35	
	MFMAN1	
R VECTOR	/555/hh mm ss/X/Y/Z/	IOMANI looks for duplications on the 3 inserted messages. Time is stored in Fx.Pt. secs B35 during re-entry and Fx.Pt. mins B35 during orbit. X, Y, Z, $\dot{X}$, $\dot{Y}$, and $\dot{Z}$ stand for signed 12 character floating point octal numbers.
	Time into TIOTRV, X into TIOTRV+1, etc.	
	MFMAN5 (non zero AC)	
V VECTOR	/777/hh mm ss/$\dot{X}$/$\dot{Y}$/$\dot{Z}$/	
	$\dot{X}$ into TIOTRV+4, $\dot{Y}$ into TIOTRV+5, etc.	
	MFMAN5 (zero AC)	
RETROFIRE	/888/NNN/hh mm ss/hh mm ss/hh mm ss/	N is the number of retros fired. IOMANI looks for a 2 out of 3 agreement. If N = 0, return is to MFMAN4.
	N into MCREEN – Fx. Pt. B35	
	Time of RF into MCTOFS – Fx. Pt. secs B35	
	MFMAN2 or MFMAN4	

NOTE: Each of the above formats must be preceded by at least 6 figure shifts and followed by at least 15 letter shifts.

g) The GMTLO of message 4 and the time of retro fire in message 8 is duplicated, but contains a non-numeric character.

h) There isn't a duplication of the number of retros fired in message 8.

i) There isn't a 2 out of 3 agreement for each item on message 5. The comparisons are made after the message has been inserted 3 times. Similarly for message 7.

j) Message 7 is inserted three times before an immediately preceding successful insertion of message 5 (3 times).

k) The time associated with message 5 isn't exactly the same as that associated with message 7.

2.2.3 Usage

Transfer Information

When the correct number of words have been read in, transfer should be made to I0MANI. If the input message does not contain an entire message, I0MANI will recognize the fact and will transfer to MFMANI with the number of words to collect before transferring back. This number will be placed in the accumulator, fixed point B17.

Space Required:

Working Storage	7
Instructions	246
Constants	18
Erasable Storage	45
Subroutines (Internal)	78
Tables	85
Total	479

Testing

I0MANI was tested by feeding it 107 different messages (both good and bad) by means of a debug program. The 107 messages were of every type conceivable designed to cause the execution of every instruction at least once under the various conditions possible.

Flow Chart Symbols

"T_o" is the first word of the block holding the input from monitor to I0MANI.

"N_o" is the first word of the block into which I0MANI transfers the input.

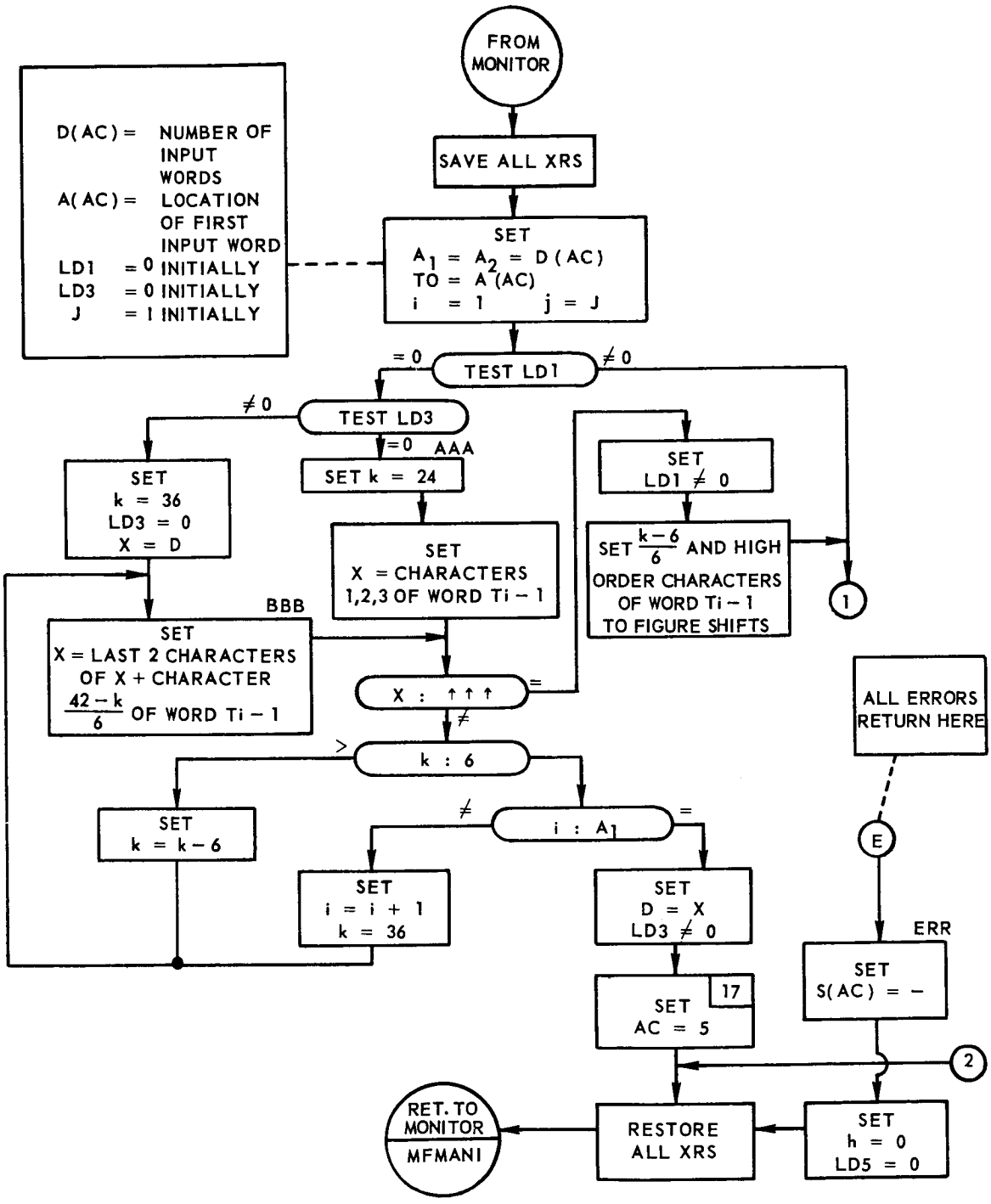


FIGURE 2-3. IOMANI PROGRAM FLOW CHART *(Sheet 1 of 6)*

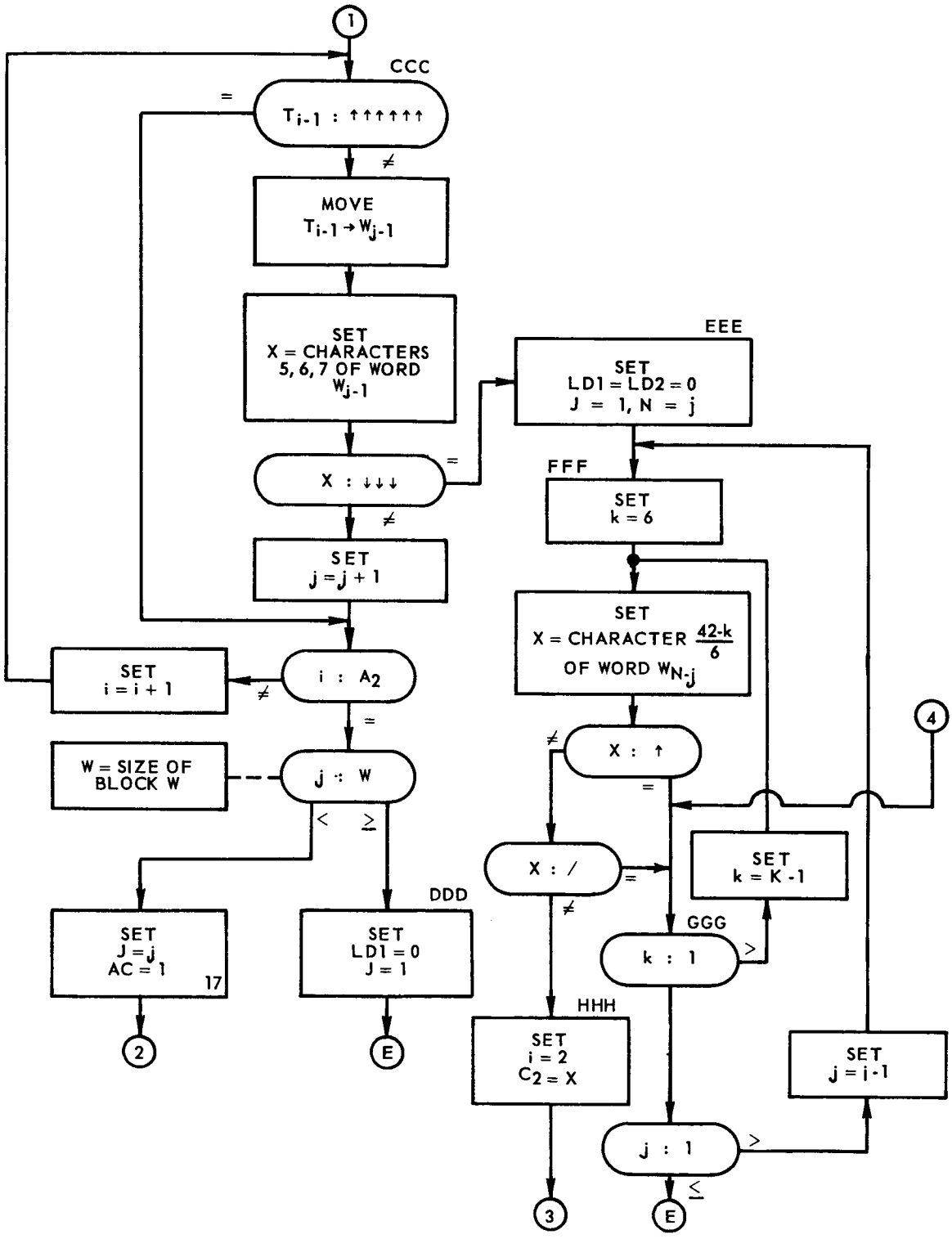


FIGURE 2-3. IOMANI PROGRAM FLOW CHART *(Sheet 2 of 6)*

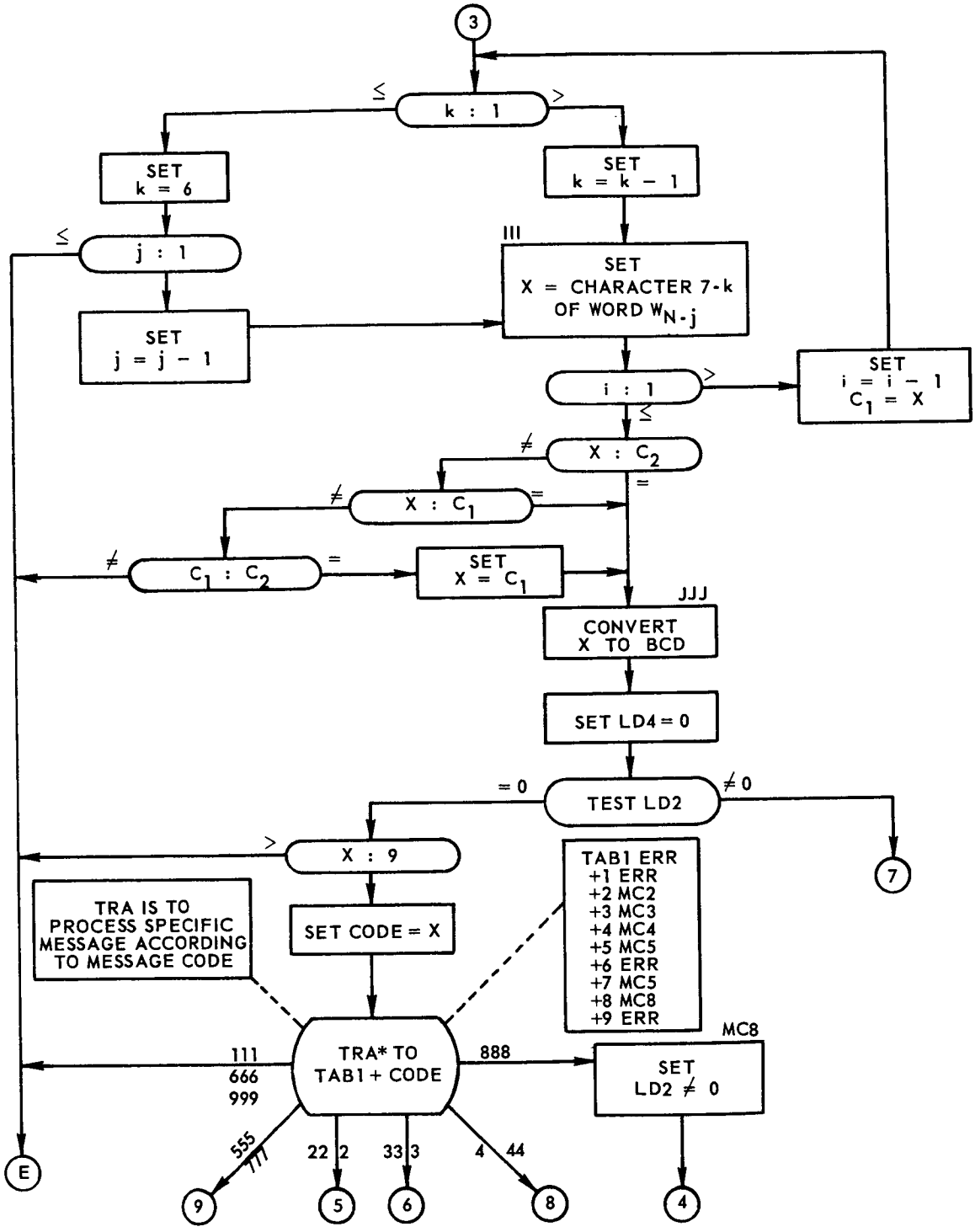


FIGURE 2-3. 10MANI PROGRAM FLOW CHART *(Sheet 3 of 6)*

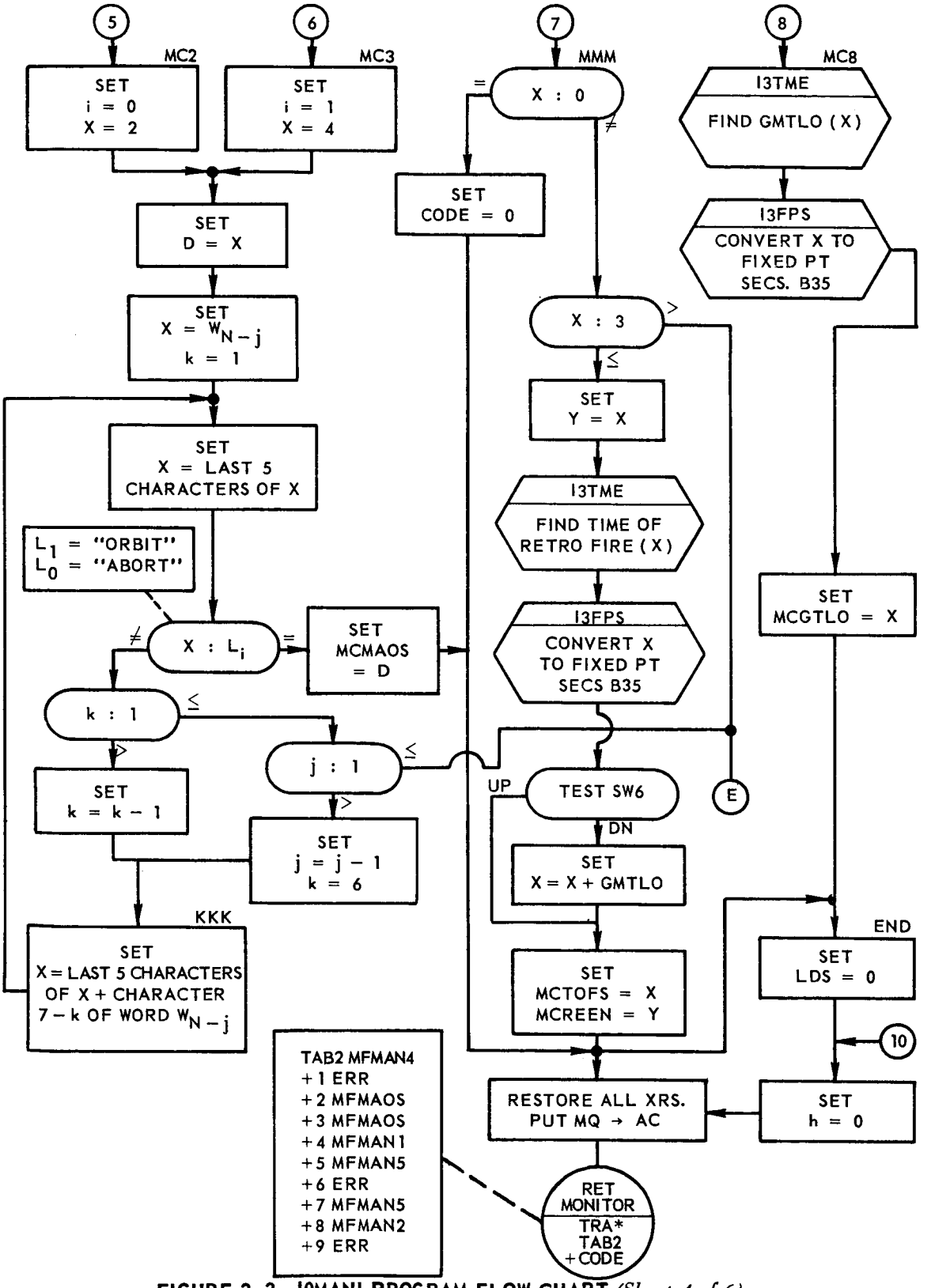


FIGURE 2-3. IOMANI PROGRAM FLOW CHART *(Sheet 4 of 6)*

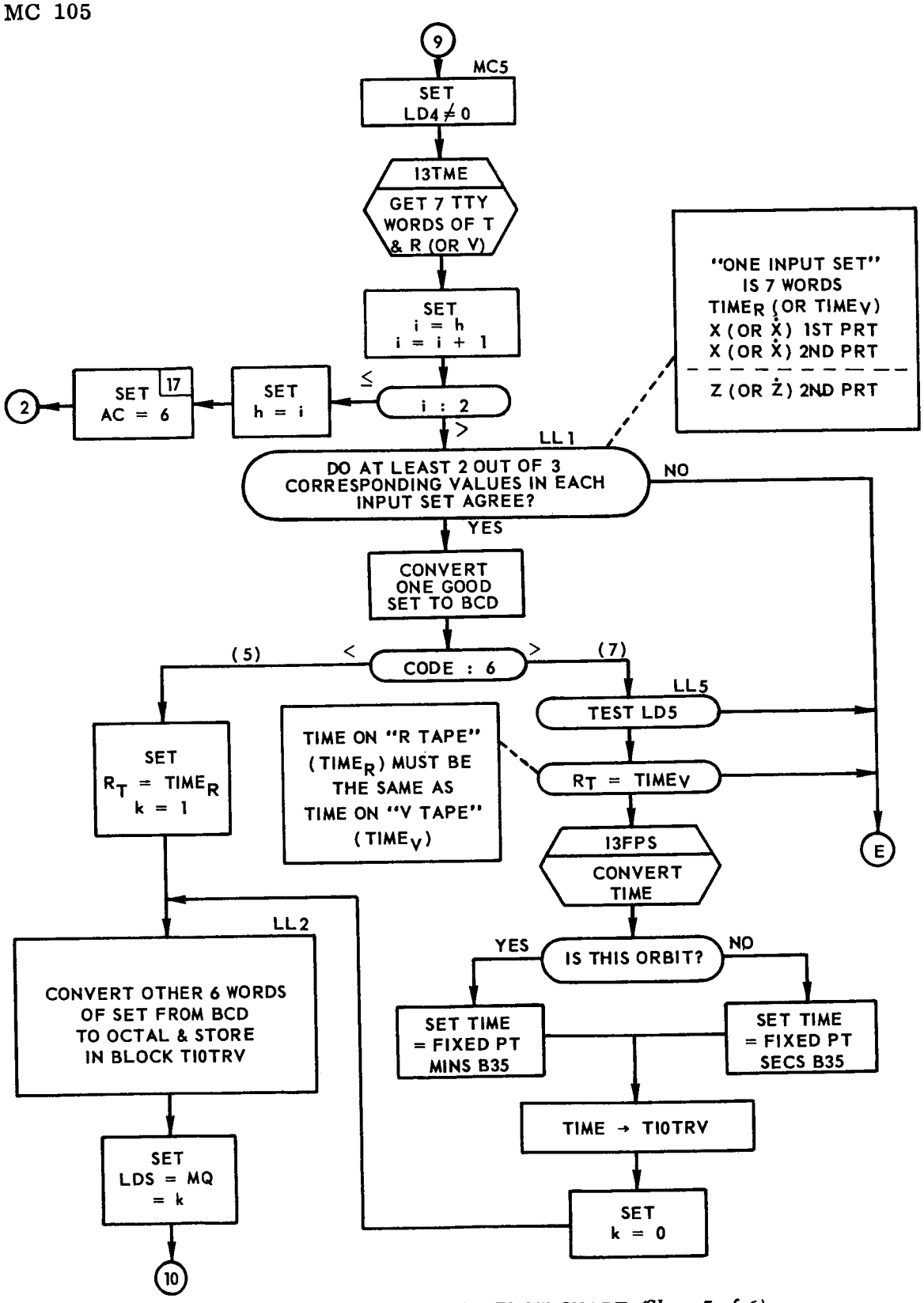


FIGURE 2-3. IOMANI PROGRAM FLOW CHART *(Sheet 5 of 6)*

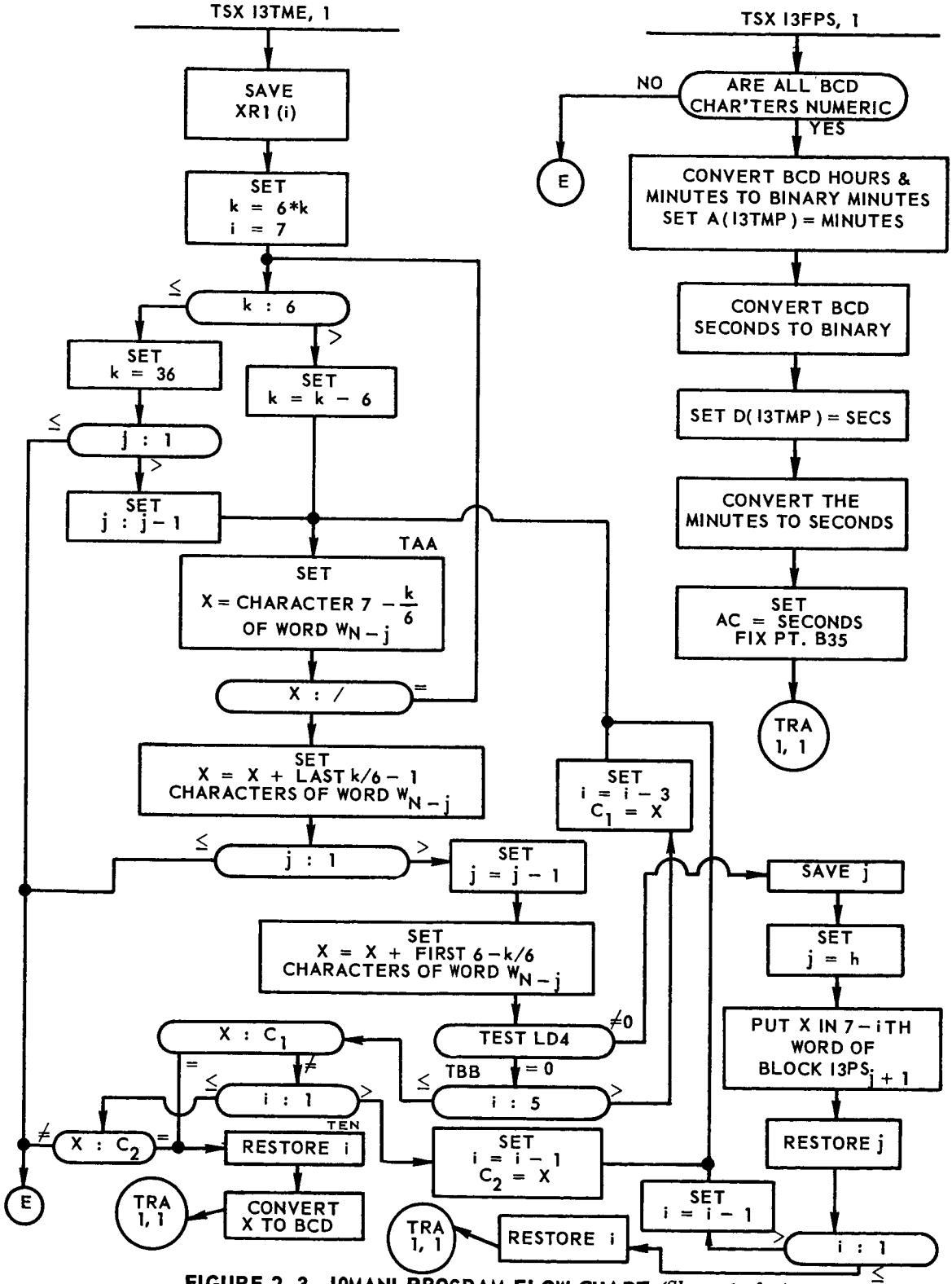


FIGURE 2-3. IOMANI PROGRAM FLOW CHART *(Sheet 6 of 6)*

2.3 B-GE HIGH-SPEED INPUT PROGRAM (I0HSGB)

I0HSGB processes B-GE computed data and associated telemetry summaries transmitted to Goddard from Cape Canaveral during the launch phase of all Mercury missions. The flow chart for I0HSGB is shown in Figure 2-4.

2.3.1 Input Requirements

Input to I0HSGB is stored in the Monitor communications cell, MCHST1, and in the 48-word data table, TMHSL1, each word of which contains an eight-bit entry right justified. On each entry to the program,

a) the 48 words of the data table contain a 24 word first or second subframe and its duplicate (the format of each of these subframes is specified in the message format shown in Figure 2-5).

b) the cell MCHST1 contains, in bits 0 through 26, the integral number of half second increments which have elapsed since midnight (GMT) preceding launch and, in bits 27 through 35, the integral number of 8-1/3 milliseconds increments which have elapsed since the last one-half second.

2.3.2 Output Requirements

The type and form of output from the program are shown below.

a) TMH1DB: Seven entry computed data table with entries as follows:

1) Time from two-inch lift-off in floating point seconds

2) Inertial position (X)

3) Inertial position (Y)

4) Inertial position (Z)

5) Inertial velocity ($\dot{X}$)

6) Inertial velocity ($\dot{Y}$)

7) Inertial velocity ($\dot{Z}$)

Entries 2 through 7 above are stored in floating point B-GE units.

b) TMH1TM: Four entry discrete data table with entries as follows:

1) Time from two-inch lift-off in floating seconds

2) B-GE discrete signal word with the following format:

Bits 0-27 are not used and are set to zero

Bit 28—GO NO-GO recommendation

Bit 29—δ_1 data flag

Bit 30—δ_2 data flag

Bit 31—δ_3 data flag

Bit 32—γ_n data flag

Bit 33—two-inch lift-off signal

Bit 34—booster engine cut-off (BECO)

Bit 35—sustainer engine cut-off (SECO)

3) GMT computed time tag of the telemetry data in floating point seconds.

4) Telemetry discrete signal word with the following format:

Bits 0-22 are not used and are set to zero

Bit 23—two-inch lift-off

Bit 24—escape tower released

Bit 25—tower escape rockets fired

Bit 26—capsule released from sustainer

Bit 27—one of three posigrades fired

Bit 28—two of the three posigrades fired

Bit 29—three of three posigrades fired

Bit 30—abort sequence initiated

Bit 31—retrorocket 1 fired

Bit 32—retrorockets 1 and 2 fired

Bit 33—retrorockets 1, 2, and 3 fired

Bit 34—abort phase started

Bit 35—orbit phase started

A one in any of the discrete signal positions of entries 2 and 4, above, indicates that the event has occurred; a one in any of the data quality flag positions indicates an acceptable data condition.

c) TMHSIT: Three entry telemetry time data table with the following entries:

Capsule elapsed time

Retrofire mechanism setting

Elapsed GMT time tag of telemetry data transmission

Each of these entries is stored in fixed point integer format, minutes in the address and seconds in the decrement of the respective words.

d) MCCOMI: Computed data validity indicator. A one in this cell indicates the rejection of the processed computed data and retention of the previous contents of TMHIDB.

e) MCTEL1: Telemetry data validity indicator. A one in this cell indicates rejection of the telemetry data and retention of the previous contents of TMHITM, THMHSIT, MCSGEB, and MCS709.

f) MCSGEB: B-GE selected source indicator. A one in this cell indicates B-GE as the selected source for the main launch computations.

g) MCS709: IP 7090 selected source indicator. A one in this cell indicates IP 7090 as the selected source for the main launch computations.

2.3.3 Method

A data frame consisting of 384 (8 x 48) binary bits is transmitted at a rate of 1000 bits/sec over a duplexed line from the B-GE buffer and is received as a sequence of two 192 bit (8 x 24) subframes. Each subframe and its image are stored by Monitor in the 48-word table, TMHSL1, and control is transferred to I0HSGB.

The generation of a complete set of output data as specified in 2.3.2 above, requires the reception of a complete high speed data message, i.e., a first and second subframe in sequence, and, hence, a minimum of two entries to the program I0HSGB. The processing of such a message consists of three phases:

Subframe Identification

Telemetry Data Processing

Computed Data Processing

A discussion of each of these phases follows:

Subframe Identification: Upon each entry to the program I0HSGB, the given subframe, words 1 through 24 of the data table TMHSL1, is first examined to insure that it is not "solid," i.e., that not all twenty-four entries of the subframe are identical. If the subframe is not solid, the subframe ID word, word 24 of the data table, is examined to determine its validity. If the subframe fails to meet either of these criteria, the same tests are performed on the subframe image, words 25 through 48 of TMHSL1. Control is returned to Monitor by way of the "missing data" exit if the image also fails to satisfy either of these requirements. However, processed according to the ID of the entries first satisfying both of these standards.

If the subframes are first frames, the telemetry data of the message is processed, the computed data entries of TMHSL1 are transferred to temporary storage, and a second subframe indicator is set to 1. Control is then returned to Monitor by way of a "missing data" exit.

If the received subframes are second subframes, and if the second subframe indicator is set to 1, the 48 entries of TMHSL1 are transferred to the remaining 48 words of the temporary storage block, the second subframe is reset to zero, and the computed data of the message is processed.

A second subframe received with the second subframe indicator is set to zero is rejected without processing. In this case, the indicator MCTEL1 is set to 1, and control is returned to Monitor by way of the "missing data" exit.

Telemetry Data Processing: When a first subframe is recognized, telemetry data is processed in the following manner. The set of three seven-bit-repeated-signal-sequences is first examined to insure:

a) That not all of the seven bits of the sequence are zero;

b) That not both the abort and orbit bits are set to one;

c) That all three of the seven bit sequences are identical.

If each of the above conditions is satisfied, the repeated signals are stored, and the GMT time tag of this telemetry data is computed and stored. If any one of these conditions is not satisfied, the telemetry data of the subframe image is similarly tested. Failure of the image data to satisfy the criteria causes the rejection of the telemetry data, the setting of the cell MCTEL1 to one, and the retention of the previous contents of TMH1TM, TMHSIT, MCS709 and MCSGEB.

The nonrepeated signals of the telemetry data entries satisfying the repeated signal criteria are processed in the following manner. Parity of this telemetry data is determined and compared with the transmitted parity bit. Parity disagreement causes a return to Monitor without further processing, whereas parity agreement causes the storing of the posigrade and retrograde rocket signals, the setting of the selected source indicators, MCSGEB and MCS709 and the generation of the table TMHSIT.

Upon completion of the telemetry processing, control is returned to Monitor by direct transfer to MFML6A.

Computed Data Processing: Upon receipt of a first and second subframe in sequence, the computed data entries of the message are processed. The checksum of the message and, if necessary, of the image, are successively computed and compared with the transmitted checksums. The data contained in that block which first produces checksum agreement is then processed to produce the output table TMHIDB and the locations TMHITM, THMHIT - 1, and the cell MCCOMI is set to zero. Failure to find a valid checksum causes the rejection of the computed data and the setting to one of the indicator MCCOMI. In either case, control is returned to Monitor by transfer to MFHSGB.

2.3.4 Usage

a) Space Required:

The program I0HSGB requires 525 core storage locations. The input and output data storage locations are not included in this figure and requires an additional 55 core locations.

b) Constants:

K00000	DEC	0
K00001	DEC	1
K00003	DEC	3
K00060	DEC	60
KCH200	OCT	200 000 000 000
KCH201	OCT	201 000 000 000
KCH211	OCT	211 000 000 000

c) Special Usage:

I0HSGB must be entered at least twice in order to generate the output data as specified in paragraph 2.3.2. Computed data output is generated only after receipt of a cpmplere message, i.e., a first and second subframe in sequence, whereas telemetry data output is generated each time a first subframe is received. Control is returned to Monitor by a direct transfer to MFHSGB only when a first and second subframe have been received in sequence. In all other cases, return is made to MFML6A.

d) Time Required—approximately 10 milliseconds.

e) Checkout—The program has been operational in the Mercury Program System since October of 1960.

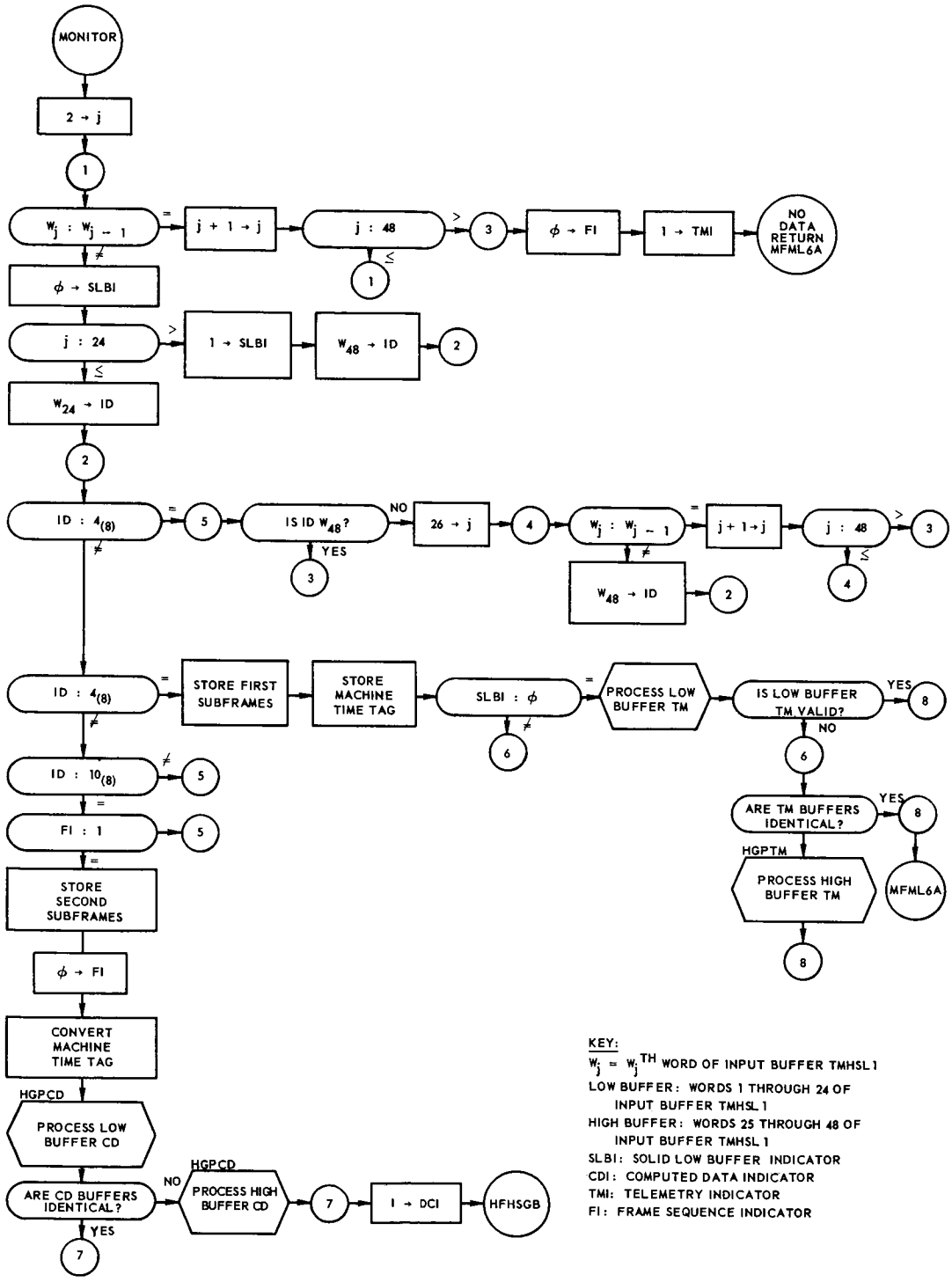


FIGURE 2 - 4. 10HSGB PROGRAM FLOW CHART *(Sheet 1 of 3).*

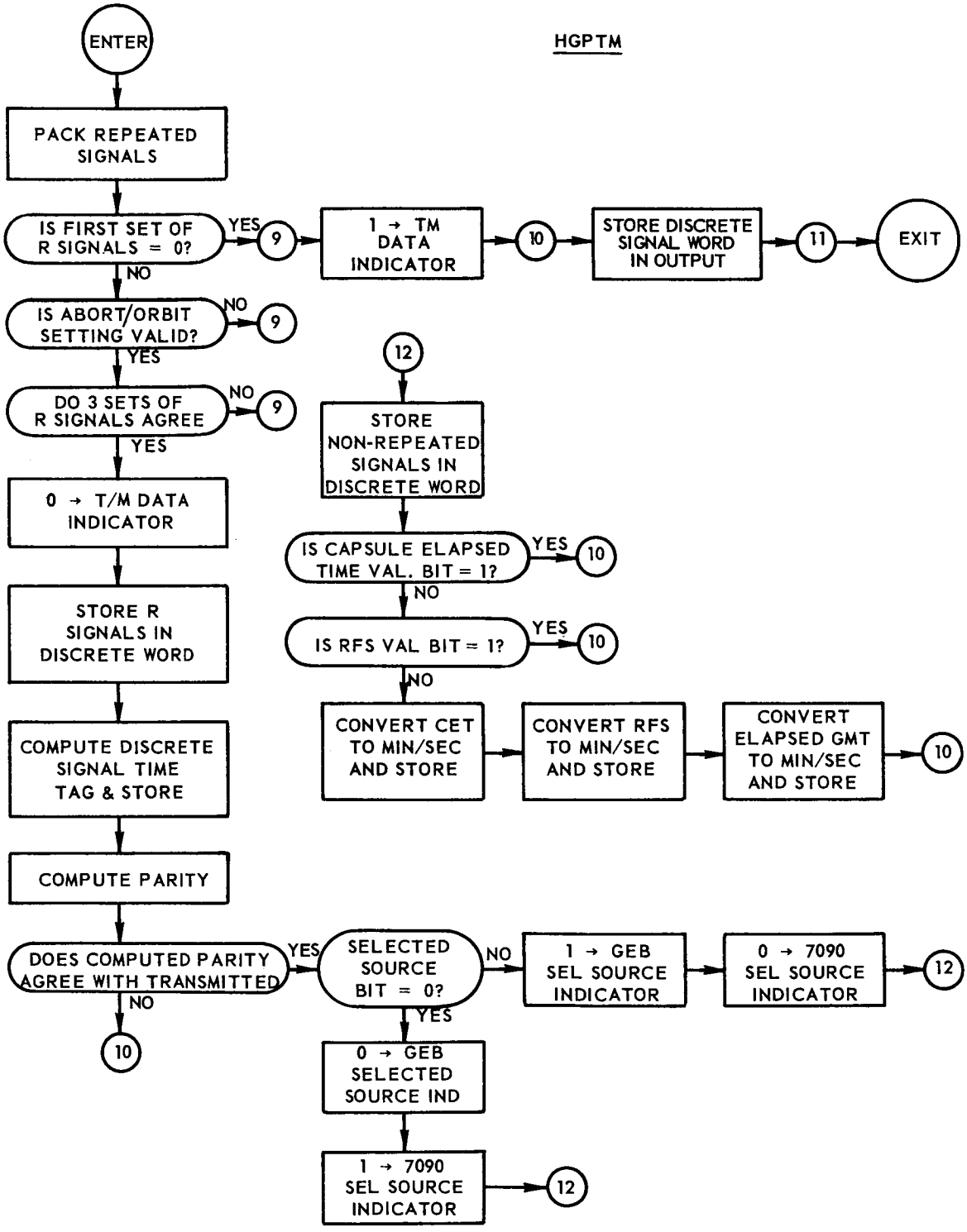


FIGURE 2-4. 10HSGB PROGRAM FLOW CHART *(Sheet 2 of 3)*

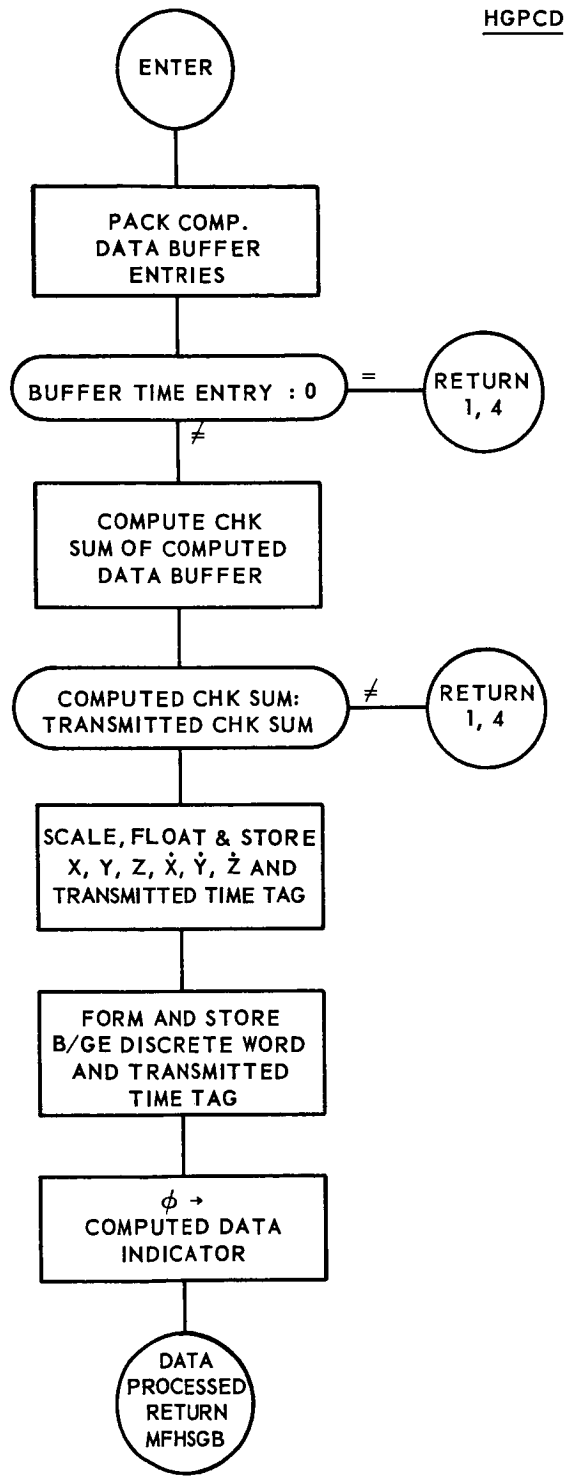


FIGURE 2-4. 10HSGB PROGRAM FLOW CHART *(Sheet 3 of 3)*

8–Bit Word Transfers from Data Receiver
First Subframe

1	1	Telemetry	8
2	9	Telemetry	16
3	17	Telemetry	24
4	25	Telemetry	32
5	33	Telemetry	40
6	41	Telemetry	48
7	49	Telemetry	56
8	57	Telemetry	64
9	65	Telemetry	72
10	1	Discrete Word	8
11	1	G	8
12	9	G	16
13	17	G	24
14	1	H	8
15	9	H	16
16	17	H	24
17	1	J	8
18	9	J	16
19	17	J	24
20	1	K	8
21	9	K	16
22	17	K	24
23	1 1 1 1 1 1 1 1		
24	0 0 0	1 ID*	5

8–Bit Word Transfers from Data Receiver
Second Subframe

1	1	L	8
2	9	L	16
3	17	L	24
4	1	M	8
5	9	M	16
6	17	M	24
7	1	N	8
8	9	N	16
9	17	N	24
10	1	Checksum	8
11	9	Checksum	16
12	17	Checksum	24
13	1 1 1 1 1 1 1 1		
14	1 1 1 1 1 1 1 1		
15	1 1 1 1 1 1 1 1		
16	1 1 1 1 1 1 1 1		
17	1 1 1 1 1 1 1 1		
18	1 1 1 1 1 1 1 1		
19	1 1 1 1 1 1 1 1		
20	1 1 1 1 1 1 1 1		
21	1 1 1 1 1 1 1 1		
22	1 1 1 1 1 1 1 1		
23	1 1 1 1 1 1 1 1		
24	0 0 0	1 ID*	5

The above constitutes a complete message frame and is transmitted with an interval of 500 ± 100 milliseconds between the start of one message and the start of the next message. Each subframe consists of 192 serial bits preceded by a sync signal. The quantities represented by G, H, J, K, L, M, and N are restricted information and are specified in other documents.

* See Note 1 for makeup of ID word.

See Note 2 for format of bits 1 to 72 of first subframe in absence telemetry data and Note 3 for format in absence of quantities G, H, J, K, L, M, and N.

See table 2–15 for format of telemetry data.

FIGURE 2-5. BURROUGHS-GE DATA, CAPE CANAVERAL TO GODDARD, MESSAGE FORMAT (SHEET 1 OF 2)

NOTES

1. The 5-bit identity (ID) word is each subframe conveys the following information:

   Date from Burroughs–GE High-Speed Buffer and Retransmitter.

   Bit 1: Always a 0.
   Bit 2: A 1 signifies second subframe.
   Bit 3: A 1 signifies first subframe.
   Bit 4: Always a 0.
   Bit 5: Always a 0.

2. In the event that no data is received from the telemetry event transmitting buffer, the high-speed buffer and retransmitters are arranged to transmit 0's in the bit positions occupied by telemetry event data bits 1 through 40 and 43 through 72. Ones are transmitted in positions 41 and 42, resulting in erroneous parity for the telemetry event data message.

3. In the absence of data quantities G, H, J, K, L, M, and N, the complete message frame is transmitted every 650 milliseconds. Telemetry data continues to be transmitted. Zeros with 1's interspersed in certain positions are transmitted in place of the missing data quantities. The following bits appear as 1's in this event:

Subframe	8–Bit Word No.	Quantity	Bit Within Quantity
1	13	G	24
1	16	H	24
1	19	J	24
1	22	K	24
2	3	L	24
2	6	M	24
2	9	N	24
2	10	Checksum	1

FIGURE 2-5. BURROUGHS-GE DATA CAPE CANAVERAL TO GODDARD, MESSAGE FORMAT (SHEET 2 OF 2)

2.4 IP 7090 HIGH-SPEED INPUT PROGRAM (I0HS09)

I0HS09 processes all IP 7090 computed data, unprocessed AN/FPS-16 radar observations, and telemetry data transmitted from the IP 7090 buffer over duplexed high-speed lines during the launch phase of every Mercury mission. The type of data being transmitted, i.e., either computed data or radar observations—is manually selected at the 7090 buffer and is indicated by a coded identification in each transmission. This data selection affects the format of the telemetry data transmitted but not the data content. The flow chart of I0HS09 is shown in Figure 2-6.

2.4.1 Input Requirements

Input to I0HS09 is stored in the Monitor communication cell, MCHST2, and in the 48-word data table, TMHSL2.

a) The 48-entry table TMHSL2 contains a 24-word first or second subframe and its duplicate. Each entry of the table contains eight data bits right justified. (Subframe format is shown in Figure 2-7.)

b) The cell MCHST2 contains, in bits 0 through 26, the integral number of half second increments which have elapsed since midnight (GMT) preceding launch and, in bits 27 through 35, the integral number of 8-1/3 milliseconds increments which have elapsed since the last 1/2 second.

2.4.2 Output Requirements

Data selection at the IP 7090 buffer determines which of the outputs listed below are generated by I0HS09, the processing of computed and radar data being mutually exclusive on any entry. The processing of telemetry data, however, is independent of this data selection, and the same output quantities are computed from all telemetry data. A description of each of the output tables and indicators follows:

a) TMH2DB—location computed data table with entries:

1) time from two-inch lift-off in floating point seconds

2) Inertial Position (X)
3) Inertial Position (Y)
4) Inertial Position (Z)
5) Inertial Velocity ($\dot{X}$)
6) Inertial Velocity ($\dot{Y}$)
7) Inertial Velocity ($\dot{Z}$)

These values are stored in floating point IP 7090 units.

b) TMHRAE—361-location table of radar observations with entries:

1) T_1—Elapsed GMT time tag of R_1, A_1, and E_1 in .1 second increments

2) R_1—slant range in fixed point feet, B35

3) A_1—Azimuth in fixed point radians, B14

4) E_1—Elevation in fixed point radians, B14

5) ⋮

6) R_i

7) A_i

8) E_i

where R_i, A_i and E_i represents the range, azimuth, and elevation entries, respectively, of the ith radar observation, $i = 1, 2, \ldots 120$.

c) TMHEDP—7-location table of edit parameters. The first entry of this table contains that location of TMHRAE containing the range entry of the last observation inserted by I0HS09. This value is stored by I0HS09, and each of the seven values of the table is adjusted by I0HS09 when the data table TMHRAE is moved to accommodate more recent observations.

d) TMH2TM—2-location table of telemetry discrete signals with entries:

1) T—elapsed GMT time tag of the telemetry discrete signal word in floating point seconds.

2) Discrete Signal Word with following format:

Bits 0 through 22 are not used and are set to zero.

Bit 23—two-inch lift-off signal

Bit 24—escape tower released

Bit 25—tower escape rockets fired

Bit 26—capsule released from sustainer

Bit 27—one of three posigrades fired

Bit 28—two of three posigrades fired

Bit 29—three of three posigrades fired

Bit 30—abort sequence initiated

Bit 31—retrorocket 1 fired

Bit 32—retrorockets 1 and 2 fired

Bit 33—retrorockets 1, 2 and 3 fired

Bit 34—abort phase started

Bit 35—orbit phase started.

A one in any of these bit positions indicates that the event has occurred.

e) TMHS2T—3-location table of telemetry time data with entries:

1) Elapsed Capsule Time

2) Retrofire Mechanism Setting

3) Elapsed GMT time tag of this telemetry data

Each of the entries of TMHS2T is stored in fixed point integer format, with minutes in the address and seconds in the decrement of the respective words.

f) MCCOM2—Computed Data Validity Indicator: A one in this cell indicates rejection of the computed data and retention of the previous contents of TMH2DB.

g) MCTEL2—Telemetry Data Validity Indicator: A one in this cell indicates rejection of the telemetry data and retention of the previous contents of TMH2TM, TMHS2T, MCS709, and MCSCEB.

h) MCWCH2—Data Indicator: This cell contains the identity of the data currently being processed by I0HS09 coded as follows:

2—unprocessed radar (AN/FPS-16)

3—IP 7090 computed data (AZUZA)

4—IP 7090 computed data (AN/FPS-16)

i) MCS709—A one in this cell indicates the IP 7090 buffer data as the main launch computations selected source.

j) MCSGEB—A one in this cell indicates the B-GE computed data as the main launch computations selected source.

k) MCSRRS—unprocessed radar source indicator. Since any one of the three AN/FPS-16 radars of the Cape Canaveral complex can be selected as the source radar for unprocessed radar data, this cell contains the identity of the transmitting radar coded as follows:

1—Cape Canaveral AN/FPS-16

2—Grand Bahama AN/FPS-16

3—San Salvador AN/FPS-16

2.4.3 Method

A data frame consisting of 384 (8 x 48) binary bits is transmitted at a rate of 1000 bits/sec over a duplexed line from the IP 7090 buffer and is received as a sequence of two duplicate 192 (8 x 24) bit subframes. Each subframe and its image are stored by Monitor in the 48-word table, TMHSL2, and control is transferred to I0HS09.

Upon each entry the given subframe, words 1 through 24 of TMHSL2, is examined to insure that it is not "solid," i.e., that not all twenty-four entries of the subframe are identical. If the subframe is not "solid," the ID word, word 24 of TMHSL2, is examined to insure that it is valid. If the subframe fails to satisfy either of these criteria, the subframe image, words 25 through 48 of TMHSL2, is similarly tested. Failure of the image to satisfy either of these requirements causes a "missing data" return to be made to Monitor without further processing.

However, the subframe and its image are both processed according to the subframe number and coded data identity of that valid ID word first detected. A description of the processing of the telemetry, computed, and unprocessed radar data of these frames follows:

2.4.3.1 Telemetry Data Processing

Reference to the subframe formats of Table 2-2 shows that a first subframe of either computed or unprocessed radar data contains 72 bits of telemetry information which is processed according to the procedure described below.

The set of three seven-bit-sequences of repeated signals of the subframe is first examined to insure (a) that not each of the seven bits of the sequence is a zero, (b) that both the abort and orbit signal bits are not set to one, and (c) that the three seven bit sequences are identical.

TABLE 2-2. MERCURY CONTROL CENTER (TELEMETRY EVENT TRANSMITTING BUFFER) TO HIGH-SPEED RETRANSMITTERS, MESSAGE FORMAT

Bits	Quantity	Comment
1–20	Capsule elapsed time reading (Binary–coded decimal - as follows) Bits 1 – 2 Tens of hours Bits 3 – 6 Units of hours Bits 7 – 9 Tens of minutes Bits 10 – 13 Units of minutes Bits 14 – 16 Tens of seconds Bits 17 – 20 Units of seconds	
21–40	Retrofire mechanism setting (Binary–coded decimal - as follows) Bits 21 – 22 Tens of hours Bits 23 – 26 Units of hours Bits 27 – 29 Tens of minutes Bits 30 – 33 Units of minutes Bits 34 – 36 Tens of seconds Bits 37 – 40 Units of seconds	
41	Capsule elapsed time not valid	1=not valid
42	Retrofire mechanism setting not valid	1=not valid
43	Liftoff	1=has occurred
44	Staging	1=has occurred
45	Escape tower separated	1=has occurred
46	Tower escape rockets fired	1=has occurred
47	Capsule separated from sustainer	1=has occurred
48	One of three posigrades fired	1=has occurred
49	Two of three posigrades fired	1=has occurred
50	Three of three posigrades fired	1=has occurred
51	Abort sequence initiated	1=has occurred
52	Retro 1 fired	1=has occurred
53	Retro 1 and 2 fired	1=has occurred
54	Retro 1, 2, and 3 fired	1=has occurred
55	Spare	1=was commanded
56	Burroughs–GE data selected	0=Azusa/FPS-16 data selected

TABLE 2-2. MERCURY CONTROL CENTER (TELEMETRY EVENT TRANSMITTING BUFFER) TO HIGH-SPEED RETRANSMITTERS, MESSAGE FORMAT (cont'd)

Bits	Quantity	Comment
57	Liftoff signal manually reversed	1=was reversed
58	Staging signal manually reversed	1=was reversed
59	Escape tower separated signal manually reversed	1=was reversed
60	Tower escape rockets fired manually reversed	1=was reversed
61	Capsule separated from sustainer manually reversed	1=was reversed
62	One of three posigrades fired manually reversed	1=was reversed
63	Two of three posigrades fired manually reversed	1=was reversed
64	Three of three posigrades fired manually reversed	1=was reversed
65	Abort sequence initiated manually reversed	1=was reversed
66	Retro 1 fired manually reversed	1=was reversed
67	Retro 1 and 2 fired manually reversed	1=was reversed
68	Retro 1, 2, and 3 fired manually reversed	1=was reversed
69	Abort phase has started	1=has started
70	Orbit phase has started	1=has started
71	Spare	
72	Parity for previous 71 bits	1=An even number of 1's in the previous 71 bits.

If each of these criteria is satisfied by the subframe data, the repeated signals are stored; and the telemetry discrete word time tag is computed and stored. If, however, one of these criteria is not satisfied, the same testing is applied to the telemetry data of the subframe image; failure of the image to satisfy each of the repeated signal criteria causes the telemetry data to be rejected and the telemetry output locations, TMH2TM, TMHS2T, MCS709 and MCSGEB to remain unchanged.

The non-repeated signals of the telemetry data entries satisfying the repeated signal criteria are processed in the following manner. Parity of this telemetry data is determined and compared with transmitted parity. Parity disagreement causes a return without further processing, while parity agreement results in the storing of the posigrade and retrograde rocket signals, the setting of the selected source indicators, MCS709 and MCSGEB, and the generation of the table TMHS2T, ending the processing of the telemetry data.

2.4.3.2 Computed Data

Where the coded data identity of a first subframe indicates that the subframe contains computed data, the entries are transferred to temporary storage, the frame sequence indicator is set to one, and a return is made to Monitor to await the arrival of the remaining computed data entries in a second subframe.

Upon receipt of a first and second subframe sequence of computed data, the check sum of the message and, if necessary, of the image, are successively computed and compared with the transmitted checksums. The data contained in that block which first produces checksum agreement is then processed to produce the output table-TMH2DB. Failure to find a valid checksum causes the rejection of the computed data and the retention of the previous contents of TMH2DB.

A second subframe of computed data received with the frame sequence indicator set to 0 causes the rejection of the subframe and initiates the search for another computed data subframe sequence.

2.4.3.3 Radar Observation Processing

The processing of the radar observations of first and second subframes is identical and is executed on the reception of each radar subframe. This processing is described below in terms of the following procedures:

Entry Processing

Establishing Data Tables

Observation Processing

Moving Data Table

Entry Processing: On each radar entry, the coded identity of the transmitting radar contained in the valid ID word is first examined for indication of a change in radar source. If a different radar of the complex has begun transmission, the time entry of the data table, TMHRAE, is examined. A negative time entry indicates that editing and smoothing are in progress and, therefore, the data table is not to be disturbed. In this case, control is returned to Monitor without further processing. If, however, the table contains a positive time entry, the new radar source code is stored in MCSRRS and a zero is stored as the time entry of TMHRAE to indicate the absence of a valid data table. The contents of the time cell MCHST2 is then converted to .1 second increments and transfer is made to the processing of Establishing Data Table.

If, however, no change of radar source is indicated, MCHST2 is converted immediately and transfer is made to Observation Processing.

Establishing Data Table: A zero time entry in the table, TMHRAE, indicates that the table does not contain currently valid data. When a radar subframe is received with the table in this condition, the observations are processed in the following manner.

The two observations of the subframe are examined to insure that they are both "on track" observations, i.e., that the transmitting radar was actively tracking the capsule when the observations were read—and that both observations contain an elevation entry, E, where $3^{\circ} \leq E \leq 90^{\circ}$. If either of these requirements is not satisfied by either observation of the subframe, the subframe image is similarly tested. Failure of both the subframe and image to meet these two criteria results in the rejection of the radar data and an immediate return to Monitor.

However, the observations of those entries satisfying these standards are inserted as the initial entries of the data table. The time converted from MCHST2 is then stored in TMHRAE as the time tag of the first observation, and a normal return is made to Monitor.

Observation Processing: Once the data table, TMHRAE, has been established according to the above procedure, the observations of subsequent subframes showing no change in radar source are processed as outlined below.

The time of present entry, T_N, is compared to the time of last entry, T_L, where T_L and T_N represent the successive contents of MCHST2 expressed in .1 second increments. If $T_N - T_L > 2$, k observations are extrapolated and inserted into TMHRAE, where $k = T_N - T_L - 2$; the observations of the received subframe are then processed. The coordinates of the inserted observations are extrapolated according to the formula $C_i = 2C_{i-1} - C_{i-2} + M$, where $i = 1, 2 \ldots k$; $M = 5000$, for $C_i = R_i$; and $M = .025$, for $C_i = A_i$ and $C_i = E_i$.

If, however, $T_i - T_L \leq 2$, the two observations of the subframe are processed immediately.

The first of the two received observations is examined in the subframe and, if necessary, in the subframe image. The entries of the data first indicating an on-track observation are scaled and stored in successive locations of TMHRAE. If the elevation entry is such that $3^o > E > 90^o$, it is replaced with a value extrapolated according to the above formula. If an on-track observation is found in neither the subframe nor image, the coordinates are again extrapolated as above. Upon completing the processing of the first observation, the second of the received observations is processed in an identical manner.

The location, TMHEDP, containing the location of the range entry of the last observation stored by I0HS09 is updated as each observation—either received or extrapolated—is stored.

It is to be noted that the datatable, TMHRAE, contains an observation for every .1 second increment of elapsed time after the time of the first observation as stored in location TMHRAE. There are, then, no missing data exits from this processing after the table has been established.

Moving the Data Table: After the storing of each observation in the table, TMHRAE, the location, TMHEDP, is updated and examined. If, according to this storage index, fewer than 100 valid observations appear in the table, processing continues as outlined above. If, however, there are 100 or more observations in the table, the program attempts to move the table to accommodate more recent data.

The time entry of TMHRAE is first examined. If, according to a negative time entry, edit and smoothing are in progress, a move suppression indicator is set and this data to be processed is stored in the move buffer, locations TMHRAE + 301 through TMHRAE+360, provided there are free locations in this buffer. A return is made to Monitor without further processing if this buffer is also full.

However, when the location, TMHRAE, contains a positive non-zero time entry, the data table is moved. The most recent set of edited observations as indicated by the table of edit parameters—not to exceed 31 in number—is moved to the beginning of the data table, the 7 parameters of TMHEDP and the time entry of TMHRAE are adjusted to reflect this new positioning and the appropriate observation processing is resumed.

2.4.4 Usage

a) Space Required:

The program I0HS09 requires 976 core storage locations. The storage required by the input and output locations of paragraph 2 above are not included in this figure and require an additional 435 locations.

b) Constants:

K00000	DEC	0
K00001	DEC	1
K00002	DEC	2
K00003	DEC	3
K00031	DEC	31
K00060	DEC	60
K00063	DEC	63
K00080	DEC	80
K00120	DEC	120
K010.0	DEC	10.0
KCH232	OCT	232 000 000 000
KCH233	OCT	233 000 000 000

c) Time Required—The running time for I0HS09 is approximately 10 milliseconds.

d) Checkout—The I0HS09 program has been operational in the Mercury Program system since October of 1960.

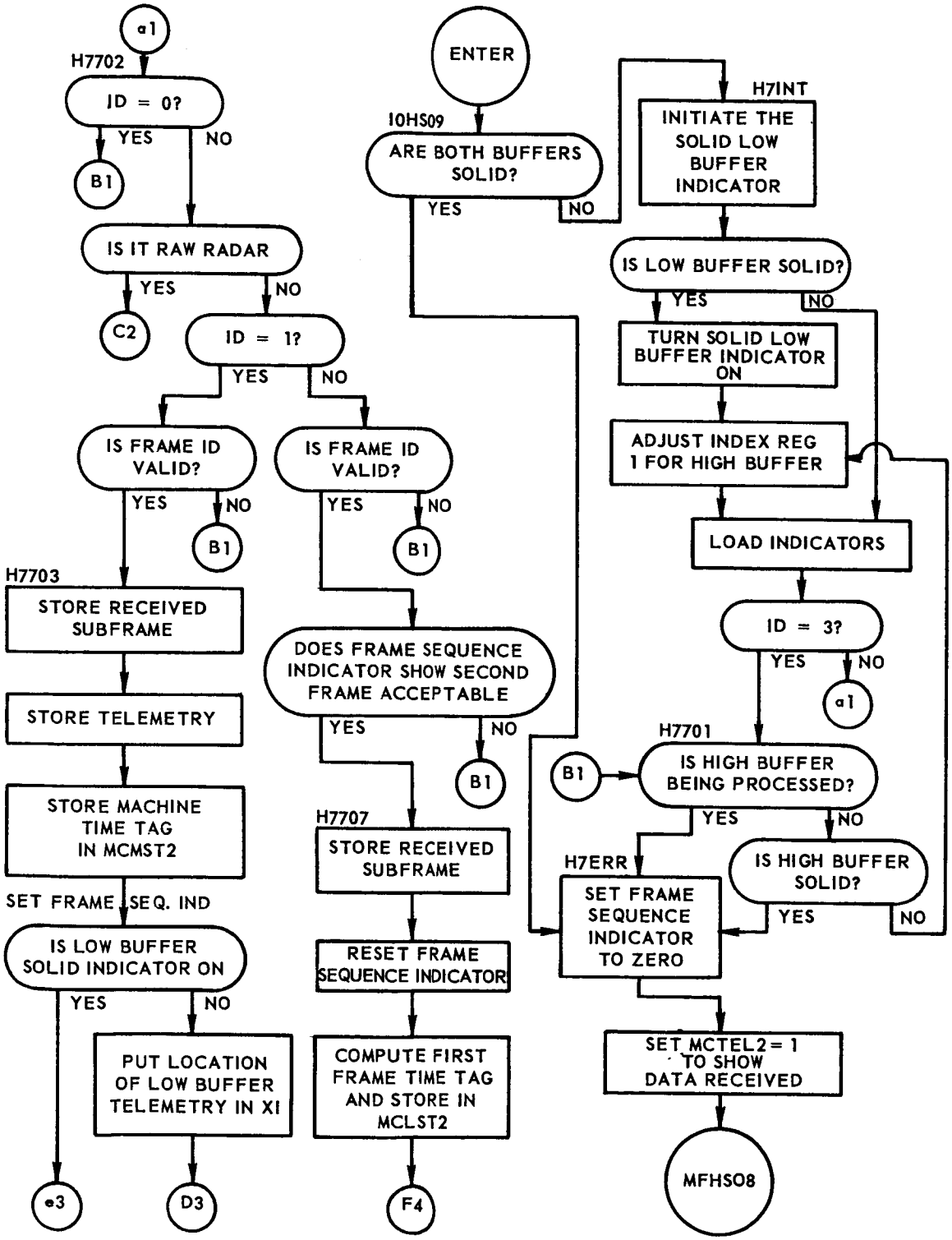


FIGURE 2-6. 10HS09 PROGRAM FLOW CHART *(Sheet 1 of 10)*

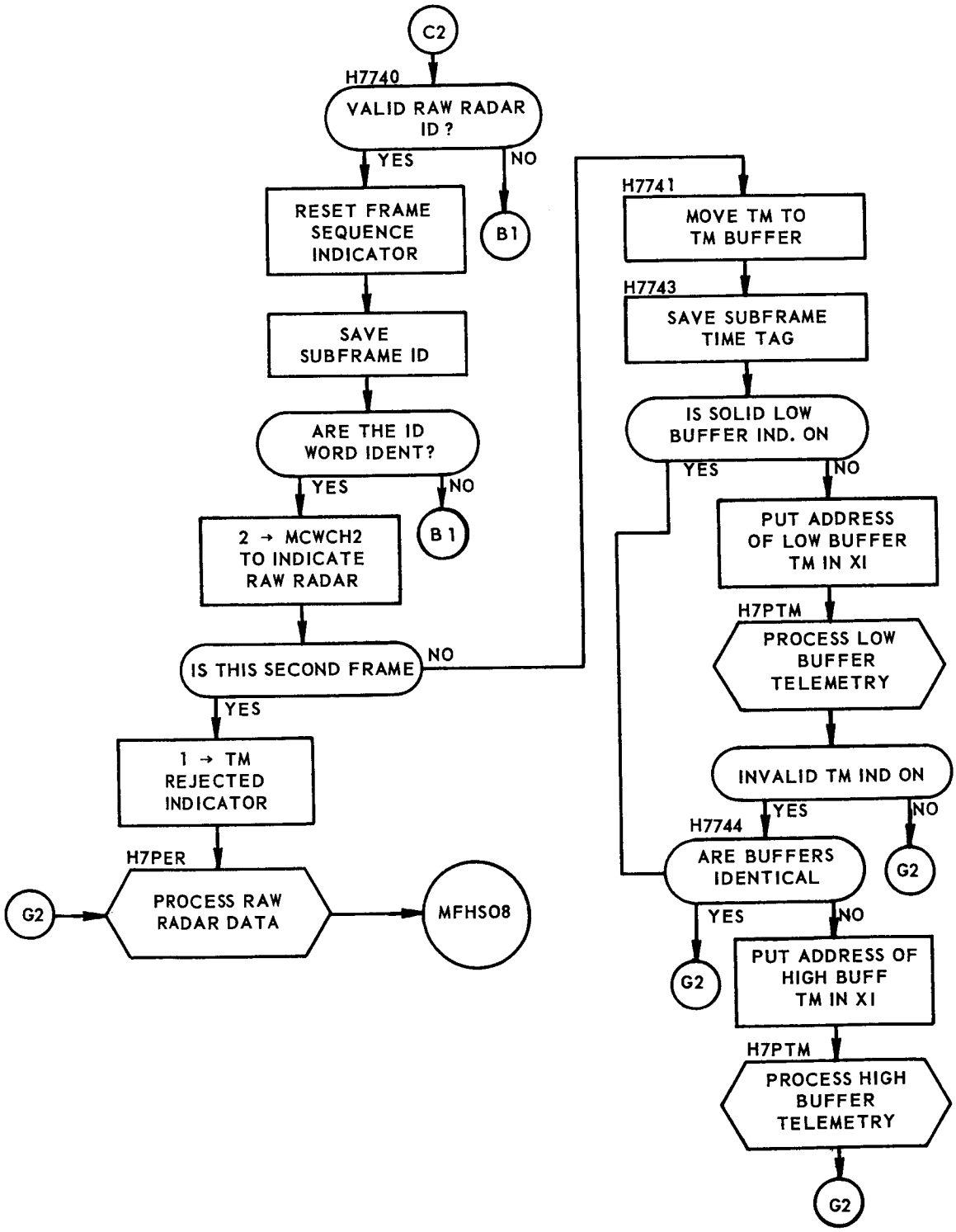


FIGURE 2-6. 10HS09 PROGRAM FLOW CHART *(Sheet 2 of 10)*

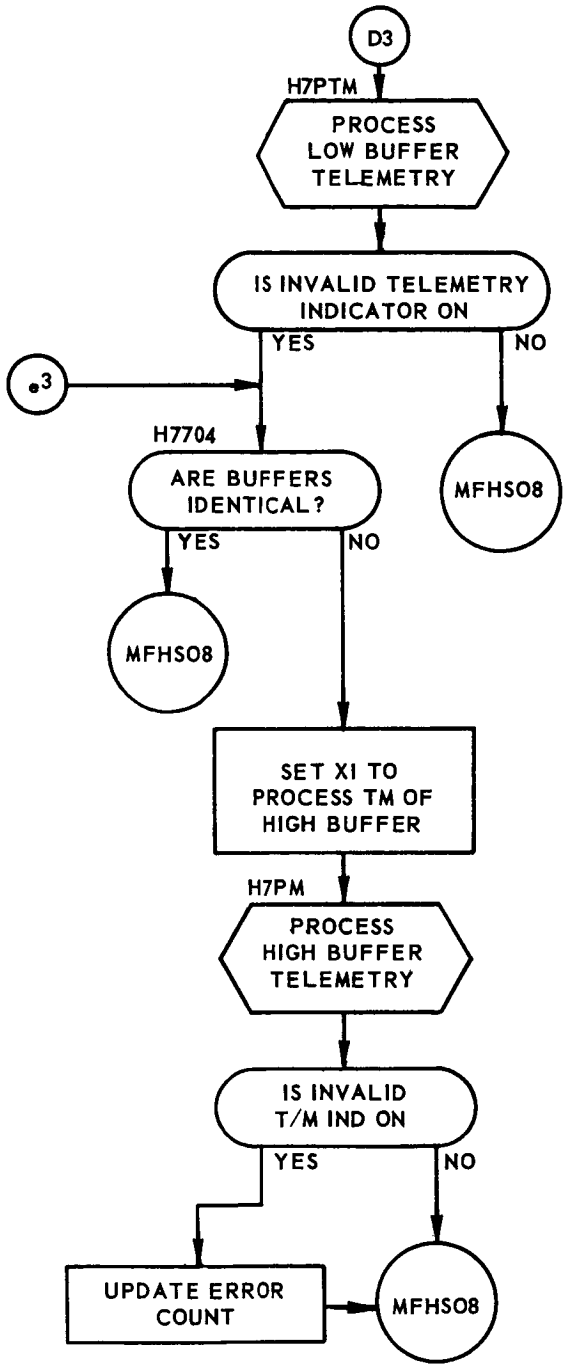


FIGURE 2-6. 10HSO9 PROGRAM FLOW CHART *(Sheet 3 of 10)*

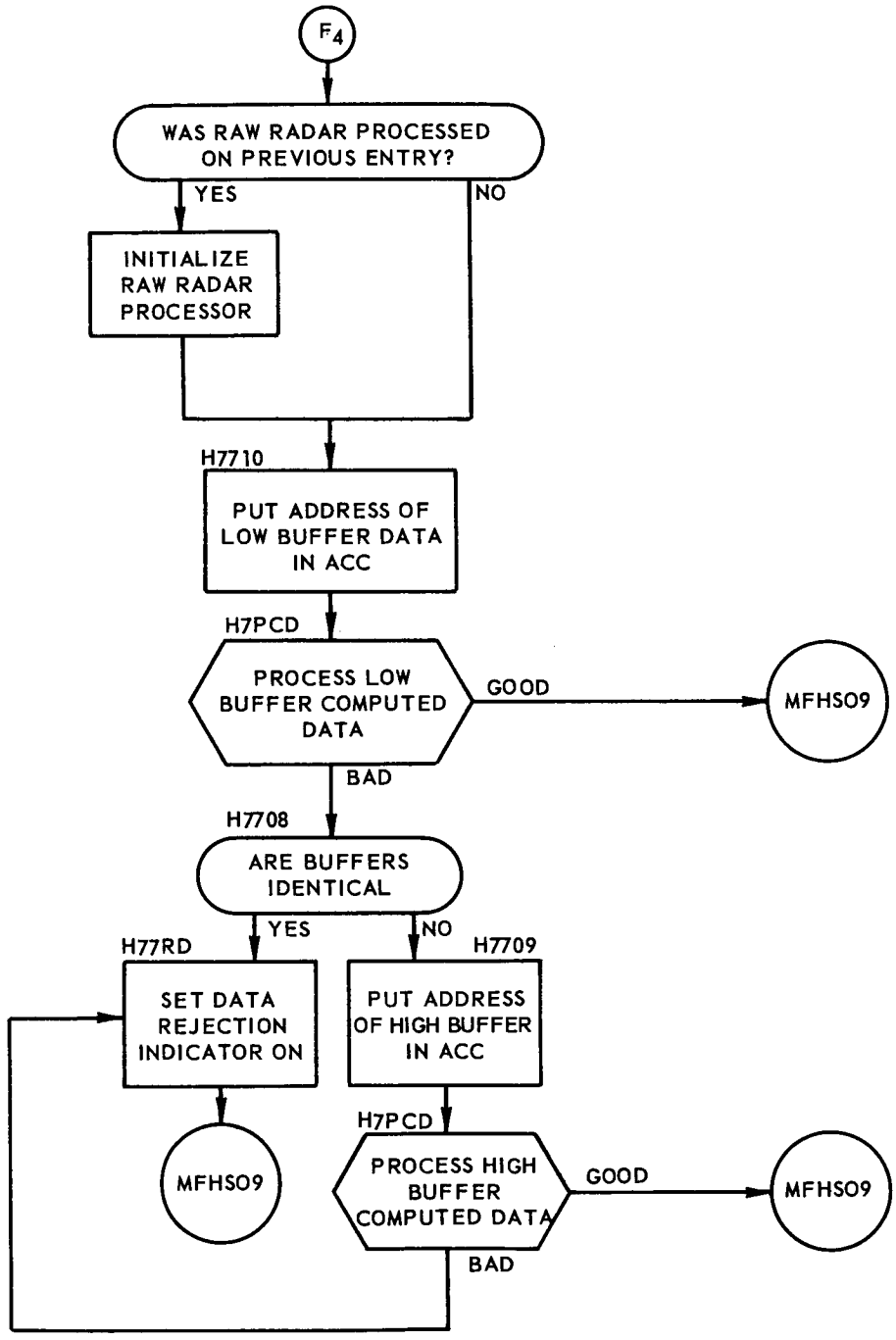


FIGURE 2-6. 10HSO9 PROGRAM FLOW CHART *(Sheet 4 of 10)*

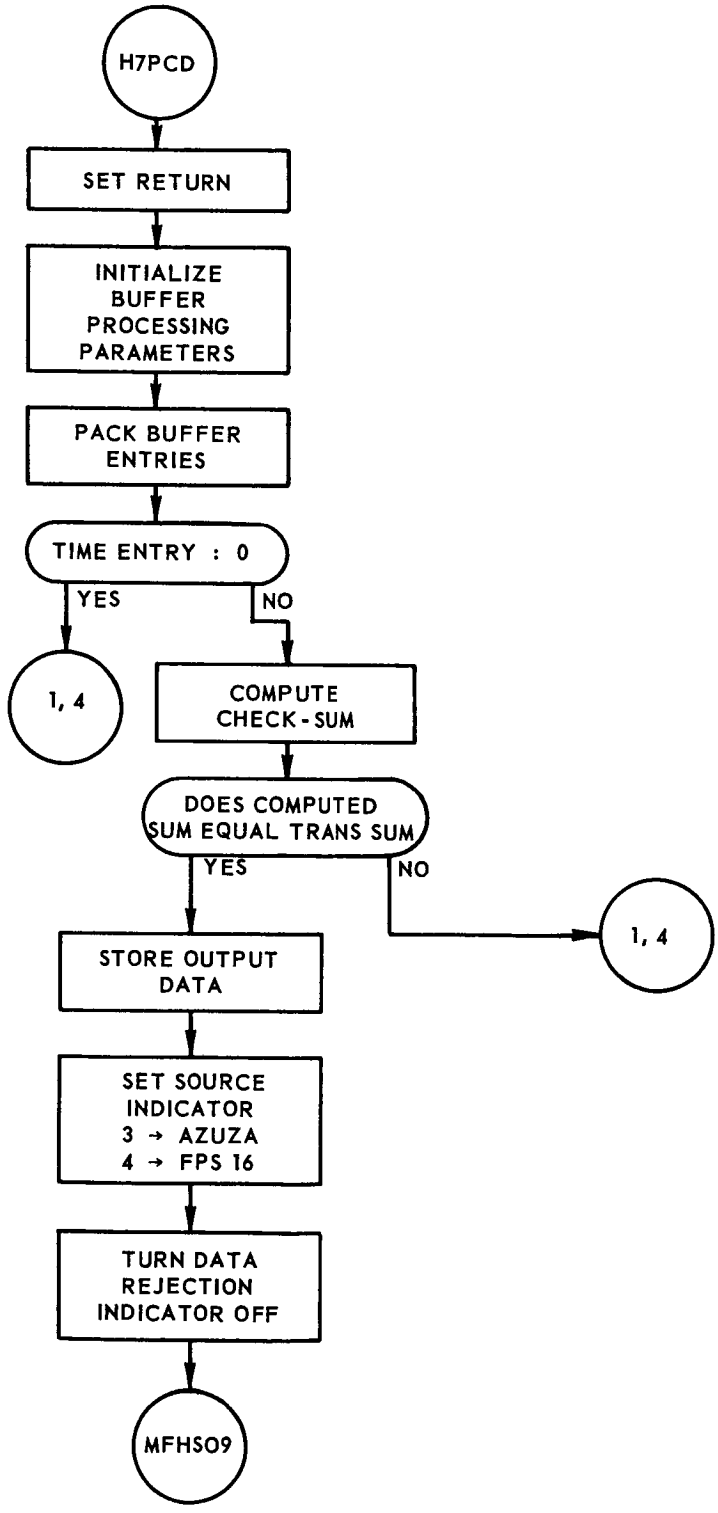


FIGURE 2-6. 10HS09 PROGRAM FLOW CHART *(Sheet 5 of 10)*

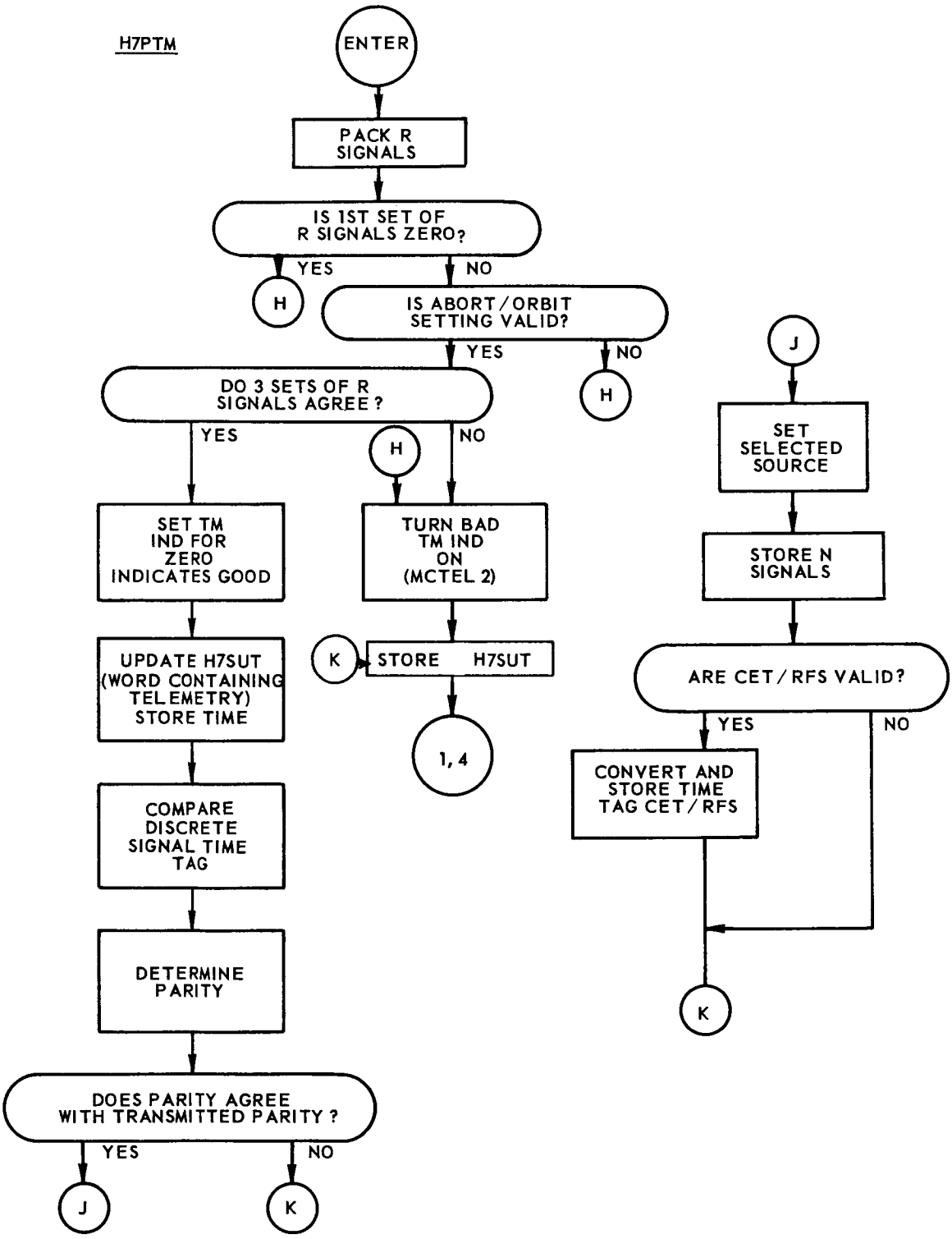


FIGURE 2-6. 10HS09 PROGRAM FLOW CHART *(Sheet 6 of 10)*

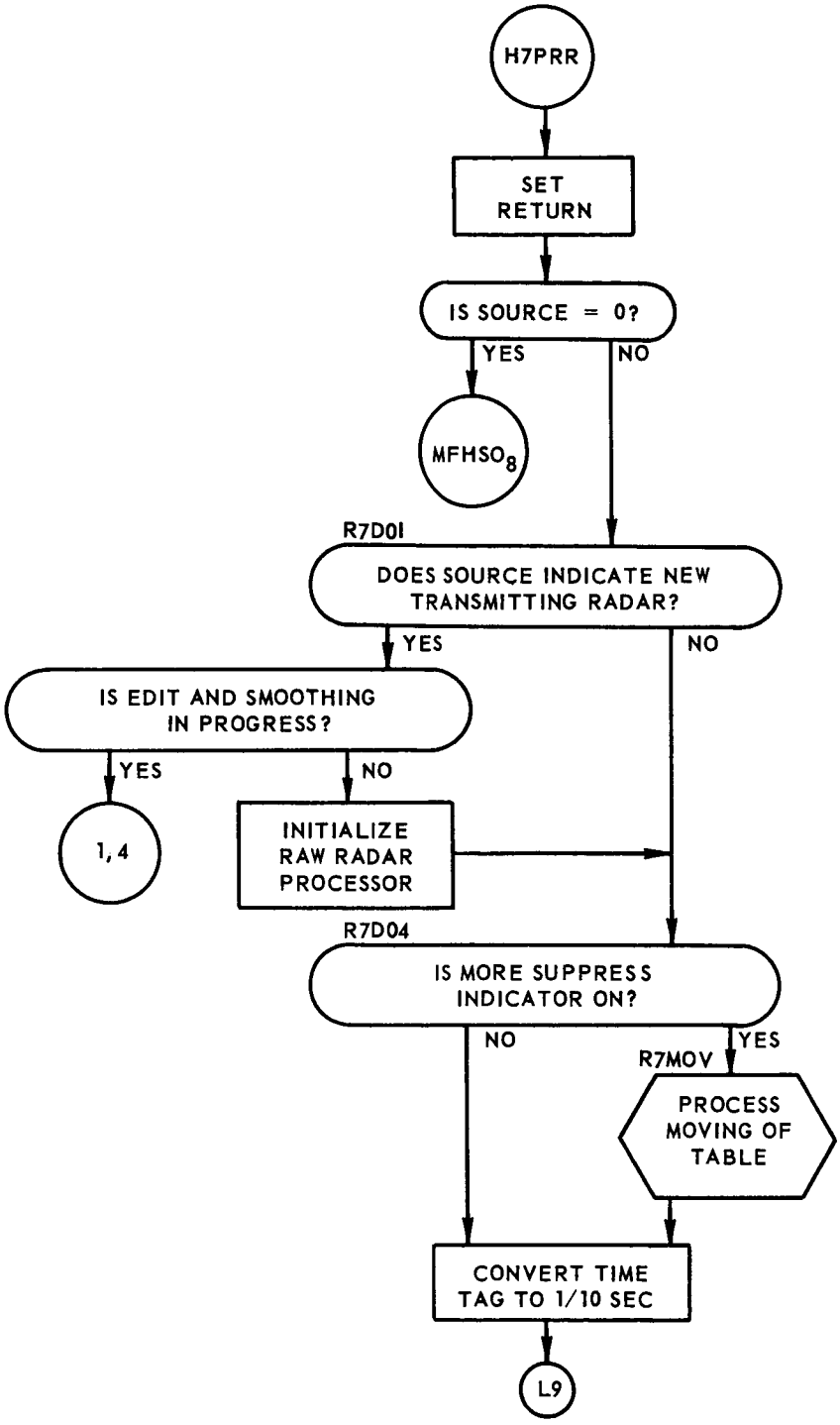


FIGURE 2-6. 10HS09 PROGRAM FLOW CHART *(Sheet 7 of 10)*

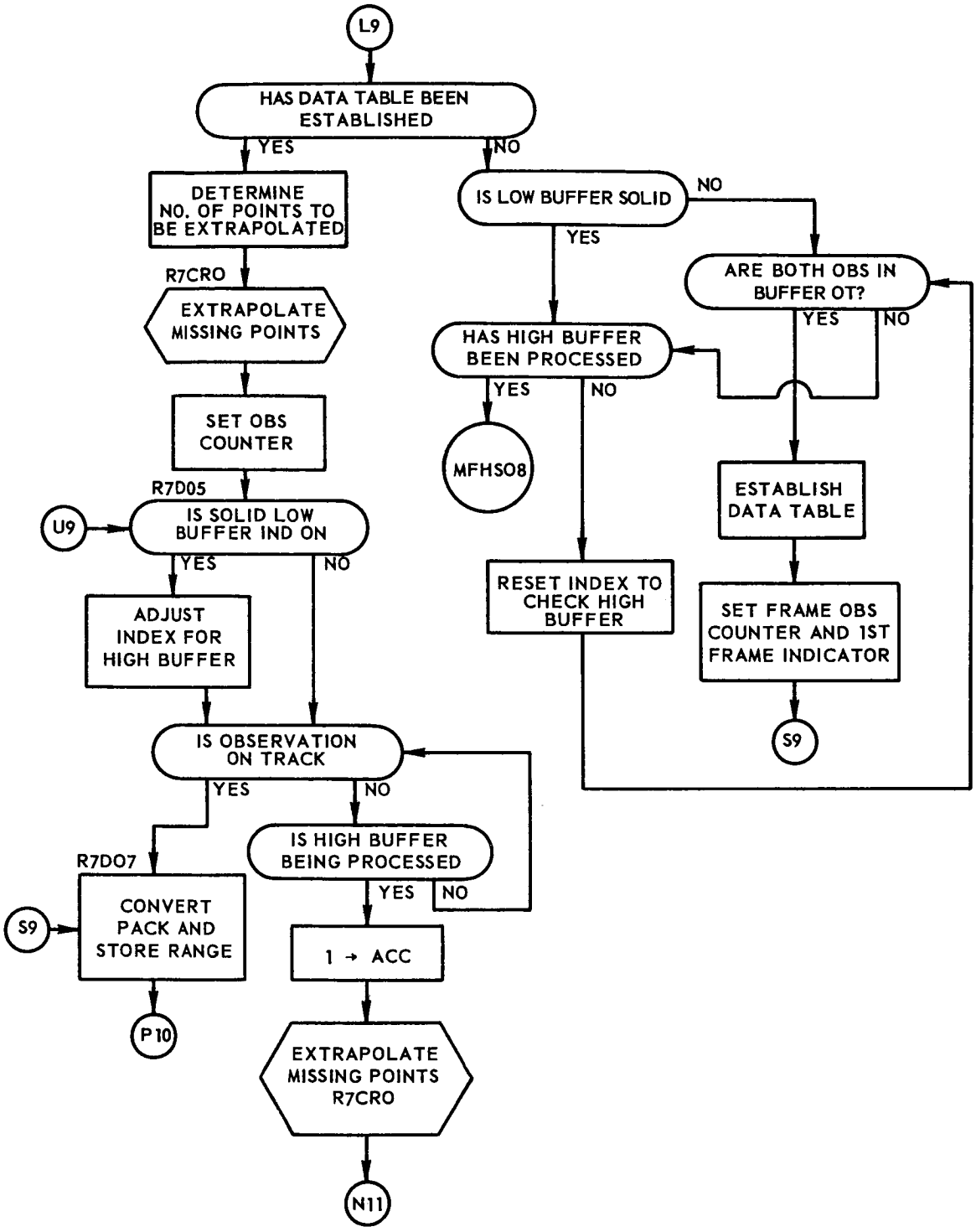


FIGURE 2-6. 10HS09 PROGRAM FLOW CHART *(Sheet 8 of 10)*

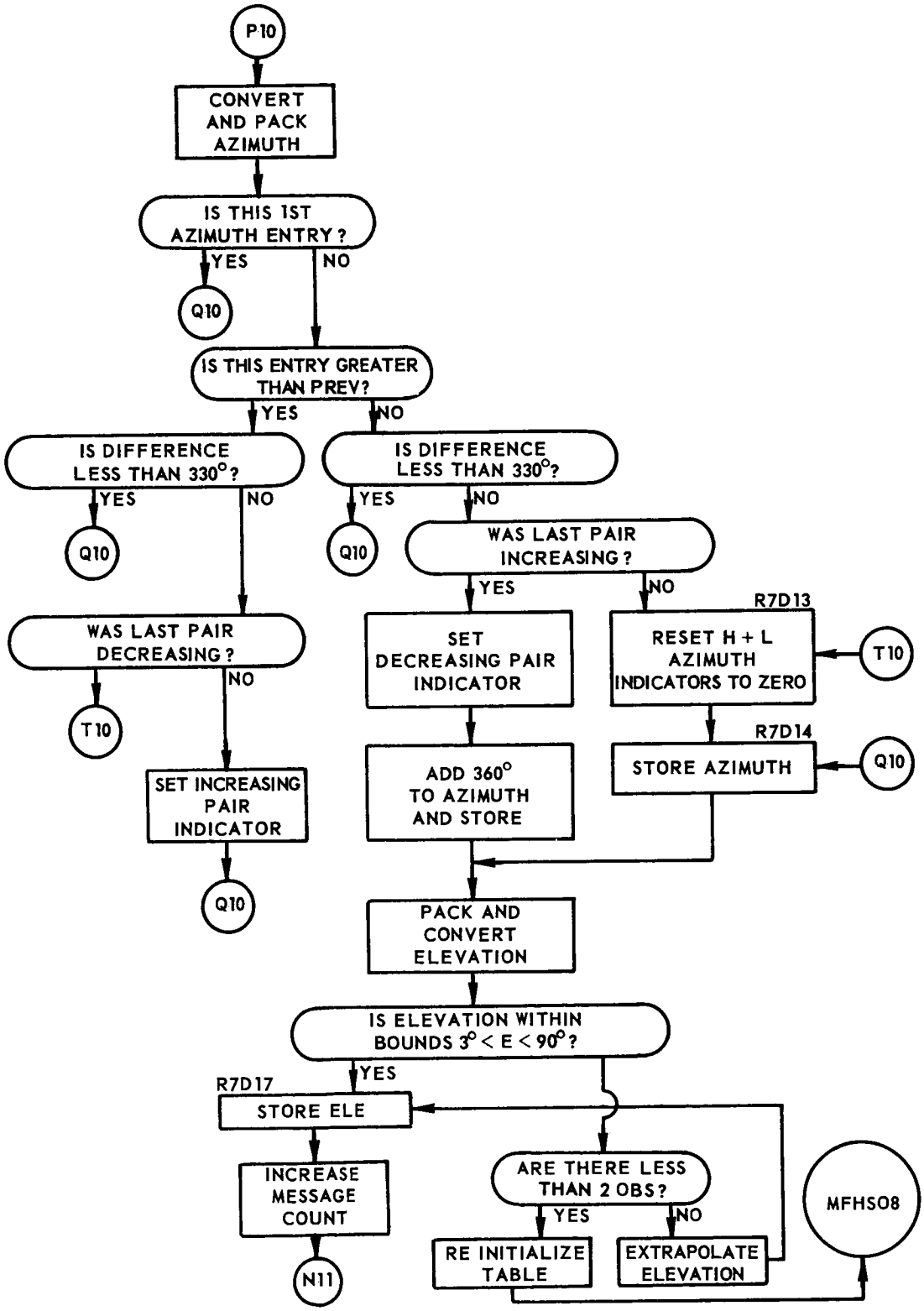


FIGURE 2-6. 10HS09 PROGRAM FLOW CHART *(Sheet 9 of 10)*

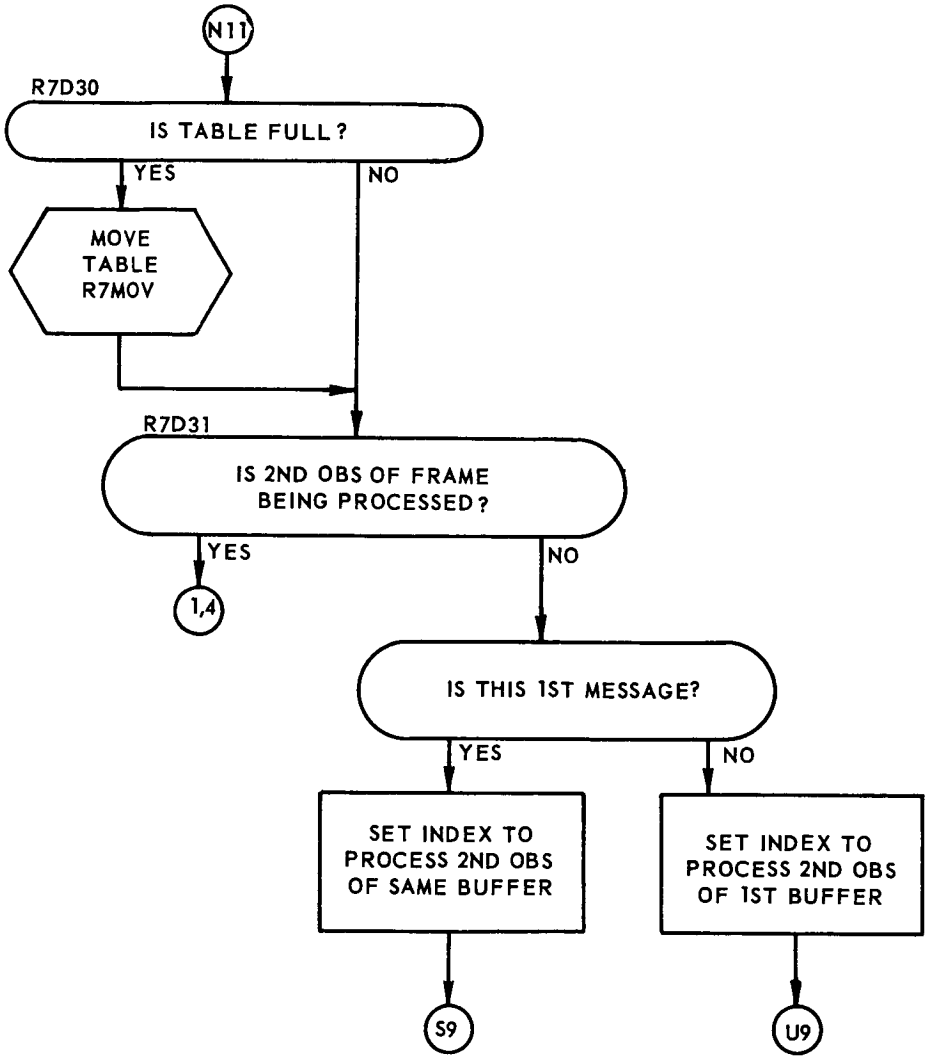


FIGURE 2-6. 10HS09 PROGRAM FLOW CHART *(Sheet 10 of 10)*

8-Bit Word Transfers from Data Receiver
First Subframe

1	1	Telemetry				8
2	9	Telemetry				16
3	17	Telemetry				24
4	25	Telemetry				32
5	33	Telemetry				40
6	41	Telemetry				48
7	49	Telemetry				56
8	57	Telemetry				64
9	65	Telemetry				72
10	1	A				8
11	9	A				16
12	17	A				24
13	25	A				32
14	33	A	36	1	B	4
15	5	B				12
16	13	B				20
17	21	B				28
18	29	B				36
19	1	C				8
20	9	C				16
21	17	C				24
22	25	C				32
23	33	C	36	0 0 0		0
24	0 0 0			1	ID*	5

8-Bit Word Transfers from Data Receiver
Second Subframe

1	1	D				8
2	9	D				16
3	17	D				24
4	25	D				32
5	33	D	36	1	E	4
6	5	E				12
7	13	E				20
8	21	E				28
9	29	E				36
10	1	F				8
11	9	F				16
12	17	F				24
13	25	F				32
14	33	F	36	1	N	4
15	5	N				12
16	13	N				20
17	21	N				28
18	29	N				36
19	1	Checksum Σ				8
20	9	Checksum Σ				16
21	17	Checksum Σ				24
22	25	Checksum Σ				32
23	33	Σ	36	0 0 0		0
24	0 0 0			1	ID*	5

The above constitutes a complete message frame and is transmitted every 400 milliseconds. Each subframe consists of 192 serial bits preceded by a sync signal. The quantities represented by A, B, C, D, E, F, and N are restricted information and are specified in other documents.

***See Note 1 for makeup of ID word.**

FIGURE 2-7. IP 7090 DATA, CAPE CANAVERAL TO GODDARD, MESSAGE FORMAT (SHEET 1 OF 2)

See Note 2 for format of bits 1 to 72 of first subframe in absence of telemetry data and Note 3 for format in absence of quantities A, B, C, D, E, F, and N.
See table 2-15 for format of telemetry data.

NOTES

1. The 5-bit identity (ID) word in each subframe conveys the following information

   Data from IP 7090 High-Speed Buffer and Retransmitter

   Bit 1: A 0 signifies IP 7090 data format.
   A 1 signifies raw radar format.
   Bit 2: A 1 signifies second subframe.
   Bit 3: A 1 signifies first subframe.
   Bit 4 and 5: Indicate source of raw radar data.

2. In the event that no data is received from the telemetry event transmitting buffer, the high-speed buffer and retransmitters are arranged to transmit 0's in the bit positions occupied by telemetry event data bits 1 through 40 and 43 through 72. Ones are transmitters in positions 41 and 42, resulting in erroneous parity for the telemetry event data message.

3. In the absence of data quantities A, B, C, D, E, F, and N, the complete message frame is transmitted every 400 milliseconds. Telemetry data continues to be transmitted. Zeros with 1's interspersed in certain positions are transmitted in place of the missing data quantities. The following bits appear as 1's in this event:

Subframe	8-Bit Word No.	Quantity	Bit within Quantity
1	12	A	24
1	15	B	12
1	18	B	36
1	21	C	24
2	3	D	24
2	6	E	12
2	9	E	36
2	12	F	24
2	15	N	12
2	18	N	36
2	19	Checksum	1

FIGURE 2-7. IP 7090 DATA, CAPE CANAVERAL TO GODDARD, MESSAGE FORMAT (SHEET 2 OF 2)

2.5 SUMMARY

To process and prepare for entry into the Goddard IBM 7090 computers the huge amounts of data received from Cape Canaveral and the worldwide tracking sites during a Mercury mission, the following input programs were designed and are now operational in the Mercury Program System:

a) Low-Speed Teletype Input Program (I0TTIN)—processes capsule positional radar data received at Goddard via the 16 input teletype lines from the remote radar sites.

b) Manual Input Program (I0MANI)—converts specified data from teletype code input, as received on the 16 input lines, to the format required by the processing programs. The data is punched on paper tape by teletype machines and is inserted manually via the Data Communications Channel subchannel reserved for paper tape input.

c) High-Speed B-GE Input Program (I0HSGB)—processes Burroughs computer data and the associated telemetry data transmitted from the Mercury Control Center to Goddard over the high-speed lines.

d) High-Speed IP 7090 Input Program (I0HS09)—processes IP 7090 data and associated telemetry data transmitted from the Mercury Control Center via the high-speed duplexed lines.

Outputs from these programs are stored in computer memory for use by the processors described in this document.

Section 3
LAUNCH/ABORT PROCESSORS

During the launch/abort phase of a Mercury mission the IBM 7090 computers at Goddard receive pertinent launch vehicle-capsule flight data from the Mercury Control Center. The Goddard computers process this data and retransmit it for display at the Control Center. From this and other data, the Mercury Control Center is then equipped to make the all-important GO or NO-GO decision.

Because the Goddard computers are required to receive, process, and transmit this vast amount of data during the relatively short (approximately 5 minutes) duration of a normal launch, a general discussion is first presented on the interrelationship of the programs that are used during this phase. This discussion is then followed by the individual program write-ups.

The main processors, their functions, and their use by the Mercury Program System during the launch/abort phase are shown in Table 3-1.

3.1 PROGRAM SYSTEM OPERATION DURING LAUNCH

All input to the programs used during the launch/abort phase is made on two high-speed input lines from Cape Canaveral. Line 1 carries B-GE data. Line 2 carries IP 7090 data or AN/FPS-16 raw radar data. Both lines will also carry a bit indicating which line is the selected source for Goddard. The choice of which line is to be the selected source is made by the Data Quality Monitor operator located at Cape Canaveral. The operator can also select that raw radar data be sent over line 2. Figure 3-1 depicts the organization of the launch program from lift-off up to Hold. Figure 3-2 illustrates the program organization during Hold. In both figures the launch programs are divided into three levels of processing:

a) First level—Coordinate conversion and raw radar data editing and smoothing.

b) Second level—Gross editing and outputs for the Data Quality Monitor strip charts. (Not used during Hold or Abort)

c) Third level—Main processing to compute outputs for plotboards and digital displays.

The following discussion presents the interrelationship of the main launch programs used during powered flight when either line 1 or line 2 is selected. The

TABLE 3-1 LAUNCH/ABORT MAIN PROCESSORS

Program	Function	WHEN USED		
		Launch Prior To Hold	Launch During Hold	Abort
CCGEB1	Co ordinate conversion of B-GE data	X		
CC7091	Co ordinate conversion of IP 7090 data	X	X	X
CCRAWR	Raw Radar edit and co ordinate conversion	X	X	X
CCSTGE	Strip chart program for B-GE data	X		
CCSTIP	Strip chart program for IP 7090 data	X		
CCST16	Strip chart program for AN/FPS-16 data	X		
CCMAIN	Launch phase (before Hold) main calculation program	X		
CCHOLD	Launch phase (during Hold) main calculation program		X	
CCMISS	Missing data computation during launch (before Hold)	X		
CCHOMI	Missing data computation during launch (after Hold)		X	
CCRTYL	Low abort - below 100,000 feet.			X
CCMEAB	Medium Abort - above 100,000 feet and prior to TOWS			X
CCABRT	High Abort - after TOWS and after the abort switch is set at Cape Canaveral			X

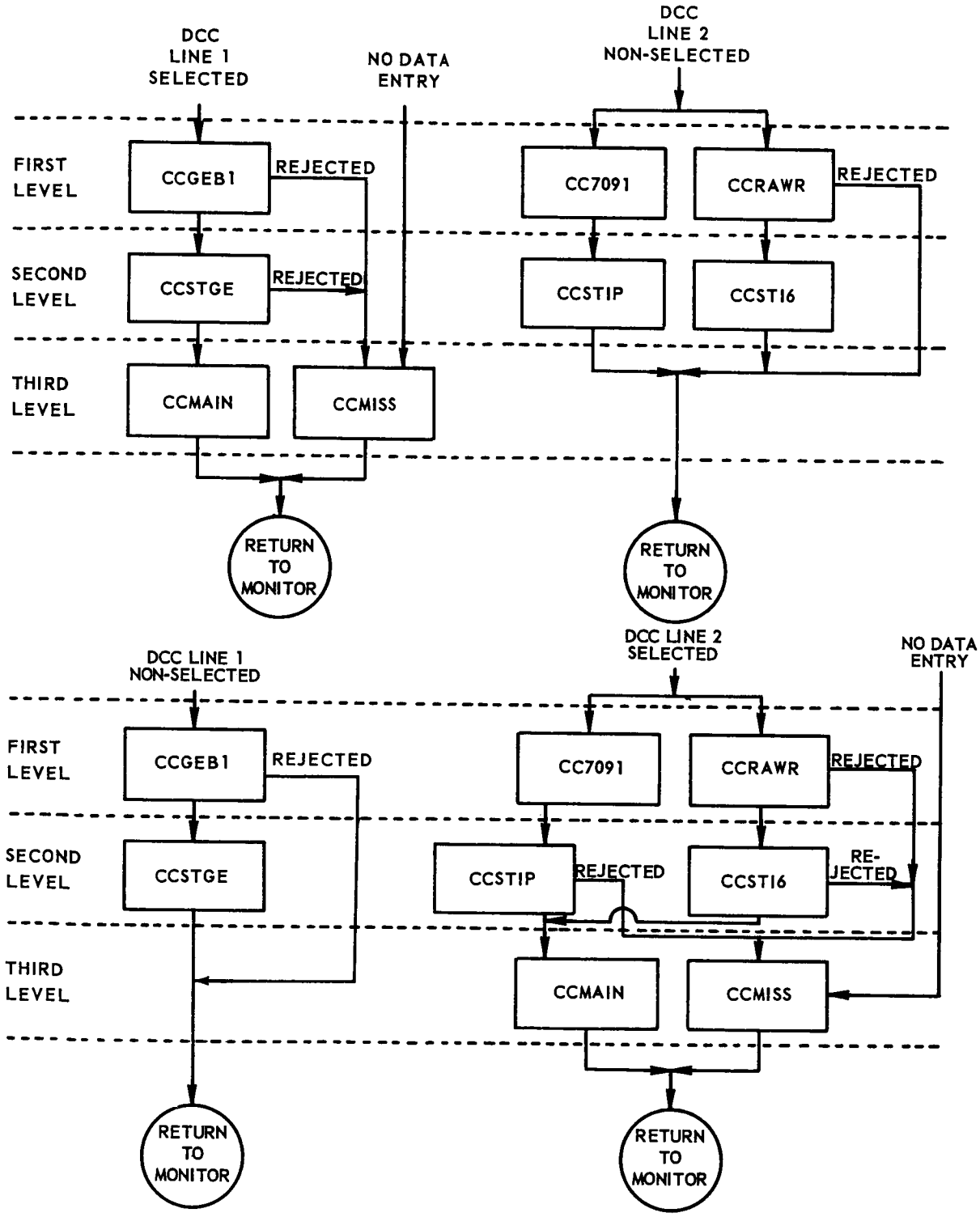


FIGURE 3-1. LAUNCH PROGRAM ORGANIZATION (FROM LIFT-OFF UP TO HOLD)

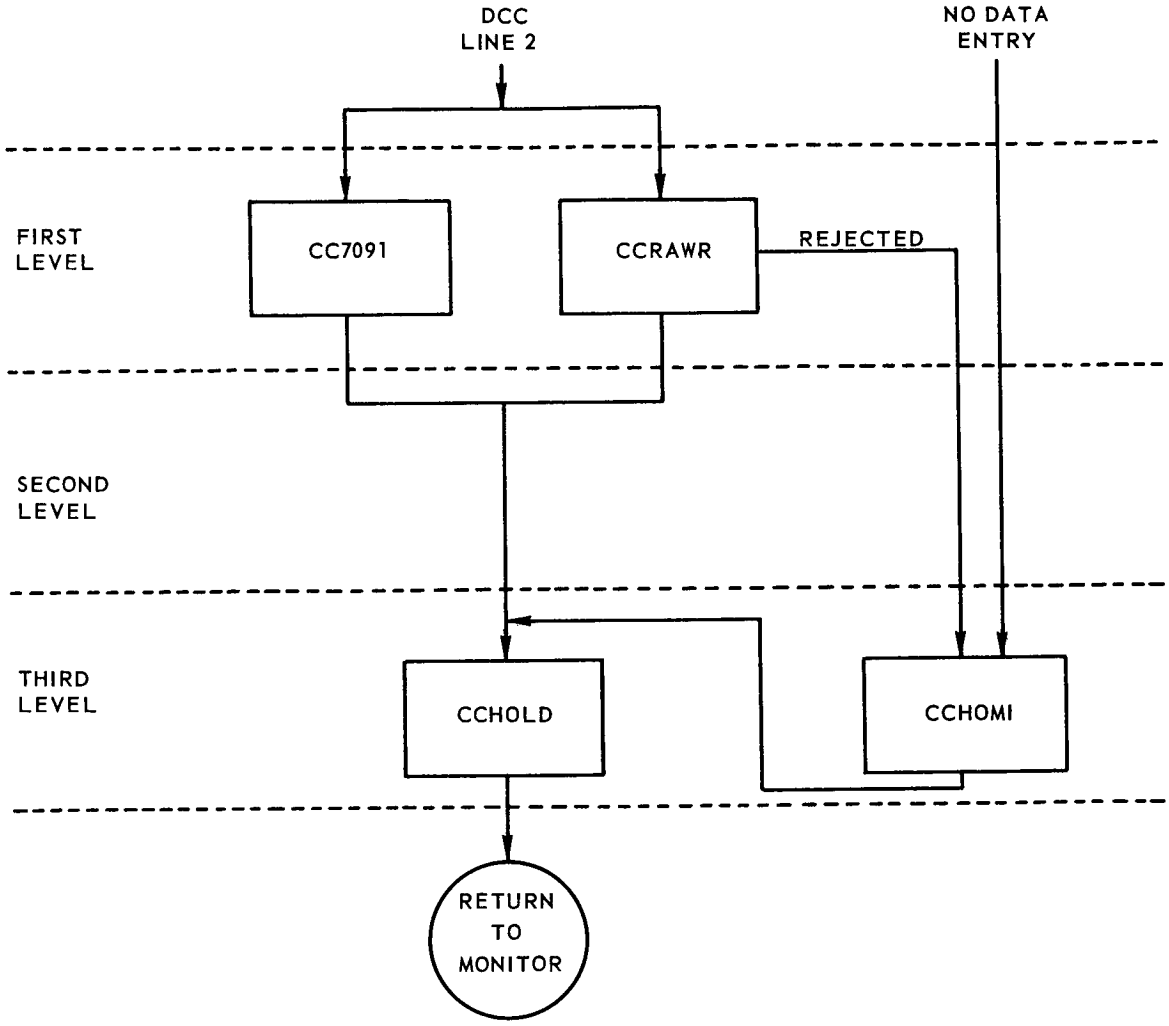


FIGURE 3-2. LAUNCH PROGRAM ORGANIZATION (DURING HOLD)

discussion on the programs used and their organization should an abort situation occur any time before insertion is included in Subsection 3-2.

3.1.1 DCC Line 1 - Selected

Under normal operating conditions, DCC line 1 is the selected source (upper half of Figure 3-1). The unselected data entering line 2 is processed in the first two levels only, control is then returned to Monitor.

Launch vehicle position and velocity vectors and time since lift-off from the B-GE system enter the Goddard computers and the first level of processing via DCC line 1. These parameters are converted to the Mercury coordinate system by CCGEB1. The output of CCGEB1 together with the time enters the second level of processing which is the strip chart program CCSTGE. The primary purpose of this processor is to compute the deviation of the velocity and flight path angle from their respective nominals for display on the strip charts at Cape Canaveral. If the data entering CCGEB1 or CCSTGE is of poor quality the data is rejected and control is passed to the Missing Data During Launch program, CCMISS in the third level, bypassing the second level processor. Therefore, only selected and good data is entered into the Main Calculations During Launch program, CCMAIN in the third level. The main function of CCMAIN is to compute the GO, NO-GO recommendation for transmission to Cape Canaveral. Control is then returned to Monitor.

3.1.2 DCC Line 2 - Selected

When line 2 is the selected source (lower half of Figure 3-1) the unselected data entering on line 1 is processed in the first two levels only. Control is then returned to Monitor.

IP 7090 launch vehicle position and velocity vectors and time since lift-off enter the Goddard computers and high-speed input processor (I0HS09) via DCC line 2. The data is processed by the input program and is transferred to the first level of launch processing where the data is converted into the Mercury coordinate system by CC7091. If range, azimuth, and elevation raw radar data is entering on line 2, CCRAWR performs an editing, smoothing, and coordinate conversion on the data.

The smoothed data together with time enters the second level of processing and the strip chart processors CCSTIP or CCSTI6. Here, the deviation of the velocity and flight path angle from their respective nominals is computed for display on the Data Quality Monitor strip charts. If the data is found to be of poor quality then CCSTIP or CCSTI6 rejects the data and passes it on to CCMISS in the third level. If the data is acceptable it is passed to CCMAIN. Here again, as in Line 1 - Selected, only selected and good data is processed by CCMAIN in the third level.

3.2 PROGRAM SYSTEM OPERATION DURING HOLD

When CCMAIN or CCMISS have completed their computations at the end of powered flight the Mercury Program System shifts into the Hold (or Abort) portion of the launch phase. The time period from the end of powered flight to the Flight Dynamics Officer's ORBIT, ABORT decision is called the Hold phase. The purpose of Hold is to provide the Flight Dynamics Officer at Cape Canaveral with sufficient time to analyze the flight data and accept or reject the computed GO, NO-GO recommendation. If, at any time during Hold, the orbit signal is received at Goddard then the Mercury Program System shifts into the orbit phase (see Section 4). If the abort signal is received then the program system shifts into the High Abort program, CCABRT (see Subsection 3.3).

The Mercury Program System organization used during the Hold portion of the launch phase is shown in Figure 3-2. IP 7090 smoothed radar data or AN/FPS-16 raw radar data enters the Goddard computers via DCC line 2. (The data on line 1 will be considered unreliable since, during Hold, the launch vehicle and the capsule are proceeding along different flight paths.)

The capsule position and velocity vectors and the time tag are converted to the Mercury coordinate system by CC7091 in the first level and control is passed directly to the Main Calculation During Hold program (CCHOLD) in the third level. (CC7091 cannot reject the IP 7090 data.) If input to the first level is raw radar then CCRAWR performs an edit smoothing and conversion function on the data and passes control to CCHOLD in the third level. Unlike the IP 7090 smoothed radar data processor (CC7091), CCRAWR will reject the data if it finds the information unreliable and then passes it on to the Missing Data During Hold program (CCHOMI).

3.3 PROGRAM SYSTEM OPERATION DURING ABORT

The organization of the programs used should an abort occur anytime after lift-off is shown in Figure 3-3. Here, three abort processors, in addition to the coordinate conversion programs, are used by the program system depending on the height of the launch vehicle at the time of abort. The Low Abort (CCRTYL) program is executed if an abort occurs below 100,000 feet. If an abort occurs above 100,000 feet and before tower separation (TOWS) then the Medium Abort (CCMEAB) program is used. If an abort occurs after TOWS the High Abort (CCABRT) program is used.

In low or medium abort, high speed smoothed or raw AN/FPS-16 radar entering Goddard on line 2 is processed by CC7091 or CCRAWR respectively. (The B-GE missile tracking data entering Goddard on line 1 will not be processed during abort.) The processed data serves as input to CCRTYL or CCMEAB in the third level (there is no second level of processing during abort).

During high abort the input to the CCABRT program is the last known position and velocity vectors from powered flight. The detailed program write-ups for the launch/abort phase are included in Subsections 3-4 through 3-22.

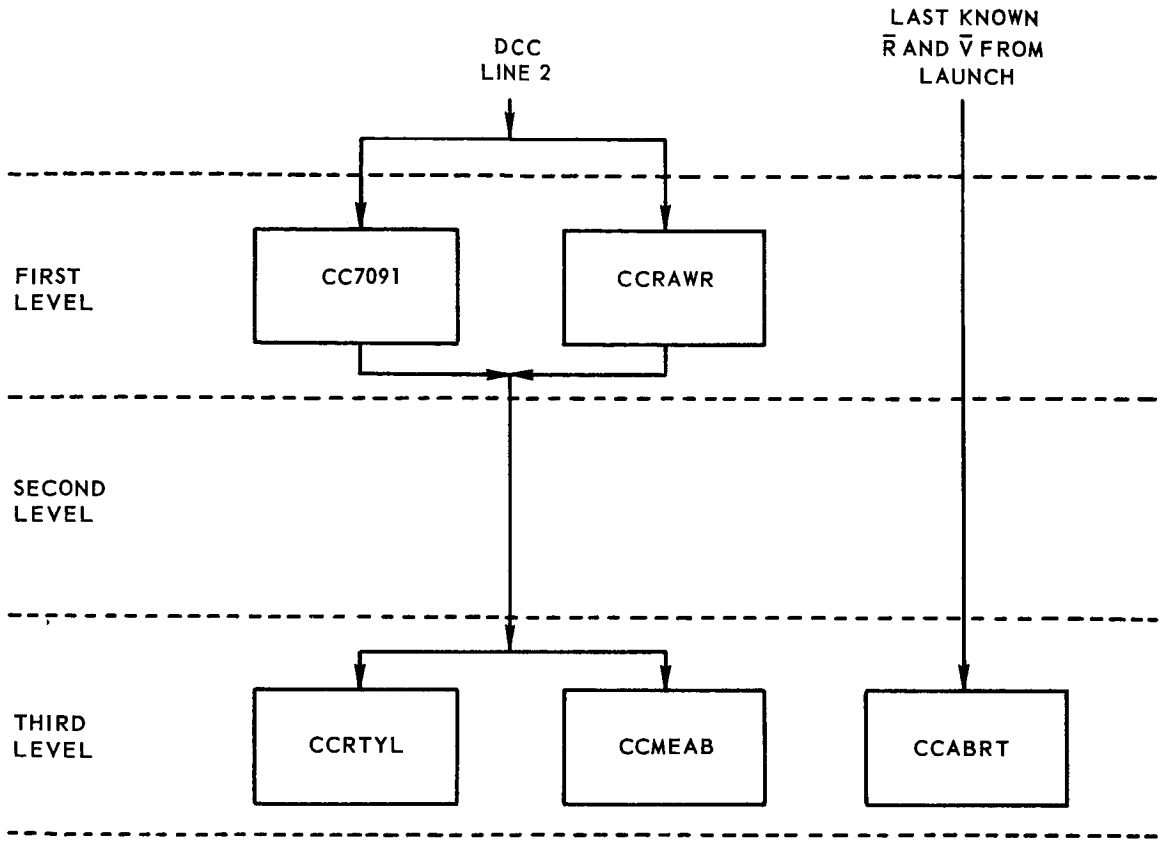


FIGURE 3-3. ABORT DURING LAUNCH PHASE PROGRAM ORGANIZATION

3.4 B-GE DATA CONVERSION PROGRAM (CCGEB1)

This is the first of three coordinate conversion programs used during launch. This program together with the following two programs (CC7091 and CCRAWR) forms the first level of processing for the launch phase. The purpose of CCGEB1 is to convert launch vehicle position and velocity data from a B-GE coordinate system to a Mercury coordinate system. This data together with an indicator set to whether this is a selected or non-selected source is entered to the Goddard computers via the high speed input line number 1. The flow chart for CCGEB1 is shown in Figure 3-4.

3.4.1 Input Requirements

This program uses the sine/cosine library subroutine U1SICO. The following data from the table, TMGEB1, serves as input to the CCGEB1 program:

TMGEB1	T_i (Time since lift-off in seconds)	
+ 1	X_1	$\bar{R}_{1G}$ (B-GE units)
+ 2	Y_1	
+ 3	Z_1	
+ 4	$\dot{X}_1$	$\bar{V}_{1G}$ (B-GE units)
+ 5	$\dot{Y}_1$	
+ 6	$\dot{Z}_1$	

Another input to CCGEB1 is the number of seconds from midnight to lift-off in floating point seconds. This information is contained in communication cell MCLFTM.

The additional information required is contained in the discrete word, TMGEDS+1, bits 30, 31, and 32. These bits are the locations of the data flags. If B-GE data is selected, CCGEB1 will process the data only if all three data flags (δ_2, δ_3, γ_N) are non-zero. If any one of these bits is zero, control passes to MFCCGE.

3.4.2 Output Requirements

The output of the CCGEB1 program consists of the following information in normalized floating point format and is stored in the table TCBEG1 which serves as one input to the second level of launch processing:

TCBEG1	T_i (Time since lift-off in seconds)	
+ 1	X_1	$\bar{R}_{1G}$ (B-GE units; same as input)
+ 2	Y_1	
+ 3	Z_1	
+ 4	$\dot{X}_1$	$\bar{V}_{1G}$ (B-GE units; same as input)
+ 5	$\dot{Y}_1$	
+ 6	$\dot{Z}_1$	
+ 7	X_3	$\bar{R}_3$ (Mercury units)
+ 8	Y_3	
+ 9	Z_3	
+10	$\dot{X}_3$	$\bar{V}_3$ (Mercury units)
+11	$\dot{Y}_3$	
+12	$\dot{Z}_3$	

3.4.3 Method

Convert $\bar{R}_{1G}$ and $\bar{V}_{1G}$ to $\bar{R}_3$ and $\bar{V}_3$ in B-GE units, by using the following formulas:

$$X_3 = X_{1G} \cos \alpha - Y_{1G} \sin \alpha$$

$$Y_3 = X_{1G} \sin \alpha - Y_{1G} \cos \alpha$$

$$Z_3 = Z_{1G}$$

$$\dot{X}_3 = \dot{X}_{1G} \cos \alpha - \dot{Y}_{1G} \sin \alpha$$

$$\dot{Y}_3 = \dot{X}_{1G} \sin \alpha - \dot{Y}_{1G} \cos \alpha$$

$$\dot{Z}_3 = \dot{Z}_{1G}$$

where

$$\alpha = (T_{\text{Lift-off}} + T_i)\,\omega + \lambda_0 + \Lambda$$

where

Λ is the longitude of the GE radar.

$\overline{R}_3$ (Mercury units)=.9557638 $\overline{R}_{1G}$ (B-GE units)

$\overline{V}_3$ (Mercury units)=1.022883455 $\overline{V}_{1G}$ (B-GE units)

3.4.4 Usage

The CCGEB1 program is used in the Mercury system in the following manner:

a) Entry to the program is made from MPCCGB.

b) Return to Monitor is normally through MFCCGB. If any of the data flags is zero, return is made to Monitor suffix, MFCCGE.

c) Upon entry, if the accumulator = 0, B-GE data is selected source.

d) Space Required—100_{10} locations, excluding subroutine, tables and constants.

e) Constants Used:

1) KLAMDO—longitude of Greenwich at midnight preceding launch

2) K.0MEG DEC .058833543

3) TCCONS—table of constants used by the launch programs.

f) Erasable Locations—the only erasable locations within the program are in TCMANR, a storage block reserved exclusively for the launch phase programs, and must not be used by any other groups.

3.4.5 Checkout

The program has been checked out both in the FAP and SOS systems using data provided by NASA, Langley Field, Virginia.

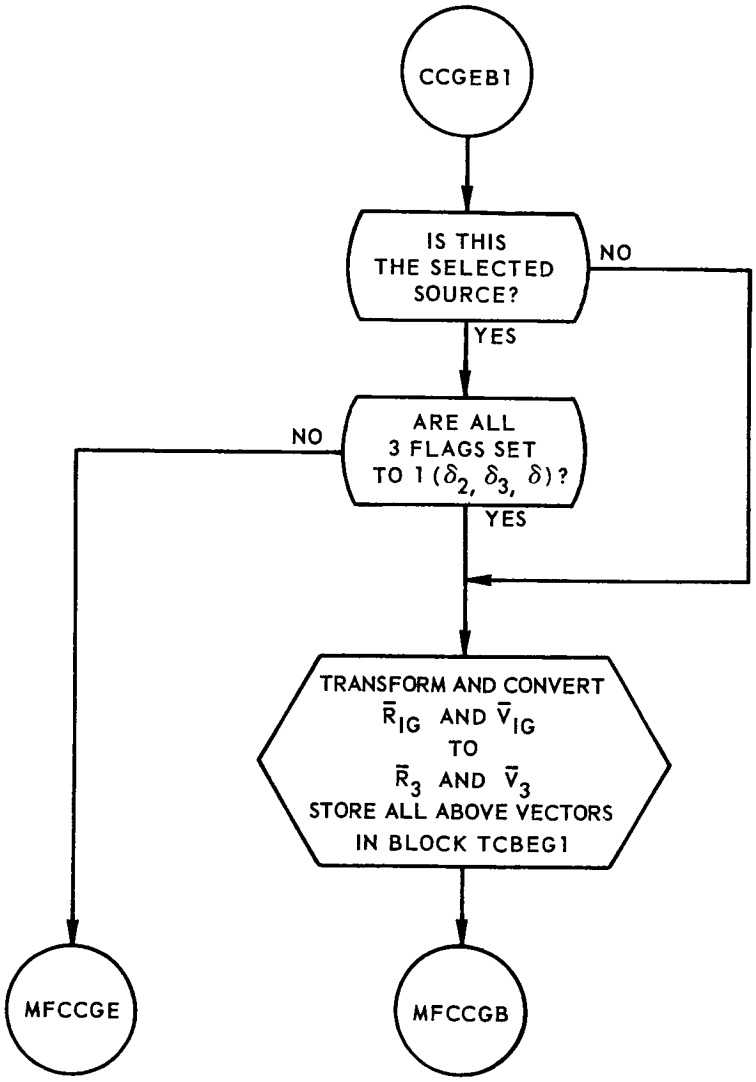


FIGURE 3-4. CCGEB1 PROGRAM FLOW CHART

3.5 IP 7090 COORDINATE TRANSFORMATION PROGRAM (CC7091)

This program is used the entire launch phase and for input uses data processed by the IP 7090 at Cape Canaveral. CC7091 converts position and velocity data from an IP 7090 coordinate system to a Mercury coordinate system. This data enters the high-speed input program, I0HS09, via input line number 2. The output of I0HS09 is fed to the CC7091 program in the first level of launch processing. The flow chart for CC7091 is shown in Figure 3-5.

3.5.1 Input Requirements

This program uses the sine/cosine library program (U1SICO).

The following data from the table TMIP71 serves as one input to the CC7091 program (all input and output to the CC7091 program is in normalized floating point format):

TMIP71	T_i (Time since lift-off in seconds)	
+ 1	X_1	$\overline{R}_1$ (Mercury units)
+ 2	Y_1	
+ 3	Z_1	
+ 4	$\dot{X}_1$	$\overline{V}_1$ (Mercury units)
+ 5	$\dot{Y}_1$	
+ 6	$\dot{Z}_1$	

The second input to CC7091 is the number of seconds between midnight preceding launch and time of lift-off in floating point. This information is located in the communication cell MCLFTM.

3.5.2 Output Requirements

The following data is generated by the CC7091 program and outputted in table TCIP71 which serves as input to the second level of launch processing:

TCIP71	T_i (Time since lift-off in seconds)	
+ 1	X_1	$\overline{R}_{1I}$ (B-GE units)
+ 2	Y_1	
+ 3	Z_1	

TCIP71	T_i (Time since lift-off in seconds)	
+ 4	$\dot{X}_1$	$\overline{V}_{1I}$ (B-GE units)
+ 5	$\dot{Y}_1$	
+ 6	$\dot{Z}_1$	
+ 7	X_3	$\overline{R}_3$ (Mercury units)
+ 8	Y_3	
+ 9	Z_3	
+ 10	$\dot{X}_3$	$\overline{V}_3$ (Mercury units)
+ 11	$\dot{Y}_3$	
+ 12	$\dot{Z}_3$	

3.5.3 Method

The CC7091 program converts $\overline{R}_{0I}$ and $\overline{V}_{0I}$ to $\overline{R}_{1I}$ and $\overline{V}_{1I}$ B-GE units, and then to $\overline{R}_3$ and $\overline{V}_3$ Mercury units. The following formulas are used:

$$R_{1I} = \overline{R}_{0I}\ 1.0463218$$

$$V_{1I} = \overline{V}_{0I}\ .97763793004$$

$$X_3 = X_{0I} \cos(\phi_0 + \omega t) - Y_{0I} \sin(\phi_0 + \omega t)$$

$$Y_3 = X_{0I} \sin(\phi_0 + \omega t) + Y_{0I} \cos(\phi_0 + \omega t)$$

$$Z_3 = Z_{0I}$$

$$\dot{X}_3 = \dot{X}_{0I} \cos(\phi_0 + \omega t) - \dot{Y}_{0I} \sin(\phi_0 + \omega t)$$

$$\dot{Y}_3 = \dot{X}_{0I} \sin(\phi_0 + \omega t) + \dot{Y}_{0I} \cos(\phi_0 + \omega t)$$

$$\dot{Z}_3 \quad \dot{Z}_{0I}$$

$$\overline{R}_3 \text{ (Mercury units)} = 1.00003643\ \overline{R}_{0I} \text{ (IP 7090 units)}$$

$$\overline{V}_3 \text{ (Mercury units)} = 1.000009664\ \overline{V}_{0I} \text{ (IP 7090 units)}$$

$$\phi_0 = \lambda_0 + \omega t$$

The CC7091 program is accurate to at least 7 decimal digits.

3.5.4 Usage

The CC7091 program is used in the system in the following manner:

a) Entry to the program is made to MPCCIP.

b) Return to Monitor is to MFCCIP.

c) Upon entry, if the accumulator = 0, IP 7090 data is selected source.

d) Space Required—99_{10} locations, excluding subroutine and constant.

e) Constants Used:

1) KLAMDO—longitude of Greenwich at midnight preceding launch (radius)

2) TCCONS—table of constants used by the launch programs.

f) Erasable locations—the only erasable locations within CC7091 are in TCMANR, a storage block reserved exclusively for the launch programs, and must not be used by other groups.

3.5.5 Checkout

The CC7091 program has been checked out both in the FAP and SOS systems using data provided by NASA, Langley Field, Virginia.

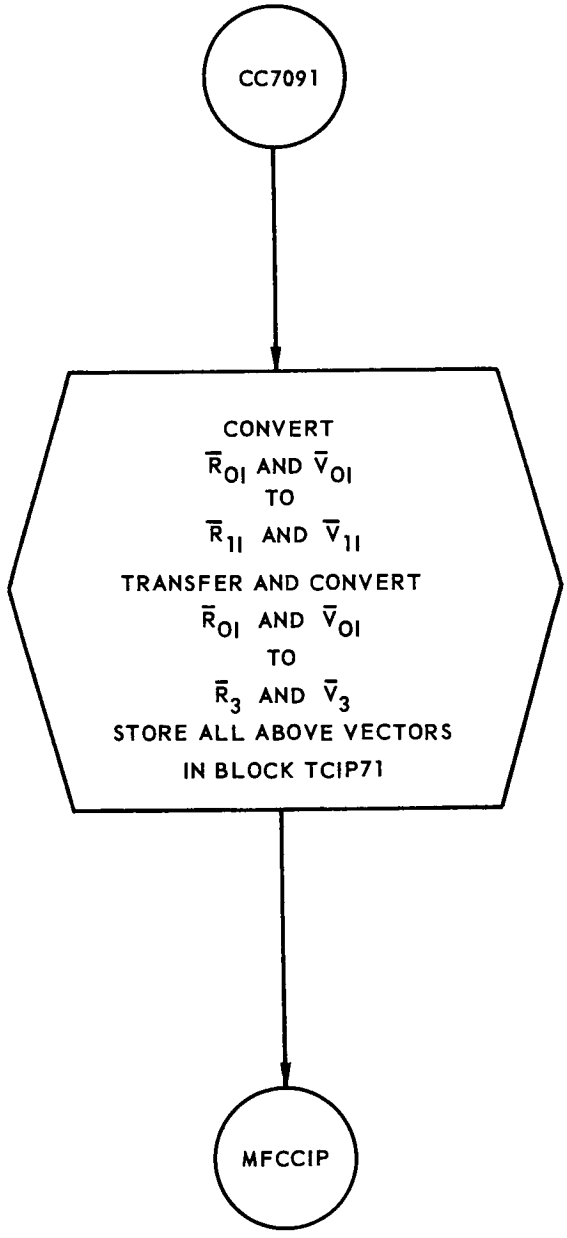


FIGURE 3-5. CC7091 PROGRAM FLOW CHART

3.6 RAW RADAR PROGRAM (CCRAWR)

CCRAWR is the third processor in the first level of the Launch/Abort processing. The purpose of CCRAWR is twofold:

a) To edit all unprocessed AN/FPS-16 radar observations received by way of the high-speed data lines from the Cape Canaveral radars.

b) To generate inertial position and velocity vectors reflecting this data for use in the main Launch/Abort computations.

The flow chart for the CCRAWR program is shown in Figure 3-6.

3.6.1 Input Requirements

Input to the program, CCRAWR, is contained in the 361-word data table TMHRAE, and in the 7-word table of edit parameters, TMHEDP.

a) The first word of the table, TMHRAE, contains the time tag (GMT) of the first observation in the table in .1 second increments, fixed point B35. The remaining 360 locations of the table contain up to 120 three-word AN/FPS-16 radar observations. The first of the three words in each of these observations contains slant range (R) expressed as fixed point feet, B35, while the second and third words contain azimuth (A) and elevation (E) respectively, expressed in fixed point radians B14. The time increment between successive observations in the table is .1 second.

b) The seven parameters of the table, TMHEDP, are used in the editing of the radar observations of the block, TMHRAE. The format of this table is as follows: TMHEDP: location of the range entry of the last observation of TMHRAE (LRTBE).

+ 1: location of the first edited range entry (FER)

+ 2: location of the last edited range entry (LER)

+ 3: location of the first edited azimuth entry (FEA)

+ 4: location of the last edited azimuth entry (LEA)

+ 5: location of the first edited elevation entry (FEE)

+ 6: location of the last edited elevation entry (LEE)

c) The entries of the table, TMHRAE, together with the cell, TMHEDP, are stored by the raw radar processor of I0HS09. The remaining entries of TMHEDP are stored in the edit processor of CCRAWR and adjusted

by the raw radar processor when the data table is moved to accommodate new data.

3.6.2 Output Requirements

Output from the program, CCRAWR, consists of a position and a velocity vector and the associated time tag. This output is stored in the 13-word table, TCFP16, in the following format: TCFP16: Time tag (Δt since L.O.) in floating point seconds.

TCFP16+ 1	X	These entries in the Mercury reference system and B/GE units.
+ 2	Y	
+ 3	Z	
+ 4	$\dot{X}$	
+ 5	$\dot{Y}$	
+ 6	$\dot{Z}$	
+ 7	X	These entries in the Mercury reference system and Mercury Units.
+ 8	Y	
+ 9	Z	
+10	$\dot{X}$	
+11	$\dot{Y}$	
+12	$\dot{Z}$	

Upon exit from CCRAWR, the P bit of the accumulator will be set to 0 to indicate newly generated output and to 1 to indicate the absence of newly generated output and the retention of the previous contents of TCFP16. In both of these cases return is made by direct transfer to MFCCRW.

3.6.3 Method

Control is first given to CCRAWR when the table of edit parameters, TMHEDP, indicates that the data table, TMHRAE, contains four or more unedited observations. Subsequent entries are made at .5 second intervals during launch phase and at 6-second intervals during abort phase. Upon each entry, the location, CESTL, contains the following information:

a) Address: Location of the table of edit parameters, TMHEDP.

b) Decrement: Location of a nine-entry table of edit criteria, CCTEL, containing the following limits.

$(1.5)^{1/2}$ time allowable range error

$(3.0)^{1/2}$ times allowable range error

$(5.0)^{1/2}$ times allowable range error

$(1.5)^{1/2}$ time allowable azimuth error

$(3.0)^{1/2}$ times allowable azimuth error

$(5.0)^{1/2}$ times allowable azimuth error

$(1.5)^{1/2}$ time allowable elevation error

$(3.0)^{1/2}$ times allowable elevation error

$(5.0)^{1/2}$ times allowable elevation error

The CCRAWR program first edits all previously unedited observations in the data table TMHRAE. The edit procedure used in CCRAWR is composed of six distinct parts:

a) Initialization

b) No Previous Editing

c) Previous Editing

d) One or Two Consecutive Rejected Points

e) Three or More Consecutive Rejected Points

f) Test for Other Unedited Coordinates

Discussion of these six parts follows:

Initialization: The following quantities are placed in the address portions of individual core locations:

Last Range to be Edited (j)

First Edited Range (h)

Last Edited Range (i)

First Edited Elevation

If $h = i$, the data table contains no edited observations and a transfer is made to No Previous Editing. If $h \neq i$, the data table does contain edited observations and transfer is made to Previous Editing.

No Previous Editing: In this case, there exists at least four points to be edited. The first and third points are used to predict the second which is compared with the actual value. If this comparison is successful, the second and fourth points are used to predict the third which is again compared to its actual value; linear almost least squares formulae are used: $X_2 = (X_1 + X_3)/2$, $X_3 = (X_2 + X_4)/2$. By considering overlapping sets of four points, this test is continued until a set of four points is found satisfying both of these formulae or until no more data is available; h and i are updated at each step. If four satisfactory points are found, i is further updated and transfer is made to Previous Editing. If the data is exhausted before four such points are found, transfer is made to Test for Other Unedited Coordinates.

Previous Editing: This is the usual calculation made. Four established points are used to predict a fifth, as yet unedited point which is compared with its actual value. If the point is accepted, i is updated and the process continued until a point is rejected or until data is exhausted. A transfer is then made to One or Two Consecutive Rejected Points or to Test for Other Unedited Coordinates, respectively. The following linear almost least square formula is used to predict the fifth point:

$$X_5 = X_4 + (X_3 - X_1)/2$$

One or Two Consecutive Rejected Points: A sixth point is predicted from the four established points; if accepted, the fifth point is interpolated and a return is made to Previous Editing.

$$X_6 = X_4 + X_3 - X_1 \; ; \quad X_5 = (X_6 + X_4)/2$$

If the sixth point is rejected and no more data is available, a transfer to Test for Other Unedited Coordinates is made. If more data is available, the seventh point is predicted from the four points. If the seventh is accepted, both the fifth and sixth points are interpolated and return then made to Previous Editing. Here,

$$X_7 = 2X_4 - X_1$$

$$X_5 = (X_7 + X_4 - X_3 + X_2)/2$$

$$X_6 = (X_7 + X_4 + X_3 - X_2)/2$$

If the seventh point is rejected, a transfer is made to Three or More Consecutive Rejected Points.

Three or More Consecutive Rejected: In this case, three or more points may actually be in error or the launch trajectory has been distinctly altered and these points lie on this trajectory. If there is no more data, a transfer to Test for Other Unedited Coordinates is made. If there is more data, successive sets of four points are again used to establish a straight line, as in No Previous Editing, until a set of four acceptable points is found, data is exhausted, or a preset maximum number of attempts to find such a set is exceeded. If an acceptable set is found, the rejected points are interpolated. If the data is exhausted before a satisfactory set is found, a transfer is made to Test for Other Unedited Coordinates. If the preset maximum number of attempts is exceeded, the table of edit parameters, TMHEDP, is reset to show the observation containing the offending coordinate as the first edited observation. Editing of this coordinate is then resumed by transfer to No Previous Editing, if the data table contains four or more unedited entries of this coordinate, or by transfer to Test for Other Unedited Coordinates if fewer than four entries remain. The required linear almost least squares formulae are as follows:

$$X_{i+k-1} = (X_{i+k} + X_{1+k-2})/2$$

$$X_{i+k} = (X_{i+k+1} + X_{i+k-1})/2$$

$$X_{i+N} = X_i + \frac{N(X_{i+k-2} - X_i)}{k-2} \qquad N = 1, 2, \ldots k-3$$

Test for Other Unedited Coordinates: Upon completing the editing of the range or azimuth coordinates, the h and i criterion locations are updated and a return made to process the remaining coordinates.

Upon completion of the editing procedure, the table of edit parameters is examined. If these parameters indicate that the data table contains fewer than 31 edited observations, an insufficient data return is made to Monitor. If however, the table contains 31 or more edited points, the time tag of that observation lying at the midpoint of the most recent 31 point edited span is computed and compared with the last time tag similarly computed. This comparison must indicate a more recent present span in order to continue processing. Otherwise an insufficient data return is made to Monitor.

If, according to the above time computations, a more recent set of 31 observations now appears in the data table, the data is smoothed; and inertial position and velocity vectors, $\overline{R}_3$ and $\overline{V}_3$ respectively, are generated representing the computed time tag in the following manner.

a) Smoothed local position and velocity coordinates are first determined:

1) Let R_i, A_i, and E_i be the coordinates of the last observation of the 31 observations span.

2) The smoothed local position coordinates, $\overline{R}_{i-15}$, $\overline{A}_{i-15}$, and $\overline{E}_{i-15}$, are computed using 11 point almost least squares smoothing about the midpoint according to the following formula:

$$\overline{C}_{i-15} = K\Big\{2\Big[4(C_{i-16} + C_{i-15} + C_{i-14}) - C_{i-20} - C_{i-10} + C_{i-18} + C_{i-17} + C_{i-13} + C_{i-12}\Big] + C_{i-19} + C_{i-11}\Big\}$$

where C_i assumes the values R_i, A_i, and E_i respectively, on successive computations $K = (1/30)(MLU/ft)$ for $C_i = R_i$, and $K = (1/30)$ for $C_i = A_i$ and $C_i = E_i$.

3) In a similar manner, smoothed local velocity coordinates, $\dot{R}_{i-15}$, $\dot{A}_{i-15}$, and $\dot{E}_{i-15}$, are computed using 31 point almost least squares smoothing about the midpoint according to the following formula:

$$C_{i-15} = L\left\{\left\{2\left\{2\left[2\left(\sum_{j=i}^{i-5} C_j - \sum_{j=i-25}^{i-30} C_j\right) + \sum_{j=i-6}^{i-10} C_j - \sum_{j=i-20}^{i-24} C_j\right] + \sum_{j=i-11}^{i-12} C_j - \sum_{j=i-18}^{i-19} C_j\right\} + \sum_{j=i-13}^{i-14} C_j - \sum_{j=i-17}^{i-18} C_j\right\}\right\}$$

C_i defined as above and $L = (MTU/151.4)(MLU/ft)$ for $C_i = R_i$ and $L = (MTU/151.4)$ for $C_i = A_i$ and $C_i = E_i$.

b) Position and velocity vectors, $R'' = (X'', Y'', Z'')$ and $V'' = (\dot{X}'', \dot{Y}'', \dot{Z}'')$, in the local coordinate system of the transmitting radar are then computed from the smoothed local coordinates as follows:

1) Position Vector R'':

$$X'' = \overline{R} \cos \overline{E} \sin \overline{A}$$

$$Y'' = \overline{R} \cos \overline{E} \cos \overline{A}$$

$$Z'' = \overline{R} \sin \overline{E}$$

2) Velocity Vector V'':

$$\dot{X}'' = \dot{R} \cos \overline{E} \sin \overline{A} + \dot{A}\overline{R} \cos \overline{E} \cos \overline{A} - \dot{E}\,\overline{R} \sin \overline{E} \sin \overline{A}$$

$$\dot{Y}'' = \dot{R} \cos \bar{E} \cos \bar{A} - \dot{A} \bar{R} \cos \bar{E} \sin \bar{A} - \dot{E} \bar{R} \sin \bar{E} \cos \bar{A}$$

$$\dot{Z}'' = \dot{R} \sin \bar{E} + \dot{E} \bar{R} \cos \bar{E}$$

c) The translation given by the equations below is then applied to the position vector R'' determining the position vector $R' = (X', Y', Z')$ in a geocentric coordinate system.

$$X' = X''$$

$$Y' = Y'' - R \sin (\phi - \phi')$$

$$Z' = Z'' + R \cos (\phi - \phi') + H$$

where R is the radius of the earth, ϕ the geodetic latitude of the transmitting radar, ϕ' the geocentric latitude of the radar, and H its height above sea level.

d) Finally, the position vector R' and the velocity vector V'' are converted to inertial position and velocity vectors R_3, V_3 and stored as output in TCFP16 + 7 thru TCFP16 + 12.

1) Geocentric position $R' = (X', Y', Z')$ is converted to inertial position $R_3 = (X_3, Y_3, Z_3)$ by first rotating the coordinate system about the X axis to give a geocentric system with X and Y axes in the equatorial plane and then about the Z axis to the true Mercury inertial coordinate system as follows:

$$\lambda = \lambda_q + \lambda_{sta} + \omega t$$

where

λ_q = inertial longitude of Greenwich at t_o

λ_{sta} = longitude of transmitting radar

ω = angular velocity of the earth's rotation

t = elapsed time since t_o

$$X_3 = -X' \sin \lambda - Y' \sin \phi \cos \lambda + Z' \cos \phi \cos \lambda$$

$$Y_3 = X' \cos \lambda - Y' \sin \phi \sin \lambda + Z' \cos \phi \sin \lambda$$

$$Z_3 = Y' \cos \phi + Z' \sin \phi$$

2) Local velocity $V'' = (X'', Y'', Z'')$ is converted to inertial velocity $V_3 = (\dot{X}_3, \dot{Y}_3, \dot{Z}_3)$ using the following formulae:

$$\dot{X}_3 = -\dot{X}'' \sin\lambda - \dot{Y} \sin\phi \cos\lambda + \dot{Z}'' \cos\phi \cos\lambda - \omega Y_3$$

$$\dot{Y}_3 = \dot{X}'' \cos\lambda - \dot{Y}'' \sin\phi \sin\lambda + \dot{Z}'' \cos\phi \sin\lambda + \omega X_3$$

$$\dot{Z}_3 = \dot{Y}'' \cos\phi + \dot{Z}'' \sin\phi$$

e) In order to drive the strip chart displays, the $\overline{R}_3$ and $\overline{V}_3$ vectors are also converted to Burroughs/GE units and stored in locations TCFP16 + 1 through TCFP16 + 6.

f) Upon completion of these computations a valid data return is made to Monitor.

3.6.4 Usage

a) Space Required: the program CCRAWR requires 592 core storage locations. The storage required for the data table TMHRAE and the table of edit parameters, TMHEDP, are not included in this figure and require an additional 368 core locations.

b) Constants:

KM0000	DEC	-0
K00001	DEC	1
K00002	DEC	2
K00003	DEC	3
K00006	DEC	6
K00009	DEC	9
K00090	DEC	90
K010.0	DEC	10.0
K.OMEG	DEC	earth's rotational velocity (radians/sec)
KCH233	OCT	233000000000

c) Subroutines: The program CCRAWR uses the sin and cos subroutine U1SICO.

d) Time Required: Normal entry to the program requires the edit processing described in Previous Editing. On this assumption the timing of the program is as follows:

1) Edit: 196 machine cycles for first point
93 machine cycles for each additional point

2) Smoothing and conversion: .02 sec.

e) Checkout:

1) Several test cases were run generating vectors on the CADFISS I Launch trajectory at 10 second intervals. All of these cases agreed with listings of the nominal trajectory.

2) The program has been an operational processor of the Mercury System since March of 1961.

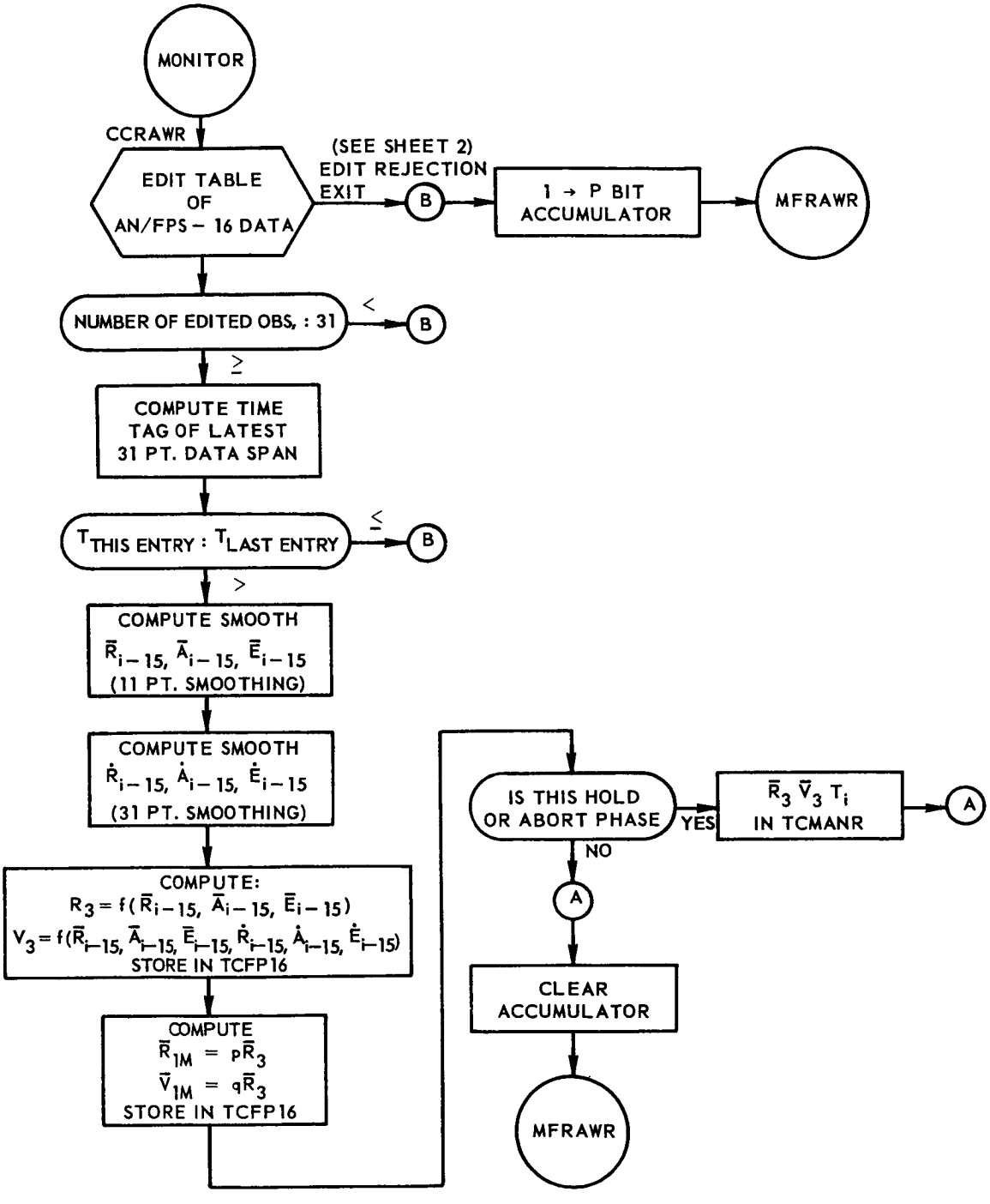


FIGURE 3-6. CCRAWR PROGRAM FLOW CHART *(Sheet 1 of 4)*

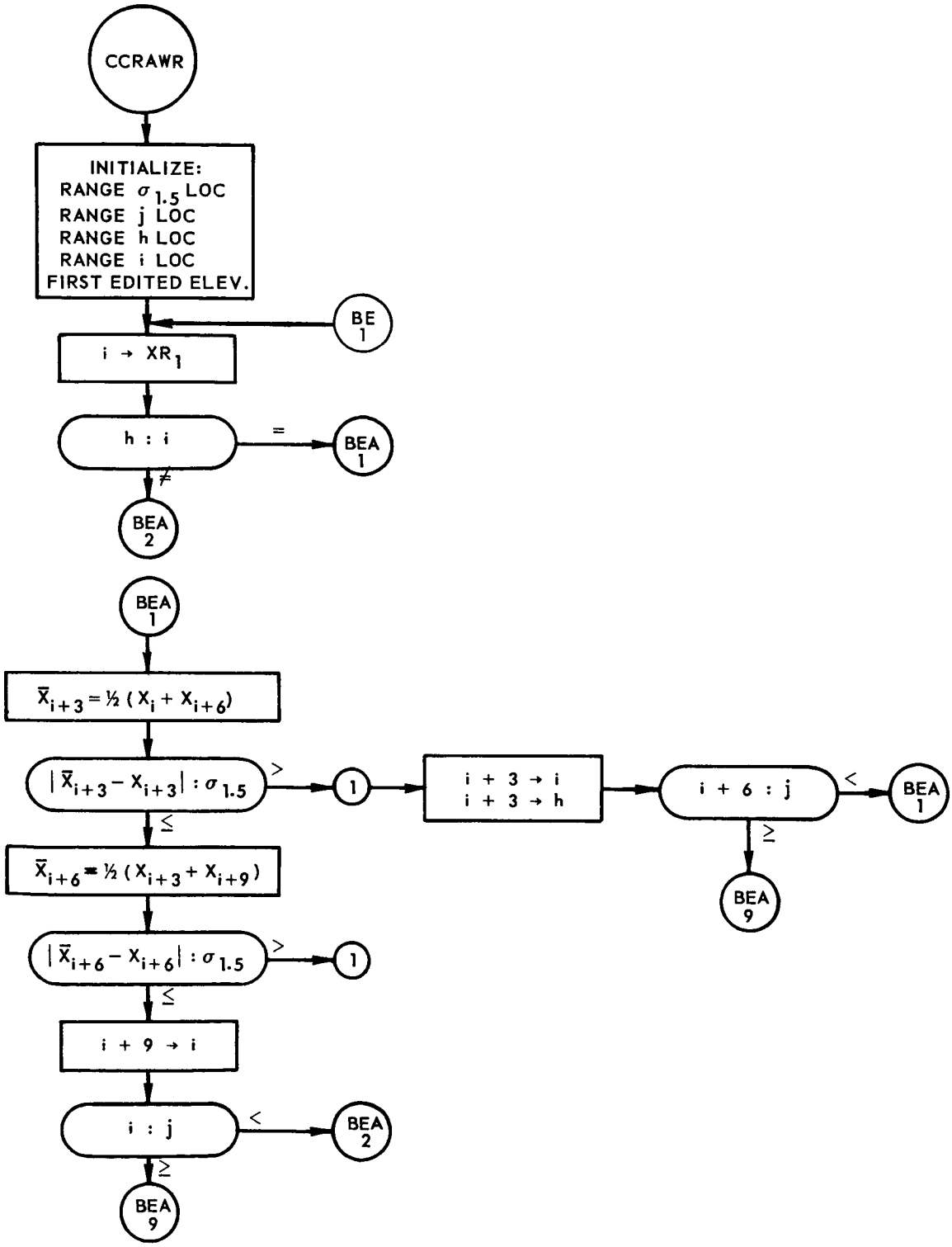


FIGURE 3-6. CCRAWR PROGRAM FLOW CHART *(Sheet 2 of 4)*

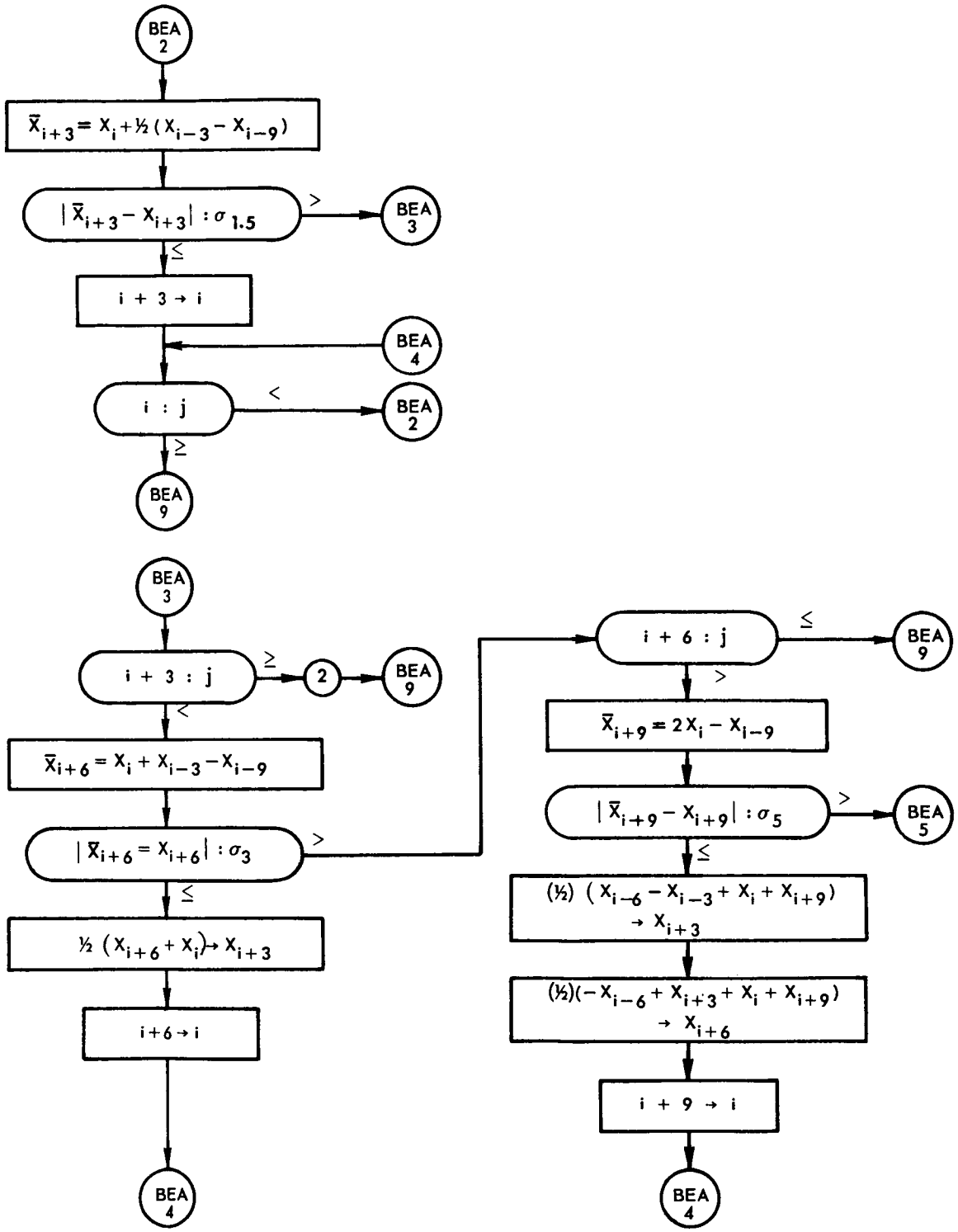


FIGURE 3-6. CCRAWR PROGRAM FLOW CHART *(Sheet 3 of 4)*

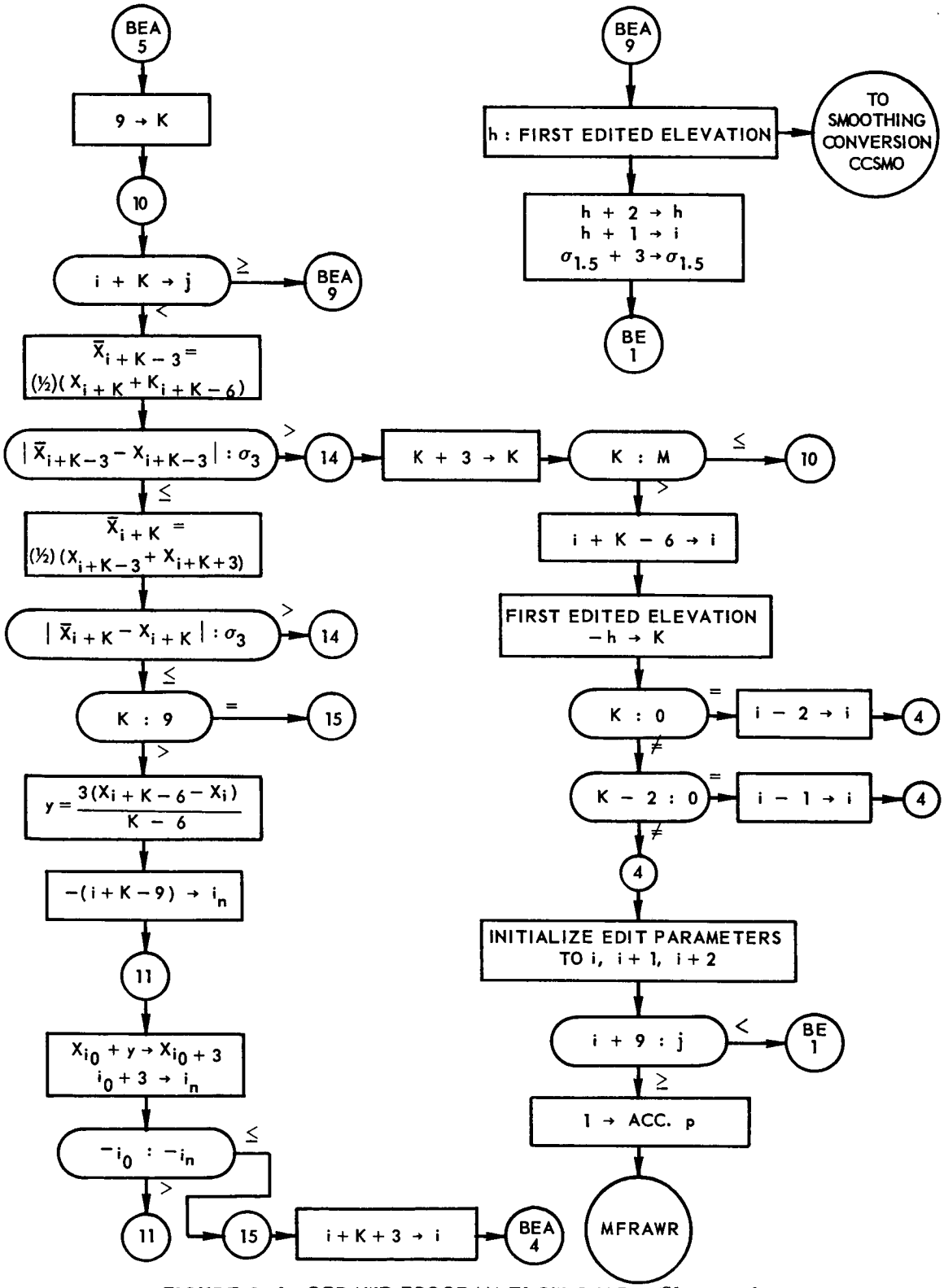


FIGURE 3-6. CCRAWR PROGRAM FLOW CHART *(Sheet 4 of 4)*

3.7 STRIP CHART CALCULATION PROGRAMS (CCSTIP, CCSTGE, and CCST16)

These programs form the second level of launch processing and are used only from lift-off to hold. The main purpose of the strip chart programs is to:

a) Prepare output for the Data Quality Monitor at Cape Canaveral using all data received from the B-GE and IP 7090 systems and the AN/FPS-16 radar.

b) Perform a gross edit check on the selected source data.

c) Computes and stores the following parameters in the output table (TCCOUT) from the selected source data.

1) height (h)

2) inclination angle (i)

3) flight path angle (γ)

4) velocity ratio (V/V_r)

5) down range distance (d)

6) cross range distance (s, sometimes referred to as y)

The flow chart for the Strip Chart programs is shown in Figure 3-7.

3.7.1 Input Requirements

The Strip Chart programs use the following library routines:

a) U1SQRT—square root

b) U1SICO—sine/cosine

c) U1ASCO—arc sine/arc cosine

d) U1ATNA—arc tangent A

The coordinate conversion programs, CCGEB1 and CC7091, and the raw radar data processor program, CCRAWR, supply the Strip Chart programs with $\overline{R}$, $\overline{V}$, and t in inertial coordinates and an indicator set in the accumulator to whether this is the selected or non-selected source (0 means selected, 1 means non-selected).

The $\bar{R}$, $\bar{V}$, and t input to the Strip Chart programs will be located in one of the following tables:

B-GE Data		Entry from CCSTGE		
TCBEG1		Time Since Lift-Off	Seconds	Floating Point
+ 1, + 2, + 3	R_{1G}	B-GE Units		Floating Point
+ 4, + 5, + 6	V_{1G}	B-GE Units		Floating Point
+ 7, + 8, + 9	R_3	Mercury Units		Floating Point
+10, +11, +12	V_3	Mercury Units		Floating Point
+13	TCCOUT	Address		Fixed Point B35
+14	TCCOUT + 2	Address		Fixed Point B35

IP 7090 Data		Entry from CCSTIP		
TCIP71		Time Since Lift-Off	Seconds	Floating Point
+ 1, + 2, + 3	R_{1I}	B-GE Units		Floating Point
+ 4, + 5, + 6	V_{1I}	B-GE Units		Floating Point
+ 7, + 8, + 9	R_3	Mercury Units		Floating Point
+10, +11, +12	V_3	Mercury Units		Floating Point
+13	TCCOUT + 1	Address		Fixed Point B35
+14	TCCOUT + 3	Address		Fixed Point B35

Raw Radar Data		Entry from CCST16		
TCFP16		Time Since Lift-Off	Seconds	Floating Point
+ 1, + 2, + 3	R_{1M}	B-GE Units		Floating Point
+ 4, + 5, + 6	V_{1M}	B-GE Units		Floating Point
+ 7, + 8, + 9	R_3	Mercury Units		Floating Point
+10, +11, +12	V_3	Mercury Units		Floating Point
+13	TCCOUT +1	Address		Fixed Point B35
+14	TCCOUT +3	Address		Fixed Point B35

3.7.2 Output Requirements

The Strip Chart programs output the following data:

a) B-GE Data

TCCOUT	$\gamma - \gamma_{nom}$	Radians	Floating point
TCCOUT+2	$V/V_r - (V/V_r)_{nom}$		Floating point

b) IP 7090 or Raw Radar Data

TCCOUT+1	$\gamma - \gamma_{nom}$	Radians	Floating point
TCCOUT+3	$V/V_r - (V/V_r)_{nom}$		Floating point

c) Selected Source Data Only

TCCOUT +19	H	Nautical miles	Floating point

+ 20	γ		Radians	Floating point
+ 21	i		Radians	Floating point
+ 22	V/V_r			Floating point
+ 24	d		Nautical miles	
+ 28	h (same as +19)			Floating point
TCMANR+7	S	Cross range distance	Nautical miles	Floating point
+ 8	$\gamma - \gamma_{nom}$	Flight path angle deviation	Radians	Floating point
+ 9	$V/V_r - (V/V_r)_{nom}$	Velocity ratio deviation		Floating point
+ 10	d	Down range distance	Nautical miles	Floating point
+ 37	$\lvert V_1 \rvert$		B/GE	Floating point
+ 38	V_r		Ft/Sec	Floating point

If the input data is the selected source and if it passes the gross edit check, then the $\bar{R}_3$, $\bar{V}_3$ and t are stored in the main computation table (TCMANR) for use by the Main Calculation program (CCMAIN).

3.7.3 Method

The strip Chart programs compute two quantities output to the Data Quality Monitor: 1) $(\gamma - \gamma_{nom})$, the deviation of the flight path angle from the nominal, and 2) $\left[V/_{V_r} - (V/_{V_r})_{nom}\right]$ the deviation of the velocity ratio, are calculated for every piece of data received. The nominal values for γ and V/V_r are computed by a curve fit as a function of time since lift-off. Each of the quantities has a curve to be used below staging (booster engine cut off) and another above staging. During powered flight, that is, before sustainer engine cutoff or abort initiate has occurred, the elapsed time of each observation is used. Since the curves

for computing nominal γ and $V/_{V_r}$ are based on powered flight, any observation occurring after power cut-off uses the time of power cut off. Thus the nominal value is frozen after power cut off.

At present, the inertial $\overline{R}_3$ is converted to the pad coordinate system ($\overline{R}_2$), and output quantities for down range (d) and cross range distances (s) of capsule position are calculated (see Subsection 3.7.5).

When selected source is being processed, the data is submitted to a gross editing test which determines whether or not the data is acceptable (close to nominal). If the value for $(\gamma - \gamma_{nom})$ or $\left[V/V_r - (V/_{V_r})_{nom}\right]$ exceeds its respective maximum bound, the data is rejected and an indication is given to the Monitor suffix that the data is assumed to be missing. This gross edit is skipped if an abort has occurred before tower separation. In this case, the observations may not agree with the nominal case, but they should be processed as valid observations.

After the gross edit is completed or skipped, the inertial $\overline{R}_3$, $\overline{V}_3$, and t vectors are transferred to the Main computation program (CCMAIN). The capsule height (h) is calculated and compared with the minimum altitude for medium abort. If (h) is below this value the h altitude counter is set to zero, if above this altitude, the counter is incremented by one. Three or more consecutive observations from the selected source which are above this minimum altitude will cause the program to enter medium abort if abort were recommended.

The inclination angle (i) is computed by:

$$h_X = Y_, \dot{Z}_, - Z_, \dot{Y}_,$$

$$h_Y = Z_, \dot{X}_, - X_, \dot{Z}_,$$

$$h_Z = X_, \dot{Y}_, - Y_, \dot{X}_,$$

where $X_,$, $Y_,$, $Z_,$, $\dot{X}_,$, $\dot{Y}_,$, $\dot{Z}_,$ are vector components

$$i = \arctan \frac{\sqrt{h_X^2 + h_Y^2}}{h_Z}$$

and is stored in the output table (TCCOUT) and the program returns to Monitor.

The Strip Chart programs are accurate to at least 7 decimal digits.

3.7.4 Usage

a) Calling Sequence - Three entry points are available

B-GE Data - CCSTGE

IP 7090 Data - CCSTIP

Raw Data - CCST16

b) Space Required

354_{10} cells excluding input and output tables and subroutines

c) Coding Information

Main Launch Tables, TCMANR and TCCONS are used.

MCLFTM contains time of lift-off (floating point - seconds)

Return to Monitor - MFSTRP

d) Constant Used

KLAMDO	Longitude of Greenwich at midnight preceding launch

e) Timing—approximately 42.8 milliseconds (709).

f) Checkout—the Strip Chart programs have been checked out using data supplied by NASA personnel located at Langley Field, Va.

3.7.5 Conversion of Inertial to Pad Coordinates

$$R_2 = \begin{pmatrix} d \\ s \\ w \end{pmatrix} = A^T B^T \left[C^T \begin{pmatrix} x_3 \\ y_3 \\ z_3 \end{pmatrix} - \begin{pmatrix} R_x \\ 0 \\ R_z \end{pmatrix} \right]$$

where: $$A^T = \begin{pmatrix} -\cos\beta_o & \sin\beta_o & 0 \\ -\sin\beta_o & -\cos\beta_o & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$B^T = \begin{pmatrix} \sin L'_p & 0 & -\cos L'_p \\ 0 & 1 & 0 \\ \cos L'_p & 0 & \sin L'_p \end{pmatrix}$$

$$C^T = \begin{pmatrix} \cos\Omega & \sin\Omega & 0 \\ -\sin\Omega & \cos\Omega & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$R_x = r'_o \cos L_p$; $R_z = r'_o \sin L_p$

$\Omega = \Omega_o + \omega t$

Ω_o is the inertial longitude of the (d, s, w) origin at lift-off.

t is time measured from lift-off.

L'_p is the geodetic latitude of the (d, s, w) origin

β_o is the launch azimuth from north, positive clockwise when looking down

r'_o is the distance from the geo-center to the (d, s, w) origin

L_p is the geocentric latitude of the (d, s, w) origin

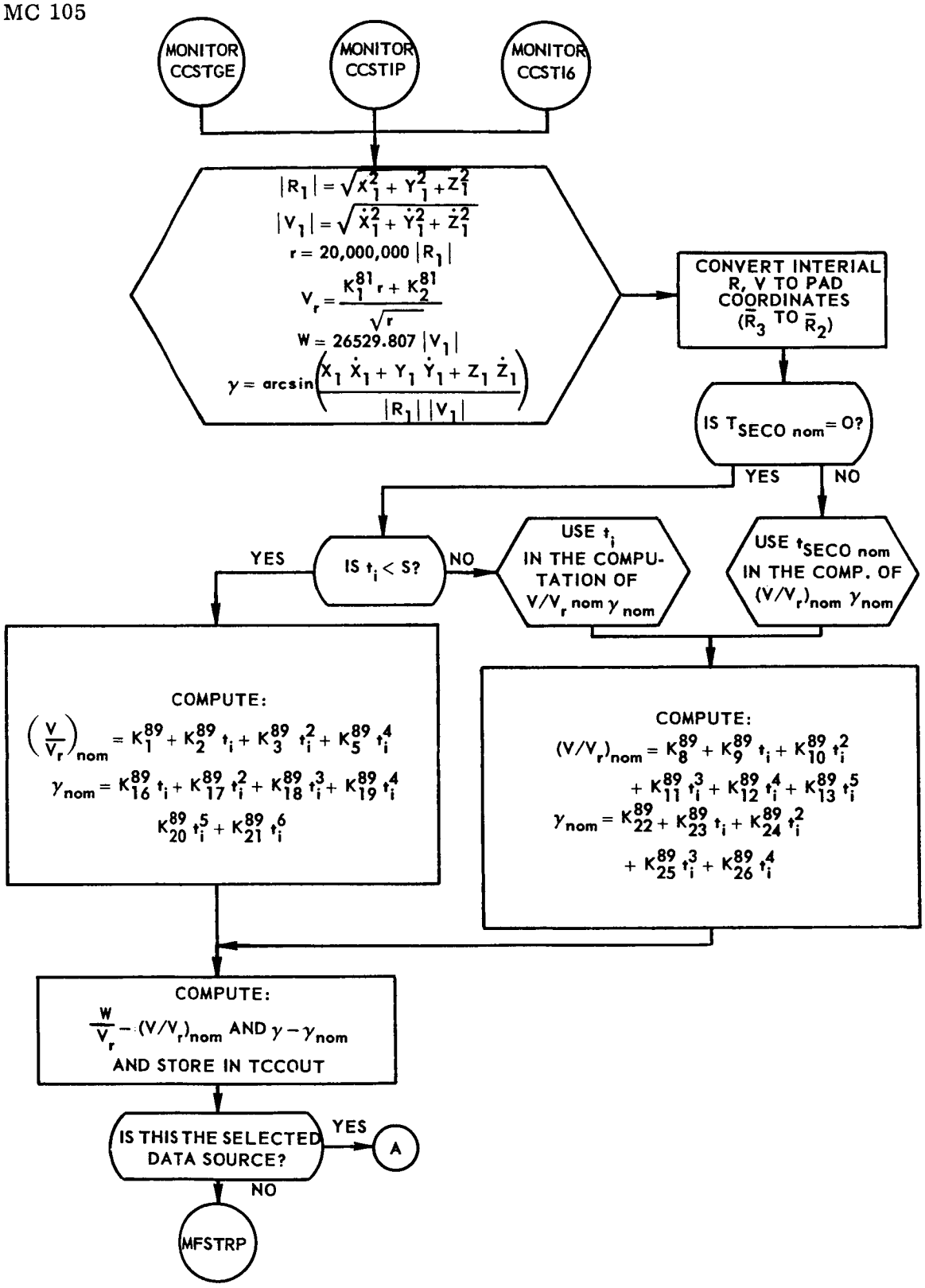


FIGURE 3-7 STRIP CHART PROCESSORS FLOW CHART *(Sheet 1 of 2)*

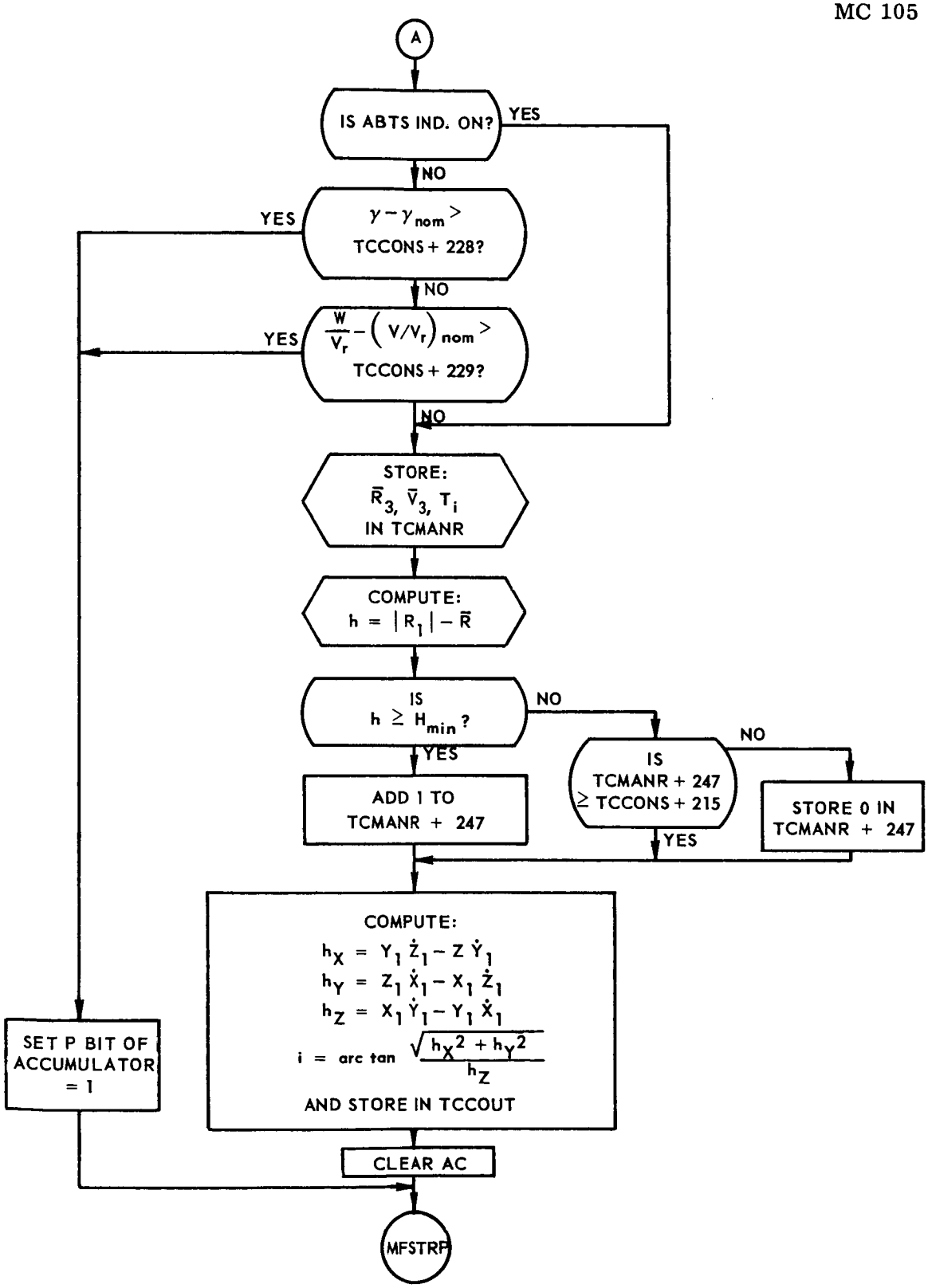


FIGURE 3-7. STRIP CHART PROCESSORS FLOW CHART *(Sheet 2 of 2)*

3.8 LAUNCH PHASE MAIN CALCULATION PROGRAM (CCMAIN)

The function of the CCMAIN program is to monitor the behavior of the launch vehicle during the powered flight phase of a Mercury mission. One main objective of CCMAIN is to compute the GO, NO-GO recommendation which provides the Flight Dynamics Officer (FDO) at the Mercury Control Center with the necessary data for the decision to let the capsule go into orbit or to abort the mission. The other objective is to compute the time to fire the retrorockets to land within a selected recovery area in the event of a capsule abort. The flow chart for the CCMAIN program is shown in Figure 3-8.

3.8.1 Input Requirements

Telemetry signals indicating the following actions serve as input to CCMAIN:

a) Lift-off

b) Tower separation

c) Power cut-off

d) Capsule separation

e) Number of posigrades fired.

In addition to these inputs are: time, position, and velocity vectors.

Provision is made for gross editing of input data to reject data which obviously is of incorrect magnitude.

3.8.2 Output Requirements

The outputs of the CCMAIN program are: present capsule position, minimum delay impact point, maximum delay impact point, recovery area, time of firing retrorockets, GO, NO-GO. The minimum delay impact point is a point at which the capsule would impact if power cut-off occurred at this instant and the retrorockets would fire within the earliest time which is usually 30 seconds. The maximum delay point is defined as an impact point with retrorockets firing at the latest possible moment which is at an altitude of slightly higher than 450,000 feet.

The outputs of CCMAIN at the end of powered flight are the GO, NO-GO recommendation and the time to fire retrorockets to have the capsule land in a selected recovery area.

3.8.3 Method

In operation, the CCMAIN program is guided in its computations by five telemetry inputs. The first signal received by the program indicates that lift-off has occurred.

The next signal indicates that the separation of the escape tower from the launch vehicle has occurred. Between lift-off and the time of tower separation only one impact point is computed since retrorockets are not involved.

The next telemetry input signal would be a power cut-off signal which could be initiated by either the sustainer engine cut-off signal or the abort initiate signal. Shortly thereafter, a capsule separation signal together with a signal indicating the number of posigrades fired to separate the capsule from the sustainer engine are input to CCMAIN. Using the posigrade signal the CCMAIN program computes a posigrade velocity increment and adds this to the known velocity of the sustainer engine. After capsule separation the program goes into the GO, NO-GO computation. This computation uses 10 consecutive inputs which include R, V, and t vectors. Ten computing cycles after capsule separation the data is statistically analyzed for deviations within the data greater than the allowable limit. When the statistical analysis is concluded the remainder of the data is averaged and used to compute the final GO, NO-GO recommendation and the final time at which retrorockets should fire if the mission is to abort. The program remains in the hold phase awaiting an abort or orbit decision by the Flight Dynamics Officer from the Mercury Control Center.

The following pages contain a more detailed discussion of the individual blocks depicted in the overall CCMAIN program flow chart. In reading the detailed method of CCMAIN operation, the flow chart (located on page 3-125) may be extended thereby allowing the reader to locate each block in relation to the overall diagram.

3.8.4 Usage

CCMAIN uses three tables, TCCOUT (Table 3-2), TCCONS (table 3-3) and TCMANR (Table 3-4). The formats of these are shown on page 3-105. A listing of the nomenclatures of R and V for the launch phase is shown on page 3-123.

a) Timing

CCMAIN is in operation once every half second when data is available. The actual time of computation is approximately one fifth of one second.

b) Checkout

CCMAIN is operational in the Mercury Program System.

CCMAIN (Figure 3-9)

The purpose of this first logical block in CCMAIN is to calculate the capsule's present position ϕ_{pp}; λ_{pp}. In addition, it calculates pointing data to be sent to Bermuda a total of four times during launch, at 30 second intervals starting at 150 seconds after lift-off.

Upon entering CCMAIN with a vector and its time tag (T_i) the time tag is converted to GMT by the addition of lift-off time $T_{now} = T_i + T_{L0.}$. This GMT time and the R vector serve as input to the A3MSCP program which produces as output the latitude and longitude of the capsule's present position. These quantities are then stored in TCCOUT + 5 and + 6 for display on the wall map at Cape Canaveral. Also, the magnitude of the R and V vectors are computed using the following equations:

$$| R_3 | = \sqrt{X_3^{\,2} + Y_3^{\,2} + Z_3^{\,2}}$$

$$| V_3 | = \sqrt{(\dot{X}_3)^2 + (\dot{Y}_3)^2 + (\dot{Z}_3)^2}$$

$| R_3|$ and $| V_3 |$ are then stored in TCMANR + 31 and + 32 respectively. Then a subprogram, MMAQA (see Subsection 3.9), determines the need for the computation of pointing data for Bermuda and generates the required data if necessary, and stores them in the table TMLABF. Control then goes to CCMLS.

CCMLB (Figure 3-10)

This block determines whether the escape rockets have fired, stores the time at which firing occurred and determines the need to modify the reported velocity (V_3). This modification can only take place if the data is B-GE or Azusa.

The program checks the cell (TCMANR + 21) for the presence of the "Escape Rockets Fired" (ESCR) bit. If the bit is absent, the cell (TCMANR + 194) containing the time of the event is cleared, and control goes to CVESC + 1. If the ESCR bit is present and detected for the first time, the time of this event is stored in (TCMANR+ 194), the print message for on-line printing is set up using subroutine PSIP and control goes to CVNPR. If the ESCR bit is present now and was also present before, a test is made to determine the source of the current data. If the current data is either B-GE or Azusa control goes to PREVV, otherwise (data being either raw or processed AN/FPS-16) control is transferred to CVESC. +1.

CVESC + 1: The velocity V_8 is set equal to the velocity V_3 because there is no need to add the velocity increment due to the firing of escape rockets. Control is then transferred to CCMLD.

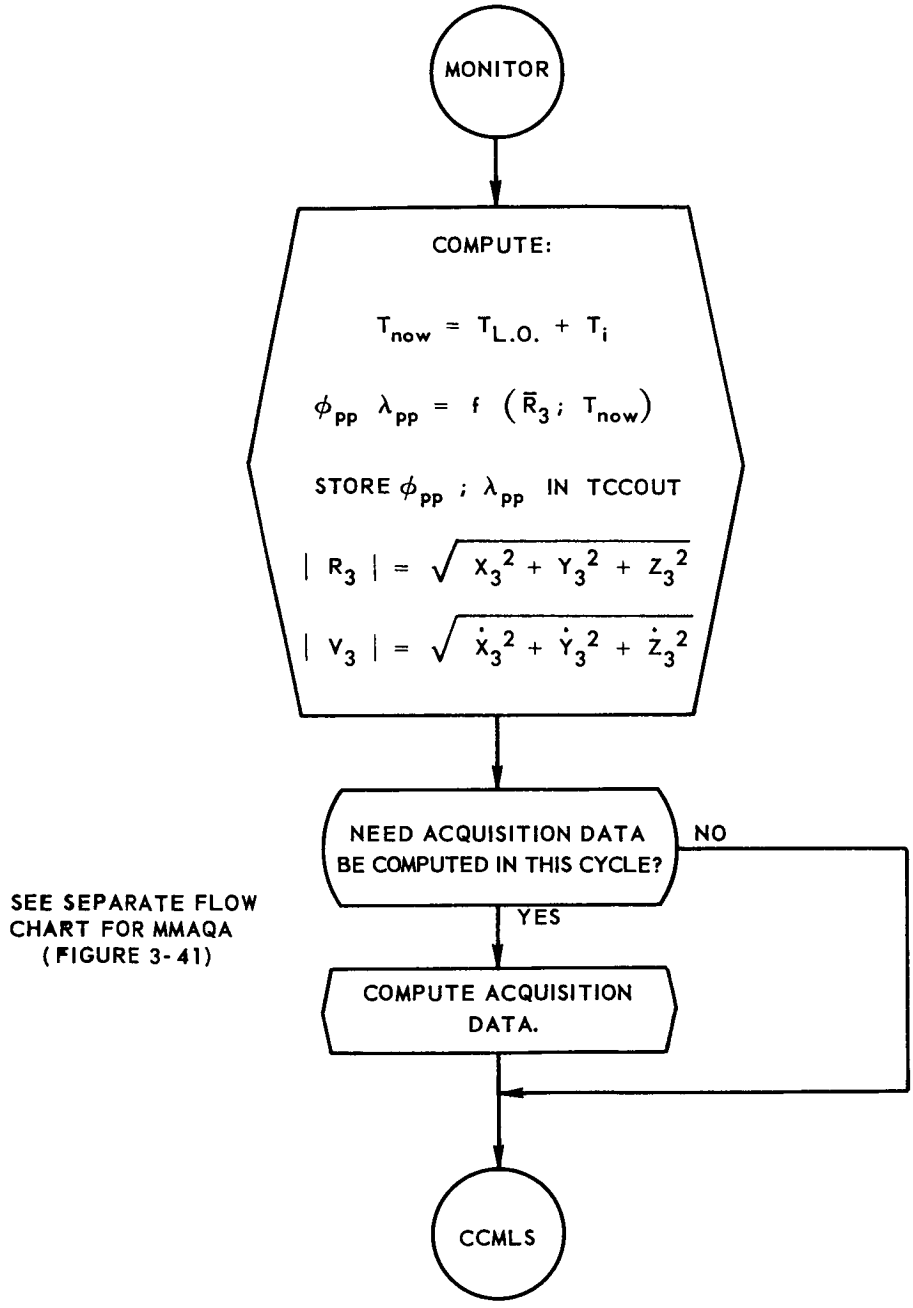


FIGURE 3-9. CCMAIN BLOCK

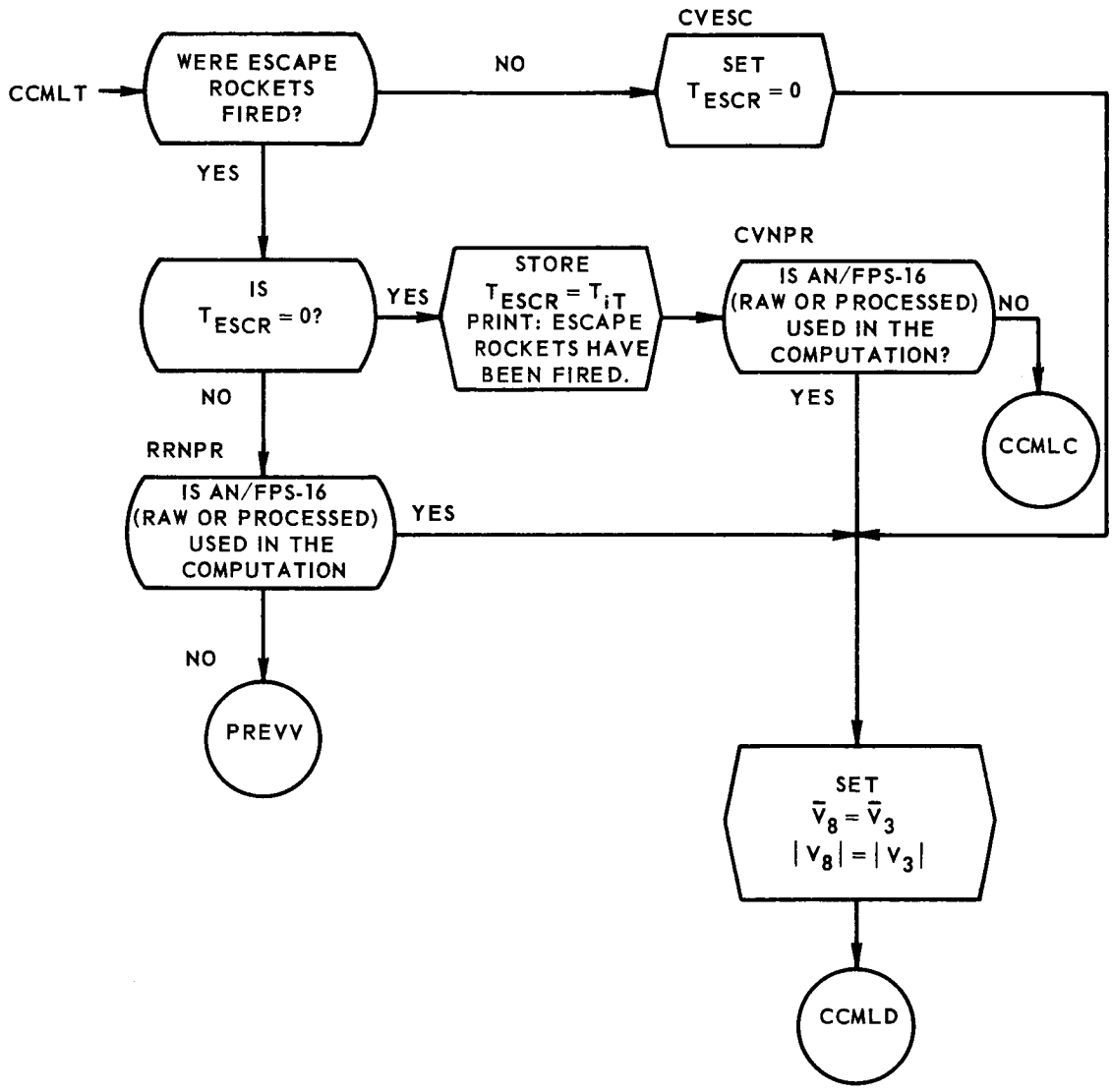


FIGURE 3-10. CCMLB BLOCK

CVNPR: If the current data is either B-GE or Azusa control goes to CCMLC, otherwise control goes to CVESC + 1.

CCMLC (Figure 3-11)

The velocity increment due to firing of escape rockets is computed as a function of altitude.

Two polynomial expressions are necessary to fit the curve properly. If the altitude is below 80,000 feet (TCCONS + 70) a third degree polynomial is solved. If the altitude is above 80,000 feet a second degree polynomial is solved. Either of the above polynomials computes the increment of velocity due to firing of escape rockets (V_{11}). The quantity (1 + n) where $n = \frac{|V_{11}|}{|V_3|}$ is computed for subsequent use (in PREVV) as a multiplier used to modify V_3. Control then goes to PREVV.

PREVV (Figure 3-12)

The velocity (V_8) incorporating the velocity due to firing of escape rockets is computed as follows:

$$V_8 = V_3 (1 + n)$$

$$|V_8| = |V_3| (1 + n)$$

where (1 + n) was computed in CCMLC. Control then goes to CCMLD.

CCMLD (Figure 3-13)

This block computes ϕ_{IP} and λ_{IP}, i.e., the latitude and longitude of a point at which the capsule would land if an abort occurred at this time.

In this computation three cases are considered:

a) The apogee height is below 5,000 feet. In this case the present subcapsule position is considered a sufficient approximation of the ϕ_{IP} and λ_{IP}.

b) The apogee height is between 5,000 and 228,000 feet. In this case the impact point at the altitude h_{f1} is computed.

c) The apogee height is above 228,000 feet. In this case the impact point at the altitude h_{f2} is computed.

The altitudes h_{f1} and h_{f2} (sometimes referred to as fictitious impact altitudes) are used because the ϕ and λ at h_{f1} or h_{f2} are a good approximation of the

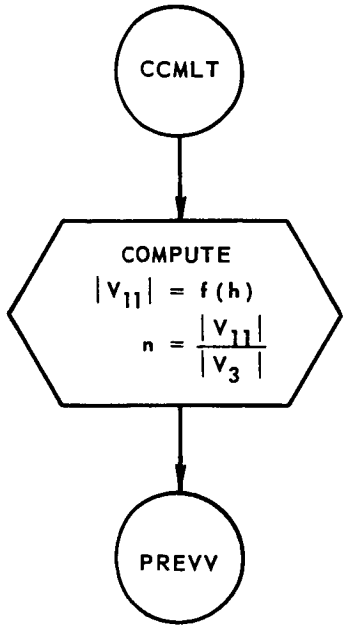


FIGURE 3-11. CCMLC BLOCK

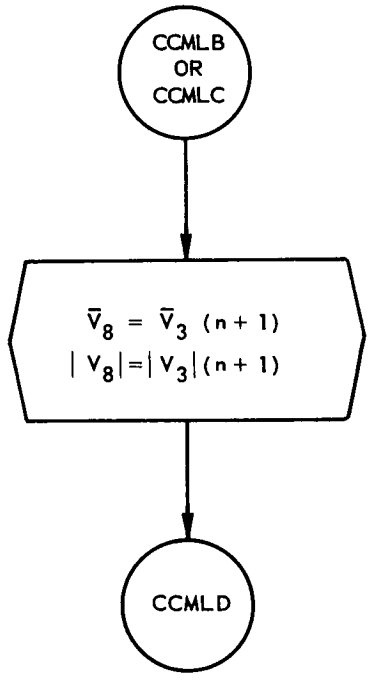


FIGURE 3-12. PREVV BLOCK

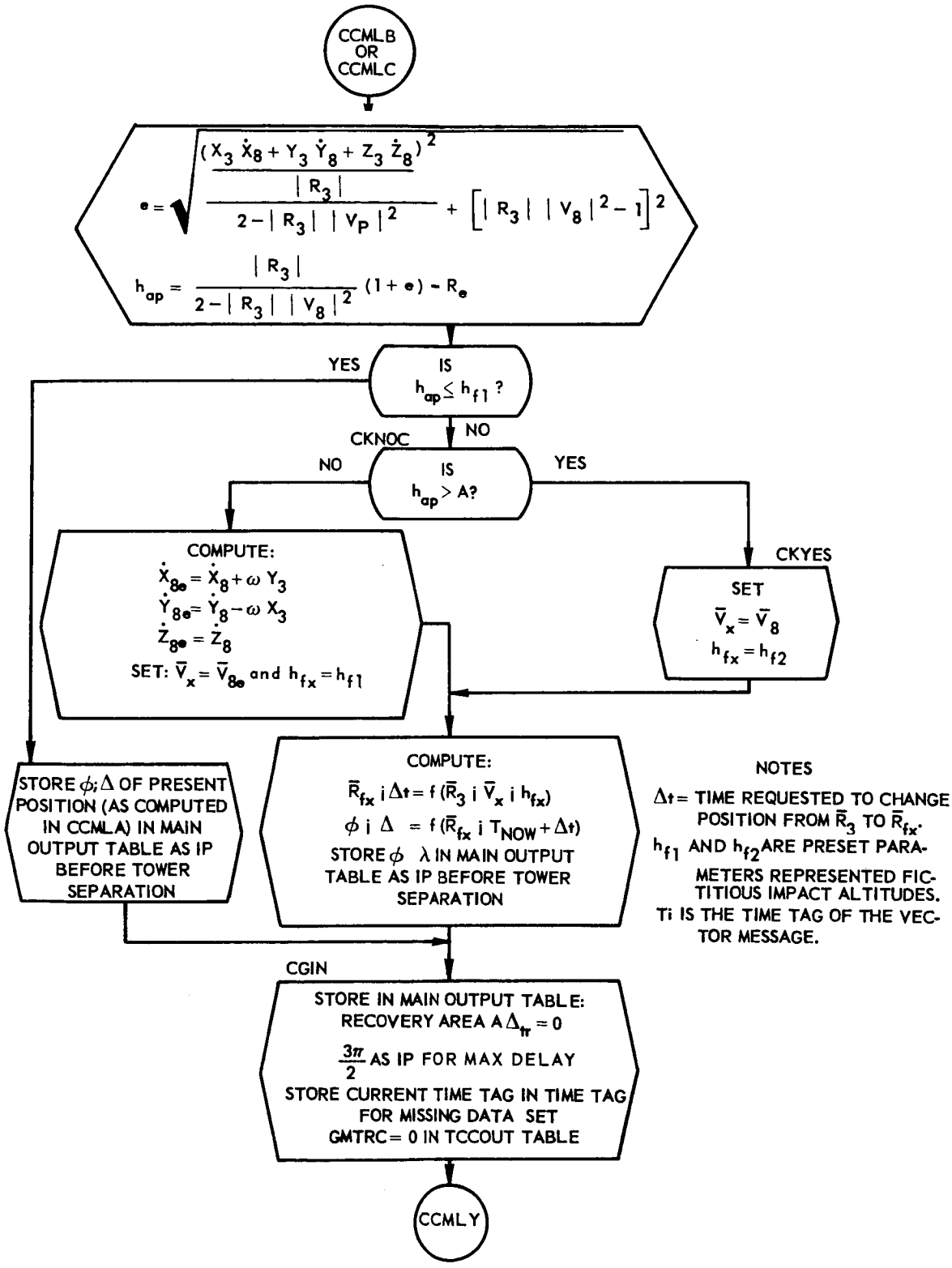


FIGURE 3-13 CCMLD BLOCK

actual IP (computed using numerical integration method). The C9RVTH program is used to simulate the path of the capsule.

The block starts with computation of apogee height:

$$h_{ap} = \frac{|R_3|}{2 - |R_3|\,|V_8|^2}(1 + e) - R_e$$

where:

$$e = \sqrt{\frac{(X_3\dot{X}_8 + Y_3\dot{Y}_8 + Z_3\dot{Z}_8)^2}{\dfrac{|R_3|}{2 - |R_3|\,|V_8|^2}} + \left[|R_3|\,|V_8|^2 - 1\right]^2}$$

and R_e is the nominal radius of the earth.

Next h_{ap} is compared with h_{f1}. If h_{ap} is greater than h_{f1} control goes to CKNOC, otherwise the ϕ_{PP} and λ_{pp} (previously computed in CCMAIN) are stored in TCCOUT + 7 and TCCOUT + 8 respectively and control goes to CGIN.

1) CKNOC

The apogee height (h_{ap}) is compared with the contents of (TCCONS + 32). If h_{ap} is greater, control goes to CKYES, otherwise the rotational velocity vector V_{8e} is computed as follows:

$$\dot{X}_{8e} = \dot{X}_8 + \omega Y_3$$

$$\dot{Y}_{8e} = \dot{Y}_8 + \omega X_3$$

$$\dot{Z}_{8e} = \dot{Z}_8.$$

Next the calling sequence for C9RVTH is set up so that $\overline{V}_{8e}$ and h_{f1} are used. Control then goes to CALS8 + 1.

2) CKYES

The calling sequence for C9RVTH is set up so that $\overline{V}_8$ and h_{f2} are used. Control then goes to CALS8 + 1.

3) CALS8 + 1

The C9RVTH program is used to compute the $\bar{R}$ at the previously selected fictitious impact altitude. The time (T) of arrival at this altitude is computed:

$$T = T_{now} + \Delta t$$

where Δt was computed by C9RVTH.

The A3MSCP program is used to compute ϕ_{IP} and λ_{IP} as a function of $\bar{R}$ and T. Control then passes to CGIN.

4) CGIN

The following quantities are stored in the output table: GMTRC and $\Delta t_r = 0$ because retrorocket velocity is not used.

The code of recovery area A is stored because this is the area in which landing would take place if an abort occurred at this time.

The quantity $\frac{3\pi}{2}$ is stored as λ_{max} to render motionless one of the arms on plotboard 1. The time tag (T_i) is saved and control then goes to CCMLY.

CCMLF (Figure 3-14)

The F, H, and G counters which are used in the computing cycles after capsule separation are initialized.

The $\bar{V}_{12}$ vector is computed on the assumption that 3 posigrades would fire if capsule separation occurred at this time.

$$\bar{V}_{12} = \bar{V}_3 \left(1 + \frac{V_p}{|V_3|}\right)$$

$$|V_{12}| = |V_3| + V_p$$

where V_p (TCCONS + 90) is the velocity increment supplied by the firing of 3 posigrades. Control then goes to CCMLG.

CCMLG (Figure 3-15)

This block computes the minimum retrofiring height ($h_{min\ ret}$) and certain other parameters which are required in subsequent computations. ($h_{min\ ret}$) is the height at which retrorockets have to be fired to ensure their burnout at 450,000 feet which is considered the beginning of the atmosphere.

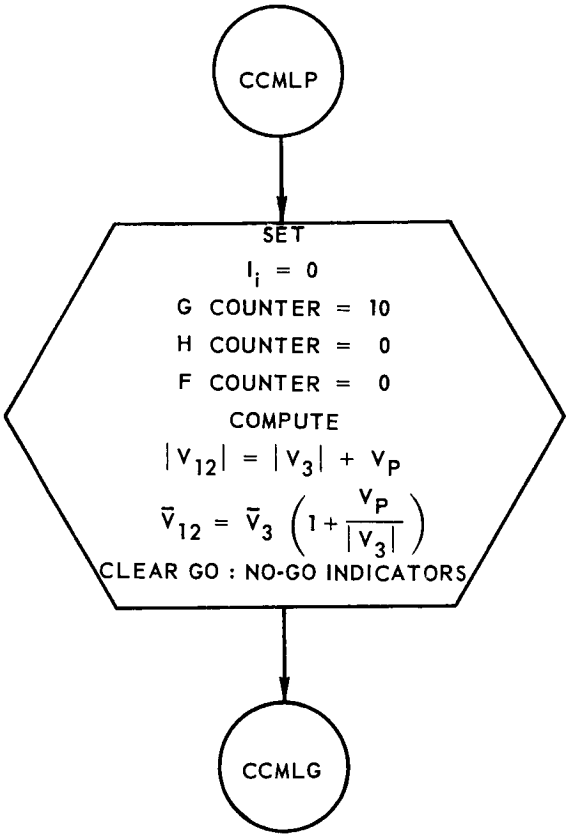


FIGURE 3-14. CCMLF BLOCK

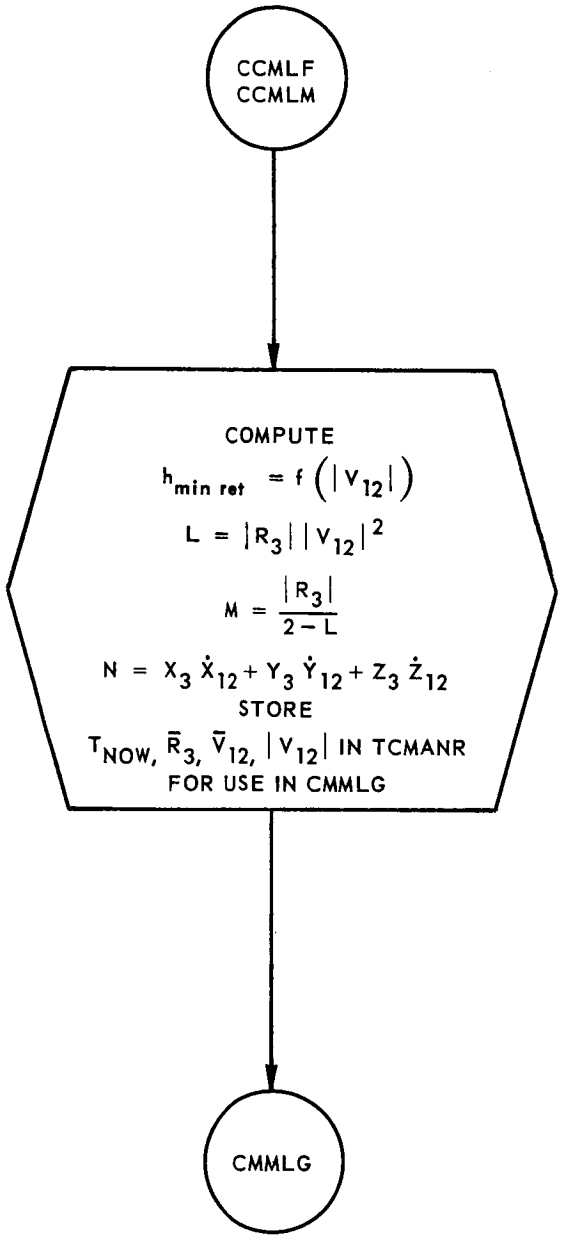


FIGURE 3-15. CCMLG BLOCK

$$h_{min\ ret} = a|V_{12}|^2 + b|V_{12}| + C$$

where a, b, c are in (TCCONS + 72 - TCCONS + 74)

The next calculations are performed to obtain the parameters required in the computation of eccentricity.

$$L = |R_3| \cdot |V_{12}|^2$$

$$M = \frac{|R_3|}{2 - L}$$

$$N = X_3\dot{X}_{12} + Y_3\dot{Y}_{12} + Z_3\dot{Z}_{12}$$

The $\bar{R}_3$ and $\bar{V}_{12}$ plus $|V_{12}|$ are then moved for use in the CMMLG section. section. Control now goes to CMMLG.

CMMLG (Figure 3-16, pages 3-54 - 3-60)

This block computes the minimum delay impact point (ϕ_{min}; λ_{min}) and maximum delay impact point (ϕ_{max}; λ_{max}), selects a recovery area, and computes the estimate of the time delay to fire retros (Δt_r) to land in the selected recovery area.

The minimum delay impact point is the point at which the capsule would impact if retrofiring took place 30 seconds after separation of the capsule from the rocket. The maximum delay impact point is the point at which the capsule would impact if retrorockets were fired so as to complete burning at 450,000 feet.

The computation of these impact points is made using the C9RVTH program (see Subsection 3.10) which simulates an elliptic trajectory, or the CHNOND program (see Subsection 3.11) in the case of low eccentricity. The EMFIT program (see Subsection 3.9) is used in addition to C9RVTH (or CHNOND) when velocities exceed 21,000 feet/sec because a more accurate impact point will be obtained. The R6BOTH program (see Subsection 3.13) is used to simulate retrorocket firing.

For velocities below 21,000 feet/sec an impact point at an altitude of 100,000 feet above a spherical earth, referred to as a fictitious impact altitude, is a good approximation of the actual impact point.

The block begins with the computation of eccentricity (e), apogee height (h_{ap}), perigee height (h_{per}).

$$e = \sqrt{\frac{N^2}{M} + (L - 1)^2}$$

$$h_{ap} = M(1 + e) - R_e$$

$$h_{per} = M(1 - e) - R_e$$

(M, N, L have been computed in either CCMLG or CMMLJ, R_e is a constant representing the nominal radius of the earth).

The "XB indicator" and the "Many areas indicator" are initialized.

If the G counter is 9 or 10, the time delay for the computation of the minimum delay IP ($\delta t_{r\ min}$) is initialized at 30 sec, otherwise:

$\Delta t_{r\ min} = 30 - (T_i - t)$ where T_i is either the time tag of the current vector or the time contained in cell MCSELM and the computation continues at CKTES.

The following is a more detailed discussion of blocks 1 through 36 that make up the overall CMMLG diagram shown on page 3-65.

CKTES (BLOCK 1)

A test of velocity is made to determine the method of computation of impact points. $h_{min\ ret}$ is the altitude at which the retrorockets should begin firing to completely burn out at 450,000 feet.

If $|V_{12}|$ is smaller than B (TCCONS + 76) control passes to CSTLU, otherwise h_{per} is compared with $h_{min\ ret}$.

If $h_{min\ ret}$ is larger than h_{per} control passes to CKNOD, otherwise control passes to CKYS.

CSTLU (BLOCK 2)

The parameter Δt_r is set equal to a nominal value (TCCONS + 30) and stored (using subroutine CKEN). Next, the calling sequence for C9RVTH is set

up in order to compute the $\bar{R}_4$ and $\bar{V}_4$ at time T_4 where $T_4 = T_{now} + t_{r\ min}$. If C9RVTH comes to the error return, $\bar{R}_4$ and $\bar{V}_4$ are computed using CHNOND.

Next, the calling sequence for R6BOTH is set up to compute $\bar{R}_5$ and $\bar{V}_5$ after retrofire. Then the calling sequence for C9RVTH is set up to compute the $\bar{R}$, T, at the fictitious impact altitude h_{f2}. If C9RVTH comes to the error return then the $\bar{R}$ and T are computed using CHNOND.

The above $\bar{R}$ and T are used to compute ϕ_{min} and λ_{min} using the A3MSCP program (see Subsection 3.14). ϕ_{min} and λ_{min} are stored in (TCCOUT + 7) and (TCCOUT + 8) respectively.

The apogee height h_{ap} is compared with $h_{min\ ret}$. If h_{ap} is larger than or equal to $h_{min\ ret}$, control is passed to CKMIP, otherwise control is passed to CKSET.

CKMIP (BLOCK 3)

This routine computes the IP for firing retros when the capsule has reached the altitude $h_{min\ ret}$ (also referred to as maximum delay IP, ϕ_{max}; λ_{max}). The calling sequence for C9RVTH is set up to compute $\bar{R}_4$ and $\bar{V}_4$ at altitude $h_{min\ ret}$ and also the time (Δt_{max}) which will elapse until this $h_{min\ ret}$ is reached.

$$T_4 = T_{now} + \Delta t_{max}$$

If C9RVTH comes to the error return, the $\bar{R}_4$, $\bar{V}_4$ and Δt are computed using CHNOND. Next, the calling sequence for R6BOTH is set up to compute $\bar{R}_5$ and $\bar{V}_5$ after retrofire. Then the calling sequence for C9RVTH is set up to compute the $\bar{R}$ and T at the fictitious impact altitude h_{f2}. If C9RVTH comes to the error return then $\bar{R}$ and T are computed using CHNOND.

The above $\bar{R}$ and T are used to compute ϕ_{max}, λ_{max} using A3MSCP. ϕ_{max} and λ_{max} are stored in (TCCOUT + 11) and (TCCOUT +12) respectively and control goes to CKST1.

CSK2P (BLOCK 4)

This block ensures that the longitude λ_{max} in (TCCOUT + 12) is less than 2π. Control then goes to CCMLQ which is outside of the block CMMLG.

CKNOD (BLOCK 5)

The calling sequence for C9RVTH is set up in order to compute $\bar{R}_4$ and $\bar{V}_4$ at time T_4 where $T_4 = T_{now} + \Delta t_{r\ min}$.

If C9RVTH comes to the error return, $\bar{R}_4$ and $\bar{V}_4$ are computed using CHNOND. Next, the calling sequence for R6BOTH is set up to compute $\bar{R}_5$ and $\bar{V}_5$ after retrofire. The magnitude ($|R_5|$) of the $\bar{R}_5$ vector is computed using library subroutine U3VMAG. Next, the altitude after retrofire (h_5) is computed:

$$h_5 = |R_5| - R_e$$

If h_5 is greater than or equal to 450,000 feet, control goes to CKABY. Otherwise, control goes to CKABN.

CKABY (BLOCK 6)

The calling sequence for C9RVTH is set up to compute $\bar{R}_6$, $\bar{V}_6$ and T_6 at the altitude of 450,000 feet. If C9RVTH comes to the error return $\bar{R}_6$, $\bar{V}_6$ and T_6 are computed using CHNOND. Next, the latitude (ϕ_6) and longitude (λ_6) at 450,000 feet are computed using the A3MSCP program. Next, the magnitude ($|R_6|$) of $\bar{R}_6$ and the magnitude ($|V_6|$) of $\bar{V}_6$ are computed using the library subroutine U3VMAG. The flight path angle (γ_6) at 450,000 feet is computed:

$$\gamma_6 = \arcsin \frac{X_6\dot{X}_6 + Y_6\dot{Y}_6 + Z_6\dot{Z}_6}{|R_6|\,|V_6|}$$

and control is transferred to the EMFIT program which computes ϕ_{min} and λ_{min}. Next, h_{ap} is compared with $h_{min\ ret}$. If h_{ap} is greater than or equal to $h_{min\ ret}$ control is passed to CKABZ, otherwise control passes to CKRTN.

CKABZ (BLOCK 7)

The calling sequence for C9RVTH is set up to compute $\bar{R}_4$, $\bar{V}_4$ at the altitude $h_{min\ ret}$ and also the time (Δt_{max}) which will elapse before this $h_{min\ ret}$ is reached.

$$T_4 = T_{now} + \Delta t_{max}.$$

If C9RVTH comes to the error return, $\bar{R}_4$, $\bar{V}_4$ and Δt_{max} are computed using CHNOND. Next, the calling sequence for R6BOTH is set up to compute $\bar{R}_5$ and $\bar{V}_5$ after retrofire. The magnitude ($|R_5|$) of $\bar{R}_5$ is computed using library subroutine U3VMAG. Next, the altitude after retrofire (h_5) is computed:

$$h_5 = |R_5| - R_e$$

If the magnitude of the difference between h_5 and 450,000 feet exceeds the limit specified in (TCCONS + 128), control passes to CKACZ. Otherwise, control goes to CKAAZ.

CKRTN (BLOCK 8)

The time to fire retros Δt_r is set to $\Delta t_{r\ min}$ and is stored (using subroutine CKEN). Control then passes to CKSET.

CKSET (BLOCK 9)

The latitude ϕ_{max} is set equal to the latitude ϕ_{min}. The longitude λ_{max} is set to the contents of location (TCCONS + 115). Control then passes to CKST1.

CKACZ (BLOCK 10)

The altitude (h_4) at which retros should begin firing, used in the computation of λ_{max}, is recomputed:

$$h_4 = h_{min\ ret} - (h_5 - 450{,}000 \text{ feet}).$$

The calling sequence for C9RVTH is set up to recompute $\bar{R}_4$, $\bar{V}_4$ at the altitude h_4 and also the time (Δt_{max}) which will elapse before h_4 is reached.

$$T_4 = T_{now} + \Delta t_{max}$$

If C9RVTH comes to the error return, $\bar{R}_4$, $\bar{V}_4$ and Δt are computed using CHNOND. Next, the calling sequence for R6BOTH is set up to compute $\bar{R}_5$ and $\bar{V}_5$ after retrofire. Control then passes to CKAAZ.

CKAAZ (BLOCK 11)

The calling sequence for C9RVTH is set up to compute $\bar{R}_6$, $\bar{V}_6$ and T_6 at the altitude of 450,000 feet. If C9RVTH comes to the error return, $\bar{R}_6$, $\bar{V}_6$ and

T_6 are computed using CHNOND. Next, the latitude (ϕ_6) and longitude (λ_6) at 450,000 feet are computed using the A3MSCP program. Next, the magnitude ($|R_6|$) of the $\bar{R}_6$ and the magnitude ($|V_6|$) of $\bar{V}_6$ are computed using the library subroutine U3VMAG. The flight path angle (γ_6) at 450,000 feet is computed:

$$\gamma_6 = \arcsin \frac{X_6\dot{X}_6 + Y_6\dot{Y}_6 + Z_6\dot{Z}_6}{|R_6|\ |V_6|}$$

and control is transferred to the EMFIT program which computes ϕ_{max} and λ_{max}. If the magnitude of the velocity, $|V_{12}|$, is less than the constant in location (TCCONS + 12), control passes to CSETA. Otherwise, the inclination angle is tested. If the inclination angle is larger than or equal to the constant in location (TCCONS + 129) control passes to CT340. Otherwise, the longitude λ_{min} is tested. If λ_{min} falls between the longitude of recovery area E and the longitude constant in (TCCONS), control is transferred to CSSEE (see block (35) for operational reason), otherwise control passes to CSTAE.

CKYS (BLOCK 12)

The calling sequence for C9RVTH is set up to compute the $\bar{R}_4$ and $\bar{V}_4$ at time T_4 where $T_4 = T_{now} + \Delta t_{r\ min}$.

If C9RVTH comes to the error return, $\bar{R}_4$ and $\bar{V}_4$ are computed using CHNOND. Next, the calling sequence for R6BOTH is set up to compute $\bar{R}_5$ and $\bar{V}_5$ after retrofire. Then, the calling sequence for C9RVTH is set up to compute $\bar{R}_6$, $\bar{V}_6$ and T_6 at the altitude of 450,000 feet.

If C9RVTH comes to the error return $\bar{R}_6$, $\bar{V}_6$ and T_6 are computed using CHNOND. Next, the latitude (ϕ_6) and longitude (λ_6) at 450,000 feet are computed using the A3MSCP program.

Next, the magnitude ($|R_6|$) of $\bar{R}_6$ and the magnitude ($|V_6|$) of $\bar{V}_6$ are computed using the library subroutine U3VMAG. The flight path angle (γ_6) at 450,000 feet is computed:

$$\gamma_6 = \arcsin \frac{X_6\dot{X}_6 + Y_6\dot{Y}_6 + Z_6\dot{Z}_6}{|R_6|\quad |V_6|}$$

and control is transferred to the EMFIT program which computes ϕ_{min} and λ_{min}. If λ_{min} is larger than the constant in (TCCONS) control goes to CKABW; otherwise, control passes to CSSEE.

CSSEE (BLOCK 13)

The code for recovery area E is stored in (TCCOUT + 17) and the longitude of recovery area E (λ_E) is stored in (TCMANR + 182) and control passes to CLDTM.

CKABW (BLOCK 14)

The code for recovery area XB is stored in (TCCOUT + 17) and the longitude of recovery area XB (λ_{XB}) is stored in (TCMANR + 182). The "XB indicator", interrogated in CKDTR, is set in (TCMANR + 187). If the magnitude of the difference between λ_{min} and λ_{XB} is larger than the limit stored in (TCCONS + 94), control passes to CKBW. Otherwise, control goes to CLDTM.

CLDTM (BLOCK 15)

The time to fire retros (Δt_r) is made equal to $\Delta t_{r\ min}$. Control then goes to CLCDT.

CLCDT (BLOCK 16)

The time Δt_r is converted from floating point seconds to integral minutes and seconds and stored in (TCCOUT +14). Control then passes to CLSNI.

CKBW (BLOCK 17)

The time delay to fire retros ($\Delta t_{r\ max}$) is arbitrarily set to the constant in location (TCCONS + 95).

The calling sequence for C9RVTH is set up to compute the $\bar{R}_4$ and $\bar{V}_4$ at time T_4 where $T_4 = T_{now} + \Delta t_{r\ max}$.

If C9RVTH comes to the error return, $\bar{R}_4$ and $\bar{V}_4$ are computed using CHNOND. Next, the calling sequence for R6BOTH is set up to compute $\bar{R}_5$ and $\bar{V}_5$ after retrofire. Then the calling sequence for C9RVTH is set up to compute $\bar{R}_6$, $\bar{V}_6$ and T_6 at the altitude of 450,000 feet.

If C9RVTH comes to the error return, $\bar{R}_6$, $\bar{V}_6$ and T_6 are computed using CHNOND. Next, the latitude (ϕ_6) and longitude (λ_6) at 450,000 feet are computed using the A3MSCP program.

Then the magnitude ($|R_6|$) of R_6 and the magnitude ($|V_6|$) of $\bar{V}_6$ are computed using the library subroutine U3VMAG.

The flight path angle (γ_6) at 450,000 feet is computed:

$$\gamma_6 = \arcsin \frac{X_6\dot{X}_6 + Y_6\dot{Y}_6 + Z_6\dot{Z}_6}{|R_6|\ |V_6|}$$

and control is transferred to the EMFIT program which computes ϕ_{max} and λ_{max}. If λ_{max} is greater than or equal to $\frac{\pi}{2}$ control passes to CKAC. Otherwise, 2π is added to λ_{max} and control passes to CKAC.

CKAC (BLOCK 18)

The estimate of Δt_r to land in the selected recovery area is computed by linear interpolation between $\Delta t_{r\ min}$ and $\Delta t_{r\ max}$.

$$\Delta t_r \left[\frac{(\lambda_{min} - \lambda_{sel})(\Delta t_{r\ max} - \Delta t_{r\ min})}{\lambda_{min} - \lambda_{max}} + \Delta t_{r\ min}. \right]$$

Δt_r is stored (using subroutine CKEN) and the "No iteration required" indicator is cleared which will allow an iterative procedure to be used in CCMLR (see write up of CCMLR). Control is then passed to CKDTR.

CKDTR(BLOCK 19)

If the "XB indicator" is not set control passes to CSK2P. Otherwise, the constant in location (TCCONS +115) is stored in (TCCOUT +12) (thus superseding λ_{max} which cannot be displayed on the plotboard) and control passes to CCMLQ.

CKEN (BLOCK 20)

This is a closed subroutine which converts Δt_r from floating point seconds to integral minutes and seconds and stores them in (TCCOUT +14).

CKST1 (BLOCK 21)

The code for recovery area A is stored in (TCCOUT + 17) and control passes to CLSNI.

CLSNI (BLOCK 22)

The "No iteration required" indicator is set (for use in CCMLR to indicate that nothing is to be gained by trying to refine Δt_r). λ_{max} is stored in (TCMANR + 220) and control passes to CSK2P.

CKABN (BLOCK 23)

The calling sequence for C9RVTH is set up to compute the $\bar{R}$, T at the altitude h_{f2}. If C9RVTH comes to the error return then $\bar{R}$ and T are computed using CHNOND.

The above $\bar{R}$ and T are used to compute ϕ_{min} and λ_{min} using the A3MSCP program. ϕ_{min} and λ_{min} are stored in (TCCOUT + 7 and TCCOUT + 8) respectively and control passes to CKRTN.

CLHMR (BLOCK 24)

This block determines the number of recovery areas that are located between λ_{min} and λ_{max}. There are 3 possibilities:

a) No recovery area between λ_{min} λ_{max}. In this case, control goes to CLNON.

b) One recovery area between λ_{min} and λ_{max}. This is the usual case. The code number of this area is then stored in (TCCOUT + 17) and control goes to CKONE.

c) More than one recovery area between λ_{min} and λ_{max}. In this case, control goes to CKMNY.

CLNON (BLOCK 25)

The difference (δ_2) between λ_{max} and the longitude of the adjoining recovery area east of λ_{max} is computed. The difference (δ_1) between λ_{min} and the adjoining recovery area west of λ_{min} is also computed. If δ_2 is smaller than δ_1 control

goes to CLEST, otherwise the code of the area west of λ_{min} is stored in (TCCOUT + 17) and the longitude of this recovery area is stored in TCMANR + 182 and control goes to CLDTM.

CLEST (BLOCK 26)

The code of the recovery area east of λ_{max} is stored in (TCCOUT + 17), the longitude of this recovey area is stored in (TCMANR + 182) and control passes to CLDMX.

CKMNY (BLOCK 27)

Since there is more than one recovery area that could be selected a choice must be made. Operational requirements demand selection of the easternmost area, unless it happens to be the area in the Indian Ocean (area XB). If the test shows area XB is not the easternmost area, control is transferred to CLAXB; otherwise, area E or E′ (as determined in block CLAEP or CSTAE) becomes the selected area. The code of the area is stored in (TCCOUNT + 17) and the longitude of the area is stored in TCMANR + 182. Control then passes to CK2F.

CLDMX (BLOCK 28)

The estimate of the time delay to retrofire (Δt_r) is set equal to the time delay to retrofire associated with the (prior) computation of λ_{max} because λ_{max} is as close to λ_{sel} as required by the tolerance limit or because this is the best approximation that can be obtained when there was no recovery area to choose from. Control then goes to CLCDT.

CKONE (BLOCK 29)

The longitude of the selected recovery area is stored in (TCMANR + 182) and control goes to CK2F.

CK2F (BLOCK 30)

If λ_{sel} is as close to λ_{min} as the tolerance limit (TCCONS + 94) requires control passed to CLDTM; otherwise, control goes to CL2F.

CL2F (BLOCK 31)

If λ_{sel} is as close to λ_{max} as the tolerance limit (TCCONS + 94) requires control passes to CLDMX; otherwise, control goes to CKAC.

CSTAE (BLOCK 32)

There are two interchangeable recovery areas, namely E and E′. The longitude of area E is stored in the list of areas for future use. This is done because the test of the inclination angle in CKAAZ determined that landing in area E′ is impossible. Control then goes to CLHMR.

CLAEP (BLOCK 33)

The longitude of area E′ is stored in the list of areas because landing in area E is impossible. (For detailed explanation see CSTAE above.) Control then goes to CLHMR.

CSETA (BLOCK 34)

Recovery area A is being selected by storing its code in (TCCOUT + 17). This is done because control reaches this routine when the velocity is not high enough to make the other recovery areas applicable. Δt_r is set equal to a nominal value (TCCONS + 30) and is stored using subroutine CKEN. Control then passes to CLSNI.

CT340 (BLOCK 35)

A test is made to determine whether λ_{min} is east of the longitude of recovery area E′. If it is, this implies that in case of abort, landing must take place east of area E . Although there is no other landing area between area E′ and the west coast of Africa, it is operationally desirable in this case to land (in case of abort) west of 347 degrees (= 13 degrees W) provided λ_{min} shows that this is possible. Then program control passes to CSSEE. If the test shows that λ_{min} is not between the longitude of area E′ and 347 degrees, program control goes to CLAEP.

CLAXB (BLOCK 36)

The easternmost of the recovery areas between λ_{min} and λ_{max} is selected by storing the code of the area in (TCCOUT + 17) and the longitude of the area in (TCMANR + 182).

The "many areas indicator" is set to signify to the CCLOK program that another recovery area is also located between λ_{min} and λ_{max}. Control then passes to CL2F.

CCMLH (Figure 3-17)

The velocity increment due to the firing of posigrade rockets is computed and added to the input velocity. The flight path angle is recomputed to reflect this change of velocity.

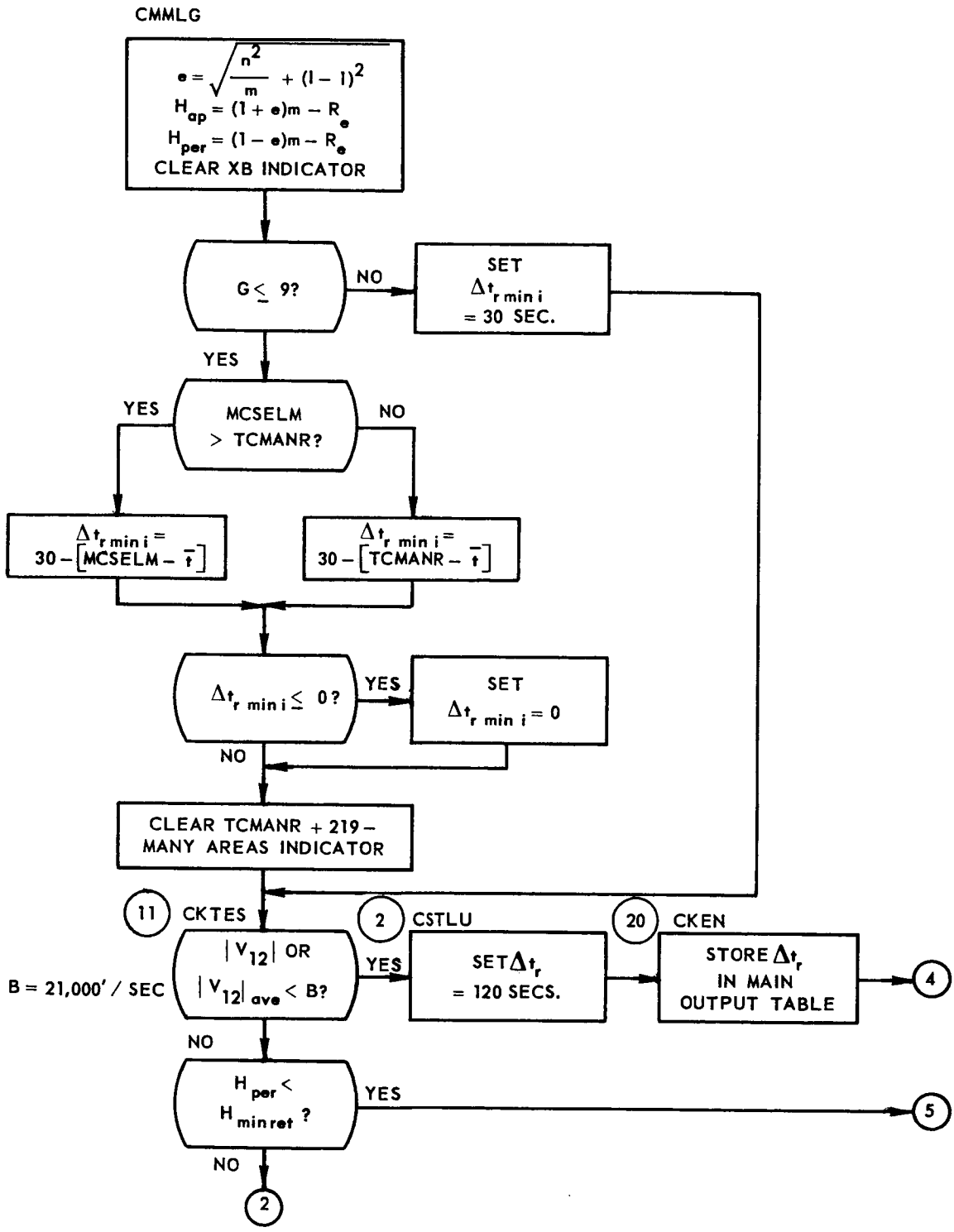


FIGURE 3-16. CMMLG BLOCK *(Sheet 1 of 7)*

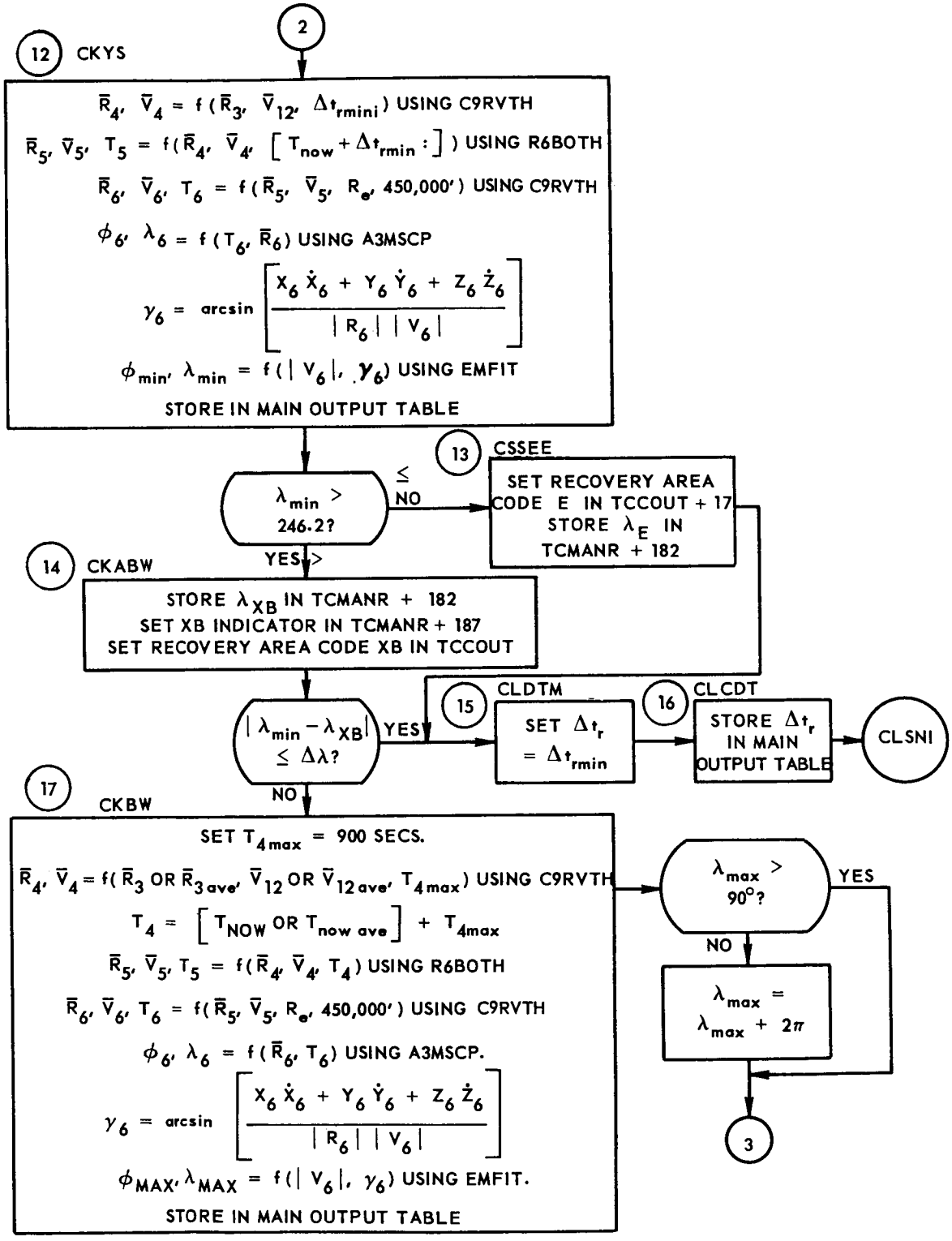


FIGURE 3-16. CMMLG BLOCK *(Sheet 2 of 7)*

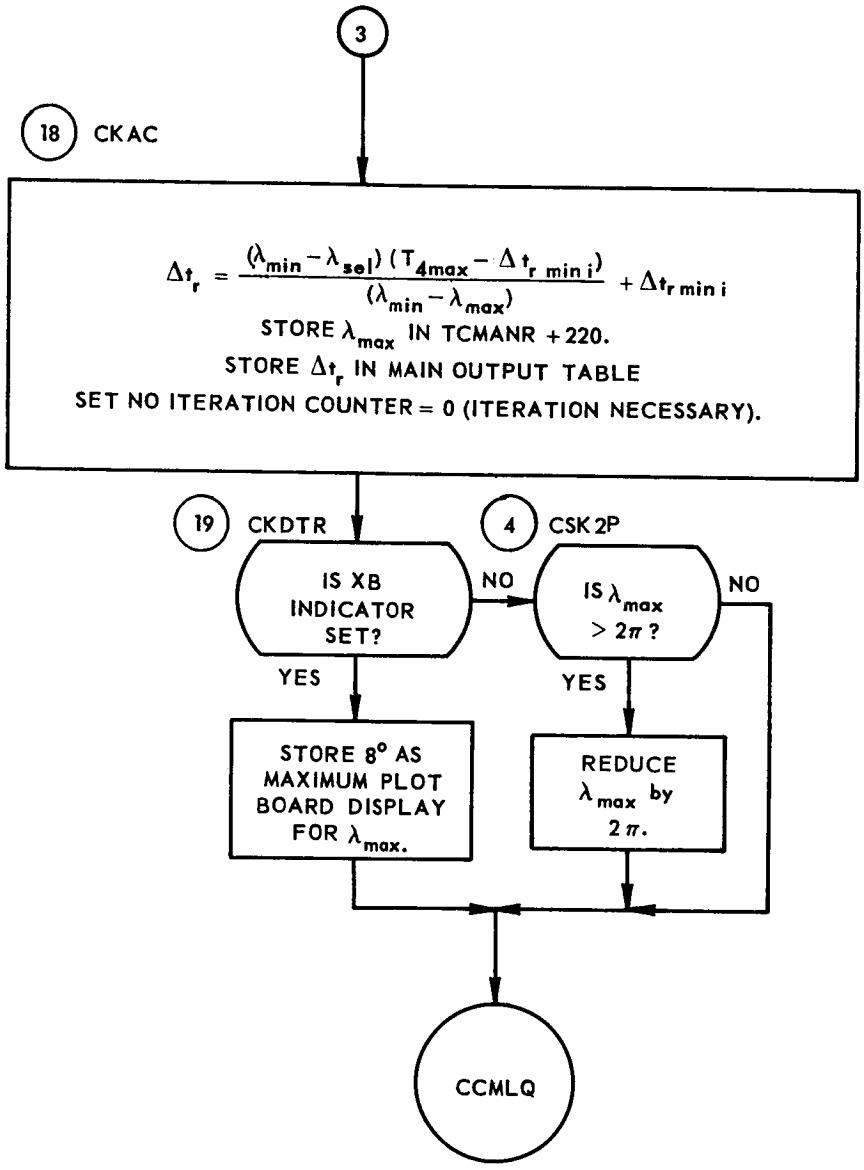


FIGURE 3-16. CMMLG BLOCK *(Sheet 3 of 7)*

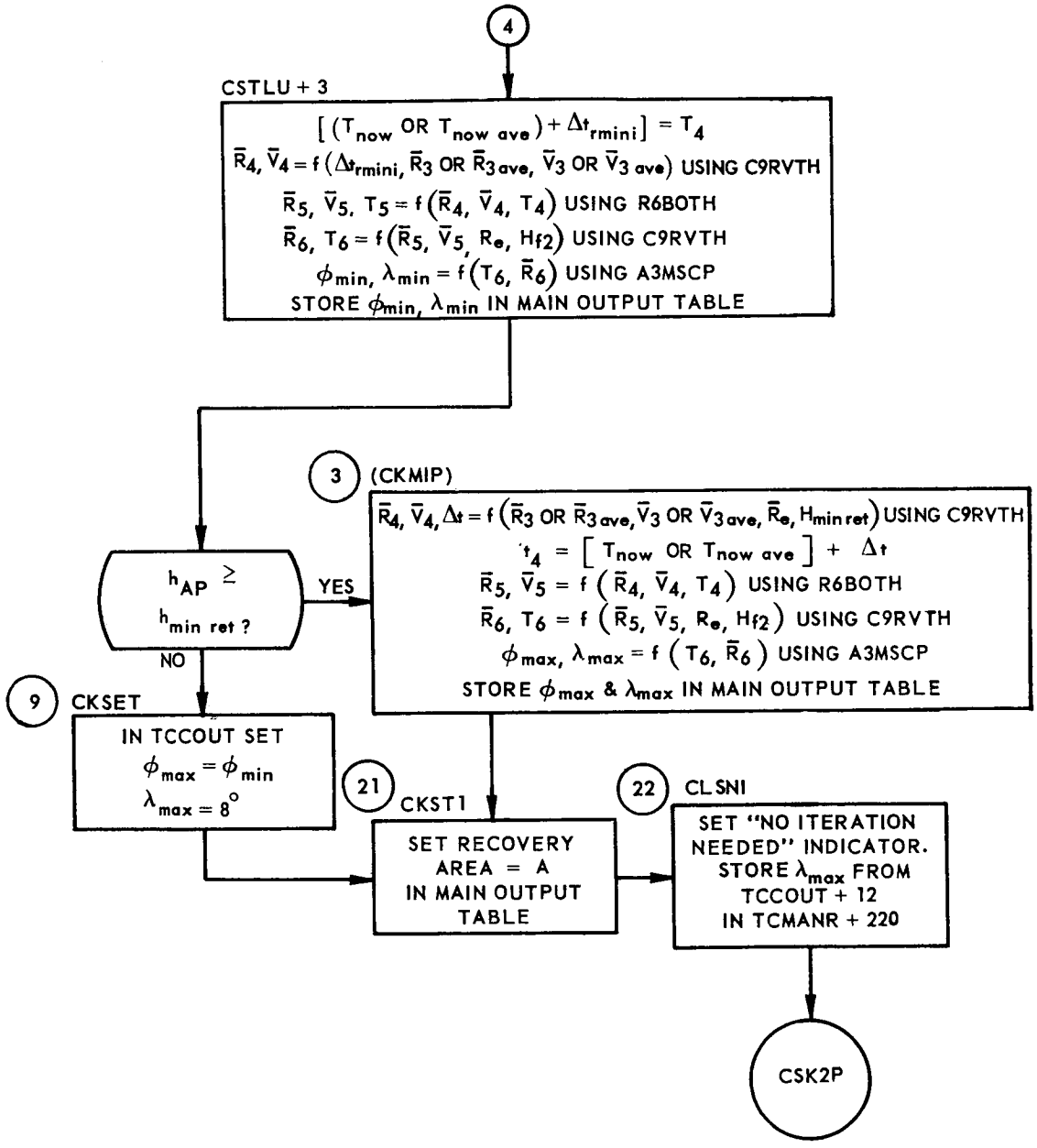


FIGURE 3-16. CMMLG BLOCK *(Sheet 4 of 7)*

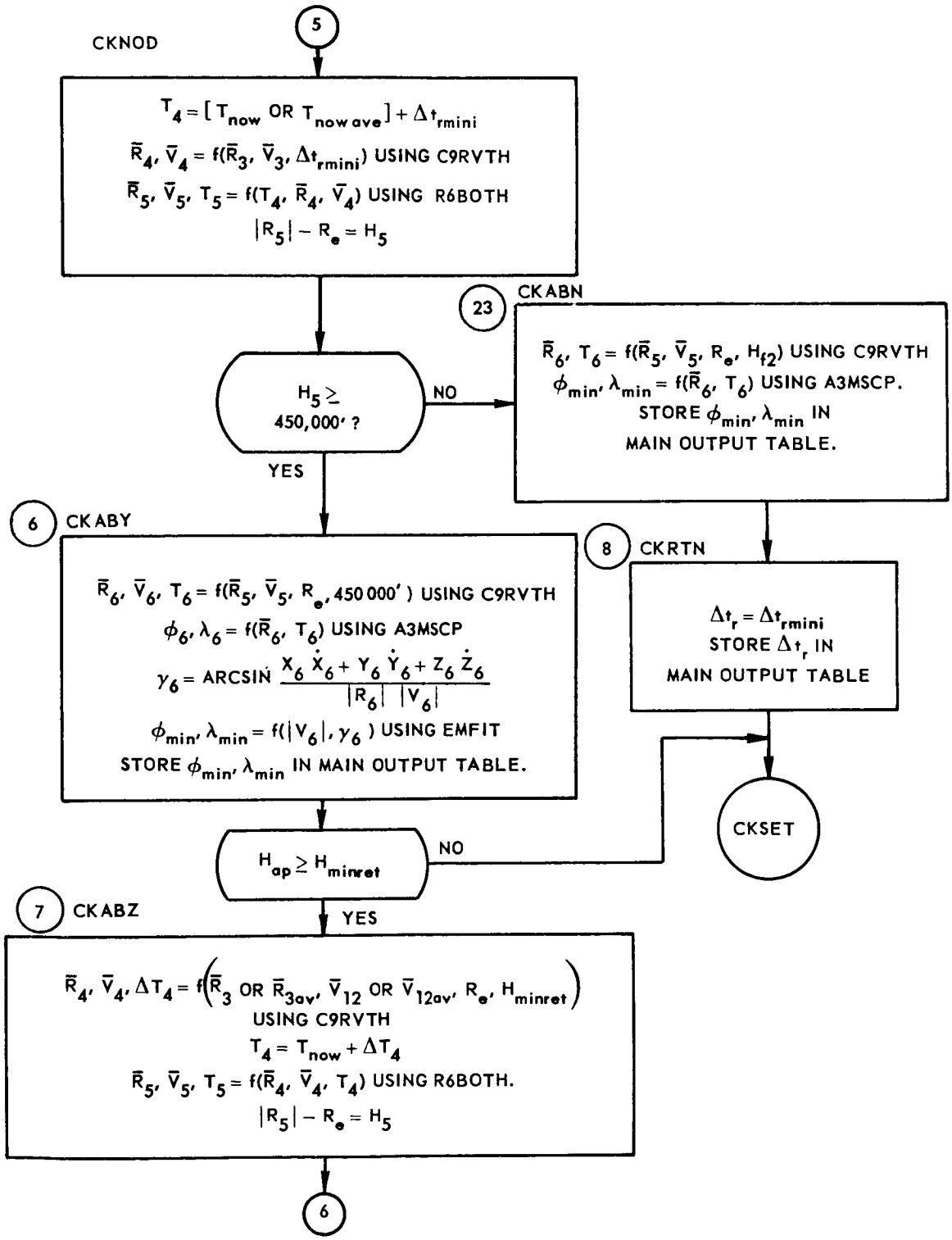


FIGURE 3-16. CMMLG BLOCK *(Sheet 5 of 7)*

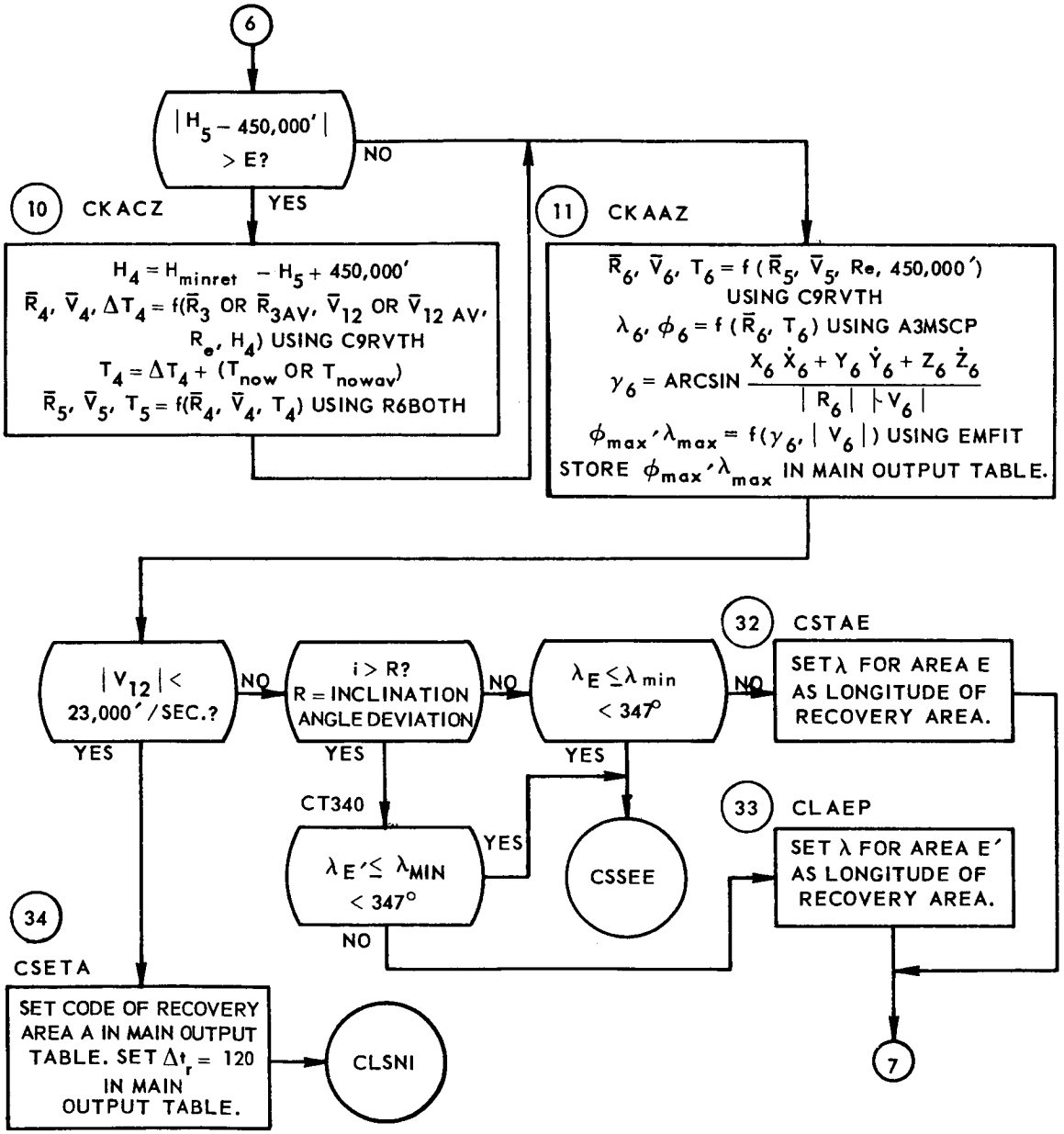


FIGURE 3-16. CMMLG BLOCK *(Sheet 6 of 7)*

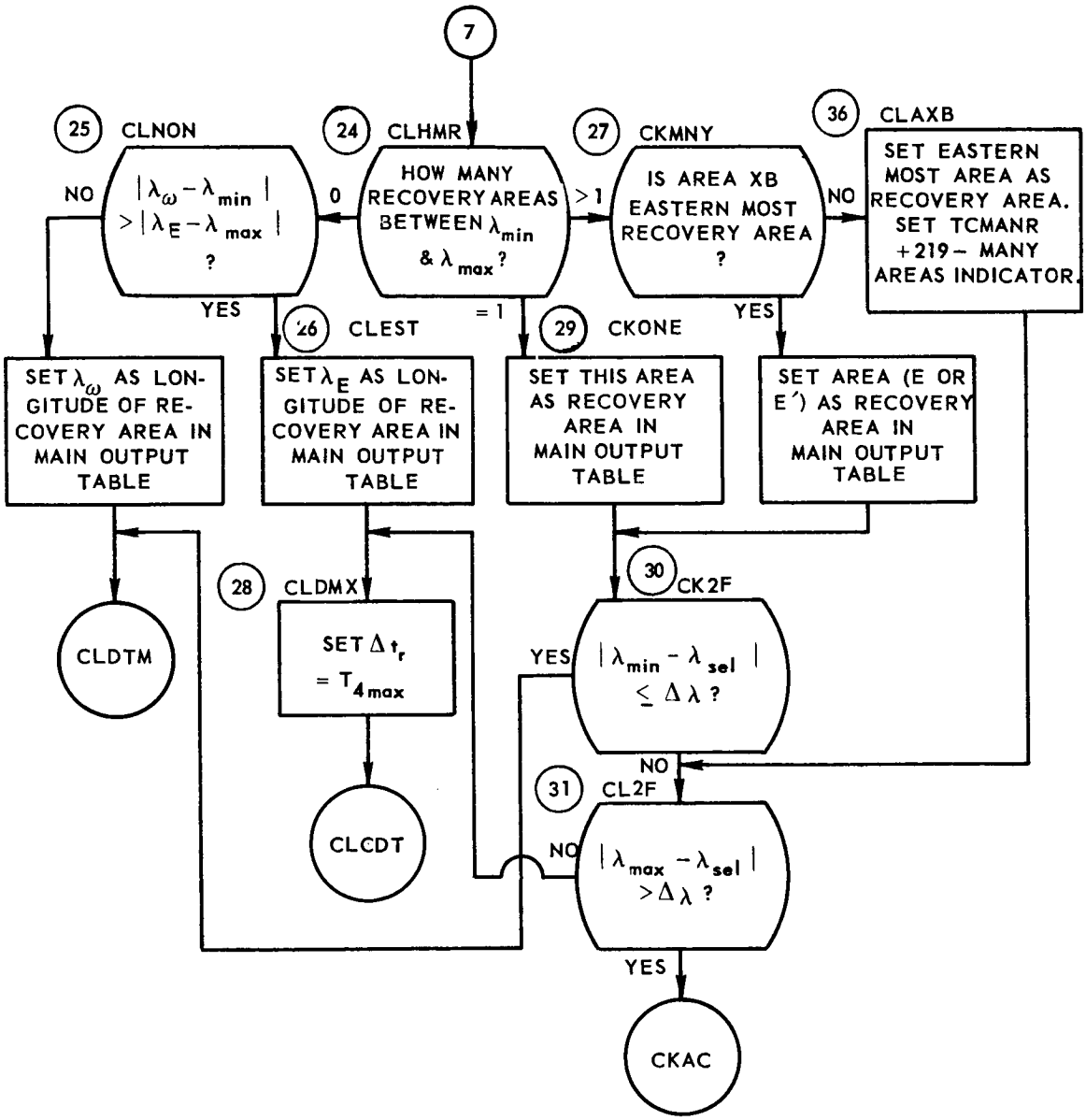


FIGURE 3-16. CMMLG BLOCK *(Sheet 7 of 7)*

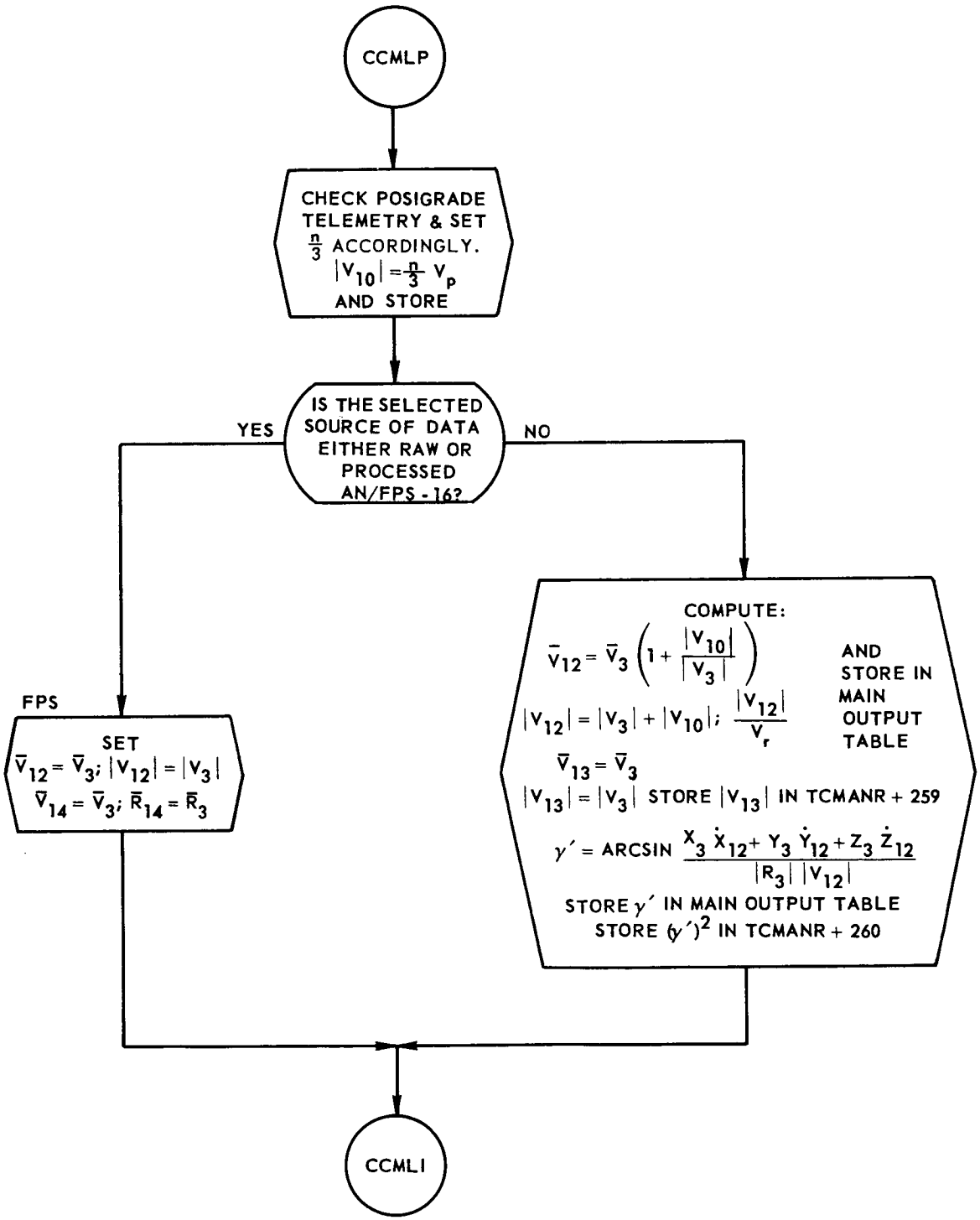


FIGURE 3-17 CCMLH BLOCK

$$| V_{10} | = \frac{n}{3} | V_p |$$

where "n" is the number of posigrades fired and V_p is the total velocity increment supplied by the firing of 3 posigrades.

A test is made to determine the source of the input. If the data is AN/FPS-16, the $\bar{R}_3$ and $\bar{V}_3$ are saved and control goes to CCMLI, otherwise $\bar{V}_{12}$ is computed using the following equation:

$$\bar{V}_{12} = \bar{V}_3 \left(1 + \frac{| V_{10} |}{| V_3 |} \right)$$

The following equation is then used to recompute the flight path angle:

$$\gamma' = \arcsin \frac{X_3\dot{X}_{12} + Y_3\dot{Y}_{12} + Z_3\dot{Z}_{12}}{| R_3 | \; | V_{12} |}$$

γ' is then stored in (TCCOUT + 20) for display on the Flight Dynamics Officer's Console and plotboard 1 at the Mercury Control Center. Control then goes to CCMLI.

CCMLI

CCMLI determines if this is the first pass after CAPS by testing the G counter (TCMANR + 16) for a value greater than 9. If this is the first pass, a transfer is made to CCMLK, otherwise, a transfer is made to CCMLL.

CCMLJ (Figure 3-18)

In this block the $\bar{R}_{3ave}$ and $\bar{V}_{12ave}$ are extrapolated to the nearest minute for use by the orbit program. Based on velocity and flight path angle a GO, NO-GO recommendation is made. If the recommendation is GO, the orbit capability is also computed.

The GO or NO-GO recommendation is computed using a polynomial expression in γ to compute the minimum velocity required to enter orbit (V_{go}).

The velocity vector V_{13ave} and the velocity magnitude $| V_{13} |_{ave}$ are incremented by the velocity due to the firing of posigrade rockets in the following manner:

$$\bar{V}_{12ave} = \bar{V}_{13ave} \left(1 + \frac{| V_{10} |}{| V_{13} |_{ave}} \right)$$

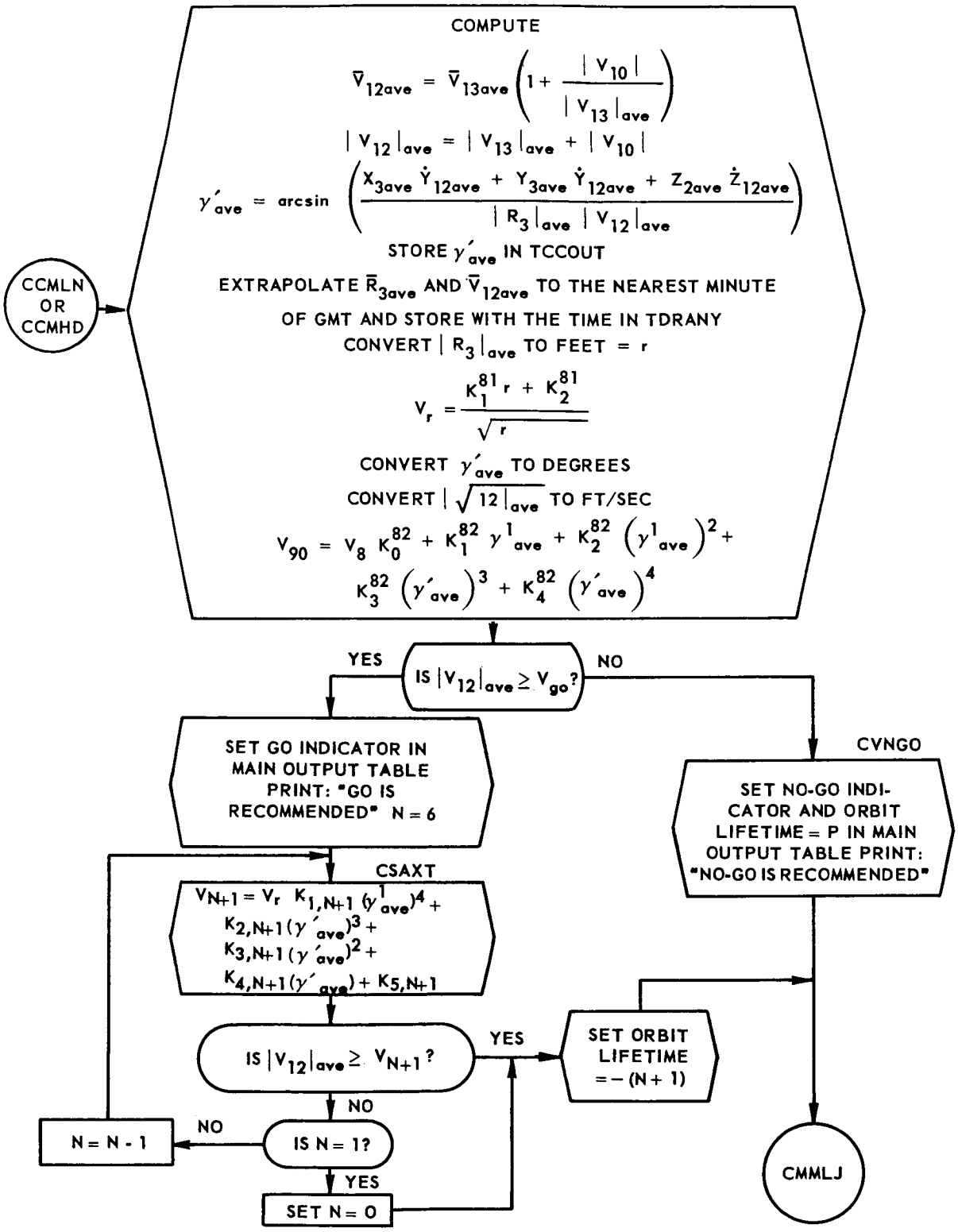


FIGURE 3-18 CCMLJ BLOCK

$$|V_{12\ ave}| = |V_{13\ ave}| + |V_{10}|$$

where $|V_{13\ ave}|$ is the magnitude of the average input vector $|V_3|$ plus the tail-off velocity, which is currently assumed to be zero.

The average flight path angle (γ'_{ave}) must then be updated to the change in velocity:

$$\gamma'_{ave} = \arcsin \frac{X_{3ave}\dot{X}_{12ave} + Y_{3ave}\dot{Y}_{12ave} + Z_{3ave}\dot{Z}_{12ave}}{|R_{3\ ave}|\ |V_{12\ ave}|}$$

The average position and velocity vectors are then extrapolated forward to the next minute of GMT using the subroutine CVEXT and stored in the TDRANV table for use by the orbit progrms.

The speed V_R is computed as a function of the distance from the center of the earth to the capsule.

$$V_R = \frac{K_1^{81} r + K_2^{81}}{\sqrt{r}}$$

where $r = |R_{3ave}|$ in feet

$$K_1^{81} = -2.026218$$

$$K_2^{81} = 1.6229852E8$$

A minimum GO velocity is then computed as a function of V_R and the flight path angle as follows:

$$V_{go} = V_r \left[K_0^{82} + K_1^{81}\gamma'_{ave} + K_2^{82}(\gamma'_{ave})^2 + K_3^{82}(\gamma'_{ave})^3 + K_4^{82}(\gamma'_{ave})^4\right]$$

The computed average velocity $|V_{12ave}|$ is then compared with the minimum GO velocity described above. If the average velocity is greater than or equal to the minimum GO velocity, a GO is recommended, the orbit capability is then determined in the following iterative manner.

The capability speed V_{N+1} is computed where N is set to 6 or N + 1 is the orbit capability, making the initial test for a 7 orbit capability.

$$V_{N+1} = V_r \left[K_{1,N+1} (\gamma'_{ave})^4 - K_{2,N+1} (\gamma'_{ave})^3 + K_{3,N+1} (\gamma'_{ave})^2 + K_{4,N+1} (\gamma'_{ave}) + K_{5,N+1} \right.$$

The speed V_{N+1} is then compared with $|V_{12\ ave}|$. If it is larger than V_{N+1} the orbit capability is set to N + 1; if smaller, N is decreased by one and a capability speed is computed using the above equation with the decreased value of N. This is repeated until $|V_{12\ ave}|$ is larger than V_{N+1} or N becomes zero. In either case, the orbit capability is set to N + 1. If a NO-GO decision was made, the orbit capability is set minus to freeze the output displays and clear the plotboards during the hold phase. The only values which are not frozen are the orbit capability and the GO, NO-GO indicator.

Control is transferred to CCMLR after the on-line message for the GO, NO-GO decision has been set up to be printed.

CMMLJ (Figure 3-19)

The height $h_{min\ ret}$ at which retrofiring must be initiated so that the retro-rockets will burn out at 450,000 feet is computed. Certain quantities needed in the computation of eccentricity are also calculated based on the average position and velocity vectors.

$$h_{min\ ret} = a\,|V_{12\,ave}|^2 - b\,|V_{12\ ave}|^{-c}$$

where a, b, c are in (TCCONS + 72 - TCCONS + 74)

$$L = |R_{3\ ave}| \cdot |V_{12\ ave}|^2$$

$$M = \frac{|R_{3\ ave}|}{2 - L}$$

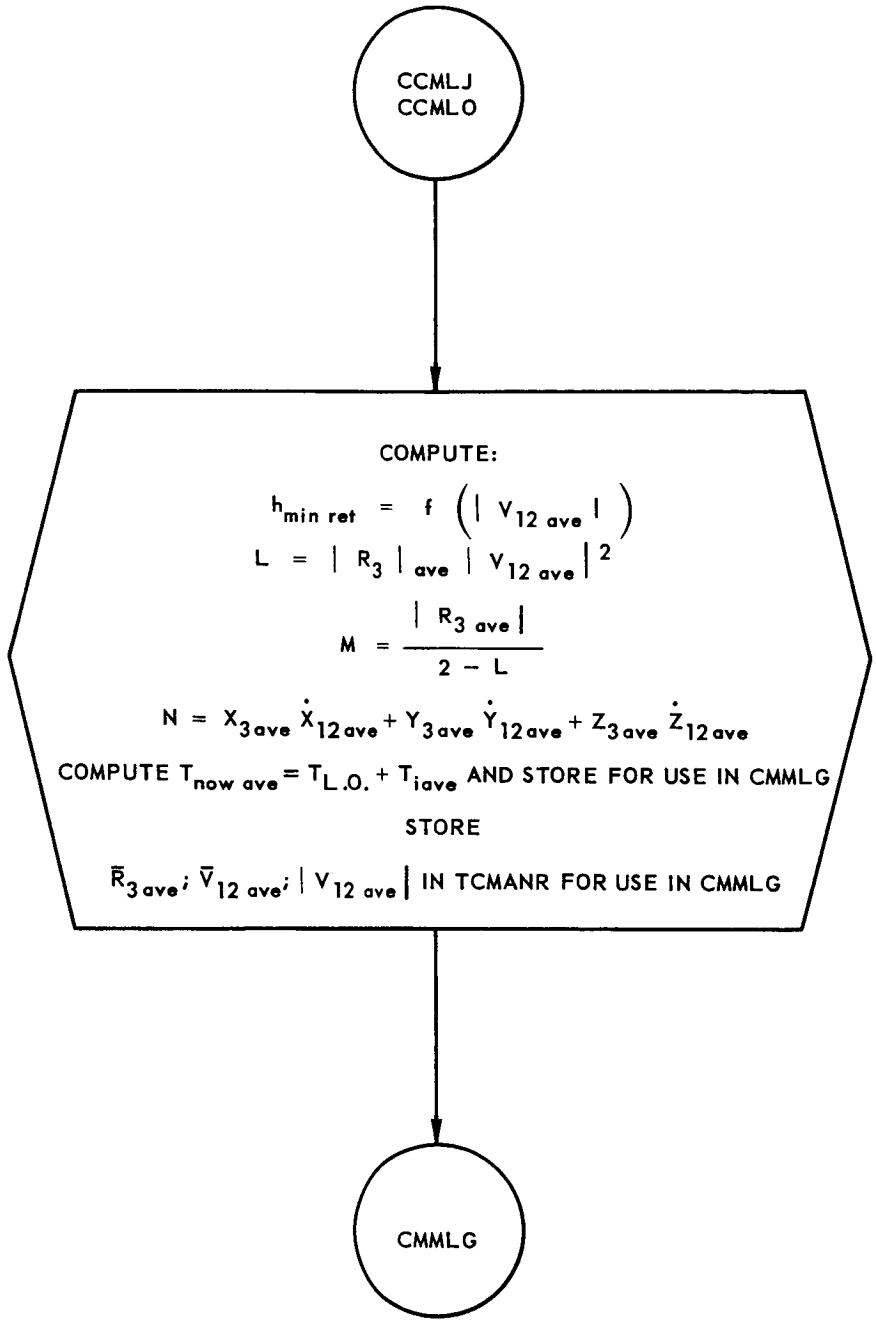


FIGURE 3-19. CMMLJ

$$N = X_{3ave}\dot{X}_{12ave} + Y_{3ave}\dot{Y}_{12ave} + Z_{3ave}\dot{Z}_{12ave}$$

$$T_{now\ ave} = T_{L.O.} - T_{i\ ave}$$

The computations in this section are similar to CCMLG. Mean quantities, however, are used here. Transfer is then made to CMMLG.

CCMLK (Figure 3-20)

This block performs the initializations required for the accumulation of the vector components obtained during the 10 passes after CAPS. This accumulation is needed so that the average $\overline{R}$ and $\overline{V}$ can be computed in subsequent blocks. Control then goes to CCMLL.

CCMLL (Figure 3-21)

In CCMLL the vector components are accumulated and the counters used in the 10 passes are updated after CAPS. The program determines if the data is AN/FPS-16, and if it is, it sums the $\overline{R}_{14}$, $\overline{V}_{14}$, and T_{14} by the addition of the new values to the previous sum (TCMANR + 206 to + 212). Also the F counter for AN/FPS-16 data (TCMANR + 192) received during these 10 passes is incremented by 1 . Control then goes to INCG.

If the data is B-GE or Azusa $|V_{13}|$ and γ' are saved and a running sum of $|V_{13}|$, γ', $|V_{13}|^2$, $(\gamma')^2$ is maintained. Also, $\overline{R}_3$, $\overline{V}_{13}$ and T_i are saved in the RVT70 table. A count of the number of points saved is maintained by the H counter (TCMANR + 17). Control then goes to INCG.

INCG

The count of computing cycles after CAPS is maintained by the G counter and control is transferred to CCMLW.

CCMLN (Figure 3-22)

The computation of mean R and V vectors is made. Statistical methods (t distribution) are used to eliminate vector components which would bias the mean. The inputs to the computation are the $\overline{R}_3$ and $\overline{V}_{13}$ accumulated during the 10 cycles after capsule separation.

If five or more points are received during the 10 cycles following capsule separation, a standard deviation (δ) is calculated for both the velocity and the flight path angle, according to the following formulas:

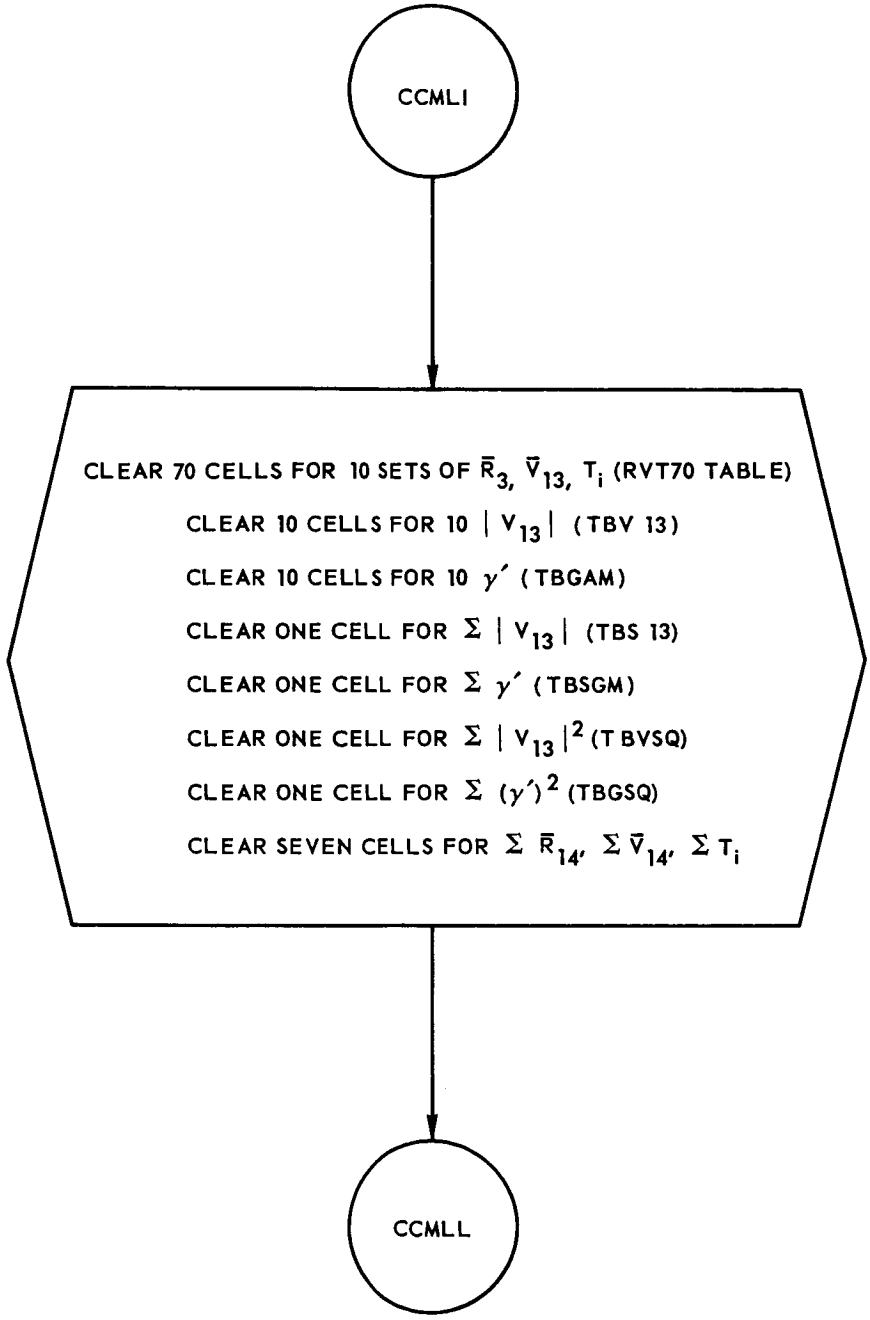


FIGURE 3-20. CCMLK BLOCK

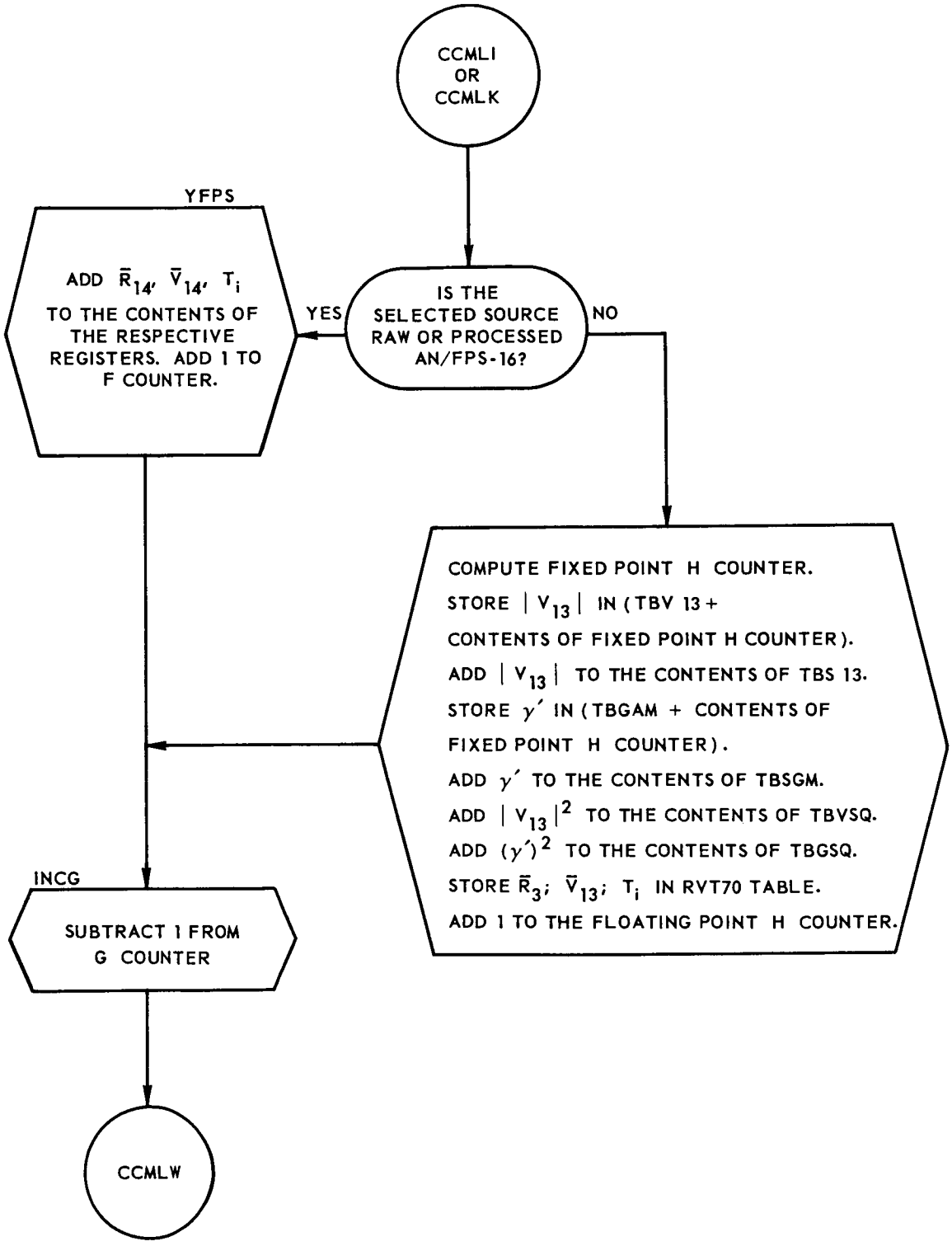


FIGURE 3-21. CCMLL BLOCK

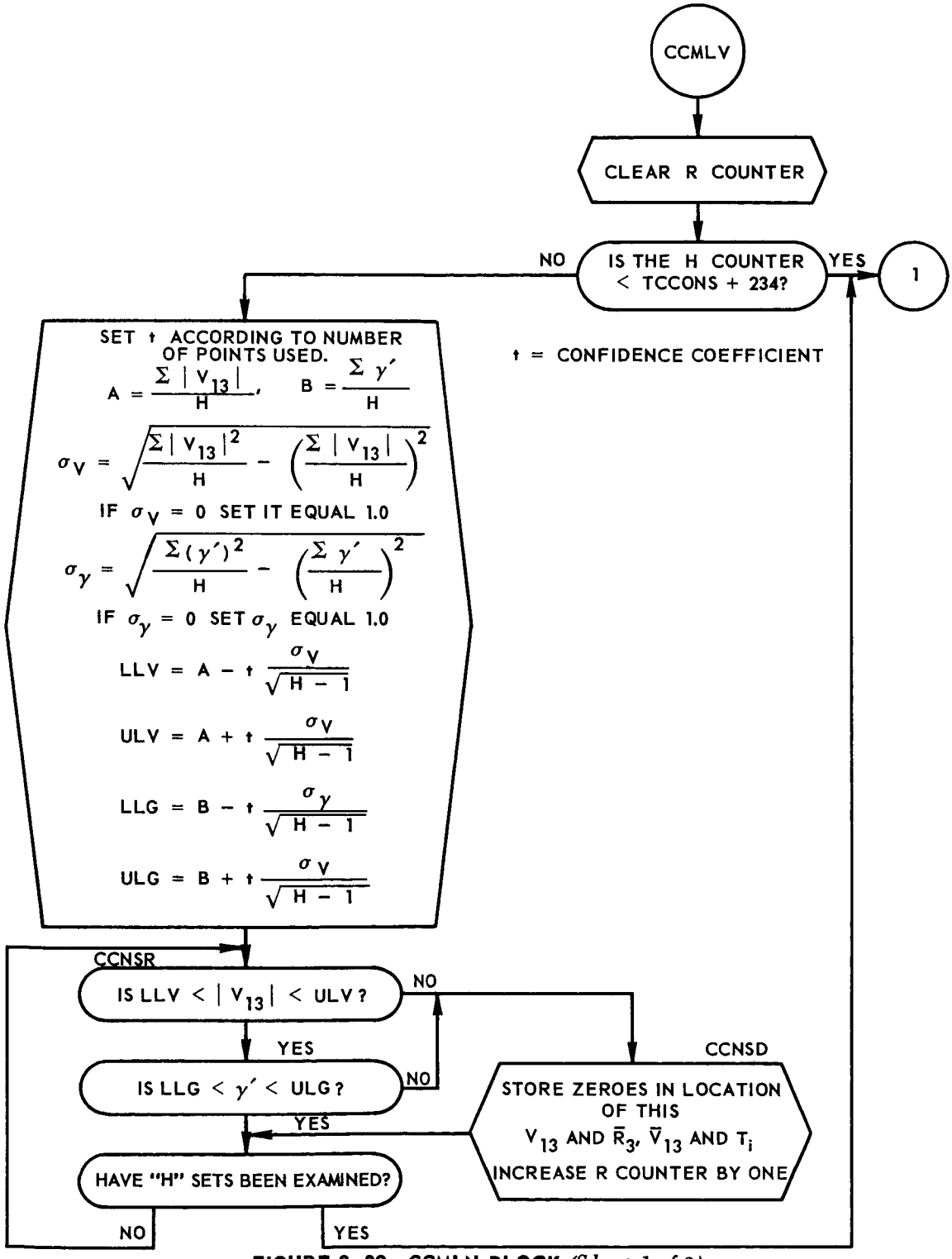


FIGURE 3-22. CCMLN BLOCK *(Sheet 1 of 2)*

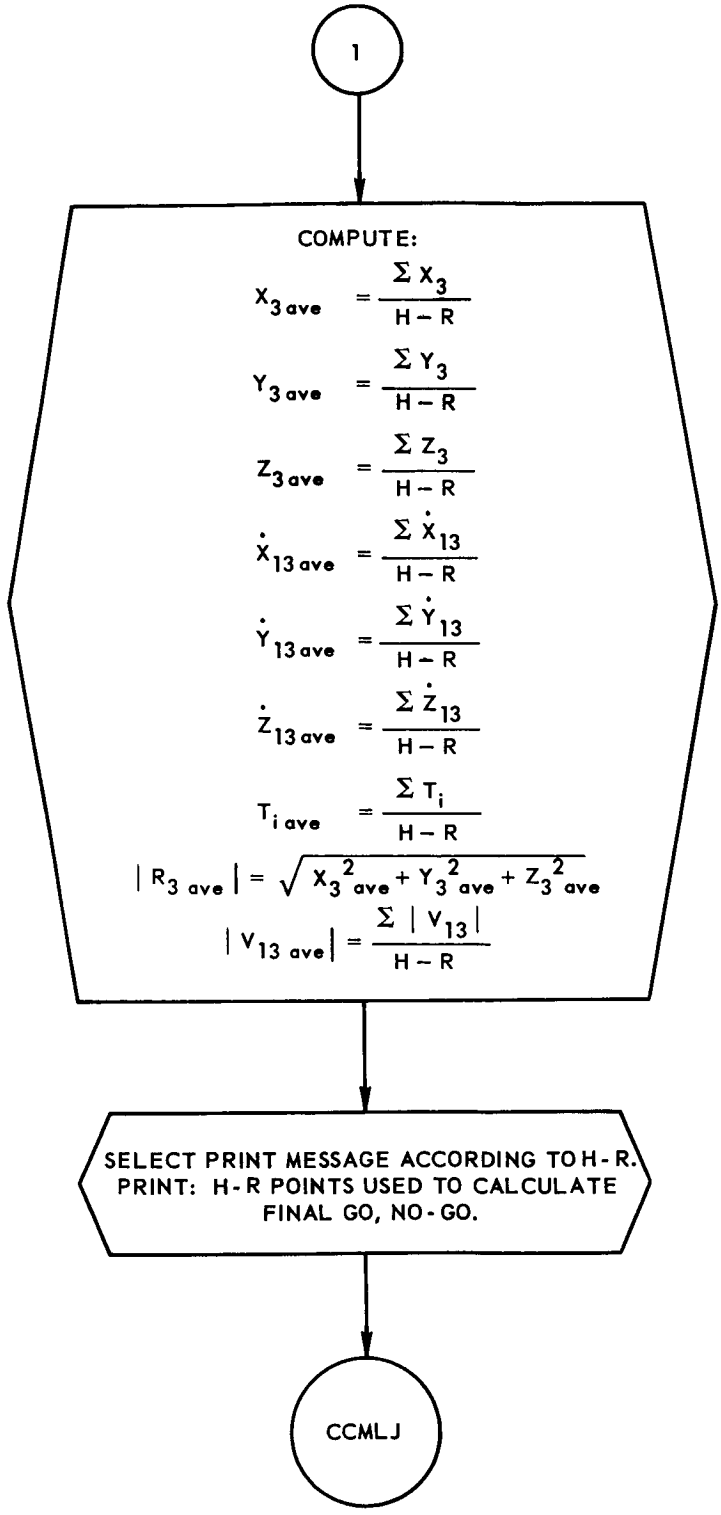


FIGURE 3-22. CCMLN BLOCK *(Sheet 2 of 2)*

$$\sigma_V = \sqrt{\frac{\Sigma(|V_{13}|)^2}{H} - \left(\frac{\Sigma\ |V_{13}|}{H}\right)^2}$$

$$\sigma_{\gamma'} = \sqrt{\frac{\Sigma\ (\gamma')^2}{H} - \left(\frac{\Sigma\ \gamma'}{H}\right)^2}$$

where H = the number of points received (H counter).

An upper and lower bound for velocity and flight path angle are then computed from the δ values as follows:

$$\text{lower limit of velocity} = \frac{|V_{13}|}{H} - \frac{t \cdot \delta_v}{\sqrt{H-1}}$$

$$\text{Upper limit of velocity} = \frac{|V_{13}|}{H} + \frac{t\ \delta_v}{\sqrt{H-1}}$$

$$\text{lower limit of gamma} = \frac{\Sigma\ \gamma}{H} - \frac{t\ \delta_{\gamma'}}{\sqrt{H-1}}$$

$$\text{Upper limit of gamma} = \frac{\Sigma\ \gamma}{H} + \frac{t\ \delta_v}{\sqrt{H-1}}$$

where "t" is the confidence coefficient established by the t distribution.

Values for this coefficient are listed below:

3.25 - confidence coefficient for 10 points

3.355 - confidence coefficient for 9 points

3.499 - confidence coefficient for 8 points

3.707 - confidence coefficient for 7 points

4.032 - confidence coefficient for 6 points

4.604 - confidence coefficient for 5 points

When the limit values have been computed, the $|V_{13}|$ and γ' for each point are compared with the corresponding upper and lower limits, and if they fall outside the limits, the corresponding $\bar{R}$, $\bar{V}$, time, velocity, position magnitudes

and gamma are all rejected and the rejected points counter (R) is incremented by one.

In the rare case where δ_v or δ_γ is zero the upper and lower limits become identical causing rejection of the $|V_{13}|$ and γ' quantities. To prevent this, δ_v or δ_γ are set to 1.

A message is then set up using the subroutine PSIP to print the number of points used to calculate the mean values.

If the number of points received during the 10 cycles following capsule separation is less than 5, the computation of the upper and lower limits is bypassed and the mean values are computed using all available points. It is considered that the t distribution criteria does not hold for less than 5 points. Control then goes to CCMLJ.

CCMLO (Figure 3-23)

In this block, the mean position and velocity vectors and the associated time are obtained from the vectors received during the 10 computing cycles after capsule separation as follows:

$$\overline{R}_{3ave} = \frac{\Sigma \quad \overline{R}_{14}}{\text{contents of F counter}}$$

$$\overline{V}_{12ave} = \frac{\Sigma \quad \overline{V}_{14}}{\text{contents of F counter}}$$

$$T_{i\ ave} = \frac{\Sigma \quad T_i}{\text{contents of F counter}}$$

$$|R_{3\ ave} = \sqrt{X^2_{3ave} + Y^2_{3ave} + Z^2_{2ave}}$$

$$|V_{12\ ave} = \sqrt{\dot{X}^2_{3ave} + \dot{Y}^2_{3ave} + \dot{Z}^2_{3ave}}$$

The $\overline{R}_{3ave}$ and $\overline{V}_{12ave}$ are then extrapolated to the next minute of GMT using the subroutine CVEXT (see write-up on page 3-103) and stored with GMT in the TDRANV table for the orbit programs. Control then transfers to CMMLJ.

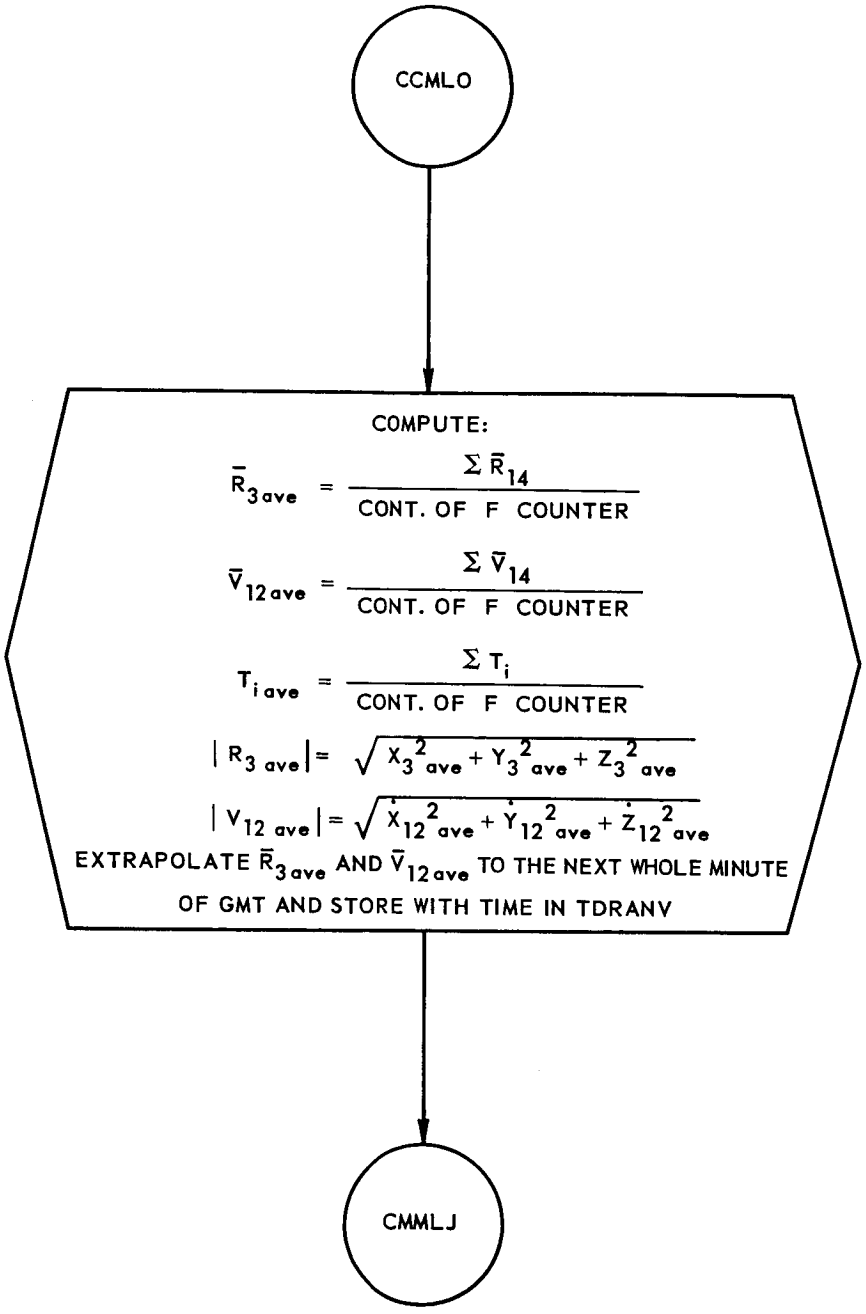


FIGURE 3-23. CCMLO BLOCK

CMMLO (Figure 3-24)

When no data is received during the 10 cycles following capsule separation, and since it is always necessary to supply an average $\bar{R}$ and $\bar{V}$ to the orbit programs, the last $\bar{R}$ and $\bar{V}$ received are extrapolated and these are stored in the TDRANV table.

It is necessary to check the "Hold Phase" indicator (MCRUDY). If this indicator is non-zero indicating that the program has returned to the Launch Phase from the Hold Phase, computation of an extrapolated $\bar{R}$ and $\bar{V}$ is prevented because $\bar{R}$ and $\bar{V}$ have been computed and stored in the TDRANV table in a previous computing cycle. Control then is transferred to CCMLX.

If this indicator is zero, the last known $\bar{R}_3$ and $\bar{V}_{12}$ are extrapolated to the next minute of GMT using the subroutine CVEXT (see write-up on page 3-103) and stored with GMT in the TDRANV table for the orbit programs. Control is then transferred to CCMLR.

CCMLP (Figure 3-25)

This block determines whether powered flight has ended. This determination is made by testing for the presence of the SECO (Sustainer Engine Cut-Off) or ABOI (Abort Initiated) bit. If neither of them is present powered flight is still continuing.

The cell MCLLDT is set $\neq 0$ when the end of powered flight is detected to indicate to Monitor that the low speed radar lines should now be activated.

The presence of the CAPS (Capsule Separation) bit is then tested. This bit would normally appear within 1 second of the appearance of SECO or ABOI. If CAPS is not present after 3.5 seconds, it is assumed to have occurred. The time t of the first appearance of the CAPS bit or the time at which this assumption is made is stored in TCMANR + 13.

The time of the first appearance of the SECO bit (T_{seco}) is stored in TCMANR + 11, the time of the first appearance of the ABOI bit (T_{ABOI}) is stored in TCMANR + 12 and time of the first appearance of the CAPS bit (T_{CAPS}) is stored in TCMANR + 39. Whenever either of SECO, ABOI or CAPS is detected for the first time, i.e., after having been absent in the previous computation cycle, a corresponding on-line message is prepared using subroutine PSIP.

If the CAPS bit is present or capsule separation is being assumed, control is transferred to CCMLH, otherwise control goes to CCMLF.

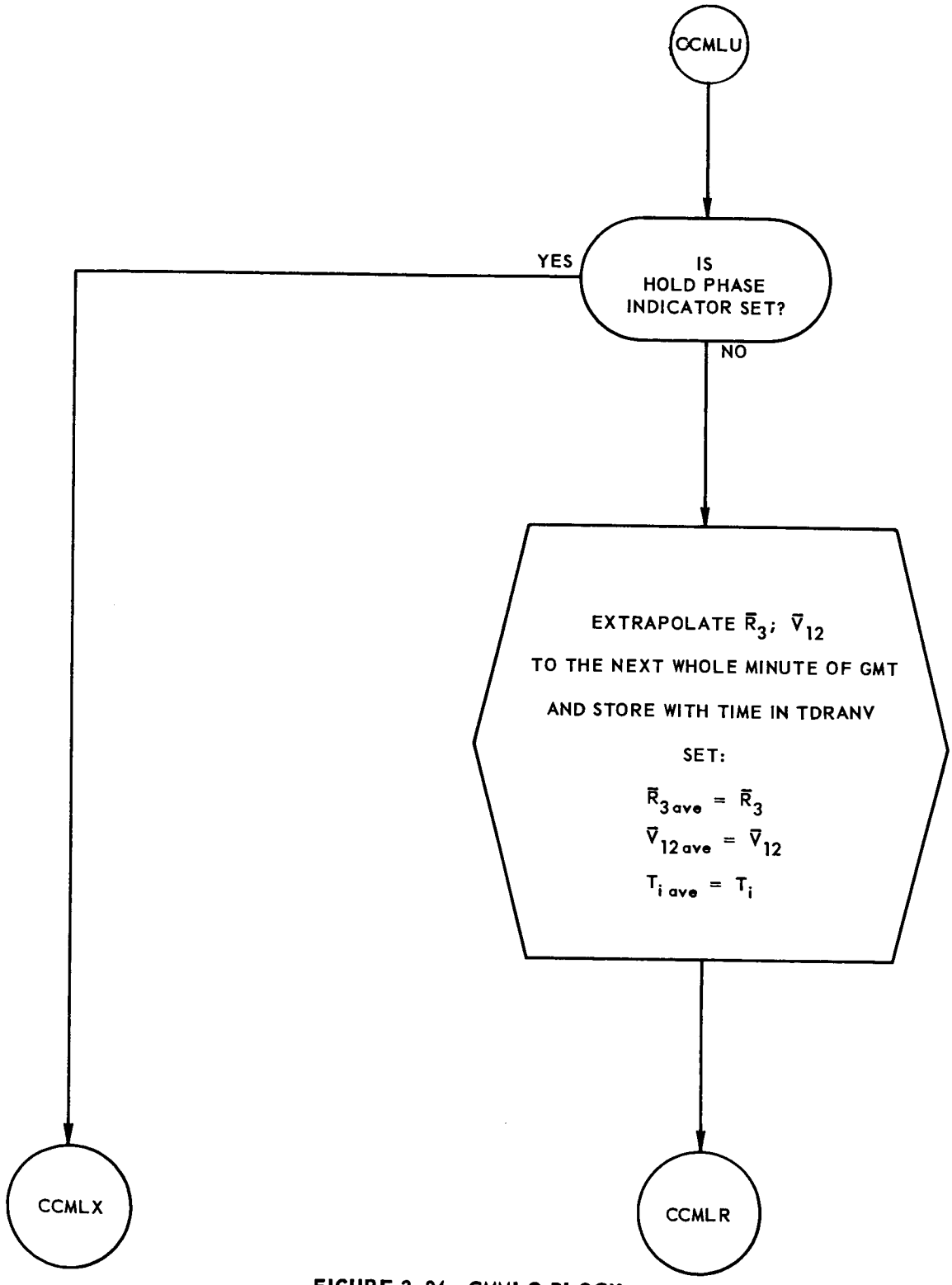


FIGURE 3-24. CMMLO BLOCK

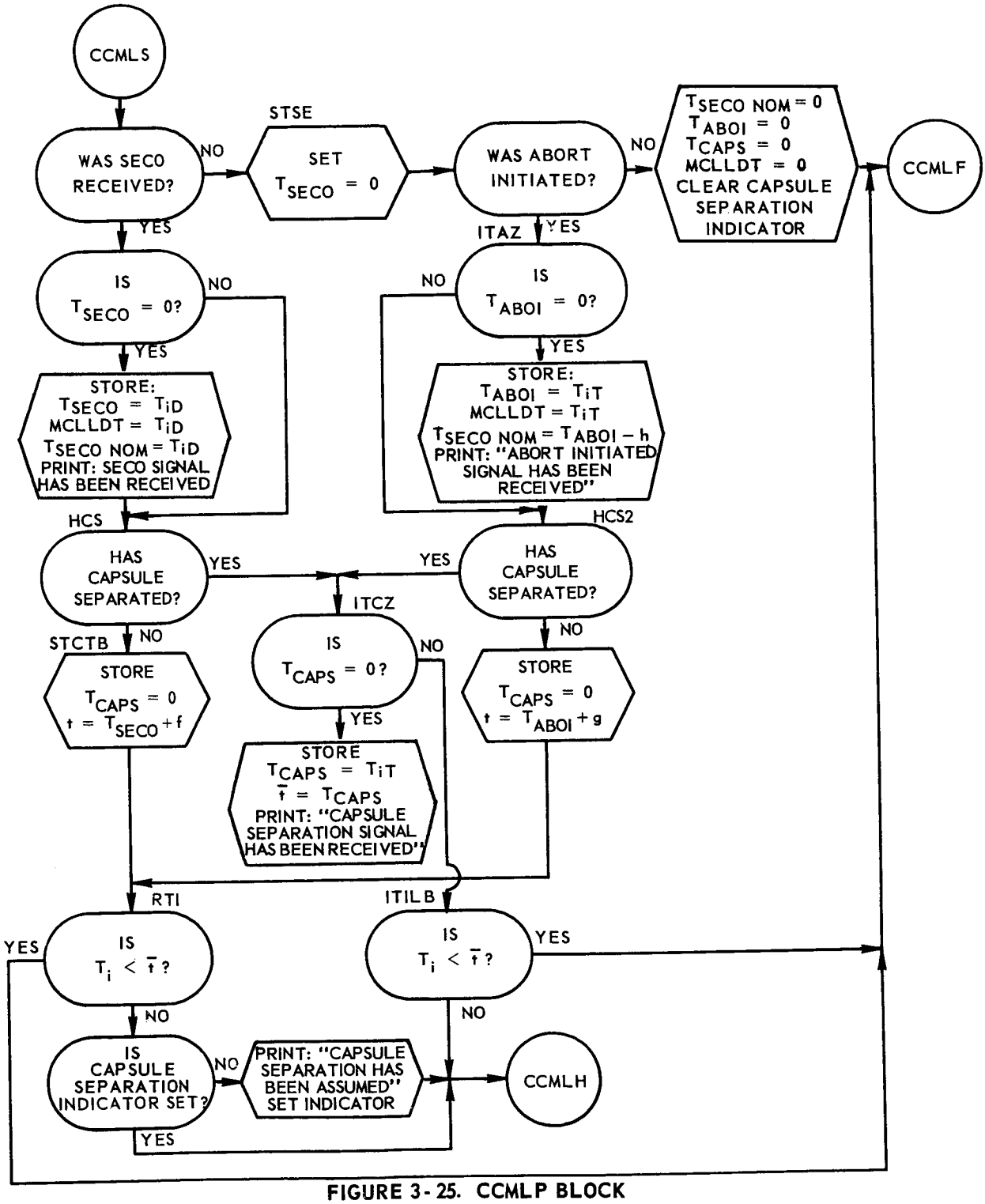


FIGURE 3-25. CCMLP BLOCK

CCMLQ

In case the next cycle should be a No Data cycle, the present time since lift-off, T_i, is saved (TCMANR + 48). Then the G counter (TCMANR + 16) (which reflects the number of passes since CAPS) is tested. If the G counter is zero, ten cycles have been completed since CAPS and transfer is made to CCMLR. Otherwise control goes to CKCCY.

CCMLR (Figure 3-26)

CCMLR is used to compute the GMT to fire retrorockets (GMTRC) during the computation of the GO, NO-GO decision. CCMLR may also be used to refine the estimate of the time delay to retrofire (Δt_r, computed in CMMLG), to impact as close as possible to a recovery area.

If it was found necessary in CMMLG to refine Δt_r, the No Iteration indicator was cleared. This indicator is tested here. Should no refinement be required, transfer is made to CGHOW, otherwise, preparations are made for the calculation as follows:

a) The "iteration counter" is set to 5 because 5 iterations give sufficient refinement.

b) T_4 is computed where $T_4 = T_{3i\ ave} + \Delta t_r$.

c) Δt_r is converted to Mercury units for later use.

The next section of the computation is called ITER, reflecting the iterative technique employed to refine Δt_r. The position and velocity vectors at the beginning of retrofire, $\overline{R}_4$ and $\overline{V}_4$, are computed using C9RVTH where

$$\overline{R}_4,\ \overline{V}_4 = f(\overline{R}_{3ave},\ \overline{V}_{12ave}, \Delta t_r).$$

An error return from C9RVTH results in the use of CHNOND to compute $\overline{R}_4$ and $\overline{V}_4$. From T_4, $\overline{R}_4$, and $\overline{V}_4$ using R6BOTH, the vectors describing the position and velocity of the capsule at the termination of retrorocket firing, $\overline{R}_5$ and $\overline{V}_5$ are computed. T_5 is also computed. The altitude above the surface of the earth at T_5 is calculated as follows:

$$h_5 = |\overline{R}_5| - R_e$$

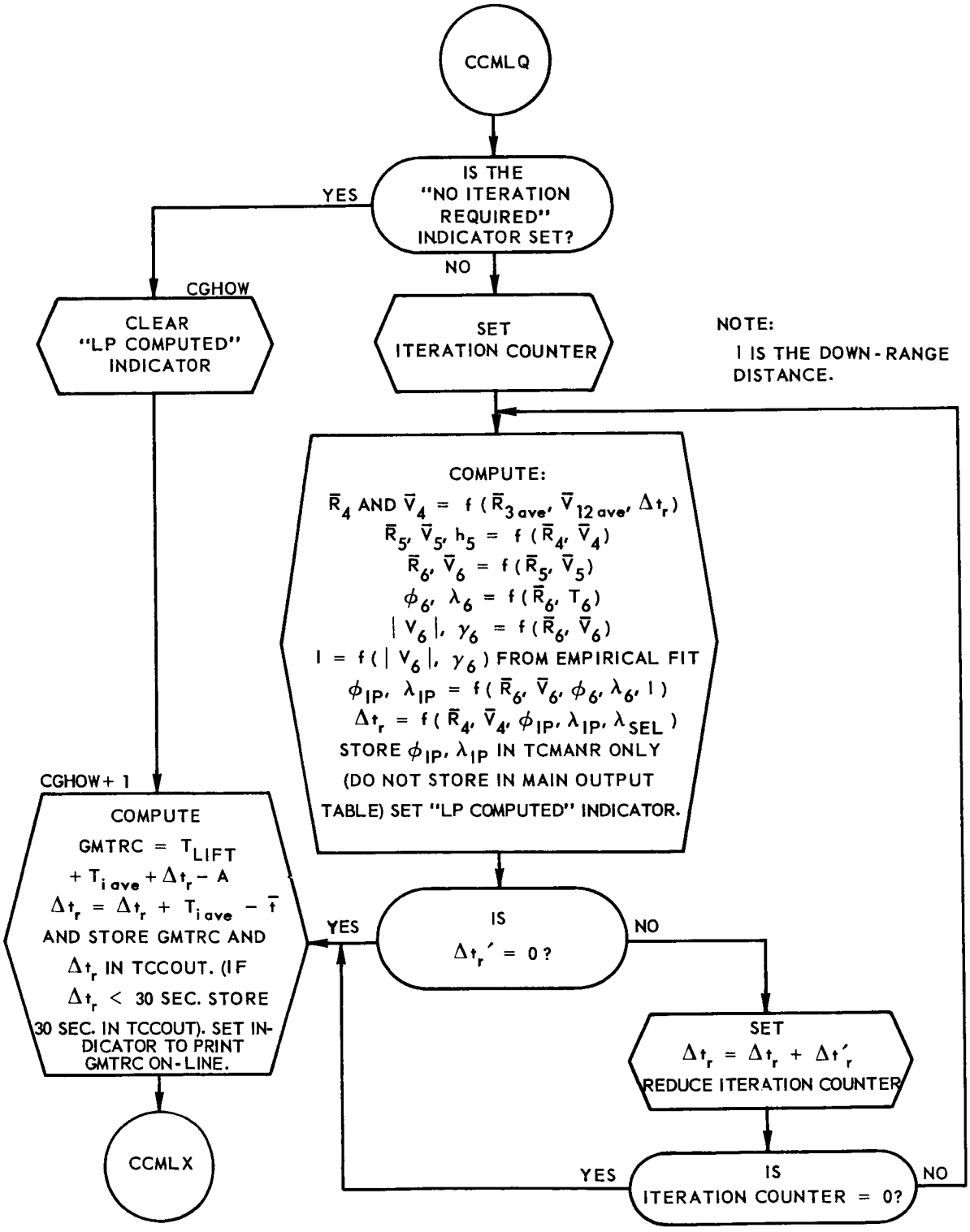


FIGURE 3-26. CCMLR BLOCK

If $h_5 = 450{,}000$ feet, then T_5, $\overline{R}_5$, $\overline{V}_5$ are identical to T_6, $\overline{R}_6$, $\overline{V}_6$ since the latter are defined as the position and velocity vectors at an altitude of 450,000 feet. T_6 is the time that this altitude will be attained (based on Δt_r).

If $h_5 \neq 450{,}000$ feet, C9RVTH is used to compute $\overline{R}_6$, $\overline{V}_6$ as a function of $\overline{R}_5$, $\overline{V}_5$, R_e and $h = 450{,}000$ feet. In this case, $T_6 = T_5 + \Delta T$, the ΔT having been computed in C9RVTH. If there is an error return from C9RVTH, CHNOND will be used to compute T_6, $\overline{R}_6$ and $\overline{V}_6$.

Conversion of T_6 to GMT time is made for use with $\overline{R}_6$ in the A3MSCP program. This program calculates ϕ_6 and λ_6, the latitude and longitude of the capsule at 450,000 feet above the surface of the earth.

To compute flight path angle at 450,000 feet (γ_6), $|V_6|$ and $|R_6|$ are computed. Then γ_6 is determined by the following formula:

$$\gamma_6 = \arcsin \frac{X_6\dot{X}_6 + Y_6\dot{Y}_6 + Z_6\dot{Z}_6}{|R_6| \quad |V_6|}$$

Having obtained all the necessary input quantities, EMFIT is used to compute the latitude and longitude of impact (ϕ_{IP} , λ_{IP}).

Having computed λ_{IP} for the known ΔT_r, C9DTRF is used to compute the correction for Δt_r ($\Delta t_r'$), if a correction is required to make the following relationship true:

$$\left|\lambda_{SEL} - \lambda_{IP}\right| \leq \Delta\lambda,$$

where $\Delta\lambda$ is defined as the acceptable longitudenal tolerance. (TCCONS +94)

At one time in the evolution of the Mercury Program System, it was necessary for the high abort program CCABRT to know if a landing point had been computed using the iterative technique described herein. An indicator is set at this point to show that this computation has been made. If the "No iteration required" indicator were set, this indicator would never have been set. However, it is never interrogated in the present system.

Returning to the output of C9DTRF, if $\Delta t_r'$ is equal to zero, then Δt_r is considered to be the refined Δt_r. In this case, since no further iterations are required, control goes to CGHOW + 1. If $\Delta t_r \neq 0$, Δt_r is corrected by Δt_r so that $\Delta t_r'$ is now:

$$\Delta t_r = \Delta t_r + \Delta t_r'.$$

In this case, all computations are repeated, beginning with the first equation and using the new value for Δt_r. If $|\lambda_{sel} - \lambda_{IP}| \leq \Delta\lambda$, then $\Delta t'_r = 0$ and control goes to CGHOW + 1. Otherwise this procedure is repeated, until five iterations have been completed. Then the value of Δt_r obtained is considered sufficiently refined and transfer is made to CGHOW + 1.

In block CGHOW the final time delay before firing retrorockets (Δt_r) is prepared for display at Cape Canaveral. This final value is defined thus:

$$\Delta t_r = \Delta t_r + T_{iave} - t .$$

Here Δt_r is the time delay computed above (or in the case of "no iteration," it is the time delay computed in CMMLG). T_{iave} is the time tag associated with $\overline{R}_{3ave}$ and $\overline{V}_{12ave}$. t is the time of capsule separation.

If Δt_r is 30 seconds or greater, it is converted from seconds to minutes and seconds and stored as output in location TCCOUT + 14. If Δt_r is less than 30 seconds, and to allow sufficient time for the capsule to attain retrofire altitude, 30 seconds is stored in TCCOUT + 14.

Finally, the GMT to fire retrorockets (GMTRC) is computed. GMTRC is defined as follows:

$$\text{GMTRC} = T_{lift\text{-}off} + T_{iave} + \Delta t_r - \alpha .$$

Here $T_{lift\text{-}off}$ is the GMT time of lift-off measured in seconds since midnight preceding launch and α is a preset constant. T_{iave} and Δt_r are as previously defined.

For output, GMTRC is converted to integral minutes and rounded integral

seconds and stored in TCCOUT + 13. To see GMTRC printed on the on-line printer, Monitor communication cell MCPNOL is cleared.

Then, with GMTRC computed, CCMLR is succeeded by CCLOK.

CCLOK (Figure 3-27)

In the event of an abort after nominal tower separation, CCLOK determines if the capsule is within the range and elevation limits of Bermuda command capability at the time when retrofire is to be triggered to impact in the recovery area selected by CMMLG. If the capsule is not within command limits, CCLOK attempts to select a recovery area which satisfies the command requirements. A test of the "Many Areas Indicator" determines the existence of an alternate area. For the case where no recovery area can be selected, the recommendation to retrofire with minimum time delay is displayed.

CCLOK is entered from CCMLR. The test, $|V_{12\ ave}|$: B, bypasses CCLOK if $|V_{12\ ave}|$ is only sufficient to land the capsule in recovery area A. Otherwise, the library subroutine U1ALSC is utilized to compute slant range and elevation from Bermuda to the capsule. If slant range and elevation are within limits, implying that retrorockets can be fired by command from Bermuda, control passes to CCMLX.

If control remains with CCLOK after the completion of the above tests, a check is made of the "Many Areas Indicator" to determine if another recovery area is available for consideration. If it is not possible to impact in any other area, e.g., if only one area falls within the minimum and maximum impact lingitudes, the recommendation to fire retros with minimum time delay (30 seconds) will be given. The recovery area which is closest to the computed impact point will be displayed. Program control then passes to CCMLX.

If another recovery area is available, a computation is made to determine the time remaining until retrofire, Δt_r, for impacting in this newly selected recovery area. The equations and logic of CCMLR are employed here. A test is again made to determine if the capsule position at retrofire time will now permit Bermuda to initiate retrofire for impact in the new area. If so, the new recovery area and its associated GMTRC will be displayed. Program control then passes to CCMLX.

If Bermuda is not able to command retrofire impacting in the new area, a minimum delay time to retrofire (30 seconds) will be displayed along with a newly computed GMTRC based on this delay. The recovery area selected for display will be the one closest to the computed impact point. Program control passes to CCMLX.

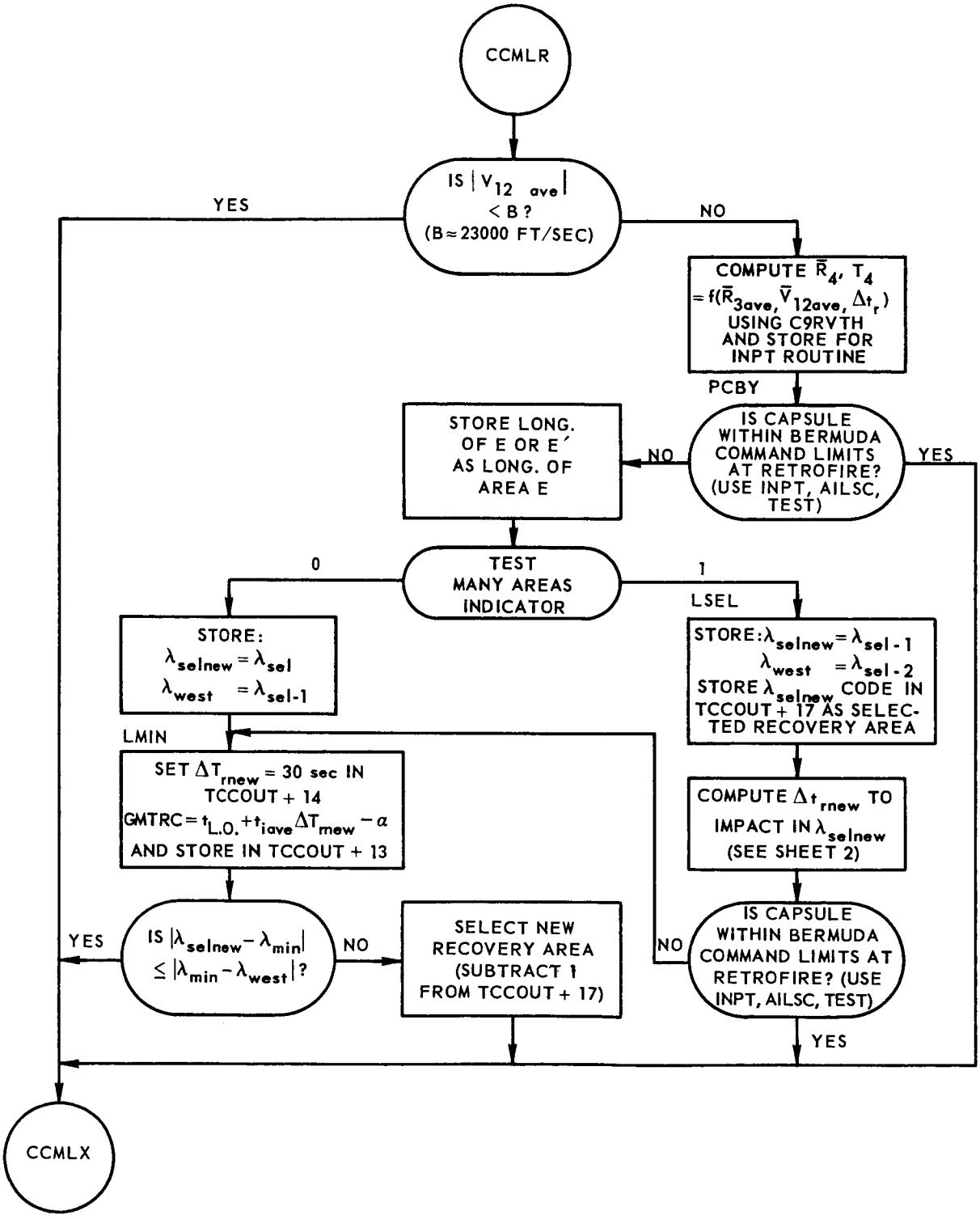


FIGURE 3-27 CCLOK BLOCK *(Sheet 1 of 2)*

COMPUTATION OF Δt_r TO IMPACT IN $\lambda_{sel\ new}$

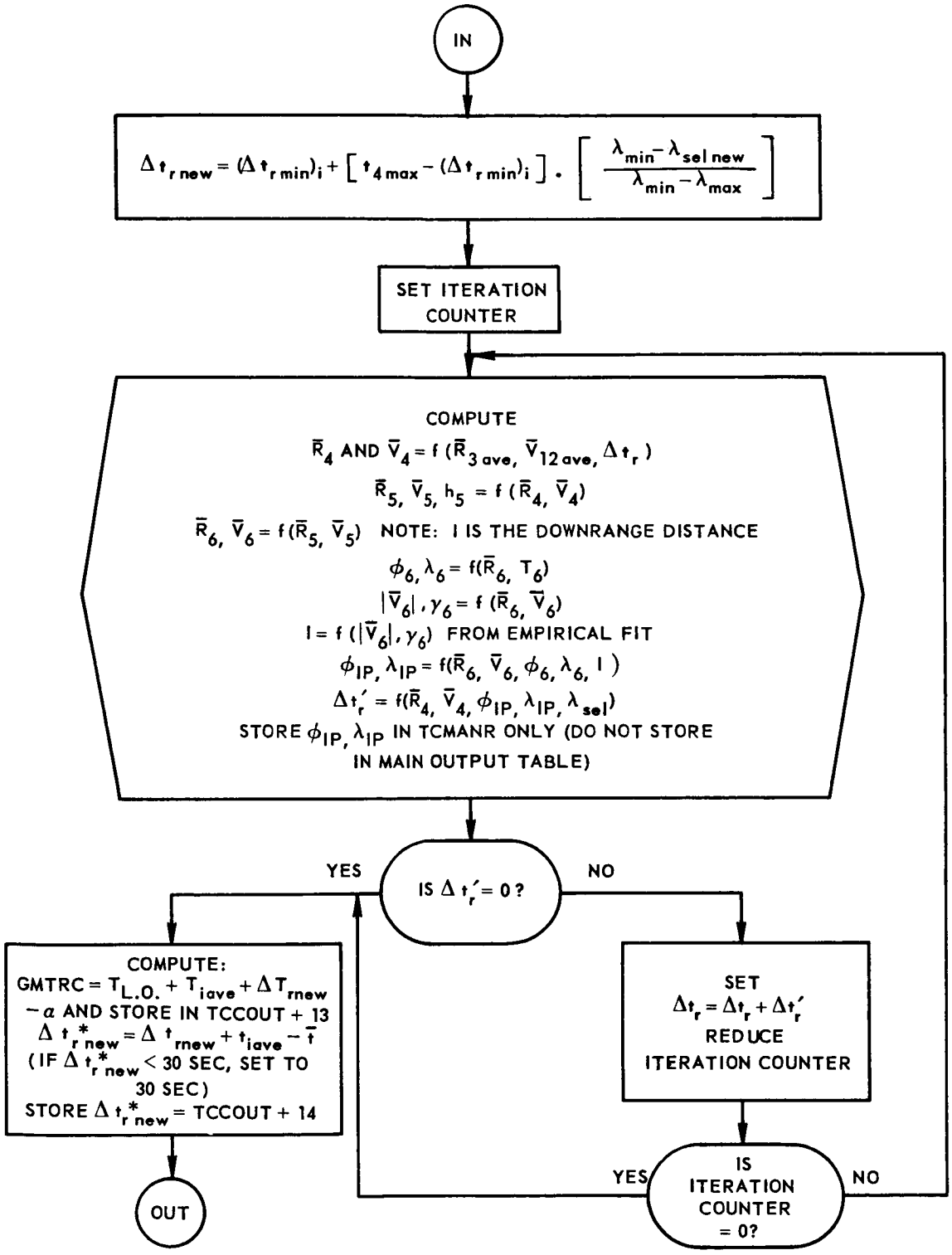


FIGURE 3-27. CCLOK BLOCK *(Sheet 2 of 2)*

CCMLS (Figure 3-28)

The purpose of CCMLS is to determine if Tower Separation (TOWS) has occurred. In a normal mission this event occurs about 158 seconds after lift-off, that is after the booster engine drops off. If the mission were to end in early abort, before the normal TOWS, the booster and sustainer engines would be separated from the capsule and tower and this event would be indicated by the telemetry event "Abort Initiated" (ABOI). The tower escape rockets would then be fired and the tower and capsule, still connected, would move away from the vehicle due to the additional velocity gained by firing the tower escape rockets. In the case of an abort after tower separation, the firing of the retro rockets, located on the capsule, can be timed to facilitate the choice of a landing area.

CCMLS tests the telemetry input word (TCMANR +21) for the presence of the TOWS bit. If the bit is not present, the location containing the time of this event (TCMANR +193) is set to zero since the signal may have been present at one time and then disappeared because it was false, spurious or manually overridden by Cape Canaveral. Control then is transferred to CCMLT.

If the TOWS bit was present, the program determines if TOWS has occurred before or after ABOI. It does this by checking the "Abort Before Tower Separation" word (ABTS - TCMANR +179). The contents of this cell being non-zero indicates that ABOI occurred before tows and, therefore, the TOWS indication is essentially ignored and then transfers control to CCMLT.

If this is the first indication of TOWS, and it occurred before ABOI (ABTS equals zero), the time of the TOWS event is saved in TCMANR +193. Using the subroutine PSIP, the printing of the on-line message announcing this event is initiated. The program next initializes the following locations:

a) TCMANR + 15 -- Tail-off velocity - initialized at zero.

b) TCMANR + 17 -- H counter. Counts the number of B-GE or Azusa data points received during the 10 computing cycles immediately after capsule separation. Initialized at zero.

c) TCMANR + 192 -- F counter. Counts the number of AN/FPS-16 data points received during the 10 computing cycles immediately after capsule separation. Initialized at zero.

d) TCMANR + 16 -- G counter. Counter for the 10 passes after capsule separation. Initialized at 10. Control then transfers to CCMLP.

CCMLT (Figure 3-29)

The purpose of CCMLT is to determine if the "Abort Initiated" signal has occurred. In determination of the occurrence of an event, if the indication of the event should appear and disappear a number of times, each reappearance is

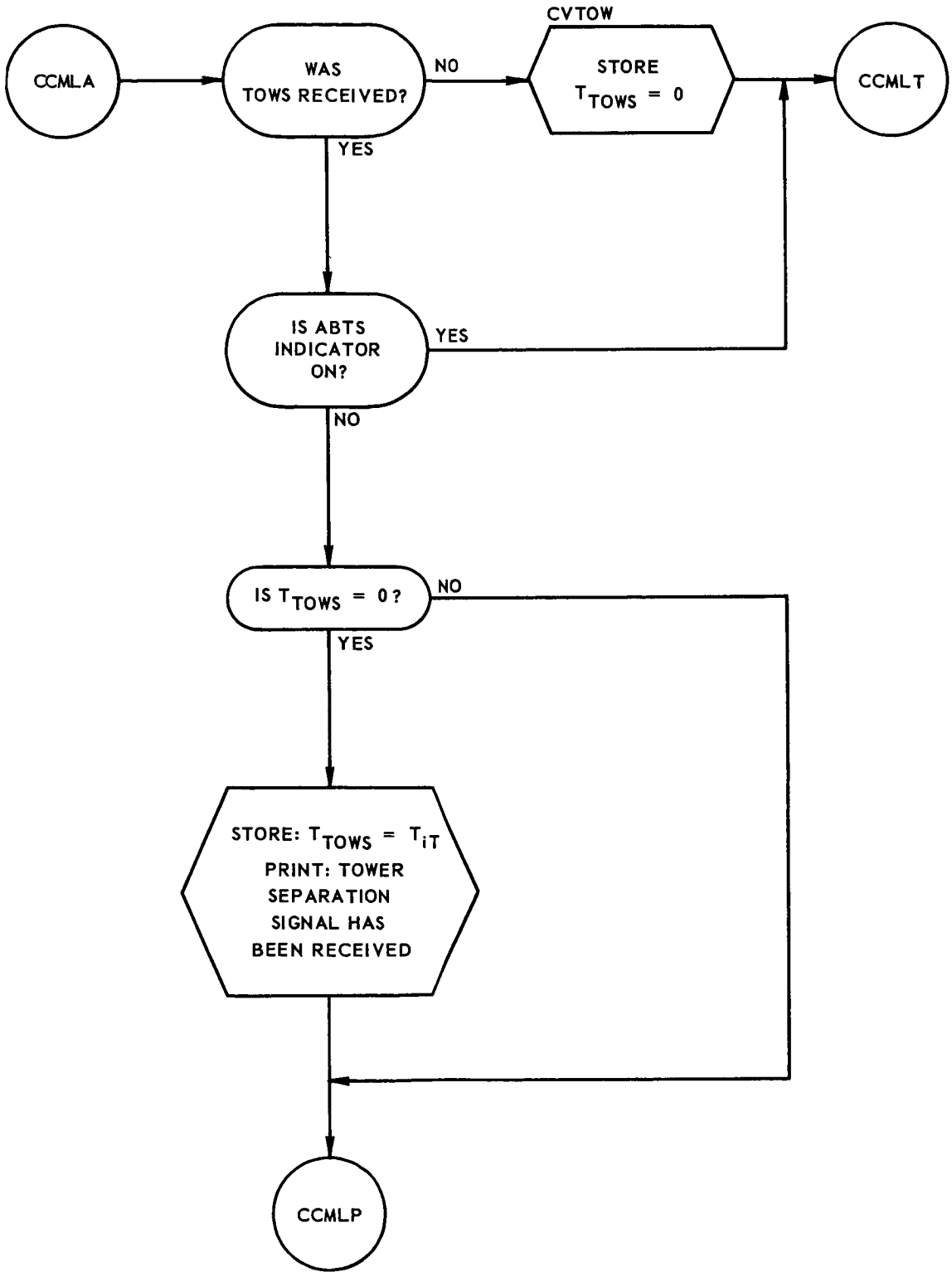


FIGURE 3-28. CCMLS BLOCK

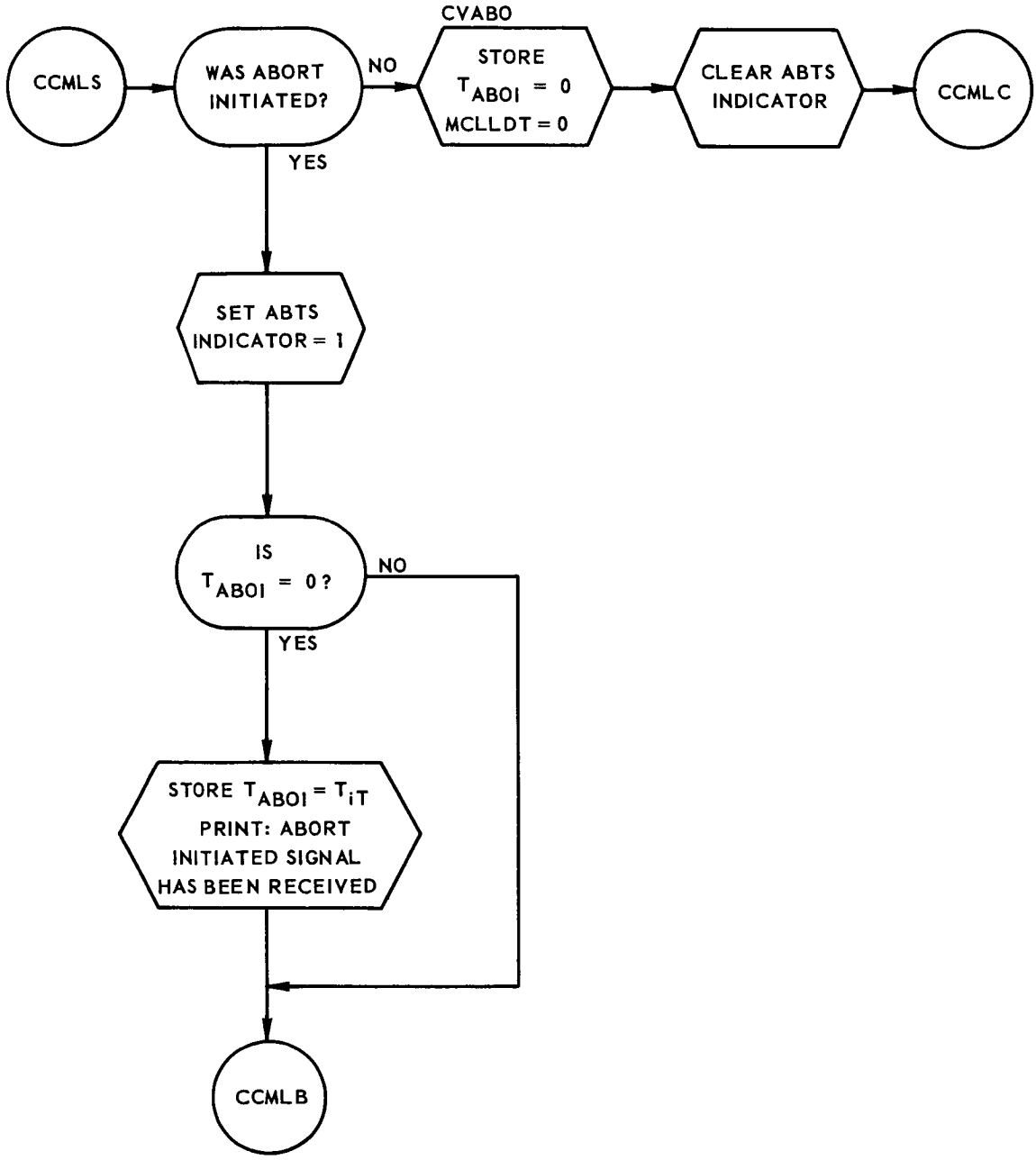


FIGURE 3-29. CCMLT BLOCK

understood to be a first appearance and a new time for the event is saved and the event announcement is printed on-line. This philosophy is adhered to in the consideration of all telemetry events except lift-off.

The input telemetry word (TCMANR + 21) is checked for the ABOI bit. If it is present, the program sets the "Abort Before Tower Separation" (ABTS) indicator (TCMANR +179) unequal to zero. In addition, if this is the first indication of ABOI the time of this event is stored in location TCMANR +12 and a transfer is made to the subroutine PSIP to initiate the printing of the on-line message announcing this event. Control then passes to CCMLB.

If the ABOI bit is not present, the time of this event (TCMANR +12), the ABTS cell (TCMANR +170), and MCLLDT -- the cell which indicates to Monitor that it can activate the low speed data lines now that powered flight has ended -- will all be set to zero. Control then goes to CCMLC.

CCMLU (Figure 3-30)

Both a GO and a NO-GO indication is given to Cape Canaveral because no B-GE or Azusa data was received during 10 cycles after CAPS. The orbit capability is made -0, the minus signifying that the orbit capability is determined. The on-line message of the NO-GO recommendation is set up using subroutine PSIP. A test of the F counter is made. If the counter is zero, no AN/FPS-16 data was received during the 10 cycles after CAPS and control goes to CMMLO, otherwise control goes to CCMLO.

CCMLV (Figure 3-31)

The on-line printing of the number of posigrade rockets fired is initiated using subroutine PSIP. A check is then made to see if the H counter (TCMANR +17) is non-zero which indicates that some B-GE or Azusa data points have been accumulated during the 10 passes after CAPS. If so, control goes to CCMLN, otherwise, control goes to CCMLU.

CCMLW

A test is made to determine if this is the last of the 10 passes after CAPS. If the G counter is equal to zero, this is the last pass and control goes to CCMLV, otherwise a transfer is made to CCMLG.

CCMLX

CCMLX is the location of the transfer to the hold suffix, MFLHLD. This transfer indicates the end of launch processing.

CCMLY (Figure 3-32)

The purpose of this block is to examine the input telemetry word for the "Abort Phase Entered" bit. This bit would appear in the telemetry as a result

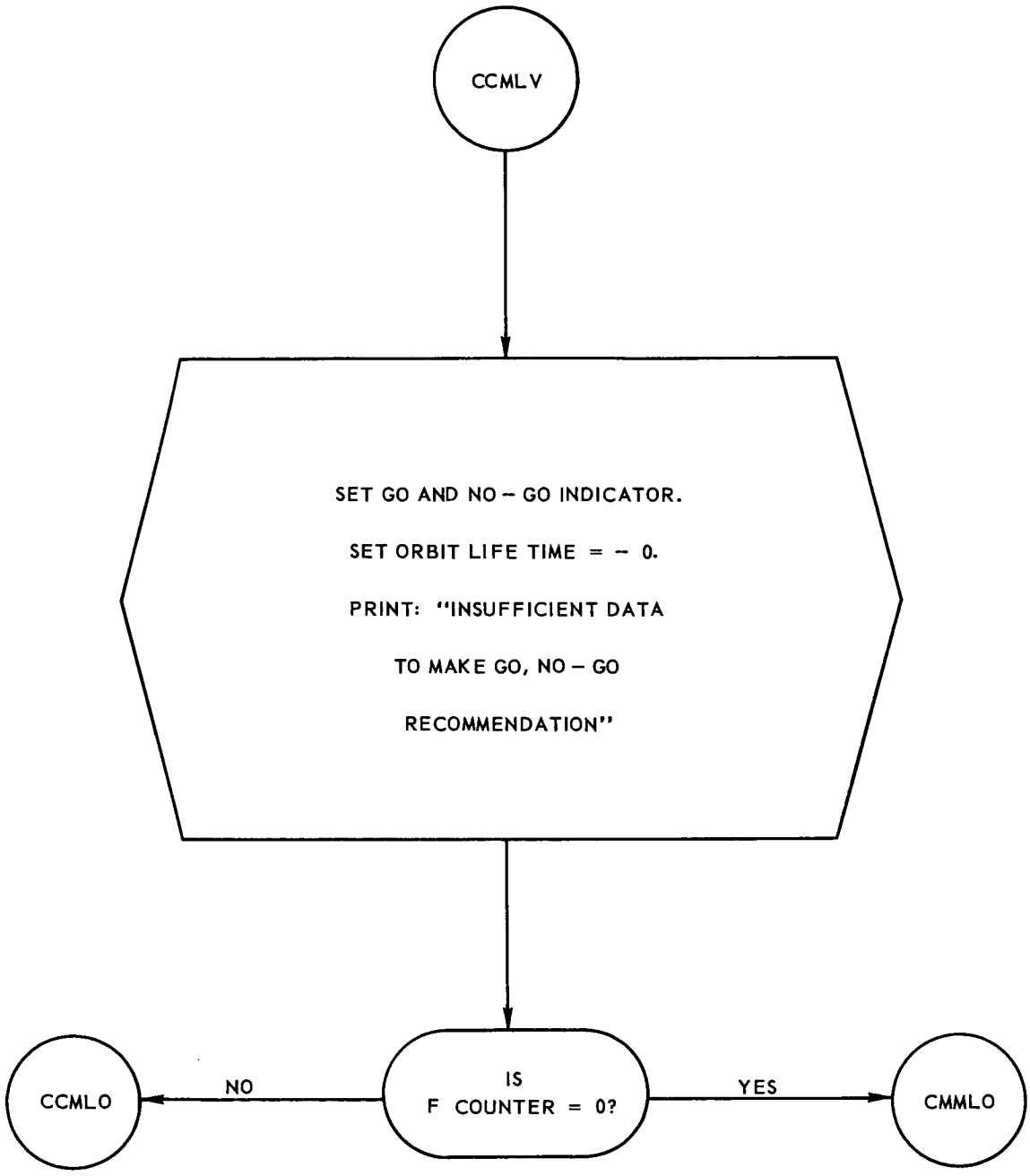


FIGURE 3-30. CCMLU BLOCK

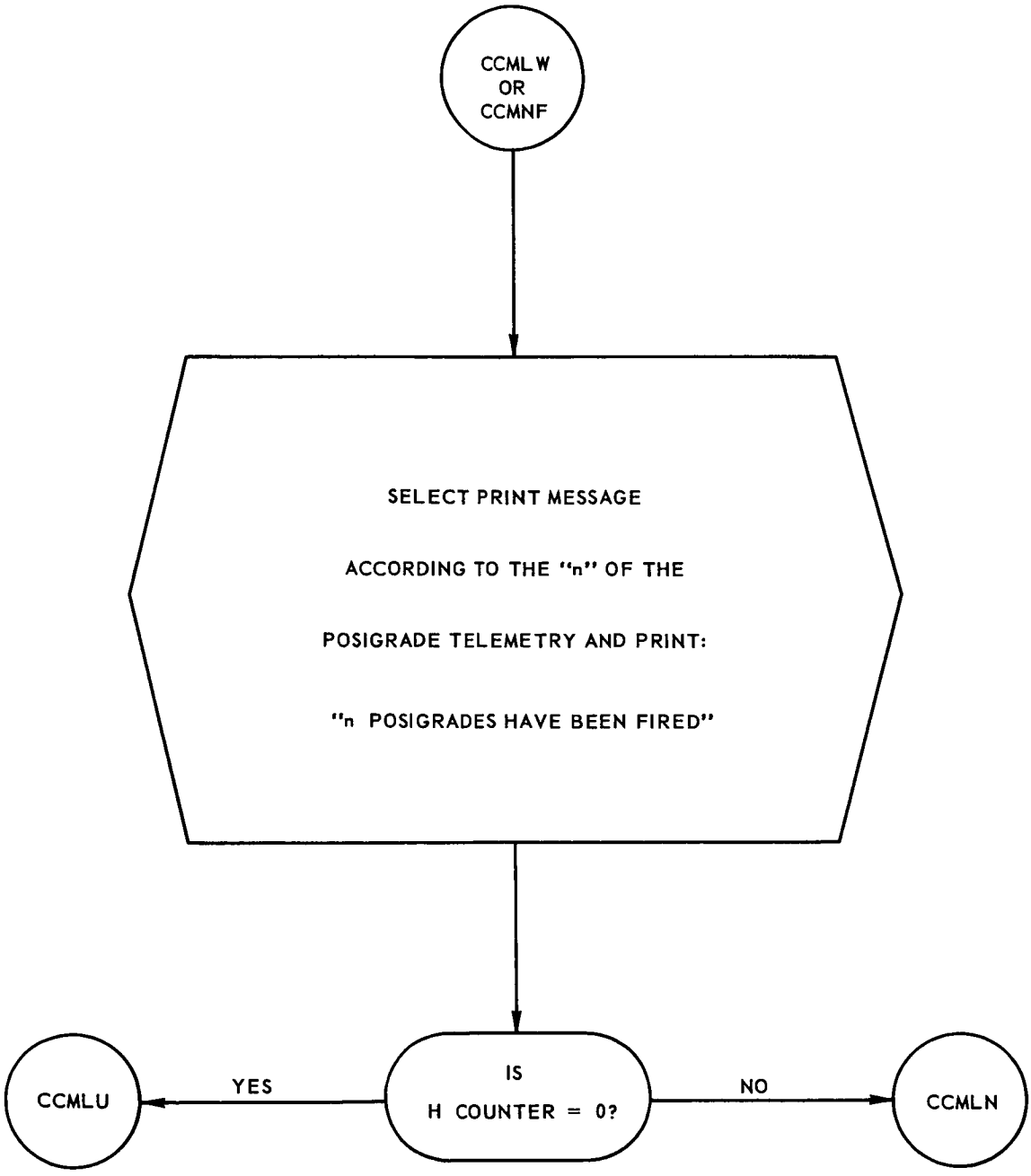


FIGURE 3-31. CCMLV BLOCK

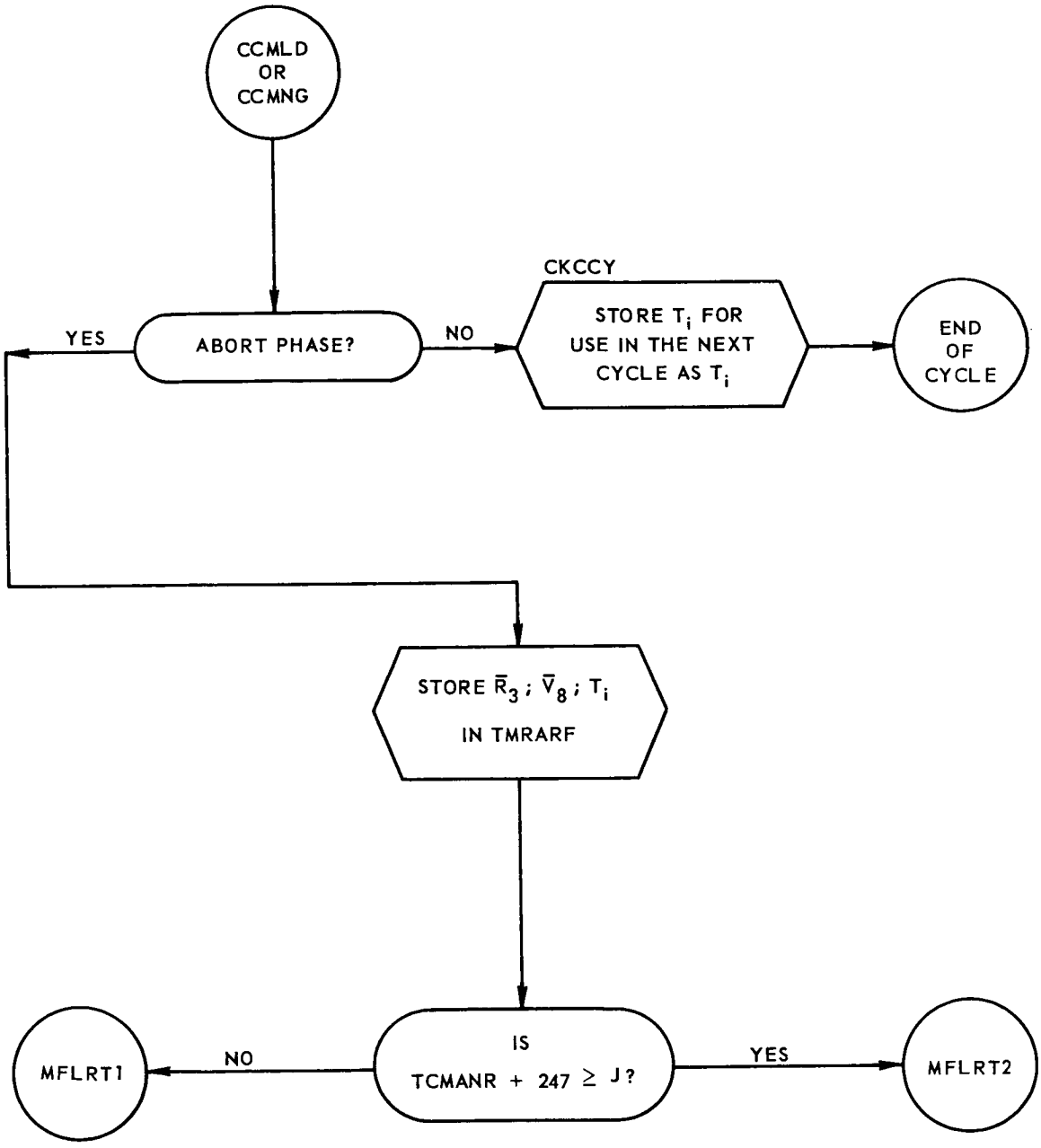


FIGURE 3-32. CCMLY BLOCK

of the three position switch at the Mercury Control Center being thrown into the "Abort" position. When the bit is absent, i.e., during a normal mission, the launch computing cycle is complete and control is transferred to the Monitor suffix MFLNML.

If the "Abort Phase Entered" bit is present the program moves the current $\overline{R}_3$ (TCMANR + 1 to+3) and $\overline{V}_8$ (TCMANR + 33 to +35) to the TMRARF table for use by the abort programs. The timetag of the vector is converted to fixed point seconds and stored in TMRARF 6.

A test is then made based upon altitude to see which abort program is to be used. If there have been three or more successive data points with altitudes above 100,000 feet, control then goes to the medium abort suffix MFLRT2; otherwise control goes to the low abort suffix MFLRT1.

CVEXT (Figure 3-33)

The average vectors $\overline{R}_{3ave}$ and $\overline{V}_{12ave}$ are extrapolated to the next whole minute of GMT. The difference (Δt) between GMT associated with T_{iave} and the next whole minute of GMT is computed.

The calling sequence for C9RVTH is set up to compute $\overline{R}$ and $\overline{V}$ associated with the time $T = T_{iave} + \Delta t$. If C9RVTH comes to the error return, CHNOND is used for the computation.

The above $\overline{R}$ and $\overline{V}$ and its associated T are stored in the TDRANV table. If Δt is zero, no extrapolation is needed and $\overline{R}_{3ave}$ and $\overline{V}_{12ave}$ are stored in the TDRANV table. In either case, control returns to the part of the program which used this subroutine.

CKCCY

Before returning to the normal monitor suffix, MFLNML, the time saved in case the next cycle might be a missing data cycle, that is T_i, is stored as the time tag of the previous cycle, T_{i-1} in TCMANR +60. In transferring to MFLNML, the computations in CCMAIN are completed for this cycle.

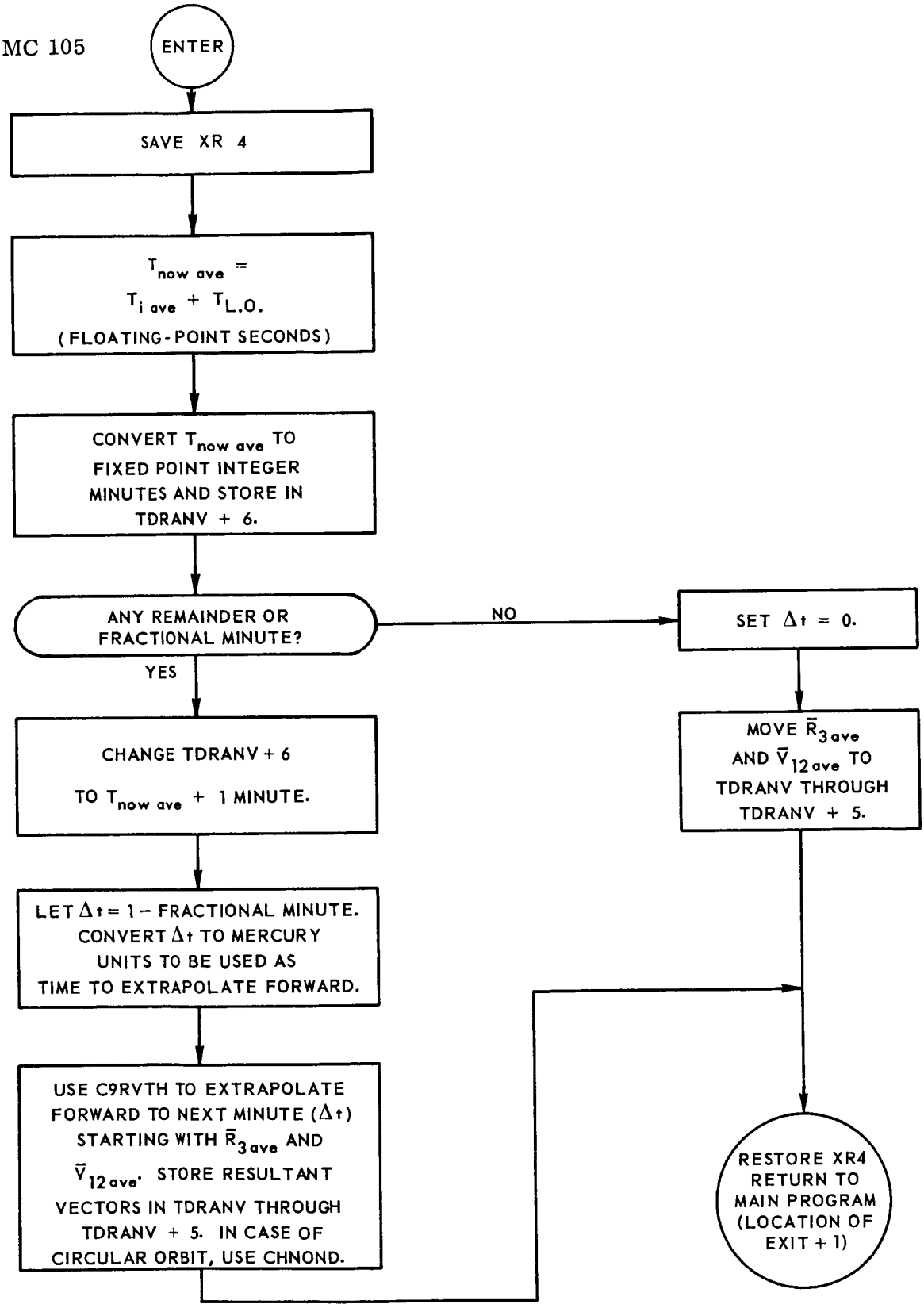


FIGURE 3-33. CVEXT FLOW CHART

TABLE 3-2. TCCOUT TABLE FORMAT

LOCATION	QUANTITY	WHERE DISPLAYED	FORMAT
TCCOUT	$\gamma - \gamma_{nom}$ (B-GE)	Strip chart	Fl. Pt. - Radians
+ 1	$\gamma - \gamma_{nom}$ (AN/FPS - 16)	Strip Chart	Fl. Pt. - Radians
+ 2	V/V_r - $V/V_{r\ nom}$ (B - GE)	Strip Chart	Fl. Pt. - Radians
+ 3	V/V_r - $V/V_{r\ nom}$ (AN/FPS-16)	Strip Chart	Fl. Pt. - Radians
+ 4			
+ 5	ϕ_{pp}	Wall Map and Plotboard 4	Fl. Pt. - Radians
+ 6	λ_{pp}	Wall Map and Plotboard 4	Fl. Pt. - Radians
+ 7	ϕ_{IP} - Before TOWS	Plotboard 4	Fl. Pt. - Radians
	ϕ_{min} - After TOWS	Plotboard 4	Fl. Pt. - Radians
+ 8	λ_{IP} - Before TOWS	Plotboard 4	Fl. Pt. - Radians
	λ_{min} - After TOWS	Plotboard 4	Fl. Pt. - Radians
+ 9			
+ 10			
+ 11	Zero - Before TOWS	Plotboard 4	------
	ϕ_{max} - After TOWS	Plotboard 4	Fl. Pt. Radians
+ 12	$(3\pi/2)$ - Before TOWS	Plotboard 4	------
	λ_{max} - After TOWS	Plotboard 4	Fl. Pt. - Radians
+ 13			
+ 14	Zero - Before TOWS	Retrofire Controller	------
	Δt_r - After TOWS	Retrofire Controller	Time
+ 15			
+ 16	Zero - Before CAPS + 5 Seconds	Flight Dynamics Officer	------
	Negative of Orbit Capability - After CAPS + 5	Flight Dynamics Officer	Fx. Pt. - B 35
+ 17	Recovery Area	Retrofire Controller	Special Code

TABLE 3-2. TCCOUT TABLE FORMAT (Cont'd)

LOCATION	QUANTITY	WHERE DISPLAYED	FORMAT
+ 18	Abort Recommendation (1 = GO, 2 = NO - GO)	Flight Dynamics Officer	Fx. Pt. - B 35
+ 19	h - Height of Capsule	Plotboard 2	Fl.Pt.-Nautical Miles
+ 20	γ - Flight Path Angle	Flight Dynamics Officer and Plotboard 1	Fl. Pt. - Radians
+ 21	i - Inclination Angle	Flight Dynamics Officer	Fl. Pt. - Radians
+ 22	V/V_r - Velocity Ratio	Flight Dynamics Officer and Plotboard 1	Fl. Pt.
+ 23	$Y - Y_{nom}$ - Cross Range Deviation	Plotboard 2	Fl. Pt. - Nautical Miles
+ 24	d - Down Range Distance	Plotboard 2	Fl. Pt. - Nautical Miles
+ 25			
+ 26			
+ 27			
+ 28	$r - \overline{R}$	Flight Dynamics Officer and Plotboard 1	Fl. Pt.-Nautical Miles
+ 29			

TABLE 3-3. TCCONS TABLE FORMAT

LOCATION			VALUE	EXPLANATION
TCCONS +	0	DEC	6.0562925	347 degrees in radians
+	1	DEC	0	Not used in launch
+	2	DEC	3.5	G seconds
+	3	DEC	0.5	H seconds
+	4	DEC	.9557638	Conversion B-GE to Mercury (R)
+	5	DEC	1.022883455	Conversion B-GE to Mercury (V)
+	6	DEC	1.00003643	
+	7	DEC	1.000009664	Conversion IP-7090 to Mercury (V)
+	8	DEC	1.0463218	Conversion IP-7090 to B-GE (R)
+	9	DEC	.97763793004	Conversion IP-7090 tp B-GE (V)
+	10	DEC	10000.0	C seconds
+	11	DEC	-.2620152644	B radians/B-GE (V)
+	12	DEC	0.92688644	24040 ft/sec velocity at which comp. of time to Retrofire begins
+	13	DEC	0	Not used in launch
+	14	DEC	0	Not used in launch
+	15	DEC	0	Not used in launch
+	16	DEC	0	Not used in launch
+	17	DEC	.4326345E-9	K5 89 (V/V_r) nom.
+	18	DEC	0.0	K4 89 below
+	19	DEC	.1357455F-4	K3 89 staging
+	20	DEC	-.1606494F-3	K2 89
+	21	DEC	.05163	K1 89
+	22	DEC	.156703E-11	K13 89 (V/V_r) nom.
+	23	DEC	-.1107524E-8	K12 89 above
+	24	DEC	.2963183E-6	K11 89 staging
+	25	DEC	-.2741455E-4	K10 89
+	26	DEC	.6924132E-3	K9 89
+	27	DEC	.37357	K8 89
+	28	DEC	1.0455	Radius of Earth B-GE
+	29	DEC	3291.58325	Conversion B-GE to nautical miles
+	30	DEC	120.0	Time to fire retros and hit area A
+	31	DEC	0	Not used in launch
+	32	DEC	0.01089570717	A 228,000 ft. in Mercury units
+	33	DEC	.058833543	Omega rad/Mercury unit of time
+	34	DEC	.2389409468E-3	HF1 (5000 ft -- Mercury units)
+	35	DEC	0.00477881893	HF2 100,000 ft. in Mercury units
+	36	DEC	.999251039	Radius of Earth Mercury
+	37	DEC	24000.	Min. velocity-for GO
+	38	DEC	20925672.5	Convert Mercury to feet
+	39	DEC	-2.0202618	K181
+	40	DEC	1.6229852E8	K281
+	41	DEC	.729211508E-4	Omega E rad./sec.

TABLE 3-3. TCCONS TABLE FORMAT (Cont'd)

LOCATION		VALUE	EXPLANATION
TCCONS + 42	DEC	0	Not used in launch
+ 43	DEC	.493652887	Geocentric latitude of radar (28.2842274)
+ 44	DEC	.999251039	Mercury distance from geocenter to pad (R_0)
+ 45	DEC	-1.406416522	LP prime (80.581731) (radians)
+ 46	DEC	1.57079632679	PI/2 radians
+ 47	DEC	1.26689536	Beta Zero
+ 48	DEC	3443.929	Conversion Mercury to nautical miles
+ 49	DEC	26529.807	Conversion B-GE to ft./sec.
+ 50	DEC	0	Not used in launch
+ 51	DEC	2048	Mask
+ 52	DEC	1024	Mask
+ 53	DEC	0	Not used in launch
+ 54	DEC	0	Not used in launch
+ 55	DEC	0	Not used in launch
+ 56	DEC	0	Not used in launch
+ 57	DEC	32	Mask
+ 58	DEC	0	Not used in launch
+ 59	DEC	0	Not used in launch
+ 60	DEC	0	Not used in launch
+ 61	DEC	0	Not used in launch
+ 62	DEC	0	Not used in launch
+ 63	DEC	-0.25902696E-4	Escape rocket curve fit coefficient
+ 64	DEC	0.71031016E-3	Below 80,000 feet altitude
+ 65	DEC	-0.4774758E-2	
+ 66	DEC	0.16404792E-1	
+ 67	DEC	-0.10283297E-4	Escape rocket curve fit coefficient
+ 68	DEC	0.70937503E-3	Above 80,000 feet altitude
+ 69	DEC	0.11619321E-1	
+ 70	DEC	.003823055149	80,000 feet in Mercury
+ 71	DEC	25936.294946	Conversion Mercury to ft./sec.
+ 72	DEC	-.032338615	Coefficients to obtain
+ 73	DEC	+.04744618	H min. ret. as a function
+ 74	DEC	+.006608992	of velocity (V12 mag) Mercury
+ 75	DEC	30.0	30 seconds
+ 76	DEC	.809676171	B (21,000 ft./sec.) Mercury units
+ 77	DEC	.02150468521	450,000 feet in Mercury units
+ 78	DEC	0.0	K21 89 gamma nominal
+ 79	DEC	-0.1312738E-9	K20 89 below
+ 80	DEC	0.4815826E-7	K19 89 staging
+ 81	DEC	-0.5744402E-5	K18 89
+ 82	DEC	0.1623070E-3	K17 89
+ 83	DEC	0.01064008	K16 89

TABLE 3-3. TCCONS TABLE FORMAT (Cont'd)

LOCATION		VALUE	EXPLANATION
TCCONS + 84	DEC	0.0	K26 89 gamma nominal
+ 85	DEC	0.0	K25 89 above
+ 86	DEC	0.1038273E-4	K24 89 staging
+ 87	DEC	-0.6830055E-2	K23 89
+ 88	DEC	1.114796	K22 89
+ 89	DEC	0.0	Alpha (seconds)
+ 90	DEC	.92534419E-3	VP (Mercury units) 24 ft./sec.
+ 91	DEC	14.747978	K3 50 coefficients to
+ 92	DEC	-11.267406	K2 50 obtain accel. vs.
+ 93	DEC	3.7876281	K1 50 velocity (B-GE)
+ 94	DEC	.00476474885	Delta long (.273 tolerance) radians
+ 95	DEC	900.0	Artificial time to fire retro rockets
+ 96	DEC	131.4	S strip seconds
+ 97	DEC	3441.35029	Radians of Earth nautical miles
+ 98	DEC	23391.3384	Mean value of velocity ft./sec.
+ 99	DEC	-4.1594	Mean value of gamma
+ 100	DEC	.28548238E-9	Empirical
+ 101	DEC	.68265288E-9	Curve
+ 102	DEC	.62475098E-6	Fit
+ 103	DEC	.59100400E-5	Constants
+ 104	DEC	.31345620E-5	
+ 105	DEC	-.43086694E-2	Empirical
+ 106	DEC	.23747061E-1	Curve
+ 107	DEC	.44627038E-1	Fit
+ 108	DEC	.66648080	Constants
+ 109	DEC	.27185713E + 1	
+ 110	DEC	.14873996E + 2	
+ 111	DEC	.86767697E + 2	
+ 112	DEC	581.242093	
+ 113	OCT	001000000000	1 in characteristic
+ 114	DEC	0.7853981634	
+ 115	DEC	0.1396263401	Max. plotboard value radians
+ 116	DEC	0	Start of table
+ 117	DEC	5.536184381	Longitude of area A radians
+ 118	DEC	5.606870216	Longitude of area B radians
+ 119	DEC	5.705481319	Longitude of area C radians
+ 120	DEC	5.832890354	Longitude of area D radians
+ 121	DEC	0	Longitude of area E or E prime
+ 122	DEC	7.033676885	Longitude of area XB radians
+ 123	DEC	50.	End of table
+ 124	DEC	0	
+ 125	DEC	0	

TABLE 3-3. TCCONS TABLE FORMAT (Cont'd)

LOCATION		VALUE	EXPLANATION
TCCONS + 126	DEC	0	
+ 127	DEC	6.00393262	Longitude of area E radians
+ 128	DEC	.334517988E-3	Epsilon (7,000) Mercury
+ 129	DEC	.570722665	(32.7°) Inclination angle deviation (R)
+ 130	DEC	806.8104	Mercury to seconds
+ 131	DEC	0	Code for area A
+ 132	DEC	1	Code for area B
+ 133	DEC	2	Code for area C
+ 134	DEC	3	Code for area D
+ 135	DEC	4	Code for area E
+ 136	DEC	9	Code for area XB
+ 137	DEC	0	
+ 138	DEC	0	
+ 139	DEC	0	
+ 140	DEC	0	
+ 141	DEC	0	
+ 142	DEC	0	
+ 143	DEC	5.93411945	Longitude area E prime (radians)
+ 144	OCT	000074000000	60 at binary scale of 17
+ 145	DEC	2.0	Delta T used in V/V_r calculations
+ 146	DEC	753.869034	Convert seconds to B-GE units
+ 147	DEC	-1.406417099	Longitude of pad (-80.581764) GE radar
+ 148	DEC	0.496492154	Geodetic latitude of pad (28.446905) GE radar
+ 149	DEC	20000000.	Convert from B-GE to feet
+ 150	DEC	0	Not used in launch
+ 151	DEC	0	Not used in launch
+ 152	DEC	0	Not used in launch
+ 153	DEC	0	Not used in launch
+ 154	DEC	-0.00044	Orbit lifetime constants K1,7
+ 155	DEC	-0.00042	Orbit lifetime constants K2,7
+ 156	DEC	+0.00625	Orbit lifetime constants K3,7
+ 157	DEC	-0.00020	Orbit lifetime constants K4,7
+ 158	DEC	+0.998142	Orbit lifetime constants K5,7
+ 159	DEC	-0.00047	Orbit lifetime constants K1,6
+ 160	DEC	+0.00013	Orbit lifetime constants K2,6
+ 161	DEC	+0.00623	Orbit lifetime constants K3,6
+ 162	DEC	-0.00013	Orbit lifetime constants K4,6
+ 163	DEC	+0.998028	Orbit lifetime constants K5,6
+ 164	DEC	-0.00010	Orbit lifetime constants K1,5
+ 165	DEC	+0.00025	Orbit lifetime constants K2,5
+ 166	DEC	+0.00595	Orbit lifetime constants K3,5
+ 167	DEC	-0.00015	Orbit lifetime constants K4,5
+ 168	DEC	+0.997925	Orbit lifetime constants K5,5

TABLE 3-3. TCCONS TABLE FORMAT (Cont'd)

LOCATIONS		VALUE	EXPLANATION
TCCONS + 169	DEC	-0.00005	Orbit lifetime constants K1,4
+ 170	DEC	+0.00023	Orbit lifetime constants K2,4
+ 171	DEC	+0.00578	Orbit lifetime constants K3,4
+ 172	DEC	-0.00014	Orbit lifetime constants K4,4
+ 173	DEC	+0.997802	Orbit lifetime constants K5,4
+ 174	DEC	-0.00007	Orbit lifetime constants K1,3
+ 175	DEC	+0.00016	Orbit lifetime constants K2,3
+ 176	DEC	+0.00571	Orbit lifetime constants K3,3
+ 177	DEC	-0.00012	Orbit lifetime constants K4,3
+ 178	DEC	+0.997665	Orbit lifetime constants K5,3
+ 179	DEC	-0.00016	Orbit lifetime constants K1,2
+ 180	DEC	+0.00005	Orbit lifetime constants K2,2
+ 181	DEC	+0.00565	Orbit lifetime constants K3,2
+ 182	DEC	-0.00007	Orbit lifetime constants K4,2
+ 183	DEC	+0.997493	Orbit lifetime constants K5,2
+ 184	DEC	4.50114449	A 210 nominal
+ 185	DEC	4.49533262	A 240 nominal
+ 186	DEC	4.48625250	A 270 nominal
+ 187	DEC	4.47028633	A 300 nominal
+ 188	DEC	4.43847700	A 330 nominal
+ 189	DEC	4.436898672	A 360 nominal
+ 190	DEC	4.12076599	A 390 nominal
+ 191	DEC	.007860792	E 210 nominal
+ 192	DEC	.040957897	E 240 nominal
+ 193	DEC	.0793944949	E 270 nominal
+ 194	DEC	.131757680	E 300 nominal
+ 195	DEC	.217293491	E 330 nominal
+ 196	DEC	.371598051	E 360 nominal
+ 197	DEC	.745107237	E 390 nominal
+ 198	DEC	0	Not used in launch
+ 199	DEC	0	Not used in launch
+ 200	DEC	0	Not used in launch
+ 201	DEC	0	Not used in launch
+ 202	DEC	0.84	Break point for tail off acceleration fit
+ 203	DEC	50.872771	Coefficients for
+ 204	DEC	-83.680596	Tail off acceleration
+ 205	DEC	34.800786	First half of curve
+ 206	DEC	.029344515	2nd Half of curve
+ 207	DEC	-.25351989	
+ 208	DEC	.55917158	
+ 209	DEC	1.0068147688	
+ 210	DEC	-6.3557E-4	K4 81 Constants for V-GO calculation
+ 211	DEC	3.6157E-4	K3 81

TABLE 3-3. TCCONS TABLE FORMAT (Cont'd)

LOCATION	VALUE	EXPLANATION
TCCONS + 212	DEC 7.0651E-3	K2 81
+ 213	DEC -4.2719E-4	K1 81
+ 214	DEC 0.996469	K0 81
+ 215	DEC 3	J (number of inputs above H min.)
+ 216	DEC 0.00477881893	H min. (min. altitude for high abort before tower Separation in Mercury units) 100,000 feet = H min.
+ 217	DEC 150.	L time acquisition data is first computed
+ 218	DEC 30.	N interval between 2 consecutive acquisition messages
+ 219	DEC 210.0	K time first message is valid
+ 220	DEC -0.93491788E-6	A coefficient for azimuth acquisition curve-Bermuda
+ 221	DEC 0.22319669E-3	B coefficient for azimuth acquisition curve-Bermuda
+ 222	DEC 4.4954723	C coefficient for azimuth acquisition curve-Bermuda
+ 223	DEC 0.75648865E-6	A coefficient for Bermuda acquisition curve fit-elevation
+ 224	DEC 0.77038945E-3	B coefficient for Bermuda acquisition curve fit-elevation
+ 225	DEC -0.18713691	C coefficient for Bermuda acquisition curve fit-elevation
+ 226	DEC 480.0	K_{f1} constants used to compute
+ 227	DEC 480.0	K_{f2}
+ 228	DEC .17453292	Limit of $\gamma - \gamma$ nominal
+ 229	DEC .15	Limit of V/V_r - V/V_r nominal
+ 230	OCT 77777	
+ 231	DEC 14	
+ 232	DEC 0.0071429	Constants for change of
+ 233	DEC 0.64287	Range when capsule weight changes
+ 234	DEC 5.0	K minimum number of points required to compute Standard deviation
+ 235	DEC 3.25	Confidence coefficient for 10 points
+ 236	DEC 3.355	Confidence coefficient for 9 points
+ 237	DEC 3.499	Confidence coefficient for 8 points
+ 238	DEC 3.707	Confidence coefficient for 7 points
+ 239	DEC 4.032	Confidence coefficient for 6 points
+ 240	DEC 4.604	Confidence coefficient for 5 points
+ 241	DEC 0.0872664625	Bermuda command minimum elevation angle in radians
+ 242	DEC 0.1451830162	Bermuda command maximum range in Mercury units
+ 243	DEC 0.88678818	Below this velocity capsule is in Cape range

TABLE 3-4. TCMANR TABLE FORMAT

LOCATION	SYMBOL		EXPLANATION OF SYMBOL	UNITS	FORM
TCMANR	T_i		Time since lift-off	Seconds	Flt. Pt.
+ 1	X_3	$\bar{R}_3$	Position coordinate	Mercury	Flt. Pt.
+ 2	Y_3		Position coordinate	Mercury	Flt. Pt.
+ 3	Z_3		Position coordinate	Mercury	Flt. Pt.
+ 4	$\dot{X}_3$	$\bar{V}_3$	Velocity coordinat	Mercury	Flt. Pt.
+ 5	$\dot{Y}_3$		Velocity coordinate	Mercury	Flt. Pt.
+ 6	$\dot{Z}_3$		Velocity coordinate	Mercury	Flt. Pt.
+ 7	s		Cross range distance	N. miles	Flt. Pt.
+ 8	$\gamma - \gamma_{nom}$		Flight path angle deviation	Radians	Flt. Pt.
+ 9	$V/V_r - (V/V_r)_{nom}$		Velocity ratio deviation	Dimensionless	Flt. Pt.
+ 10	d		Down range distance	N. miles	Flt. Pt.
+ 11	T_{SECO}		Time tag of SECO signal	Seconds	Flt. Pt.
+ 12	T_{ABOI}		Time tag of ABOI signal	Seconds	Flt. Pt.
+ 13	$\bar{t}$		= T capsule separation = T SECO + F (constant) = T ABOI + G (constant)	Seconds	Flt. Pt.
+ 14	$T_{SECO\ nom}$		Calculated time tag	Seconds	Flt. Pt.
+ 15	I_i		Tailoff velocity at T_i	Mercury	Flt. Pt.
+ 16	G		Internal counter of cycles after CAPS		Integer
+ 17	H		Internal counter of B-GE and Azusa data		Flt. Pt.
+ 18	T_i		Time tag of discrete signal	Seconds	Flt. Pt.
+ 19	- - -		Discrete signals		Binary

TABLE 3-4. TCMANR TABLE FORMAT (Cont'd)

LOCATION	SYMBOL		EXPLANATION OF SYMBOL	UNITS	FORM
TCMANR					
+ 20	T_{iT}		Time tag of telemetry signals	Seconds	Flt. Pt.
+ 21	- - -		Telemetry signals		Binary
+ 22	ΣT_{13}		Summation of time tags after CAPS	Seconds	Flt. Pt.
+ 23	ΣX_{13}	$\Sigma \bar{R}_{13}$	Components of $\Sigma \bar{R}_{13}$	Mercury	Flt. Pt.
+ 24	ΣY_{13}			Mercury	Flt. Pt.
+ 25	ΣZ_{13}			Mercury	Flt. Pt.
+ 26	$\Sigma \dot{X}_{13}$	$\Sigma \bar{V}_{13}$	Components of $\Sigma \bar{V}_{13}$	Mercury	Flt. Pt.
+ 27	$\Sigma \dot{Y}_{13}$			Mercury	Flt. Pt.
+ 28	$\Sigma \dot{Z}_{13}$			Mercury	Flt. Pt.
+ 29	T_{now}		$= T_i + T_{L.O.}$	Seconds	Flt. Pt.
+ 30	- - -		Code Word of selected data source 1) B-GE 2) Raw AN/FPS-16 3) IP-7090 (Azusa) 4) IP - 7090 (AN/FPS-16)		Integer This code Word will be stored by Monitor
+ 31	$\lvert R_3 \rvert$		Magnitude of R_3	Mercury	Flt. Pt.
+ 32	$\lvert V_3 \rvert$		Magnitude of V_3	Mercury	Flt. Pt.
+ 33	$\dot{X}_8$	$\bar{V}_8$	Coordinates of $\bar{V}_8$	Mercury	Flt. Pt.
+ 34	$\dot{Y}_8$			Mercury	Flt. Pt.
+ 35	$\dot{Z}_8$				
+ 36	$\lvert V_8 \rvert$		Magnitude of V_8	Mercury	Flt. Pt.
+ 37	$\lvert V_1 \rvert$		Magnitude of V_1	B-GE	Flt. Pt.
+ 38	V_r			Ft./Sec.	

TABLE 3-4. TCMANR TABLE FORMAT (Cont'd)

LOCATION	SYMBOL	EXPLANATION OF SYMBOL	UNITS	FORM
TCMANR				
+ 39	T_{CAPS}	Time tag of capsule separation signal	Seconds	Flt. Pt.
+ 40	a	Tailoff acceleration	Mercury	Flt. Pt.
+ 41	$\dot{X}_{8e}$		Mercury	Flt. Pt.
+ 42	$\dot{Y}_{8e}$ $\bar{V}_{8e}$	Coordinates of V_{8e} vector	Mercury	Flt. Pt.
+ 43	$\dot{Z}_{8e}$		Mercury	Flt. Pt.
+ 44	T_{fx}	Time of impact	Seconds	Flt. Pt.
+ 45	X_{fx}		Mercury	Flt. Pt.
+ 46	Y_{fx} $\bar{R}_{fx}$	Coordinates of R_{fx} position vector	Mercury	Flt. Pt.
+ 47	Z_{fx}		Mercury	Flt. Pt.
+ 48	T_{IND}	Time tag for no data	Seconds	Flt. Pt.
+ 49		Iteration required 1 - no iteration required 0 - iteration required		
+ 50	n	Number of posigrades fired		Integer
+ 51	$\lvert V_{12} \rvert$	Magnitude of V_{12}	Mercury	Flt. Pt.
+ 52	$\dot{X}_{12}$		Mercury	Flt. Pt.
+ 53	$\dot{Y}_{12}$ $\bar{V}_{12}$	Coordinates of V_{12} velocity vector	Mercury	Flt. Pt.
+ 54	$\dot{Z}_{12}$			
+ 55	T_{ave}	Time tag associated with $\lvert R_{3\ ave} \rvert$ and $\lvert V_{12\ ave} \rvert$	Seconds	Flt. Pt.
+ 56	$\lvert V_{12} \rvert_{ave}$	Magnitude of V_{12} average	Mercury	Flt. Pt.
+ 57	$\lvert R_3 \rvert_{ave}$	Magnitude of R_3 average	Mercury	Flt. Pt.
+ 58	$H_{min\ ret}$	Altitude at which retros must fire	Mercury	Flt. Pt.

TABLE 3-4. TCMANR TABLE FORMAT (Cont'd)

LOCATION	SYMBOL		EXPLANATION OF SYMBOL	UNITS	FORM
TCMANR					
+ 59	Δt_{rmini}		Minimum delay time to fire retro rockets	Seconds	Flt. Pt.
+ 60	T_{i-1}		Time tag of previous cycle	Seconds	Flt. Pt.
+ 61	Δt_r		Time to fire retro rockets	Seconds	Flt. Pt.
+ 62	n + 1		Multiplier for escape rocket velocity	Mercury	Flt. Pt.
+ 63	$\lvert V_{10} \rvert$		Magnitude of V_{10}	Mercury	Flt. Pt.
+ 64			Used as temporary storage		
.					
.					
.					
.					
+ 164			Used as temporary storage		
+ 165	T_{4max}		Time tag	Seconds	Flt. Pt.
+ 166	$\lvert V_6 \rvert$		V magnitude used in empirical fit	Mercury	Flt. Pt.
+ 167	γ_6		Gamma in empirical curve fit	Radians	Flt. Pt.
+ 168	T_6		Time tag associated with R_6, V_6	Seconds	Flt. Pt.
+ 169	X_6			Mercury	Flt. Pt.
+ 170	Y_6	$\bar{R}_6$	Coordinates of R_6 position vector	Mercury	Flt. Pt.
+ 171	Z_6			Mercury	Flt. Pt.
+ 172	$\dot{X}_6$			Mercury	Flt. Pt.
+ 173	$\dot{Y}_6$	$\bar{V}_6$	Coordinates of V_6 velocity vector	Mercuty	Flt. Pt.
+ 174	$\dot{Z}_6$				
+ 175	λ_6		Longitude of 450,000 ft.	Radians	Flt. Pt.

TABLE 3-4. TCMANR TABLE FORMAT (Cont'd)

LOCATION	SYMBOL	EXPLANATION OF SYMBOL	UNITS	FORM
TCMANR				
+ 176	ϕ_6	Geodetic latitude at 450,000 ft.	Radians	Flt. Pt.
+ 177	h_e	Height of capsule	Mercury	Flt. Pt.
+ 178		Code word for duplicating printing 1 - Message has been printed 0 - Message has not been printed binary		
+ 179	ABTS	Abort before TOWS indicator 1 = Abort before TOWS 0 = No abort before TOWS		
+ 180	δ_1	Used in CMMLG		Flt. Pt.
+ 181	δ_2	Used in CMMLG		Flt. Pt.
+ 182	λ_{SEL}	Longitude of selected recovery area	Radians	Flt. Pt.
+ 183	$\lvert V_{11} \rvert$	Increment of escape rocket velocity	Mercury	Flt. Pt.
+ 184	h_{AP}	Apogee height	Mercury	Flt. Pt.
+ 185	H_{per}	Perigee height	Mercury	Flt. Pt.
+ 186	e	Eccentricity	Dimensionless	Flt. Pt.
+ 187		XB indicator 0 - do not change computed λ_{max} 1 - make $\lambda_{max} = 8°$		
+ 188	$\lvert V_{13} \rvert$	Magnitude of V_{13}	Mercury	Flt. Pt.
+ 189	$\dot{X}_{13}$		Mercury	Flt. Pt.
+ 190	$\dot{Y}_{13}$ $\bar{V}_{13}$	Coordinates of V_{13} velocity vector	Mercury	Flt. Pt.
+ 191	$\dot{Z}_{13}$		Mercury	Flt. Pt.
+ 192	F	Internal counter for AN/FPS-16 data	Mercury	
+ 193	T_{TOWS}	Time tag of TOWS signal	Seconds	

TABLE 3-4. TCMANR TABLE FORMAT (Cont'd)

LOCATIONS	SYMBOL		EXPLANATION OF SYMBOL	UNITS	FORMAT
TCMANR					
+ 194	T_{ESCR}		Time tag of ESCR signal	Seconds	
+ 195	$\lvert V_{13} \rvert_{ave}$		Magnitude of $V_{13\ ave}$ vector	Mercury	
+ 196					
+ 197	X_{3ave}	$\bar{R}_{3ave}$	Components of R_3 average	Mercury	Flt. Pt.
+ 198	Y_{3ave}			Mercury	Flt. Pt.
+ 199	Z_{3ave}			Mercury	Flt. Pt.
+ 200	$\dot{X}_{12ave}$	V_{12ave}	Components of V_{12} average	Mercury	Flt. Pt.
+ 201	$\dot{Y}_{12ave}$			Mercury	Flt. Pt.
+ 202	$\dot{Z}_{12ave}$			Mercury	Flt. Pt.
+ 203	$\dot{X}_{13ave}$	V_{13ave}	Components of V_{13} average	Mercury	Flt. Pt.
+ 204	$\dot{Y}_{13ave}$			Mercury	Flt. Pt.
+ 205	$\dot{Z}_{13ave}$			Mercury	Flt. Pt.
+ 206	ΣT_i		Summation of time tags after CAPS		
+ 207	ΣX_{14}	$\Sigma \bar{R}_{14}$	Components of $\Sigma \bar{R}_{14}$	Mercury	
+ 208	ΣY_{14}			Mercury	
+ 209	ΣZ_{14}			Mercury	
+ 210	$\Sigma \dot{X}_{14}$	$\Sigma \bar{V}_{14}$	Components of $\Sigma \bar{V}_{14}$		
+ 211	$\Sigma \dot{Y}_{14}$				
+ 212	$\Sigma \dot{Z}_{14}$				

TABLE 3-4. TCMANR TABLE FORMAT (Cont'd)

LOCATION	SYMBOL	EXPLANATION OF SYMBOL	UNITS	FORM
TCMANR				
+ 213	X_{14}		Mercury	Flt. Pt.
+ 214	Y_{14} $\bar{R}_{14}$	Components of $\bar{R}_{14}$	Mercury	Flt. Pt.
+ 215	Z_{14}			
+ 216	$\dot{X}_{14}$		Mercury	Flt. Pt.
+ 217	$\dot{Y}_{14}$ $\bar{V}_{14}$	Components of $\bar{V}_{14}$	Mercury	Flt. Pt.
+ 218	$\dot{Z}_{14}$		Mercury	Flt. Pt.
+ 219		"Many areas" indicator used in CMMLG	Integer	
+ 220	λ_{max}	Longitude of maximum delay impact point	Radians	Flt. Pt.
+ 221	GMTRC	Greewich Mean Time of retrofire	Seconds	Flt. Pt.
+ 222	ϕ_{IP}	Latitude IP refined	Mercury	Flt. Pt.
+ 223	λ_{IP}	Longitude IP refined	Mercury	Flt. Pt.
+ 224		LP computed indicator 0 = refined IP not computed 1 = refined IP computed	Integer	
+ 225	T_i	Time of acquisition data	Seconds	Flt. Pt.
+ 226	A_{Ti}	Azimuth at time of computation	Radians	Flt. Pt.
+ 227	E_{Ti}	Elevation at time of computation	Radians	Flt. Pt.
+ 228	A_{nom}	Nominal azimuth	Radians	Flt. Pt.
+ 229	E_{nom}	Nominal elevation	Radians	Flt. Pt.
+ 230	A_K	First azimuth	Radians	Flt. Pt.
+ 231	A_{K+N}	Second azimuth	Radians	Flt. Pt.
+ 232	A_{K+2N}	Third azimuth	Radians	Flt. Pt.

TABLE 3-4. TCMANR TABLE FORMAT (Cont'd)

LOCATION	SYMBOL	EXPLANATION OF SYMBOL	UNITS	FORM
TCMANR				
+ 233	A_{K+3N}	Fourth azimuth	Radians	Flt. Pt.
+ 234	E_K	First elevation	Radians	Flt. Pt.
+ 235	E_{K+N}	Second elevation	Radians	Flt. Pt.
+ 236	E_{K+2N}	Third elevation	Radians	Flt. Pt.
+ 237	E_{K+3N}	Fourth elevation	Radians	Flt. Pt.
+ 238	T_3 or T_{3ave}		Mercury	Flt. Pt.
+ 239	X_3 or X_{3ave}		Mercury	Flt. Pt.
+ 240	Y_3 or Y_{3ave}		Mercury	Flt. Pt.
+ 241	Z_3 or Z_{3ave}		Mercury	Flt. Pt.
+ 242	$\dot{X}_{12}$ or $\dot{X}_{12ave}$		Mercury	Flt. Pt.
+ 243	$\dot{Y}_{12}$ or $\dot{Y}_{12ave}$		Mercury	Flt. Pt.
+ 244	$\dot{Z}_{12}$ or $\dot{Z}_{12ave}$		Mercury	Flt. Pt.
+ 245	$\lvert V_{12} \rvert$ or $\lvert V_{12} \rvert_{ave}$		Mercury	Flt. Pt.
+ 246	ΔT_L	Time remaining to land	Seconds	Flt. Pt.
+ 247	- - -	Count of points above H_{min}	Integer	Fix. Pt.
+ 248	- - -	Indicator to avoid duplicate printing of capsule separation 0 = print 1 = do not print	Integer	
+ 249	- - -	Not used		
+ 250	R	Reject counter		Flt. Pt.
+ 251	- - -	H Counter	Integer	
+ 252	LLV	Lower limit for V		Flt. Pt.

TABLE 3-4. TCMANR TABLE FORMAT (Cont'd)

LOCATION	SYMBOL	EXPLANATION OF SYMBOL	UNITS	FORM
TCMANR				
+ 253	ULV	Upper limit for V		Flt. Pt.
+ 254	LLG	Lower limit for γ		Flt. Pt.
+ 255	ULG	Upper limit for γ		Flt. Pt.
+ 256	σ_V	Standard deviation of V_{13}		Flt. Pt.
+ 257	σ_γ	Standard deviation of γ		Flt. Pt.
+ 258	$\lvert V_{13} \rvert$			Flt. Pt.
+ 259	$\lvert V_{13} \rvert^2$			Flt. Pt.
+ 260	$(\gamma')^2$			Flt. Pt.
+ 261	t	Confidence coefficient		Flt. Pt.
+ 262	$\lambda_{SEL\ new}$		Radians	Flt. Pt.
+ 263	$\Delta t_{r\ new}$		Seconds	Flt. Pt.
+ 264	λ_{WEST}	Long. of recovery area west of λ_{min}	Radians	Flt. Pt.
+ 265				
+ 266	T_4		Seconds	
+ 267	X_4		Mercury	Flt. Pt.
+ 268	Y_4 $\bar{R}_4$		Mercury	Flt. Pt.
+ 269	Z_4		Mercury	Flt. Pt.
+ 270	$\dot{X}_4$		Mercury	Flt. Pt.
+ 271	$\dot{Y}_4$ $\bar{V}_4$		Mercury	Flt. Pt.
+ 272	$\dot{Z}_4$		Mercury	Flt. Pt.

NOMENCLATURE OF $\overline{R}$ AND $\overline{V}$ FOR LAUNCH

$\overline{R}_{0G}$; $\overline{V}_{0G}$ = Original B-GE as received from input program, fixed point.

$\overline{R}_{0I}$; $\overline{V}_{0I}$ = Original IP 7090 input as received from input programs.

$\overline{R}_{0M}$; $\overline{V}_{0M}$ = Raw AN/FPS-16 data converted and transterred into Mercury reference system and Mercury units.

$\overline{R}_{1G}$; $\overline{V}_{1G}$ = Components in B-GE reference system in B-GE units, floating point.

$\overline{R}_{1I}$; $\overline{V}_{1I}$ = Components in IP 7090 reference system in B-GE units, floating point.

$\overline{R}_{1M}$; $\overline{V}_{1M}$ = Components in Mercury reference system in B-GE units, floating point.

$\overline{R}_2$ = Components in GE centered down range system in B-GE units (local)

$\overline{R}_3$; $\overline{V}_3$ = Components in Mercury reference system and Mercury units

$\overline{R}_4$; $\overline{V}_4$ = Vector components at beginning of retrofire in Mercury reference system, Mercury units.

$\overline{R}_5$; $\overline{V}_5$ = Vector components at end of retrofire in Mercury reference system, Mercury units.

$\overline{R}_6$; $\overline{V}_6$ = Vector components at 450,000 feet in Mercury reference system, Mercury units.

$\overline{V}_8$ = Vector components after escape rocket burnout in Mercury reference system, Mercury units ($\overline{R}_8 = \overline{R}_3$).

$|V_{10}|$ = Velocity increment due to firing of the posigrade rockets. Mercury units.

$|V_{11}|$ = Velocity increment due to firing of the escape rockets. Mercury units.

$\overline{V}_{12}$ = Vector components of velocity adjusted for posigrades and tail-off Mercury reference system, Mercury units.

$\overline{V}_{13}$ = Vector components of velocity after adjustment for tail-off only. Mercury reference system, Mercury units.

$\overline{R}_{14}$; $\overline{V}_{14}$ = Vector components computed from raw or processed AN/FPS-16 data. Mercury reference system, Mercury units.

$\overline{V}_{8e}$ = Same as $\overline{V}_8$ but in earth fixed reference system.

$\overline{R}_f$ = Vector components at fictitious altitude (h_f) in Mercury reference system, Mercury units.

FIGURE 3-8. CCMAIN PROGRAM FLOW CHART

3.9 CAPSULE IMPACT PREDICTION CALCULATION (EMFIT)

This program is used by the launch phase to compute geodetic latitude and longitude at impact, given the position and velocity vectors, $(\overline{R}_6, \overline{V}_6)$ velocity magnitude $\left(| V_b |\right)$, flight path angle(γ_6), subcapsule geodetic latitude and longitude (ϕ_6, λ_6) at altitude of 450,000 feet. The flow chart for the EMFIT program is shown in Figure 3-34.

3.9.1 Input Requirements

The following library subroutines are used by the EMFIT program:

U1SQRT -- Square Root

U1ATNA -- Arctangent

U1ATAB -- Arctangent A/B

U1SICO -- Sine/Cosine

U1ASCO -- Arcsine/Arccosine

U1TACO -- Tangent/Cotangent

The following information from the TCMANR table is used as input to the EMFIT program:

TCMANR + 166: $| V_6 |$, Mercury units

+ 167: γ_6, radians

+ 169 —— 171 : $\overline{R}_6$ components, Mercury units

+ 172 —— 174 : $\overline{V}_6$ components, Mercury units

+ 175 : λ_6, radians

+ 176 ϕ_6, radians

The following information from the TCCONS table is also used as input to the EMFIT program:

TCCONS + 71 : factor for conversion from Mercury units to ft/sec.

+ 98 : $\overline{V}$, mean value of velocity, ft./sec.

+ 99 : $\overline{\gamma}$, mean value of gamma, degrees.

+ 100 —— + 111 : C_1 —— C_{12}

+ 112 : $\bar{I}$, nautical miles

+ 232 }
+ 233 } constants for linear correction of I due to change of capsule weight.

+ 33 : ω , rotational velocity of the earth in radians/Mercury unit of time.

+ 36 : R_e , radius of the earth, Mercury units

+ 46 : $\pi/2$, radians

+ 97 : R_e , nautical miles

+ 209 : $(a_c/b_c)^2$; factor for conversion from geocentric to geodetic latitude; a_c and b_c are dimensions of the Clarke spheroid.

3.9.2 Output Requirements

The output of the EMFIT program is placed in the TCMANR table.

TCMANR + 187 location is used to indicate that maximum IP = 8° should be displayed at Cape Canaveral in case of computed λ IP east of Greenwich.

Calling sequence indicates where ϕ_{IP} and λ_{IP} should be stored.

3.9.3 Method

During the spring of 1960, a series of integration tables were generated at NASA Langley. The computations resulted in a range of values for the downrange distance traversed in descending from 450,000 feet to 60,000 feet. Knowing the nominal trajectory for the Mercury-Atlas flights, it was possible to determine the expected V's and γ 's at 450,000 feet where 21,000 ft./sec. $\leq |V| \leq$ 25,600 ft./sec. The integration program used assumed an oblate earth, rotating atmosphere with density conditions defined by ARDC (1959), and three degrees of freedom where the capsule is a point mass with zero lift. Refer to Figure 3-35 for the following discussion:

The resultant values of downrange distance, ℓ , were fitted, using the method of least squares, and resulted in the following polynominal:

$$\ell = .28548238 \times 10^{-9} V^3 \gamma^4 + .68265288 \times 10^{-9} V^3 \gamma^3$$
$$+ .62475098 \times 10^{-6} V^2 \gamma^4 + .59100400 \times 19^{-5} V^2 \gamma$$
$$+ .31345620 \times 19^{-5} V^2 - .43086694 \times 10^{-2} V \gamma^3$$
$$+ .23747061 \times 10^{-1} V\gamma + .44627038 \times 19^{-1} V$$
$$+ .66648080 \times 10^{0} \gamma^4 - .27185713 \times 10 \gamma^3$$
$$+ .14873996 \times 10^{2} \gamma^2 + .86767697 \times 10^{2} \gamma + \bar{\ell}$$

where $V = |V_6| - \bar{V}$ with $\bar{V}$ = mean value of speed = 23,391.3384 ft./sec.; $\gamma = \gamma_6 - \bar{\gamma}$ with $\bar{\gamma}$ = mean value of flight path angle = -4.1594 degrees; ℓ = 581.242093 nautical miles.

The values of downrange distance which were obtained from the integrations and used in the empirical curve fit for l are a function of capsule weight. Since the capsule weight has increased approximately 300 pounds, a linear correction has been applied to the polynomial so that:

$$\ell_o = 1.0071429\ell + .64287.$$

In generating the values of l from which the polynominal was developed, the changes in latitude and heading angle at 450,000 feet were sufficiently small over the area in question that their effect could be ignored. The application of the polynominal is therefore limited to the North Atlantic Ocean portion of the Mercury range. The curve fit is also invalid once the capsule has entered orbit.

Having computed ℓ_o (nautical miles), it is possible to compute the latitude and longitude of impacts as follows:

a) compute geocentric latitude at 450,000 feet:

$$\ell_c = \arctan \frac{Z_6}{\sqrt{X_6^2 + Y_6^2}}$$ where X_6, Y_6, Z_6 are the components of $\bar{R}_6$ and $-\pi/2 \leq \ell_c \leq /2$.

b) Compute inclination angle at 450,000 feet:

$$i = \arctan \left[\frac{\sqrt{h_X^2 + h_Y^2}}{h_Z} \right]$$ and $\phi \leq i \leq |\pi|$.

where h_X, h_Y, h_Z are the components of the vector which results from the cross products of $\bar{R}_6$ and $\bar{V}_6$:

$$h_X = Y_6\dot{Z}_6 - Z_6\dot{Y}_6$$

$$H_Y = Z_6\dot{X}_6 - X_6\dot{Z}_6$$

$$h_Z = X_6\dot{Y}_6 - Y_6\dot{X}_6$$

c) Compute the relative heading angle at 450,000 feet, that is the angle between longitude and downrange direction:

$$\Psi_R = \arctan \frac{|V_6| \cos \gamma_6 \sin \Psi_i - \omega R_e \cos \ell_c}{|V_6| \cos \gamma_6 \cos \Psi_i}$$

where $\sin \Psi_i = \dfrac{\cos i}{\cos l_c}$, Ψ_i is the inertial heading angle, $-\pi \leq \Psi_i \leq \pi$.

$$\cos \Psi_i = \sqrt{1 - (\sin \Psi_i)^2}$$

ω = .058833543 radians/Mercury units of time

R_e = .999251039 Mercury units = radius of the earth.

For further explanation of these formulae, see 3.9.5, <u>Additional Computations.</u>

d) Compute co-latitude at 450,000 feet: $\ell_1 = \pi/2 - \ell_c$

e) Using the cosine law of spherical trigonometry compute the geocentric co-latitude at impact:

$$\ell_2 = \arccos \left[\sin \ell_1 \sin \frac{\ell o}{R_e} \cos \Psi_R + \cos \ell_1 \cos \frac{\ell o}{R_e} \right]$$

f) Geodetic latitude of impact is then $\phi_{IP} = \arctan \left[1.0068147688 \tan (\pi/2 - \ell_2)\right]$

g) Longitude at impact using the sin law:

$$\lambda_{IP} = \lambda_6 + \arcsin \frac{\sin \Psi_R \sin \ell o/Re}{\sin \ell_2}$$

3.9.4 Usage

a) Various sections of launch transfer to EMFIT.

b) Storage Required:

1) Instructions -- 257 locations

2) Tables -- 80 locations

c) Tables used:

TCMANR TCCONS	Launch tables

TEMFI BSS 80,S

d) Calling Sequence:

TSX	EMFIT, 4
PZE	Location ϕ_{IP} , , location λ_{IP}

e) Constants:

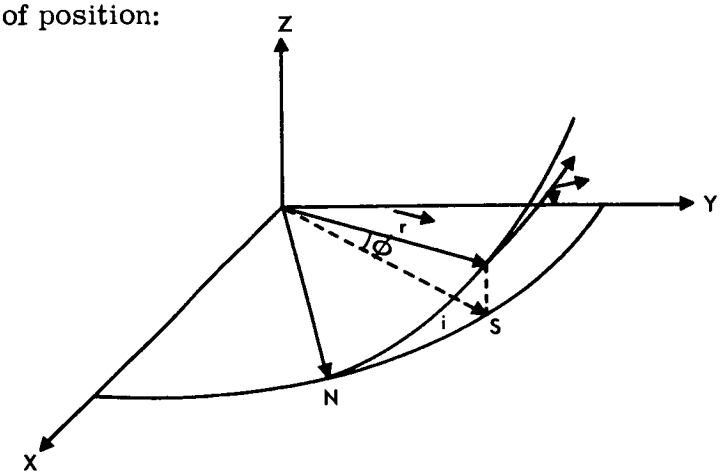
K.1RAD	DEC	57.29577951
K001.0	DEC	1.0
K00.PI	DEC	3.14159265
KKK2PI	DEC	6.283185307
K00001	DEC	1

f) Timing:

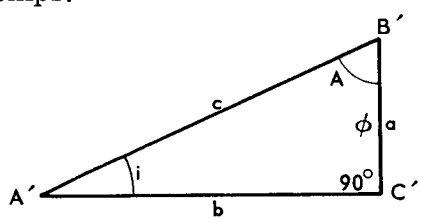
The EMFIT program uses approximately 1007 machine cycles on the 7090 or 2195.26 microseconds.

3.9.5 Additional Computations

Derivation of inertial azimuth in terms of inclination of the orbit and latitude of position:



Project $\overline{N}$, $\overline{r}$, $\overline{S}$ upon a unit sphere; the resulting spherical triangle has the relationships:



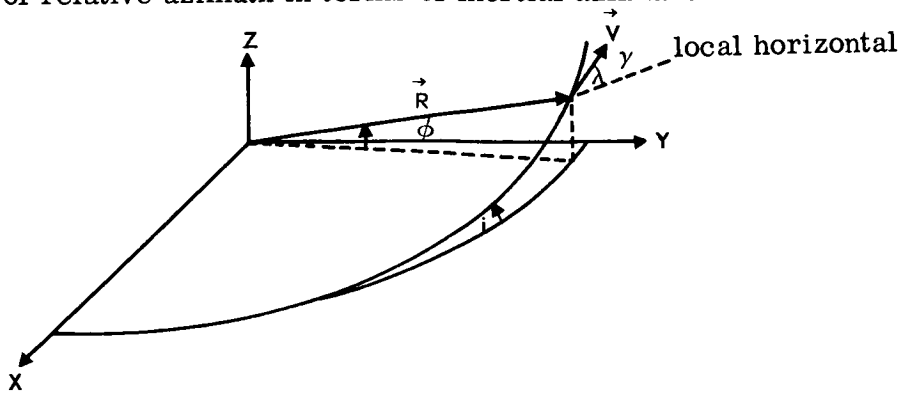
Using the law of cosines in its angular form: $\cos A' = -\cos B' \cos C' + \sin B' \sin C' \cos a$ the following is obtained:

$$\cos i = -\cos A \cos 90^{\circ} + \sin A \sin 90^{\circ} \cos \phi$$

or

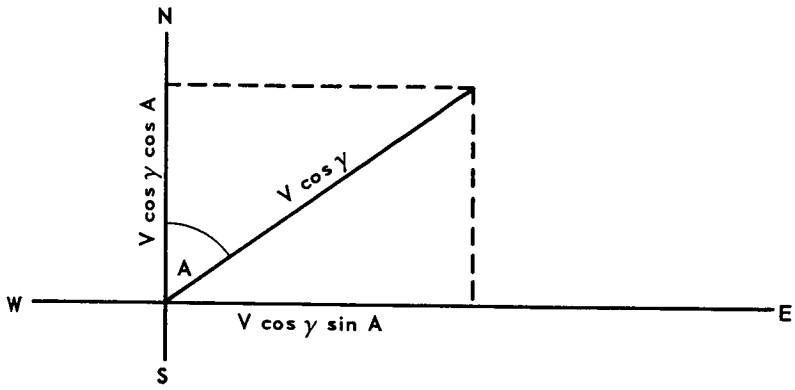
$$\sin A = \frac{\cos i}{\cos \phi}$$

Derivation of relative azimuth in terms of inertial azimuth:



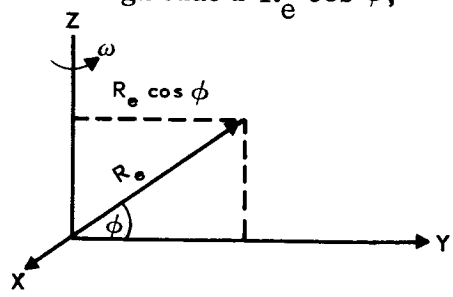
With reference to the triangle formed by dropping a perpendicular to the local horizontal from the velocity vector:

Then project the local horizontal into a plane comparable to the ZY plane, letting Z be North and the y-direction be East:

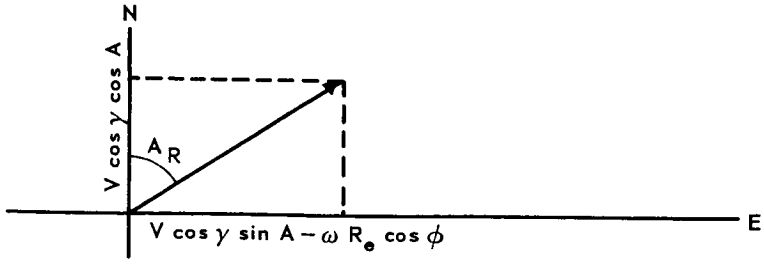


The angle between V cos γ and the N - axis is the inertial azimuth by definition.

Now, since the rotation of the earth results in a velocity in an easterly direction of magnitude $\omega R_e \cos \phi$,



This velocity is subtracted to obtain the relative azimuth of the capsule:



$$\text{and } \tan A_R = \frac{V \cos \gamma \sin A - \omega R_e \cos \phi}{V \cos \gamma \cos A}$$

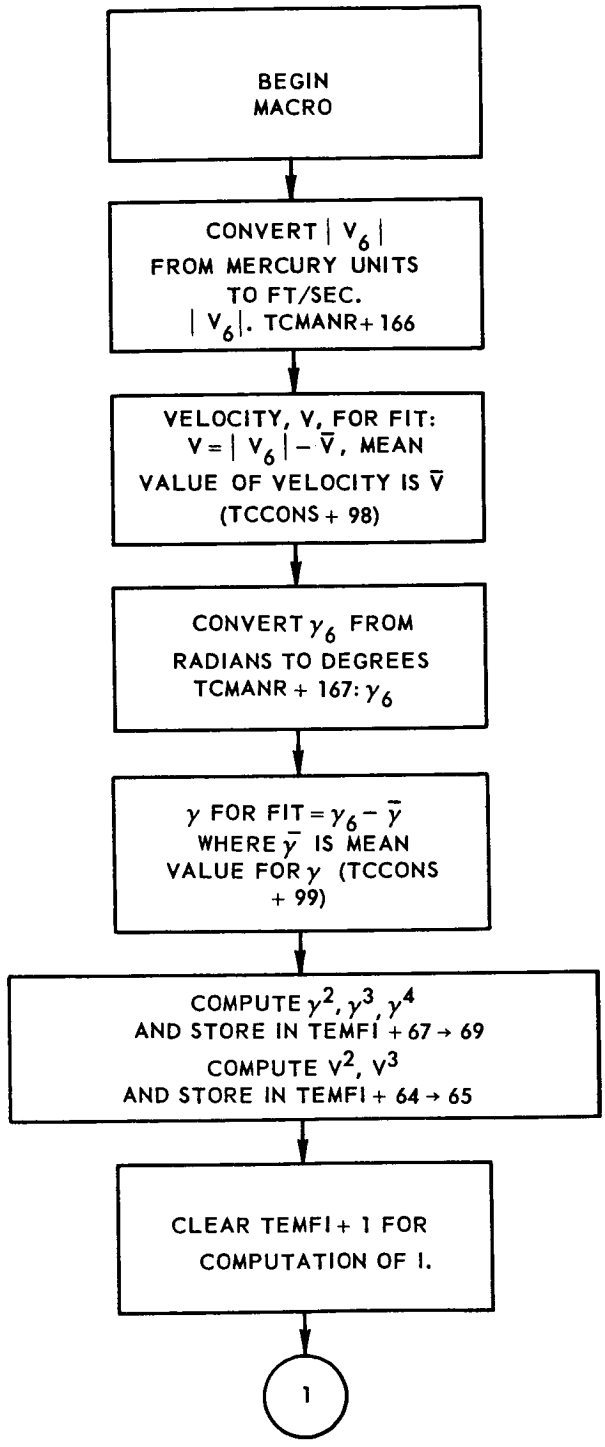


FIGURE 3-34. EMFIT PROGRAM FLOW CHART *(Sheet 1 of 4)*

1

COMPUTE $\ell = .C_1 V^3 \gamma^4 + C_2 V^3 \gamma^3 + C_3 V^2 \gamma^4 + C_4 V^2 \gamma$
$+ C_5 V^2 + C_6 V \gamma^3 + C_7 V \gamma + C_8 V + C_9 \gamma^4$
$+ C_{10} \gamma^3 + C_{11} \gamma^2 + C_{12} \gamma + \bar{\ell}$; ℓ IN NAUTICAL MILES

WHERE $V = V_{GIVEN} - 23391.3384$, V_{GIVEN} IN FT/SEC.

$\gamma = \gamma_{GIVEN} - (-4.1594)$, γ_{GIVEN} IN DEGREES

$\bar{\ell} = 581.242093$

AND

$C_1 = .28548238 \times 10^{-9}$	$C_7 = .23747061 \times 10^{-1}$
$C_2 = .68265288 \times 10^{-9}$	$C_8 = .44627038 \times 10^{-1}$
$C_3 = .62475098 \times 10^{-6}$	$C_9 = .66648080 \times 10^{0}$
$C_4 = .59100400 \times 10^{-5}$	$C_{10} = .27185713 \times 10^{+1}$
$C_5 = .31345620 \times 10^{-5}$	$C_{11} = .14873996 \times 10^{+2}$
$C_6 = -.43086694 \times 10^{-2}$	$C_{12} = .86767697 \times 10^{2}$

CORRECT ℓ FOR CHANGE IN CAPSULE WEIGHT.

$\ell = .0071429\ I + .64287 + \ell$

COMPUTE GEOCENTRIC LATITUDE:

$$\frac{\ell_c}{\phi} = \text{ARCTAN}\left[\frac{z_6}{\sqrt{x_6^2 + y_x^2}}\right]$$

WHERE x_6, y_6, z_6 ARE COMPONENTS OF $\bar{R}_6$.

2

FIGURE 3-34. EMFIT PROGRAM FLOW CHART *(Sheet 2 of 4)*

2

COMPUTE INCLINATION ANGLE AT 450,000 FT.:

$$\left.\begin{aligned} h_x &= y_6 \dot{z}_6 - z_6 \dot{y}_6 \\ h_y &= z_6 \dot{x}_6 - x_6 \dot{z}_6 \\ h_z &= x_6 \dot{y}_6 - y_6 \dot{x}_6 \end{aligned}\right\} \text{ AND } \quad i = \text{ARCTAN} \left[\frac{\sqrt{h_x^2 + h_y^2}}{h_z} \right]$$

COMPUTE Ψ_R: RELATIVE AZIMUTH

$$\sin \Psi_i = \frac{\cos i}{\cos \ell_c}, \quad \cos \Psi_i = N1 - \left(\frac{\cos i}{\cos \ell_c} \right)^2 \quad \left[\text{PUT SIGN OF } \dot{z}_6 \text{ ON } \cos \Psi_i \right]$$

$$\Psi_R = \arctan \left[\frac{|V| \cos \gamma, \sin \Psi_i - \omega R_e \cos \ell_c}{|V_6| \cos \gamma, \cos \Psi_i} \right]$$

WHERE ω = .058833543 RAD/MERCURY UNIT OF TIME

R_e = .999251039 MERCURY UNITS (RADIUS OF THE EARTH)

COMPUTE ω LATITUDE AT 450,000 FEET:

$$\ell_1 = \pi/2 - \ell_c$$

COMPUTE ω LATITUDE AT IMPACT POINT:

$$\ell_2 = \text{ARCCOS} \left[\sin \ell_1 \sin \frac{\ell}{R_e} \cos \Psi_R + \cos \ell_1 \cos \frac{\ell}{R_e} \right]$$

3

FIGURE 3-34. EMFIT PROGRAM FLOW CHART *(Sheet 3 of 4)*

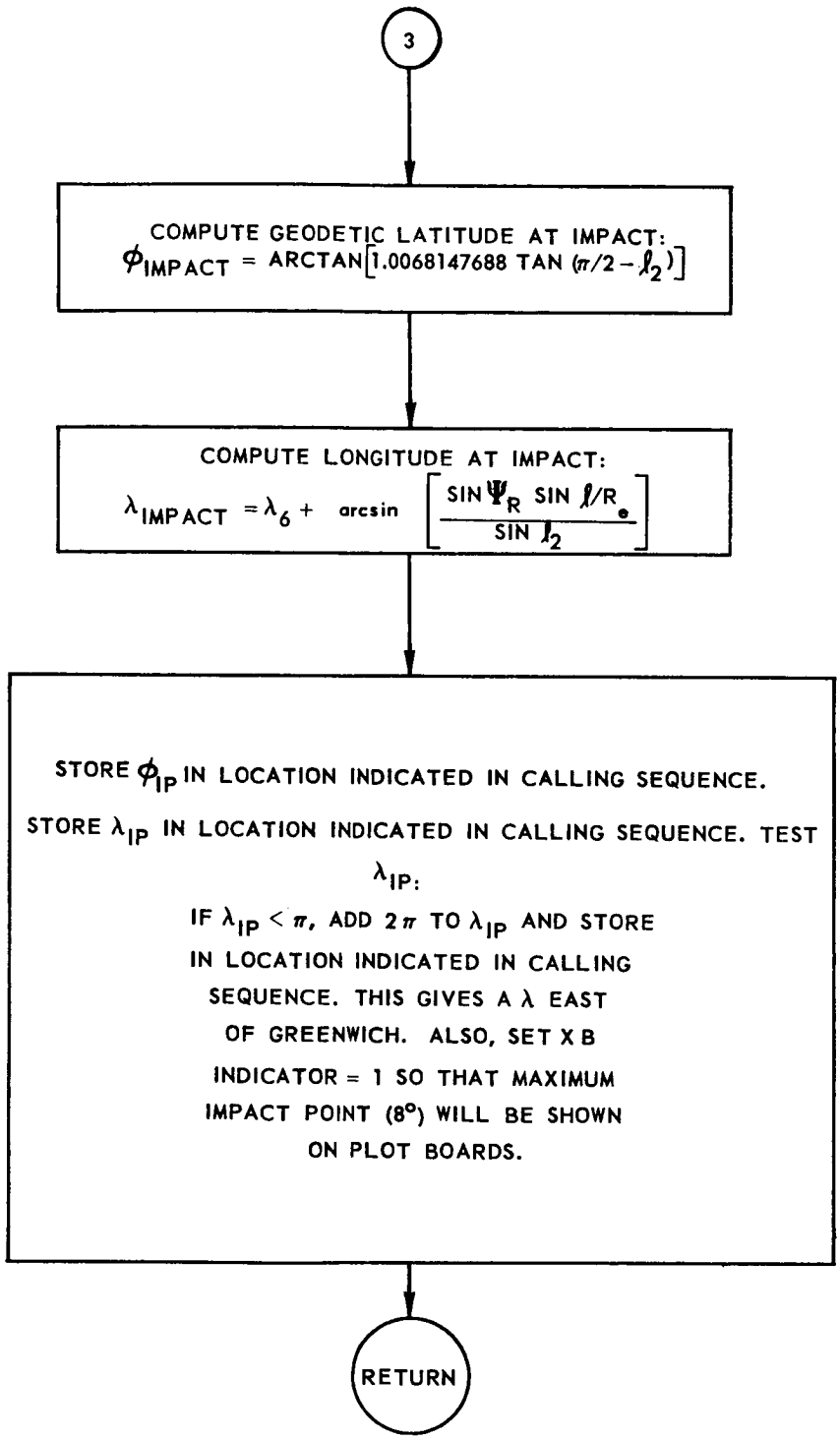


FIGURE 3-34. EMFIT PROGRAM FLOW CHART *(Sheet 4 of 4)*

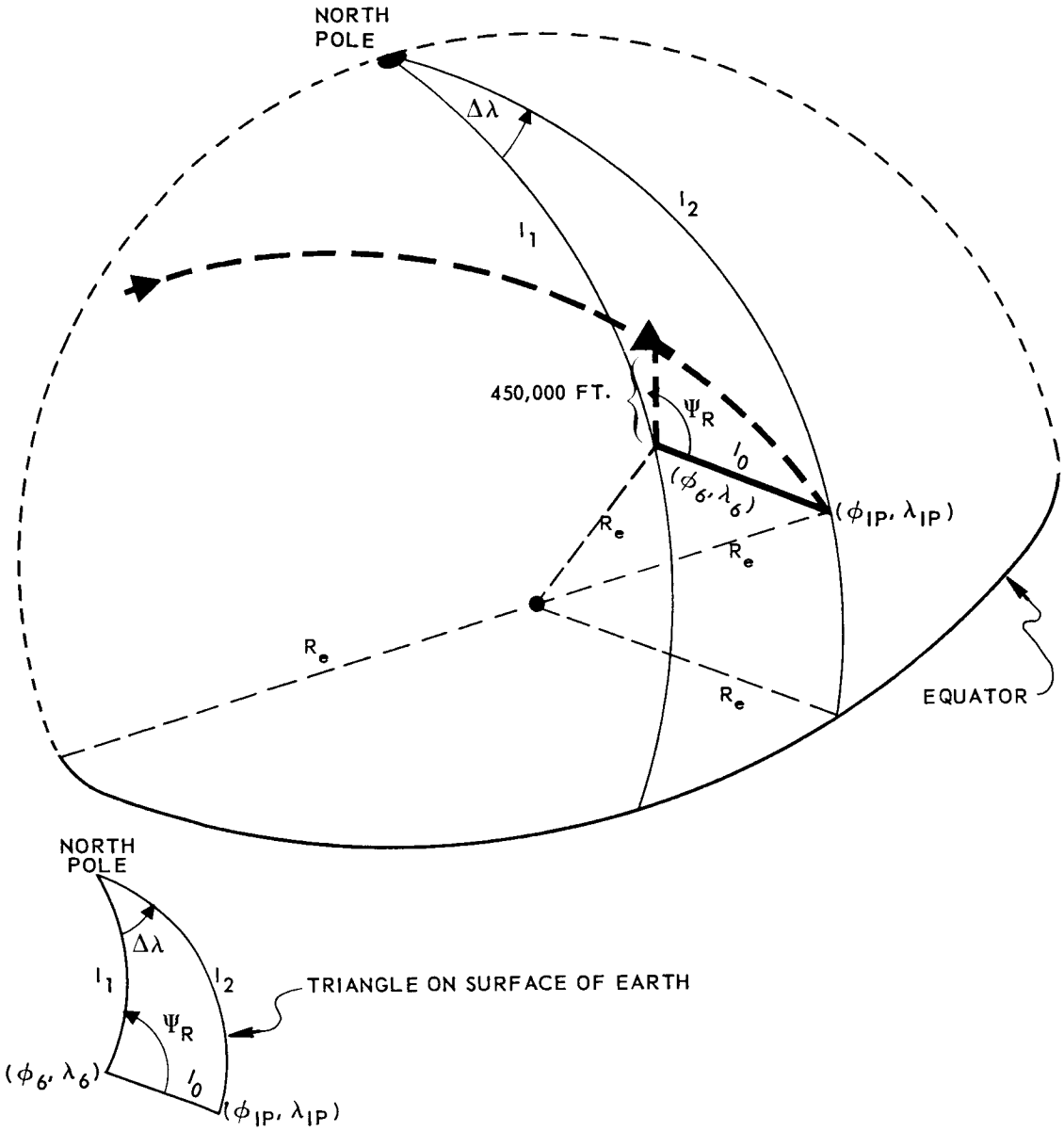


FIGURE 3-35 EMFIT DIAGRAM

3.10 COMPUTATION OF ELLIPTIC ORBIT PROGRAM (C9RVTH)

The purpose of C9RVTH is to predict the position and velocity of the Mercury capsule assuming that elliptic motion with drag and oblateness perturbations are absent. Given the position and velocity at some time t, C9RVTH computes position and velocity at time $t + \Delta t$ or at a specified height above the earth. The flow chart for C9RVTH is shown in Figure 3-36.

3.10.1 Input and Output Requirements

a) The following library subroutines are required:

U1SQRT - Square root

U1ATAB - Are tangent A/B

U1ATNA - Are tangent A

U3DOTP - Dot product of the vector

U3VMAG - Magnitude of the vector

b) Mercury subroutine C9ASKE is required.

The type of input and output generated depends upon which of the options in Table 3-1 are selected. All values are Mercury units.

Options 6-10 assume that prior entry was made at C9RVTH and that the initial $\bar{r}$ and $\bar{v}$ and the corresponding orbit elements are available in core.

a) C9RVTH may be entered in either of two places, C9RVTH or C9RVT2, depending on whether or not new orbit elements must be computed.

b) If the motion is nearly circular, i.e., eccentricity $e < e_{critical}$, the calculation of the new position and velocity is not made. A special return is provided to indicate $e < e_{critical}$.

c) C9RVTH checks to determine if the specified height h_s can be attained, i.e., it compares h_s with the computed apogee height. A special return is provided to indicate that h_s is greater than the apogee height.

d) C9RVTH assumes that the computed Δt is positive.

3.10.2 Method

The C9RVTH uses the following equations:

TABLE 3-5. C9RVTH INPUT AND OUTPUT DATA FORMAT

OPTION	AC	MQ	INPUT	OUTPUT
1. Transfer to C9RVTH	plus	minus	$\bar{r}, \bar{v}, \Delta t$	$\bar{r}'$ and $\bar{v}$
2. Transfer to C9RVTH	zero	minus	$\bar{r}, \bar{v}, h_s, R_s$	$\bar{r}', \bar{v}', \Delta t$
3. Transfer to C9RVTH	minus	minus	$\bar{r}, \bar{v}, h_s, R_s$	$\bar{r}', \bar{v}', \gamma, \Delta t$
4. Transfer to C9RVTH	plus	plus	$\bar{r}, \bar{v}, t$	$\bar{r}'$
5, Transfer to C9RVTH	minus or zero	plus	$\bar{r}, \bar{v}, h_s, R_s$	$\bar{r}'$, and Δt
6. Transfer to C9RVT2	plus	minus	Δt	$\bar{r}'$, and $\bar{v}'$
7. Transfer to C9RVT2	zero	minus	h_s, R_s	$\bar{r}', \bar{v}', \Delta t$
8. Transfer to C9RVT2	minus	minus	h_s, R_s	$\bar{r}', \bar{v}', \gamma, \Delta t$
9. Transfer to C9RVT2	plus	plus	Δt	$\bar{r}'$
10. Transfer to C9RVT2	minus or zero	plus	h_s, R_s	$\bar{r}'$ and Δt

$$r = |\bar{r}| \qquad v = |\bar{v}|$$

$$e \cos E = rv^2 - 1$$

$$a = \frac{r}{1 - e \cos E}$$

$$n = \frac{1}{\sqrt{a^3}}$$

$$e \sin E = \frac{\bar{r} \cdot \bar{v}}{\sqrt{a}}$$

$$e = \sqrt{(e \sin E)^2 + (e \cos E)^2}$$

If $e < e_{critical}$, exit from the routine.

$$E = \arctan\left(\frac{e \sin E}{e \cos E}\right), \qquad -\pi < E \leq \pi$$

$$M = E - e \sin E$$

a) <u>If H_s is given, compute</u>

$$r' = R_s + h_s$$

If $r' > a(1 + e)$, exit from the routine

$$e \cos E' = \frac{a - r'}{a}$$

$$e \sin E' = -\sqrt{e^2 - (e \cos E')^2}$$

$$M' = E' - e \sin E'$$

$$\Delta t = \frac{M' - M}{n}$$

b) <u>If Δt is given, compute</u>

$$M' = M + n \Delta t$$

Solve Kepler's equation $M' = E' - e \sin E'$ for E'

$$f = \frac{-e \cos E + \cos E' \cos E + \sin E' \sin E}{1 - e \cos E}$$

$$g = \left[(\cos E' - e) \sin E' - (\cos E' - e) \sin E \right] \frac{1}{n}$$

$$\bar{r}' = f\,\bar{r} + g\,\bar{v}$$

Either

$$\mu' = a\,(1 - e \cos E')$$

$$v' = \sqrt{\frac{2}{r'} - \frac{1}{a}}$$

$$\tan \gamma' = \frac{e \sin E'}{\sqrt{1 - e^2}}$$

Or

$$f' = \left[\frac{\sin E \cos E' - \sin E' \cos E}{(1 - e \cos E')\,(1 - e \cos E)} \right] n$$

$$g' = \left[\frac{-e \cos E' + \cos E' \cos E + \sin E' \sin E}{1 - e \cos E'} \right]$$

$$\bar{v}' = f'\bar{r} + g'\,\bar{v}$$

$$\gamma' = \arctan\,(\tan \gamma')$$

3.10.3 Usage

a) Calling Sequence

TSX C9RVTH, 4 (orbit elements have not been calculated)
or
TSX C9RVT2, 4 (orbit elements have been calculated)

+1 PZE L ($\bar{r}$), 0, L ($\bar{v}$)

+2 PZE A, 0, B

+3 PZE L ($\bar{r}'$), 0, C

+4 PZE Circular orbit - Exit 1

+5 PZE Cannot reach h_s - Exit 2

+6 Normal Return

where the symbols in the calling sequence have the following meanings:

MQ	AC	A	B	C
minus	plus	L (Δt)	O	L ($\bar{v}'$)
minus	zero	L (R_s)	L (h_s)	L ($\bar{v}'$)
minus	minus	L (R_s)	L (h_s)	L ($\bar{v}'$)
plus	(zero or minus)	L (R_s)	L (h_s)	O
plus	plus	L (Δt)	O	O

and L $(\bar{r})$ is the first location of the block containing the components of the input position vector in Mercury units.

L $(\bar{v})$ is the first location of the block containing the components of the input velocity vector in Mercury units.

L (Δt) is the location of the input quantity Δt in Mercury units.

L (R_s) is the location of the input quantity R_s, radius of the earth in Mercury units.

L $(\bar{r}')$ is the first location of the block containing the components of computed position vector.

L $(\bar{v}')$ is the first location of the block containing the components of the computed velocity vector.

L $(\bar{v}')$ is the first of two locations and contains the magnitude of the computed velocity; the second word contains the computer flight path angle.

Upon exit, C (AC) Δt in Mercury units if time was computed.

b) Space required

369_{10} locations excluding constants and subroutines.

c) Error Codes:

C9RVTH has two error returns:

1. A return to $\alpha + 4$ indicates a circular orbit.
2. A return to $\alpha + 5$ indicates the orbit will not reach h_s.

d) Constants Used

KECR.T	DEC	.003	test for circular orbit
K001.0	DEC	1.0	
K00001	DEC	1	
K002.0	DEC	2.0	
K00003	DEC	3	
KKK2PI	DEC	6.283185307	two times Pi

d) Timing

Maximum time including subroutines is 35.888 msecs.

3.10.4 Checkout

C9RVTH was tested by using the following input:

$X = 1.02900366 \qquad \dot{X} = .171182375$

$Y = 0.0 \qquad \dot{Y} = .52874836$

$Z = 0.0 \qquad \dot{Z} = .81420107$

$h_s = .021504685, \Delta t = .074366916, \; R_s = 1.0$

This input was put in the corresponding format and run with the Vanguard programs EP 2 and FMV1. Results compared to 7 places.

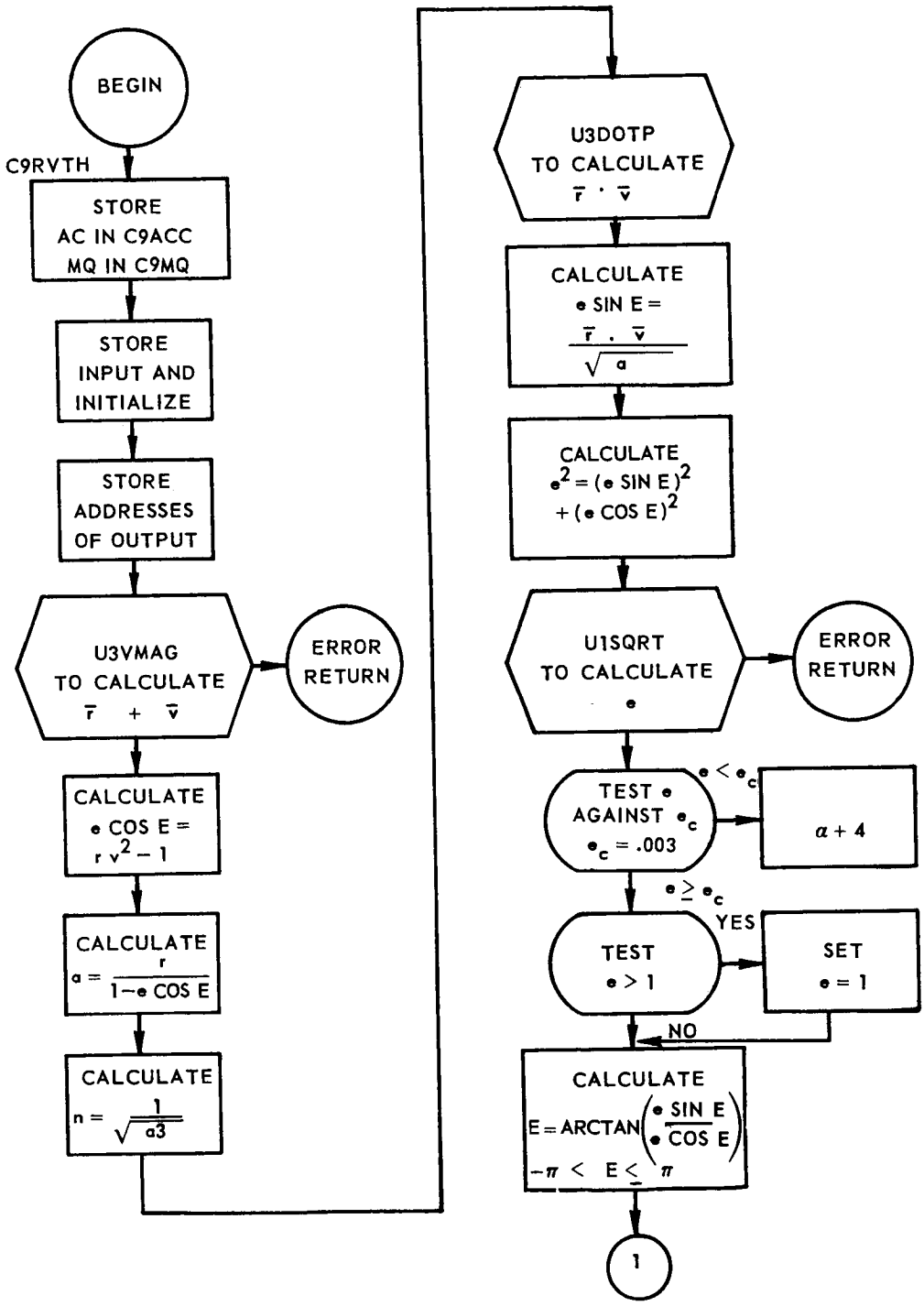


FIGURE 3-36. C9RVTH PROGRAM FLOW CHART *(Sheet 1 of 4)*

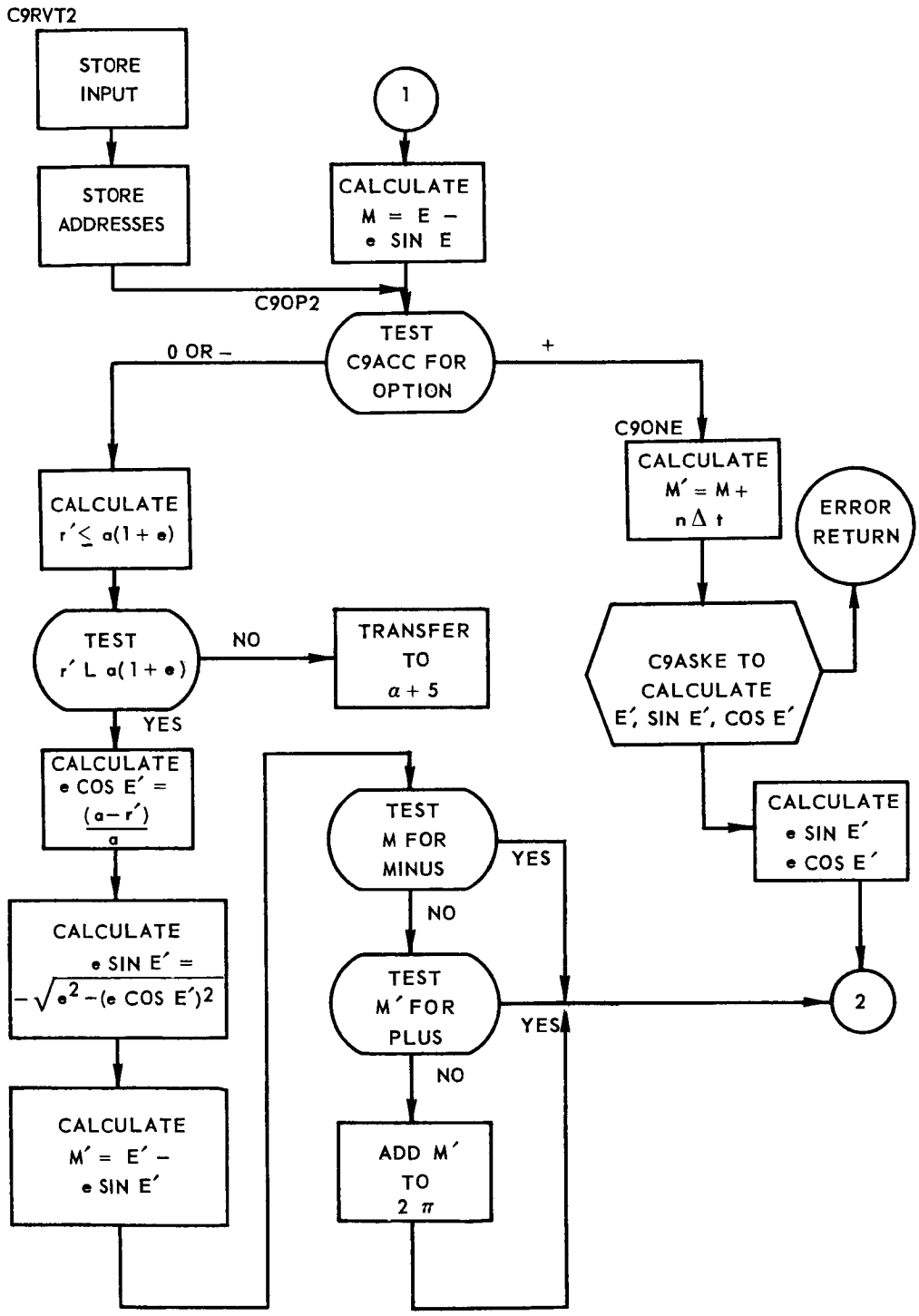


FIGURE 3-36. C9RVTH PROGRAM FLOW CHART *(Sheet 2 of 4)*

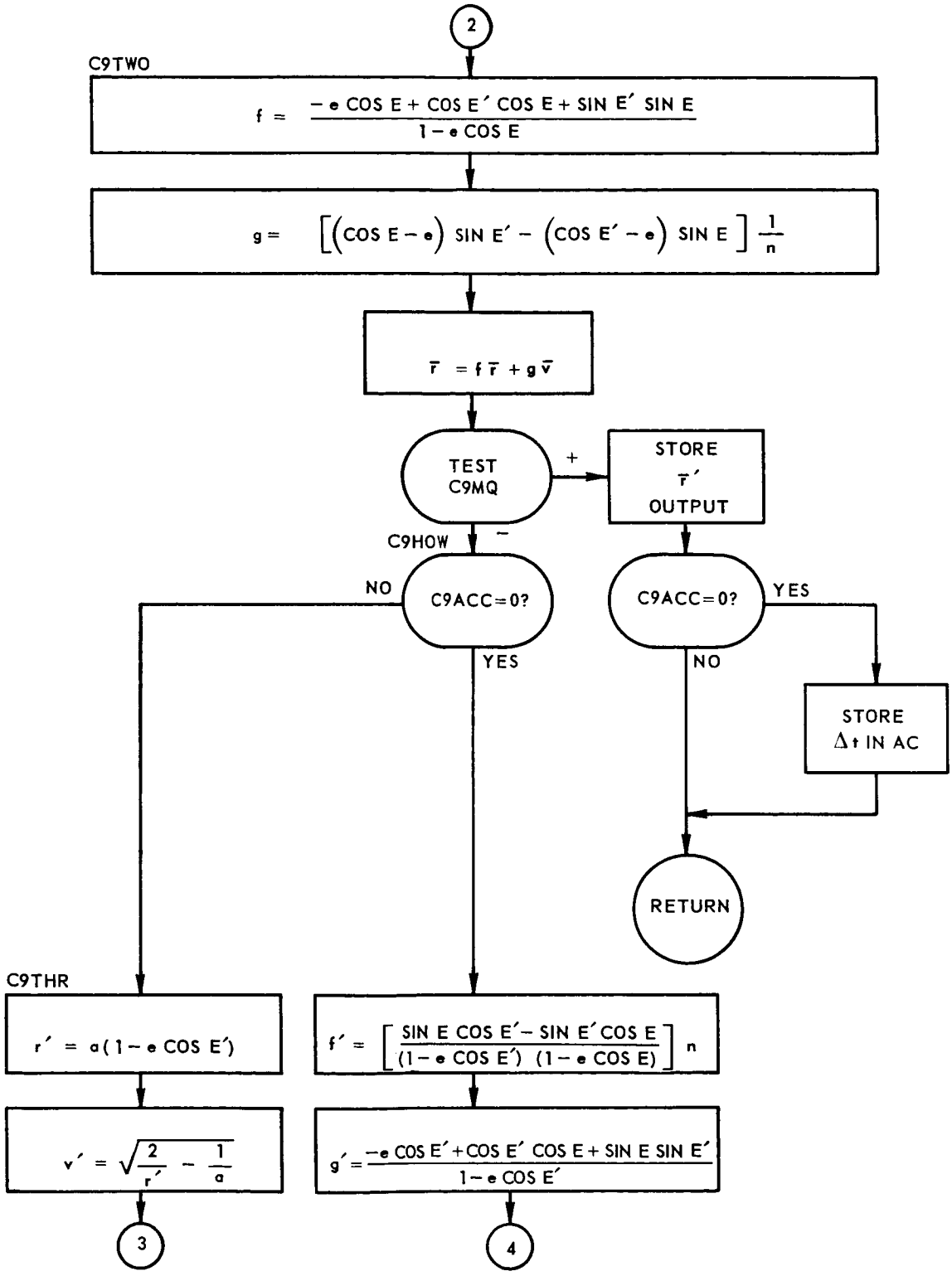


FIGURE 3-36. C9RVTH PROGRAM FLOW CHART *(Sheet 3 of 4)*

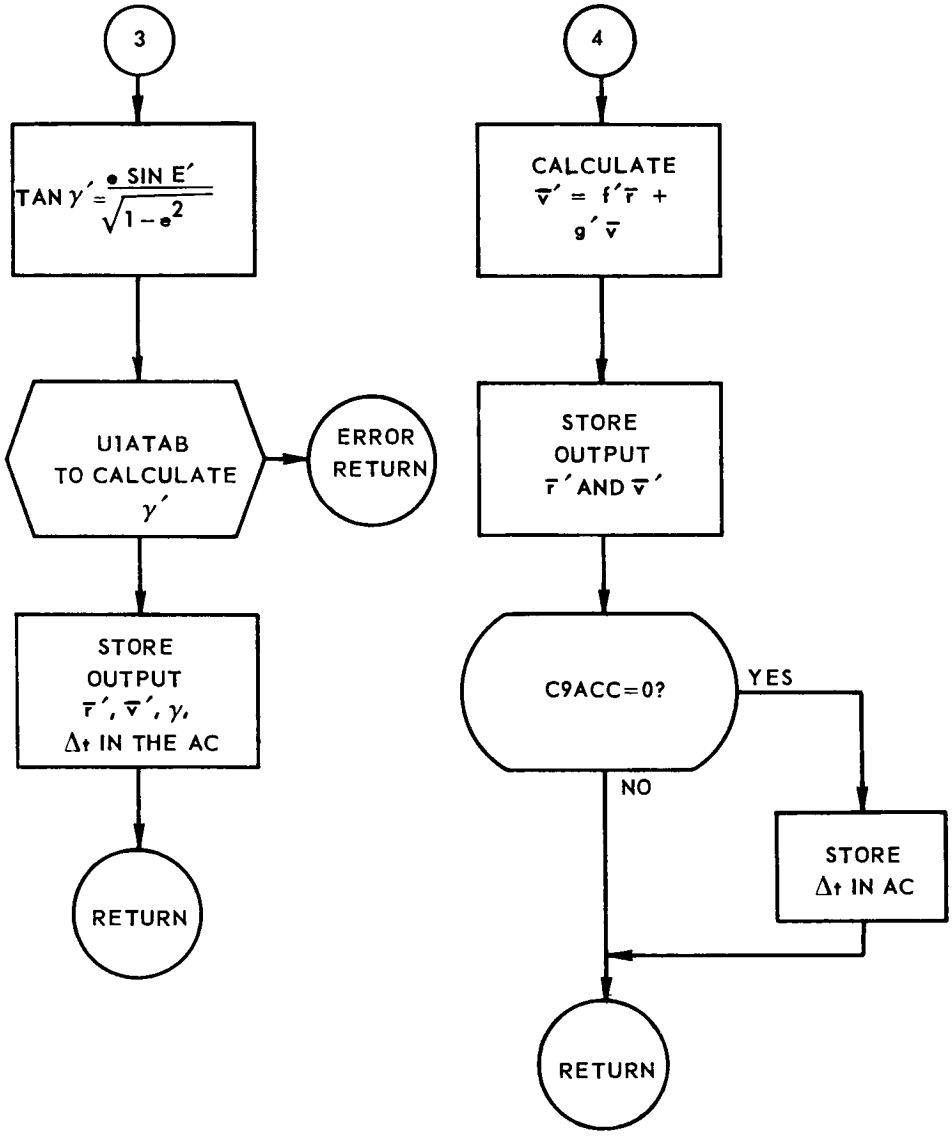


FIGURE 3-36. C9RVTH PROGRAM FLOW CHART *(Sheet 4 of 4)*

3.11 COMPUTATION OF CIRCULAR ORBIT PROGRAM (CHNOND)

This program computes new R and V vectors if Δt is given and Δt and the R and V vectors if h is given. The flow chart for the CHNOND program is shown in Figure 3-37.

3.11.1 Input Requirements

The following library subroutines are used with CHNOND:

U1SICO - sine cosine

U1SQRT - square root

U1ATAB - are tangent A/B

In the input block (in this order)

t - Time in floating point in seconds.

R - 3 locations for the R vector.

V - 3 locations for the V vector.

In the output block (first location)

$t + \Delta t$ used if a new R and V is desired at Δt.

(floating point) - not needed if R and h is given.

Other values:

R - radius of the earth

h - height above the earth

These must be placed in the accumulator if needed.

not needed if Δt is given

3.11.2 Output Requirements

The information serves as output of the CHNOND program:

$t + \Delta t$ - time (in floating point) that the capsule will be at h. (if h is given)

R - R vector at Δt+(3 locations)

V - V vector at Δt+(3 locations)

3.11.3 Method

The following equations are used by CHNOND in computing R and V components:

$$r_0 = \sqrt{X^2 + Y^2 + Z^2} \tag{1}$$

$$v_0^2 = v_X^2 + v_Y^2 + v_Z^2 \tag{2}$$

$$a_0 = \frac{r_0}{2 - r_0 v_0^2} \tag{3}$$

$$n_0 = a_0^{3/2} \tag{4}$$

$$\overline{h}_0 = \overline{r}_0 \times \overline{v}_0 \tag{5}$$

$$h_0 = \sqrt{\overline{h}_0 \cdot \overline{h}_0} \tag{6}$$

$$\overline{r} = \frac{\overline{h}_0}{h_0} \tag{7}$$

$$\cos\Omega = \frac{-h_Y}{\sqrt{h_X^2 + h_Y^2}} \tag{8}$$

$$\sin\Omega = \frac{h_X}{\sqrt{h_X^2 + h_Y^2}} \tag{9}$$

$$\overline{N} = (\cos\Omega, \ \sin\Omega, \ 0) \tag{10}$$

$$\overline{M} = \overline{r} \times \overline{N} \tag{11}$$

$$C_1 = r_0 v_0^2 - 1 = 1 - \frac{r_0}{a_0} \tag{12}$$

$$C_2 = \frac{r_0 \cdot v_0}{\sqrt{a_0}} \tag{13}$$

$$D = 1 + \sqrt{1 - C_1^2 - C_2^2} \tag{14}$$

$$A_1 = \sqrt{a_0}\,(\bar{v}_0 \cdot \bar{N}) \tag{15}$$

$$A_2 = \sqrt{a_0}\,(\bar{v}_0 \cdot \bar{M}) \tag{16}$$

$$B_1 = \frac{\bar{r}_0 \cdot \bar{N}}{r_0} \tag{17}$$

$$B_2 = \frac{r_0 \cdot \bar{M}}{r_0} \tag{18}$$

$$\xi = C_1 B_1 - C_2 A_1 \tag{19}$$

$$\eta = C_1 B_2 - C_2 A_2 \tag{20}$$

$$\sin\phi_0 = -A_1 (1 - C_1) + \frac{C}{D}\eta \tag{21}$$

$$\cos\phi_0 = A_2 (1 - C_1) + \frac{C}{D}\xi \tag{22}$$

$$\phi_0 = \tan^{-1}\left(\frac{\sin\phi_0}{\cos\phi_0}\right) \tag{23}$$

$$\mu_0 = \phi_0 - C_2 \tag{24}$$

$$r = h + R \tag{25}$$

$$C_1 = 1 - \frac{r}{a_o} \tag{26}$$

$$C_2 = \sqrt{\xi^2 + \eta^2 - C_1^2} \tag{27}$$

$$\sin_\phi = \frac{C_2\xi + C_1\eta}{\xi^2 + \eta^2} \tag{28}$$

$$\cos\phi = \frac{C_1\xi - C_2\eta}{\xi^2 + \eta^2} \tag{29}$$

New Calculations for those values (h given)

$$\mu = \phi - C_2 \tag{30}$$

$$\Delta t = \frac{\mu - \mu_o}{n_0} \tag{31}$$

$$t = t_o + \Delta t \tag{32}$$

$$\mu = \mu_o + n_o \Delta t \tag{33}$$

$$\phi_i = \mu \tag{34}$$

$$C_1 = \xi \cos \phi_i + \eta \sin \phi_i \tag{35}$$

$$C_2 = \xi \sin \phi_i - \eta \cos \phi_i \tag{36}$$

$$\phi_{i+1} = \phi_i + \frac{\mu + C_2 - \phi_i}{1 - C_1} \tag{37}$$

$$r \cos \mu = a_o \left(\cos \phi - \xi + \frac{C_2}{D} \eta\right) \tag{38}$$

$$r \sin \mu = a_o \left(\sin \phi - \eta - \frac{C_2}{D} \xi\right) \tag{39}$$

$$\overline{r} = (r \cos \ ) \ \overline{N} + (r \sin \mu \ ) \ \overline{M} \tag{40}$$

$$F_\mu = \frac{- \sin \phi + \frac{C_1}{D} \eta}{a^{1/2} \quad (1 - C_1)} \tag{41}$$

$$G_\mu = \frac{\cos \phi - \frac{C_1}{D} \xi}{a^{1/2} \ (I - C_1)} \tag{42}$$

$$\overline{V} = (\overline{N}\ F_{\mu} + \overline{M}\ G_{\mu}) \qquad (43)$$

3.11.4 Usage

a) Calling Sequence

In AC; 1) if R and V at Δt is desired, the accumulator = 0

2) if R and V and Δt are desired at height h the

AC = R, 0, h where R = L (R radius of earth),

0 = No tag, and

h = L (h, height above earth)

α	TSX	CHNOND, 4
$\alpha + 1$	PZE	I, 0, O
$\alpha + 2$	Error Return	
$\alpha + 3$	Normal Return	

where I = L (Input block)

O = L (Output block)

b) Space Required

Temporary	59
Constants	5
Subroutines	263
Instructions	421
Total	748

c) Error Codes

If the decrement of CHNOND = 1, the orbit is hyperbolic

If the decrement of CHNOND = 2, no intersection at 450,000 feet.

d) Constants Used

K001.0	DEC	1.0
K002.0	DEC	2.0
K00.UT	DEC	806.8104
K00.SH	DEC	.12394486E-2
KKK2P1	DEC	6.283185307

e) Temporary Storage

CTNOND	BSS	53 (In program)

f) Timing

To find R and V at Δt

7.8 + n (6.5) msec if old elements are used

34.6 + n (6.5) msec if new elements are calculated

where n is the number of iterations necessary for 0 to converge.

(n will be 2 in most cases)

To find Δt, R and V at h

13.8 msec if old elements are used

40.6 msec if new elements are calculated

3.11.5 Checkout

The CHNOND program has been checked by comparing answers with a numerical integration. Results were satisfactory.

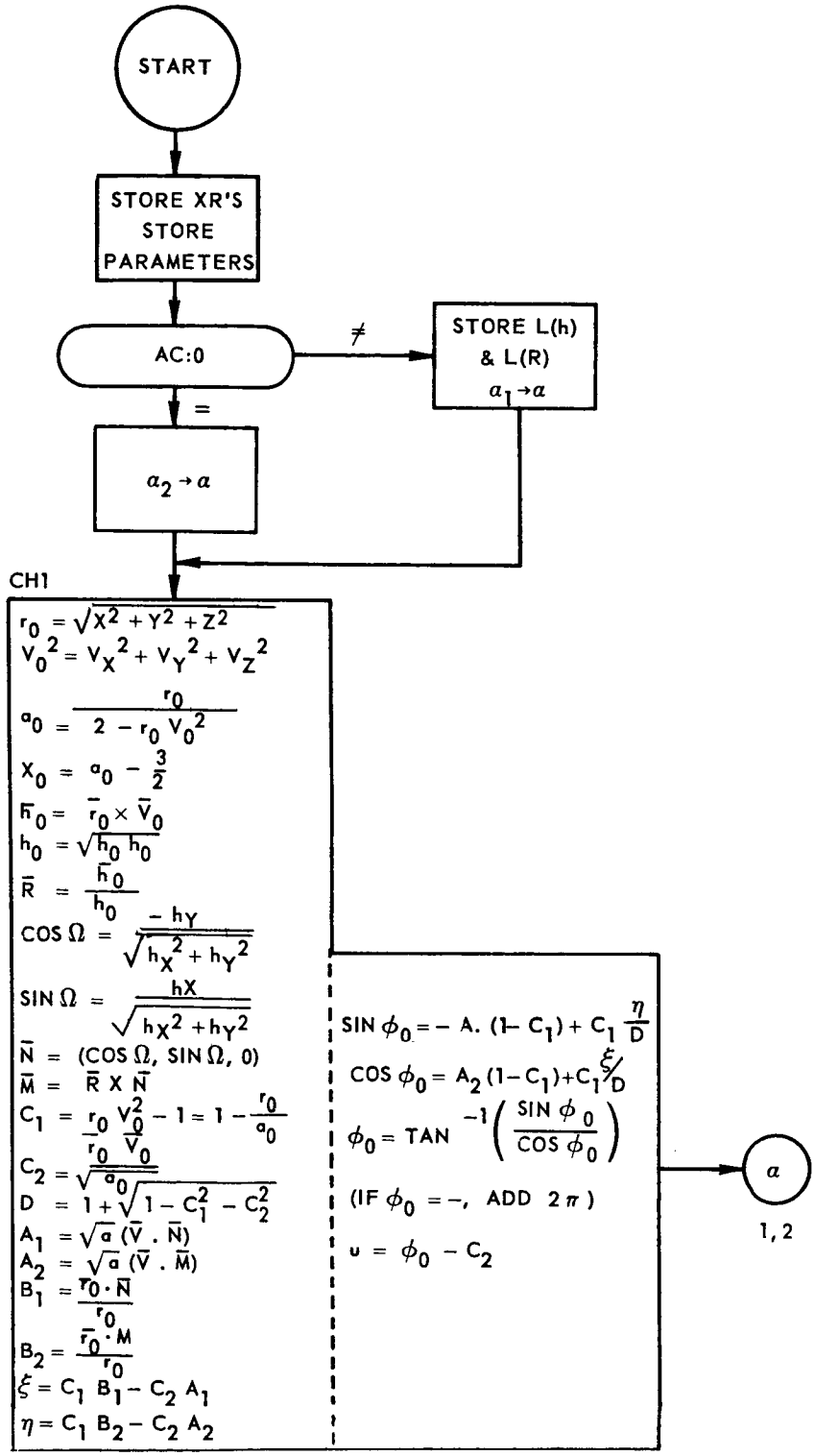
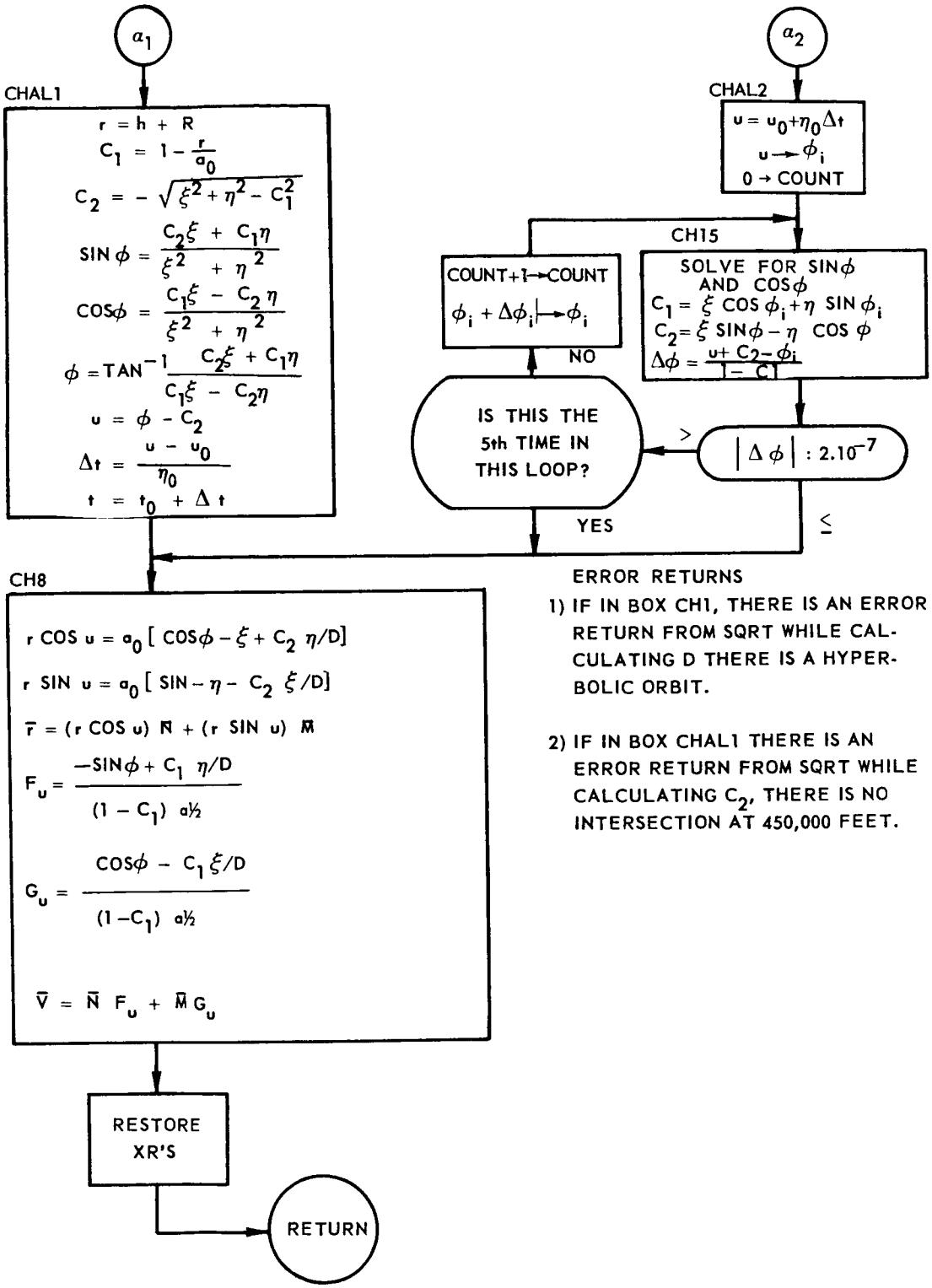


FIGURE 3-37. CHNOND PROGRAM FLOW CHART *(Sheet 1 of 2)*



ERROR RETURNS

1) IF IN BOX CH1, THERE IS AN ERROR RETURN FROM SQRT WHILE CALCULATING D THERE IS A HYPERBOLIC ORBIT.

2) IF IN BOX CHAL1 THERE IS AN ERROR RETURN FROM SQRT WHILE CALCULATING C_2, THERE IS NO INTERSECTION AT 450,000 FEET.

FIGURE 3-37. CHNOND PROGRAM FLOW CHART *(Sheet 2 of 2)*

3.12 RETROFIRE TIME CORRECTION PROGRAM (C9DTRF)

The purpose of the C9DTRF program is to compute the correction to the estimated time for firing retrorockets to land in a designated recovery area. The flow chart for the C9DTRF program is shown in Figure 3-38.

3.12.1 Input Requirements

Entry to C9DTRF is made with the longitude of the mid-point of the recovery area in floating point radians ($0 \leq \lambda \leq 2\pi$) in the AC and the computed longitude of the impact point in the MQ.

3.12.2 Output Requirements

Upon exiting from C9DTRF, the correction to the time of retrofiring in Mercury time units is in the accumulator. If the computed longitude is within range, the time correction is zero.

3.12.3 Additional Requirements

When first entering C9DTRF location T1STNO must be zero, indicating a first iteration. For the next iterations until the computed longitude has converged to the desired longitude (within the limits defined by the tolerance in longitude), C9DTRF sets location T1STNO unequal to zero. After convergence, i.e., when the time correction is computed to be zero, location T1STNO is reset to zero within C9DTRF. If any limitation is put on the number of iterations and convergence is not obtained, location T1STNO must be set equal to zero outside C9DTRF.

3.12.4 Method

Equations Used

a) To take care of the discontinuity in longitude, the minimum distance, $\Delta\lambda_{1\ min.}$ $\left(\left|\Delta_{1\ min}\right| \leq 180^{\circ}\right)$ from $\lambda_{computed}$ to λ desired is computed. See Sheet 2 of Figure 3-38, for details of this calculation.

b) $\Delta t = \Delta\lambda_{min} \Big/ \left(\omega_c - \omega_e\right)$

ω_c = component of velocity eastward

ω_e = rotational velocity of the earth

c) $\omega_c = (X\dot{Y} - Y\dot{X}) \Big/ (X^2 + Y^2)$

d) new time correction $\Delta t = (\Delta t_{previous} \cdot \Delta\lambda_{min}) \Big/ \Delta\lambda_p$

For details on calculation of $\Delta\lambda_p$, the actual change in longitude achieved by changing the firing time by the amount Δt_p, see sheet 2 of Figure 3-38.

Accuracy

7 significant digits

3.12.5 Usage

a) Calling Sequence

α	TSX	C9DTRF, 4
$\alpha + 1$	PZE	Loc. (1st component of $\overline{R}$)
$\alpha + 2$	PZE	Loc. (1st component of $\overline{V}$)
$\alpha + 3$		Tolerance in longitude
$\alpha + 4$	Normal return	

Units: longitude in floating point radians; $\overline{R}$ and $\overline{V}$ vectors in units.

It is assumed that the components of each vector are in consecutive locations. Tolerance in longitude is 5 nautical miles. The same value in radians varies with the latitude.

b) Space Required

93_{10} locations excluding constants.

c) Constants Used

K00.PI	DEC	3.14159265
KKK2P1	DEC	6.283185307
K.OMEG	DEC	.058833543
K00060	DEC	60
K00001	DEC	1

d) Tables Used

T1STNO	PZE

e) Erasable Locations

12_{10}.

ENS(2), RVEC (4), HGDT, LMDA2, LCOM, LDES, DLMIN, DLPRE.

c) Timing

.504 msec if value is within range

2.772 msec for an initial refinement

1.524 msec for any other refinement

3.12.6 Checkout

First, C9DTRF was checked out by itself testing every variation in the program. The numerical calculations proper to C9DTRF were compared with hand calculations. Then C9DTRF was checked out together with R5RARF to test how quickly its equations refined the time to fire sufficiently. For this phase of the testing, Monitor controls for the R5RARF program were simulated and the output of N0PCNI was used as input for C9DTRF.

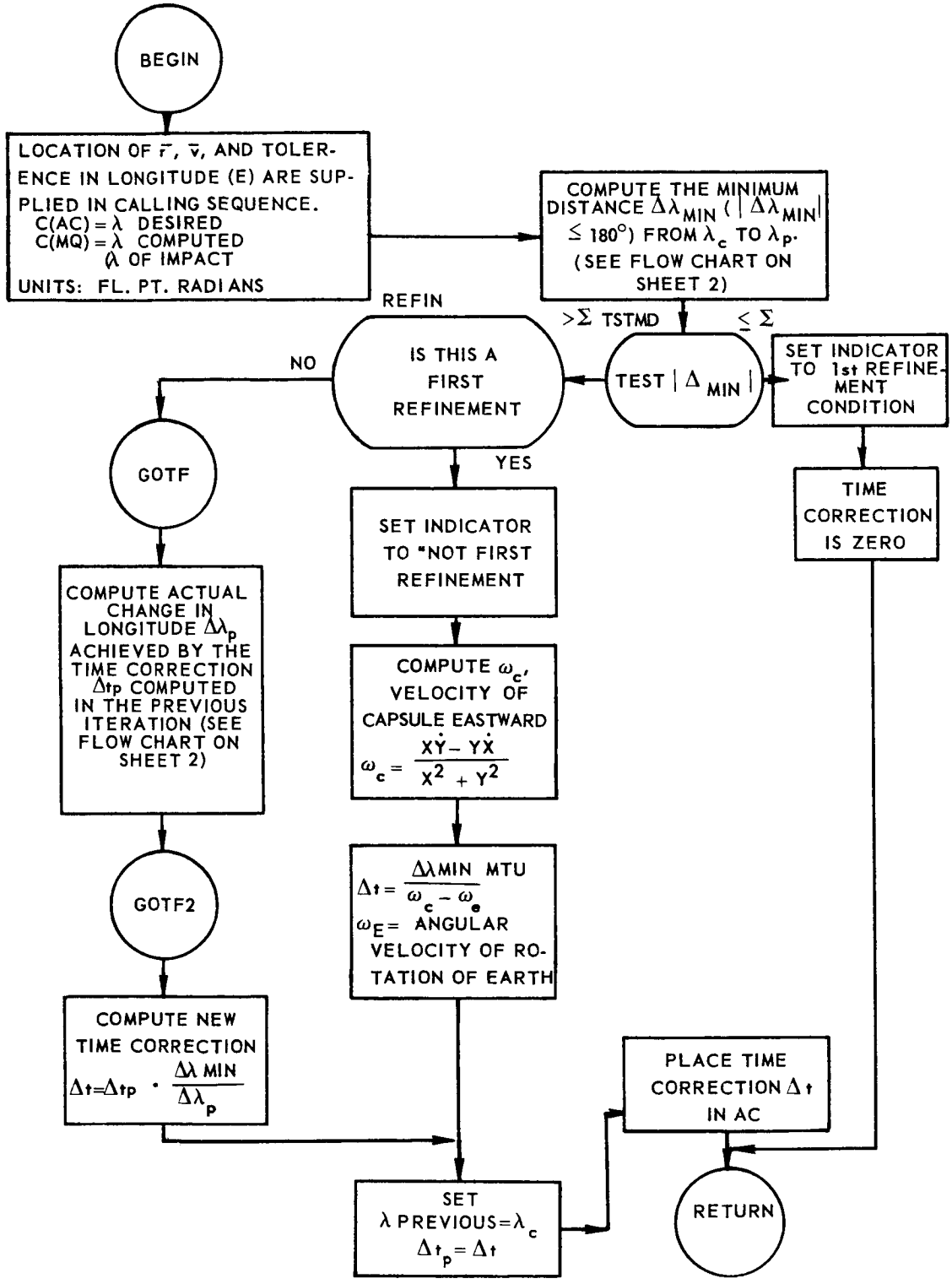
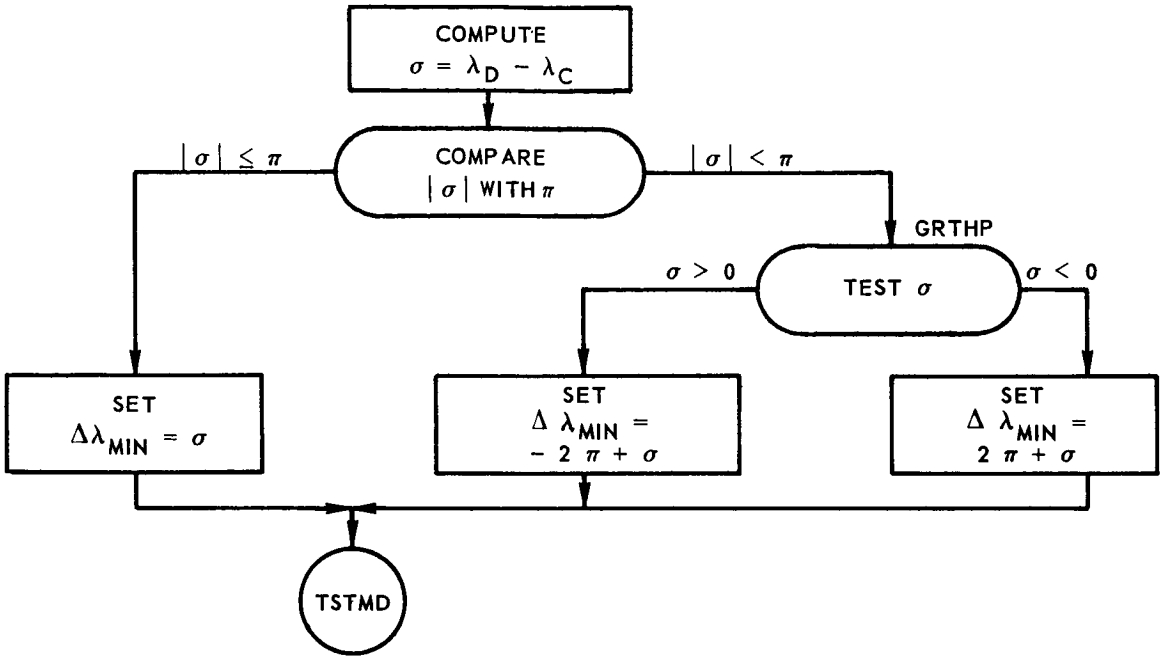


FIGURE 3-38. C9DTRF PROGRAM FLOW CHART *(Sheet 1 of 2)*

GIVEN λ_c, COMPUTED LANDING POINT LONGITUDE
λ_D, DESIRED LANDING POINT LONGITUDE
TO COMPUTE THE MINIMUM DISTANCE $\Delta\lambda_{MIN}$,
(POSITIVE OR NEGATIVE) FROM λ_C TO λ_D.



GIVEN λ PREVIOUS, LONGITUDE OF PREVIOUS LANDING POINT
λ_C, LONGITUDE OF PRESENT COMPUTED LANDING POINT.
Δt_p, TIME CORRECTION COMPUTED IN THE PREVIOUS ITERATION
TO COMPUTE THE ACTUAL CHANGE IN LONGITUDE, $\Delta\lambda_p$
ACHIEVED BY CHANGING THE FIRING TIME BY THE AMOUNT Δt_p.

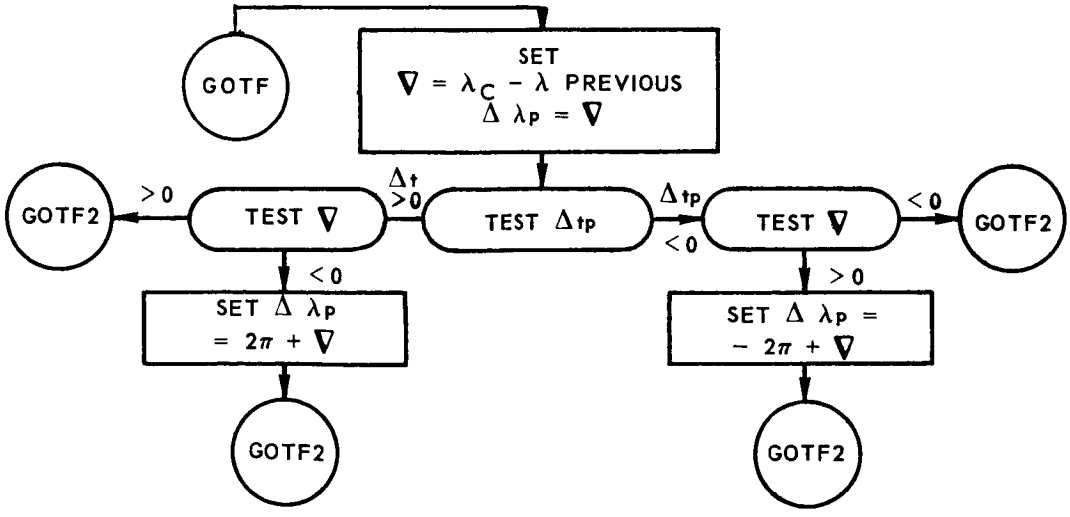


FIGURE 3-38. C9DTRF PROGRAM FLOW CHART *(Sheet 2 of 2)*

3.13 POSITION AND VELOCITY AT END OF RETROFIRE PROGRAM (R6BOTH)

Given position, velocity, and the time associated with these quantities at the start of retrofire, R6BOTH produces the position, velocity, and the associated time at the termination of retrorocket fire. The flow chart for the R6BOTH program is shown in Figure 3-39.

3.13.1 Input Requirements

The following library subroutines are required:

a) U3UNTV - Generate unit vector from vector

b) U3VPRO - Vector "triple cross" product

c) U1SQRT - Square root

The input to R6BOTH consists of seven words in consecutive ascending core locations; time (associated with given $\bar{r}$, $\bar{v}$) in seconds, floating point, $\bar{r}$ components (X, Y, Z), $\bar{v}$ components (v_X, v_Y, v_Z) all in Mercury units, floating point.

The number of retrorockets fired, N , must be given as an integer in the accumulator when entering the program.

a) 34° orientation is assumed for the thrust vector. 1080 lbs. (at surface) is the assumed (constant) thrust of rockets during retrofire. Vehicle weight is taken to be 2711 lbs. in orbit, 2635 lbs. in re-entry.

b) Retrorocket firing intervals are 5, 5,5 seconds. The duration of each burst is 12.0 seconds.

c) Retrorocket engines are assumed canted from center of gravity by 13.5^o symmetrically.

3.13.2 Output Requirements

Identical to the input in requirements and format.

3.13.3 Method

The equations in question are:

$$\bar{r} = \bar{r}_o + \bar{v}_o \ (\Delta t) - (\Delta t)^2 \ \bar{r}_o \ / \ 2 \ r_o^{\ 3} + (\Delta t)^2 \ \bar{A}/2$$

$$\bar{v} = \bar{v}_o - (\Delta t) \ \bar{r}_o / r_o^{\ 3} + (\Delta t) \ \bar{A}$$

This program has been tested using 4, 3, 2, 1, 0 rockets; 5 iterations of a single retro, with adjusted thrust and timing; and a single pass using average thrust.

3.13.4 Usage

Place integral N in the AC,

α	TSX	R6BOTH, 4
$\alpha + 1$	PZE	L(I), 0, L(O)
$\alpha + 2$	Error Return	
$\alpha + 3$	Normal Return	

where

N = number of retrorockets fired

L(I)= Location of first entry in input block

L(O) = Location of first entry in output block.

a) Locations

Program uses 200 locations of which the last 33 are erasable.

b) Error Return

Error return is made if and only if N $\neq$ 0, 1, 2, 3

c) Constants Used

K00001	DEC	1
K00003	DEC	3
K002.0	DEC	2.0

TTHRUS - 9 location table of thrust factors.

R6TMP is a block of 33 locations used for temporary storage.

d) Timing

For N = 1, 2, 3 rockets fired

Time = 25.2 + 2 . (N-I).(24.352) msec (709)

(i.e. 122.6 msec for N = 3)

3.13.5 Checkout

Program has been checked inter se and by hand computation for N = 3.

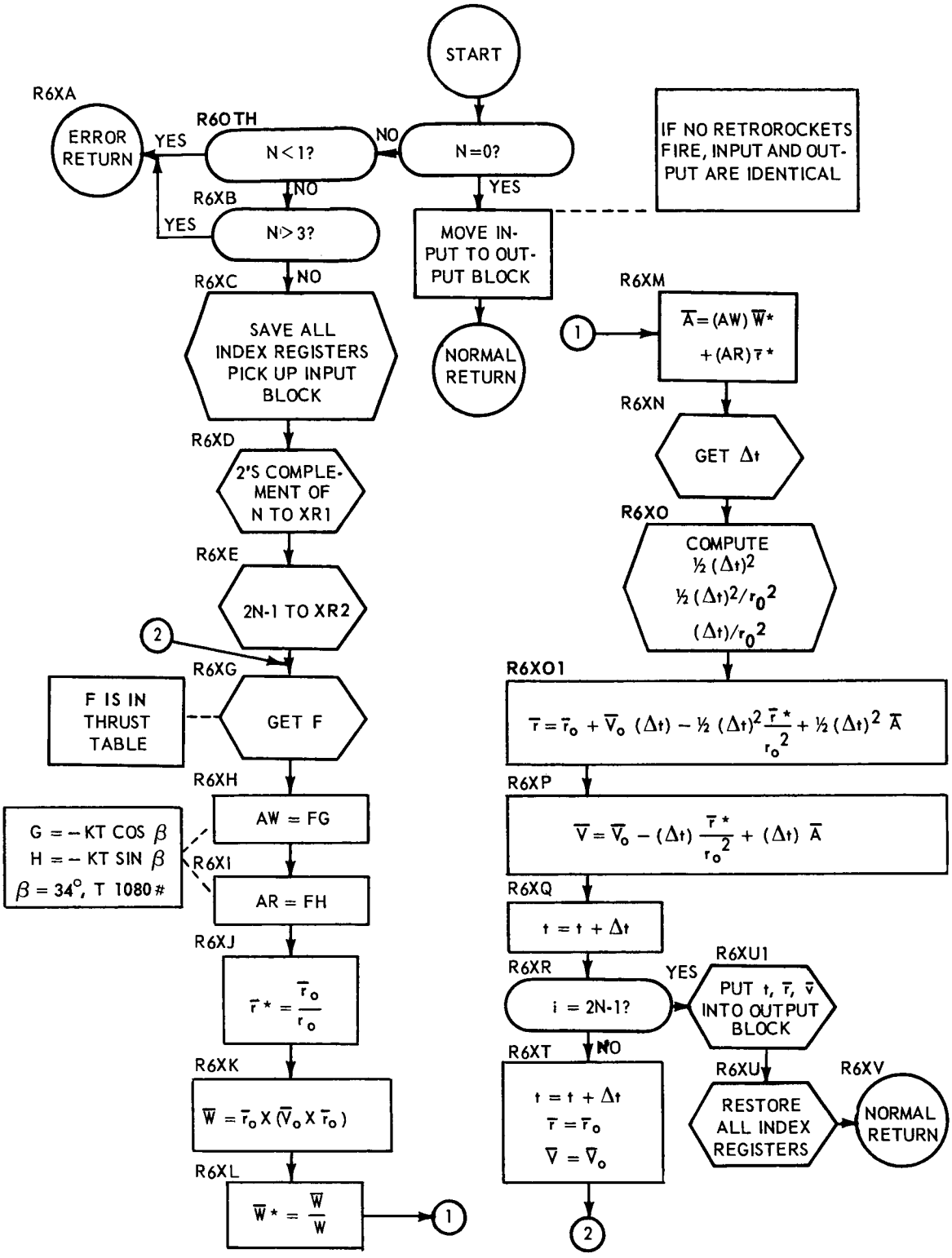


FIGURE 3-39. R6BOTH PROGRAM FLOW CHART

3.14 COMPUTE LATITUDE AND LONGITUDE PROGRAM (A3MSCP)

Given an inertial position vector R (X, Y, Z) of the capsule at a particular time, compute the geodetic latitude (ϕ) and longitude (λ) of the sub capsule position and the altitude (h) of the capsule at that time.

The flow chart for the A3MSCP program is shown in Figure 3-40.

3.14.1 Input Requirements

This program utilizes the square root library subroutine U1SQRT and the arc tangent routine U1ATAB.

The input to A3MSCP is a table of at least four locations containing the time in the first location and the X, Y, and Z components of the given R vector in the 2nd, 3rd, and 4th locations respectively. The time may be in either floating point seconds or in fixed point with seconds in the decrement and minutes in the address as explained in 3.14.4, Usage. Components X, Y, and Z are in floating point Mercury units.

3.14.2 Output Requirements

The output of A3MSCP is a three location table giving λ, ϕ, and h in that order in floating point Mercury units. λ is in the range from 0 to 2π radians. ϕ is from $-\pi/2 + \pi/2$. (The computation of h may be omitted if desired; see 3.14.4.)

3.14.3 Method

The following equations are used by the A3MSCP program:

$$\tan\phi = \frac{e^2 N \sin\phi + Z}{\sqrt{X^2 + Y^2}} \tag{1}$$

$$h = \sec\phi\sqrt{X^2 + Y^2} - N \tag{2}$$

$$\lambda = \tan^{-1}\frac{Y}{X} - \omega t - \lambda g \tag{3}$$

$$N = \frac{a}{\sqrt{1 - e^2 \sin^2\phi}} = \frac{1}{\sqrt{1 - e^2 \sin^2\phi}} \tag{4}$$

$$\tan\phi'' = \frac{b^2}{a^2}\tan\phi = b^2 \tan\phi \tag{5}$$

$$\tan \phi''' = \frac{a^2}{b^2} \tan \phi' = \frac{1}{b^2} \tan \phi' \qquad (6)$$

The method used for determining ϕ and h was developed by the IBM Space Computing Center. The derivation of equations (1) and (2) is attached. Equation (3) is a standard expression of λ in the Mercury System. Equations (4), (5), and (6) are standard geodesy textbook formulas.

If A3MSCP was entered with time in fixed point it is immediately converted to floating point seconds and stored in a temporary storage location. If the program was entered with time in floating point that too is stored in a temporary storage location.

3.14.4 Usage

a) Calling Sequences

There are two possible places to enter this program: at A3MSCP and at A3MSCP + 12. The former is used if the input contains time in fixed point (seconds in decrement and minutes in address) and the later is used if time is in floating point seconds.

α	TSX	A3MSCP, 4
$\alpha + 1$	H	L (Input table), L (Output table)
$\alpha + 2$	Error Return	
$\alpha + 3$	Normal Return	

or

α	TSX	A3MSCP + 12, 4
$\alpha + 1$	H	L (Input table), L (Output table)
$\alpha + 2$	Error Return	
$\alpha + 3$	Normal Return	

Note: H in α+1 signifies that if the sign of α + 1 is plus, H is computed. If the sign of α +1 is negative, H is not computed.

(Use PZE or MZE)

b) Space Required

Instructions	99
Erasable Storage	10
U1SQRT Routine	52
U1ATAB Routine	106
Constants	9
Output Table	3
Total . .	279 (plus an input table of at least 4 locations)

As a first estimate of ϕ the angle ϕ''' is used (geodetic latitude of the intersection of the given radius vector with the surface of the earth) and designated as ϕ_i . Thus by equation (6)

$$\tan \phi_i = \frac{1}{f^2} \tan \phi' = \frac{1}{f^2} \frac{Z}{\sqrt{X^2 + Y^2}}$$

The sec ϕ_i , sin ϕ_i , and $\frac{1}{N_i}$ values are now computed using tan ϕ_i . Tan ϕ_{i+1} may now be computed utilizing the new sin and $\frac{1}{N}$ values in equation (1).

The computed value of tan ϕ_{i+1} is used as the new estimate of tan ϕ and a second iteration of computations is made. The new value of tan ϕ_{i+1} is the true value of tan ϕ correct to eight places.

Using this value and the arctangent subroutine ϕ is determined. λ is found using equation (3). ϕ is between $-\pi/2$ and $+\pi/2$ and is between ϕ and 2π when stored.

The A3MSCP program now tests for the desire to compute h by seeing if the sign of 1,4 is positive. If it is minus, A3MSCP returns to 3,4. If the sign is plus, the program computes sec ϕ_i using the true value of tan ϕ .

Using this value and the previously computed value for $\frac{1}{N}$, h is determined and stored and the A3MSCP program returns to 3,4.

c) Error Codes

The only errors possible occur as returns from U1SQRT and U1ATAB due to bad data. A3MSCP returns control to $\alpha + 2$.

d) Constants Used

K00000	DEC	0	Fixed Point
K00060	DEC	60	Fixed Point
KCH233	OCT	233000000000	Fixed Point
K.OMES	DEC	7.29211508E-5	Rotation of earth in rad./sec.
K.ECC2	DEC	6.6934215E-3	Square of eccentricity
K001.0	DEC	1.0	Floating Point
K.INB2	DEC	1.00673852	Inverse of square of semi-minor axis
KKK2PI	DEC	6.283185307	Two Pi
KLAMDO*	DEC	5.9368535	Inertial Longitude in radians of Greenwich at midnight preceding launch

* KLAMDO is not a true constant. It is a parameter whose value changes with each day of the year.

e) Timing

Entering at A3MSCP (Time in fixed point)

1) computing H ---------- 19.7 milliseconds

2) not computing H -------- 17.5 milliseconds

Entering at A3MSCP + 12 (Time in floating seconds)

1) computing H ---------- 19.1 milliseconds

2) not computing H -------- 16.9 milliseconds

3.14.5 Checkout

The A3MSCP program was checked out using $\bar{R}$ vectors whose geodetic latitudes were approximately 15° apart from about 90°S to 90°N. Program calculations of ϕ, λ, and h were checked against hand calculations. Results were satisfactory.

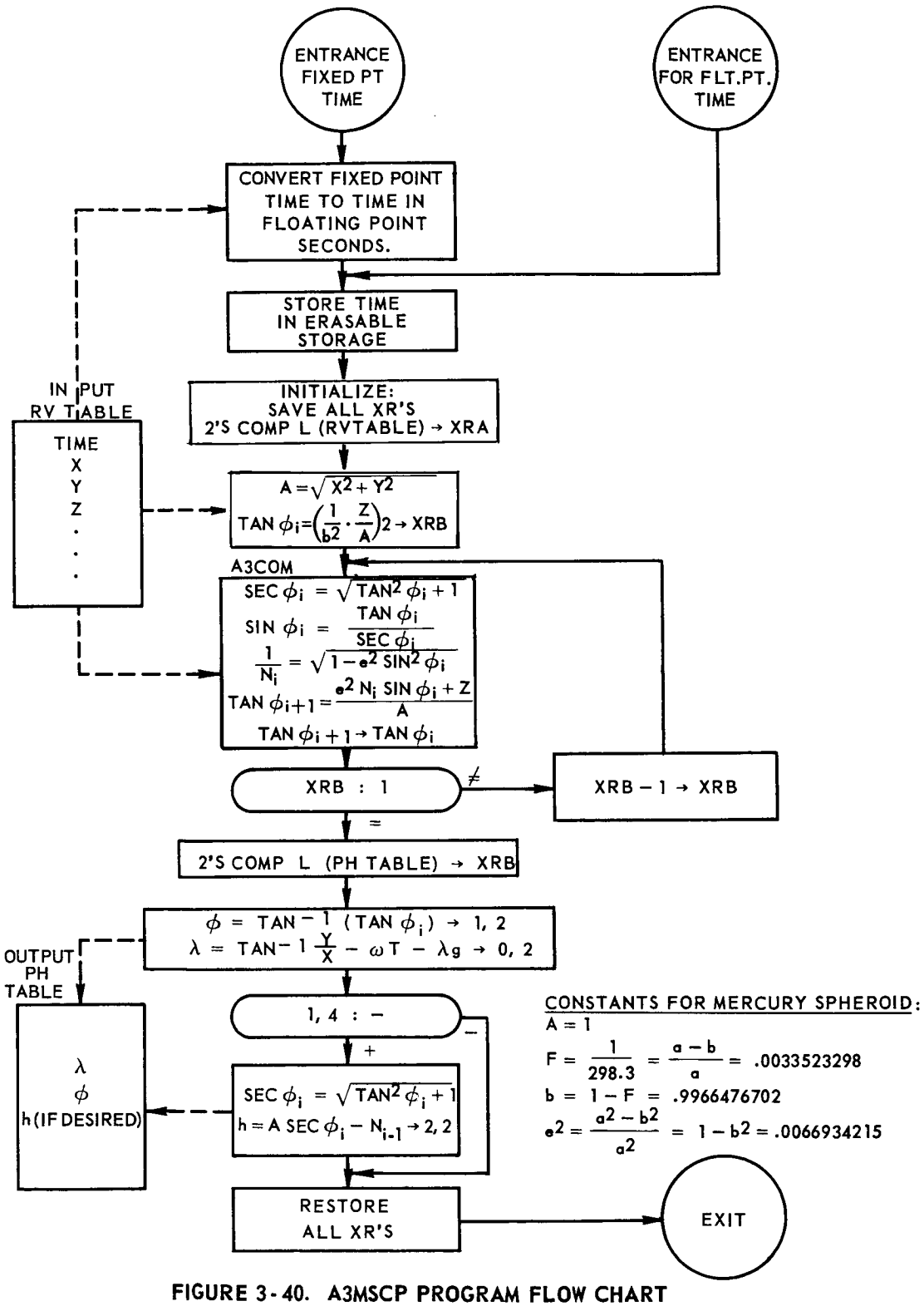
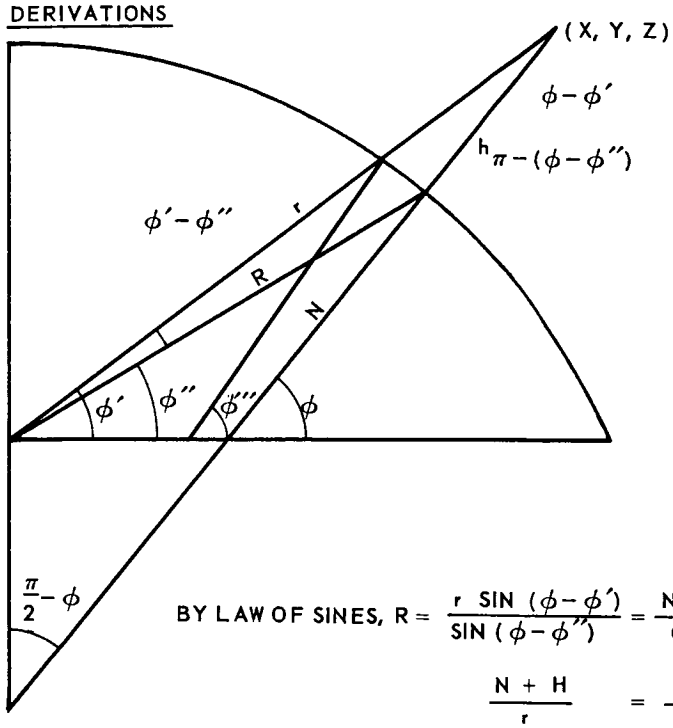


FIGURE 3-40. A3MSCP PROGRAM FLOW CHART

DERIVATIONS



BY LAW OF SINES, $R = \dfrac{r \text{ SIN } (\phi - \phi')}{\text{SIN } (\phi - \phi'')} = \dfrac{N \text{ COS } \phi}{\text{COS } \phi'''}$ (1)

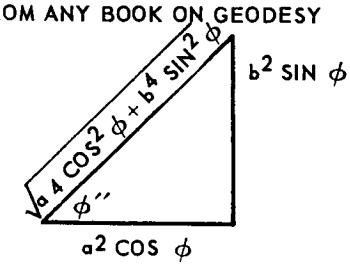
$$\frac{N + H}{r} = \frac{\text{COS } \phi'}{\text{COS } \phi} \quad (2)$$

WHERE $N = \dfrac{a}{\sqrt{1 \quad e^2 \text{ SIN}^2 \phi}}$ (3)

AND $\text{TAN } \phi'' = \dfrac{b^2}{a^2} \text{ TAN } \phi$ (4)

AND $\text{TAN } \phi''' = \dfrac{a^2}{s^2} \text{ TAN } \phi'$ (5)

FROM ANY BOOK ON GEODESY



$$\text{SIN } (\phi - \phi'') = \text{SIN } \phi \text{ COS } \phi'' - \text{COS } \phi \text{ SIN } \phi''$$

$$= \text{SIN } \phi \frac{a^2 \text{ COS } \phi}{\sqrt{a\,4 \text{ COS}^2 \phi + b\,4 \text{ SIN}^2 \phi}} - \frac{\text{COS } \phi \, b^2 \text{ SIN } \phi}{\sqrt{a\,4 \text{ COS}^2 \phi + b\,4 \text{ SIN}^2 \phi}} = \frac{(a^2 - b^2) \text{ SIN } \phi \text{ COS } \phi}{\sqrt{a\,4 \text{ COS}^2 \phi + b\,4 \text{ SIN}^2 \phi}}$$

$$r \text{ SIN } (\phi - \phi) = r \text{ SIN } \phi \text{ COS } \phi' - r \text{ COS } \phi \text{ SIN } \phi' = \sqrt{X^2 + Y^2} \text{ SIN } \phi - Z \text{ COS } \phi$$

HENCE THE 1ST EXPRESSION ABOVE GIVES:

$$\frac{\sqrt{X^2 + Y^2} \text{ SIN } \phi - Z \text{ COS } \phi}{(a^2 - b^2) \text{ SIN } \phi \text{ COS } \phi} = \frac{N}{a^2}$$

$$\sqrt{X^2 + Y^2} \text{ TAN } \phi - Z = \left(\frac{a^2 - b^2}{a^2}\right) N \text{ SIN } \phi = e^2 N \text{ SIN } \phi$$

$$\boxed{\text{TAN } \phi = \frac{e^2 N \text{ SIN } \phi + Z}{\sqrt{X^2 + Y^2}}}$$

$$\boxed{\begin{aligned} h &= \frac{\sqrt{X^2 + Y^2}}{\text{COS } \phi} - N \\ &= \text{SEC } \phi \sqrt{X^2 + Y^2} - N \end{aligned}}$$

3.15 INERTIAL POSITION VECTOR CONVERSION PROGRAM (MMAQA)

This program converts an inertial position vector from the launch phase to local spherical coordinates and generates pointing data (azimuth, and elevation) at various time intervals for use at Bermuda. The MMAQA program flow chart is shown in Figure 3-41.

3.15.1 Input Requirements

This program uses the UA1LSC library routine. The following information serves as input to MMAQA:

		Location	Format
X_3	Position vector	TCMANR + 1	Floating point Mercury units
Y_3		TCMANR + 2	
Z_3		TCMANR + 3	
Time since launch		TCMANR + 29	Floating point seconds

3.15.2 Output Requirements

MCTADL: A control word indicating to Monitor when data is ready to be converted and packed for transmission.

If 1 = data ready

If 0 = data not ready

The following data is stored in table TCMANR by MMAQA:

Location	Contents	Description	Format
TCMANR + 225	T sub I	Time of Acquisition data	Floating Point Radians
+ 226	A sub T I	Azimuth at time of computation	
+ 227	E sub T I	Elevation at time of computation	
+ 228	A nominal	nominal azimuth	Floating Point Radians
+ 229	E nominal	nominal elevation	
+ 230	AK	first azimuth at time K	
+ 231	AK + N	second azimuth at K + N	

Location		Contents	Description	Format
TCMANR	+ 232	AK + 2N	third azimuth at K + 2N	Floating Point Radians
	+ 233	AK + 3N	fourth azimuth at K + 3N	
	+ 234	EK	first elevation at time K	
	+ 235	EK + N	second elevation at K + N	
	+ 236	EK + 2N	third elevation at K + 2N	
	+ 237	EK + 3N	fourth elevation at K + 3N	

TMLABF - A 16 location output block is provided for O0LSTY

TMLABF		Time (K)	K seconds after Life-off	Min → address Sec → decrement
	+ 1			
	+ 2	azimuth	at time K	Flt. Pt. Radians
	+ 3	elevation	at time K	Flt. Pt. Radians
	+ 4	Time (K+ N)	K+N seconds after lift-off	Min in address Sec in decrement
	+ 5			
	+ 6	azimuth	at time (K+ N)	Flt. Pt. Radians
	+ 7	elevation	at time (K + N)	Flt. Pt. Radians
	+ 8	time (K+2N)	K+2N Sec 1 after lift-off	Min. in address Sec. in decrement
	+ 9			
	+ 10	azimuth	at time (K+ 2N)	Flt. Pt. Radians
	+ 11	elevation	at time (K+2N)	Flt. Pt. Radians
	+ 12	time (K+3N)	K+3N sec after lift-off	Min in address Sec in decrement

Location		Contents	Description	Format
TMLABF	+ 13			
	+ 14	azimuth	at time (K+ 3N)	Flt. Pt. Radians
	+ 15	elevation	at time (K + 3N)	Flt. Pt. Radians

a) If elevation is negative zeros will be stored in the specified location in TMLABF.

b) Locations TMLABF + 1, + 5, + 9, + 13 will contain zeros since program is not concerned with computing range.

c) The MMAQA program computes azimuth and elevation at a given time and a nominal azimuth and elevation, and then computes ΔA and ΔE from the difference between the given and the nominal values. These deltas are used to correct the nominal azimuth and elevation which are stored in the TCCONS table. This data is stored in TCMANR and in TMLABF to be converted and sent to Bermuda via the teletype line.

d) Constants which should be taken into consideration are coefficients of the functions used to compute nominal azimuth and nominal elevation. These are located in table

TCCONS + 220 --- TCCONS + 222 for azimuth

TCCONS + 223 --- TCCONS + 225 for elevation

e) Also, the nominal azimuth and elevation at the following times:

Location		Contents	Format
TCCONS	+ 184	nominal azimuth at 210 sec	Flt. Pt. Radians
	+ 185	nominal azimuth at 240 sec	Flt. Pt. Radians
	+ 186	nominal azimuth at 270 sec	Flt. Pt. Radians
	+ 187	nominal azimuth at 300 sec	Flt. Pt. Radians
	+ 188	nominal azimuth at 330 sec	Flt. Pt. Radians
	+ 189	nominal azimuth at 360 sec	Flt. Pt. Radians
	+ 190	nominal azimuth at 390 sec	Flt. Pt. Radians
	+ 191	nominal elevation at 210 sec	Flt. Pt. Radians

Location	Contents	Format
+ 192	nominal elevation at 240 sec	Flt. Pt. Radians
+ 193	nominal elevation at 270 sec	Flt. Pt. Radians
+ 194	nominal elevation at 300 sec	Flt. Pt. Radians
+ 195	nominal elevation at 330 sec	Flt. Pt. Radians
+ 196	nominal elevation at 390 sec	Flt. Pt. Radians

Time at which this program is entered from CCMAIN is located in TCCONS + 217.

K which is the time of the first predicted value is located in TCCONS + 219.

N which is the interval at which pointing data is computed for Bermuda is located in TCCONS + 218.

The following values from the Bermuda station characteristics block are needed:

Position	Description	Format	Units
8	$\sin \phi$	Floating point	Radians
9	$\cos \phi$	Floating point	Radians
10	$R \sin (\phi - \phi')$	Floating point	Earth radii
11	$R \cos (\phi - \phi')$	Floating point	Earth radii
18	Inertial λ at Reference Time	Floating Point	Radians

3.15.3 Method

The MMAQA program uses the following equations:

$A_{nom} = A\ T_i^2 + B\ T_i + C$: To compute nominal azimuth

(A, B, C being coefficients; T_i being time)

$E_{nom} = D\ T_i^2 + E\ T_i + F$: To compute nominal elevation

3.15.4 Usage

To enter program CCMAIN should transfer to location MMAQA, a test will then determine flow of program.

The only error return in program will be from subroutine UA1LSC. If this occurs, MMAQA will consider data being computed to be bad and will transfer out to CCMAIN.

Space Required:

Instructions	109	
Constants	7	
Common	29	
Subroutine	209	(U1ALSC)
	267	(U1SICO, U1ATNA, U1ATAB, U1SQRT)
	621	Locations (including subroutines)
	145	Locations (excluding subroutines)

Constants used:

K00001	DEC	1
K00060	DEC	60
K0.375	DEC	0.375
K004.0	DEC	4.0
K0060.	DEC	60.0
KD0001	PZE	, , 1 (1 in decrement)
KCH233	OCT	233000000000

Erasable and common locations:

MMTAD	PZE
MMTMP	BSS3
MMCON	BSS 6
MMDLA	PZE
MMDLE	PZE

MMIPA	PZE
MMIPB	PZE
MMIPC	PZE
MMIPD	PZE
MMIPE	PZE
MMOPT	BSS 12

Timing:

CCACQ	0.03 milliseconds
UA1LSC	3.7 milliseconds
Total	3.73 milliseconds

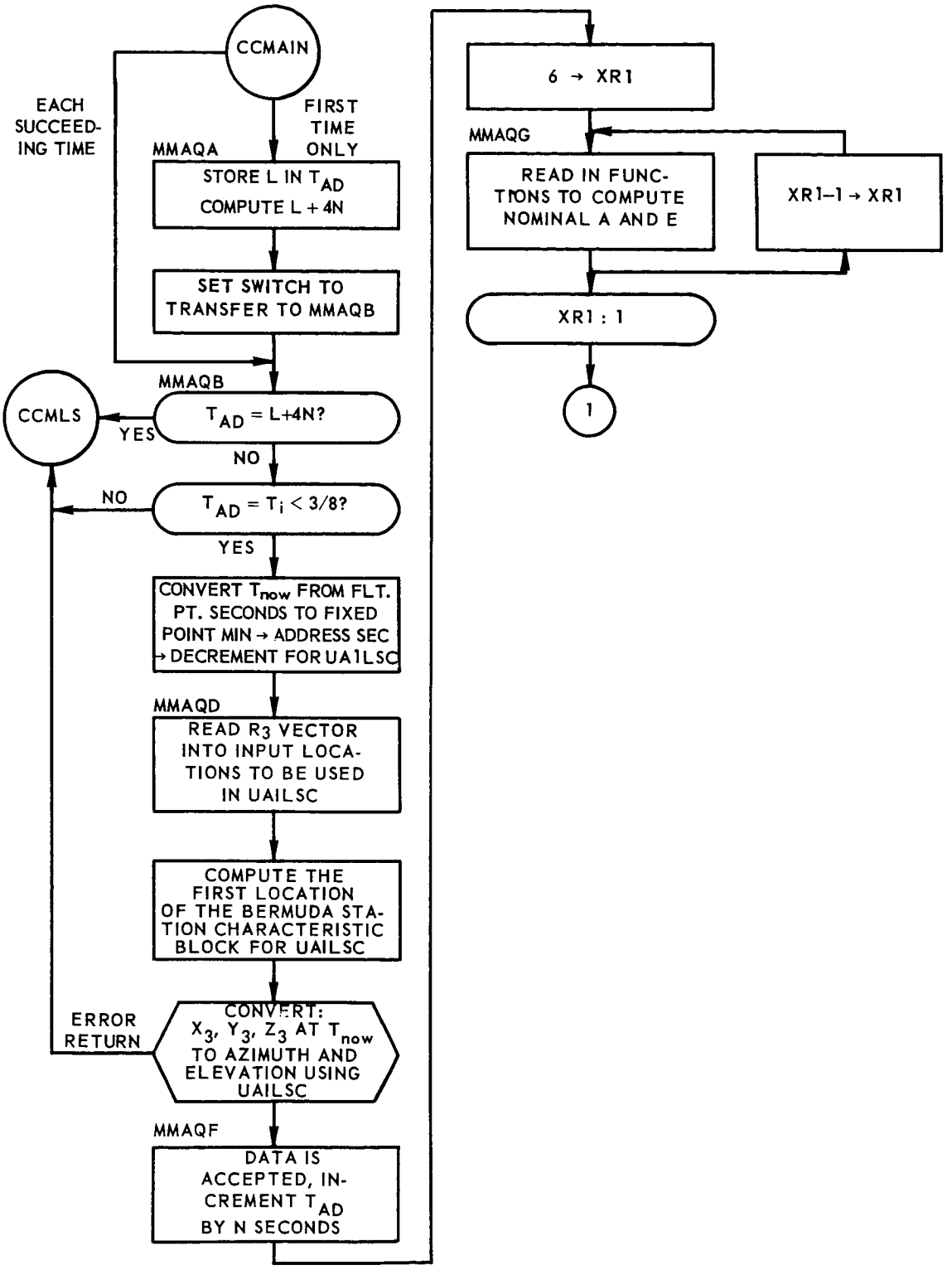


FIGURE 3-41. MMAQA PROGRAM FLOW CHART *(Sheet 1 of 2)*

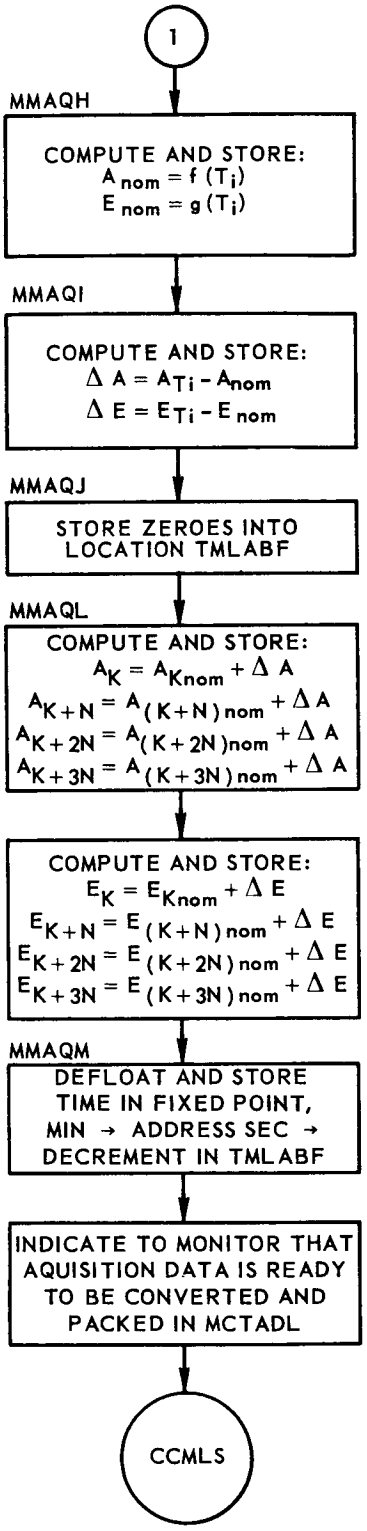


FIGURE 3-41. MMAQA PROGRAM FLOW CHART *(Sheet 2 of 2)*

3.16 PRINT ON-LINE MESSAGE ROUTINE (PSIP)

This is a subroutine used exclusively by CCMAIN to print on-line messages. The flow chart for the PSIP routine is shown in Figure 3-42.

3.16.1 Input and Output Requirements

a) Communication cell MCLMSG.

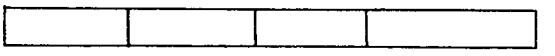
b) When entering PSIP the AC must contain a number related to the number of the message which is to be printed. (All messages are numbered on the message tape.)

c) There are two returns from PSIP: a normal return and an error return. The normal return means that the number in the AC was packed into MCLMSG and the message will be printed under Monitor control. The error return indicates that cell MCLMSG was full and the message will be lost. In this case, a 1 is stored in the decrement of location PSIP.

3.16.2 Method

Monitor Communication cell MCLMSG is divided into four sections of 9 bits:

MCLMSG:



S B_8 B_{17} B_{26} B_{35}

Message numbers are right justified in each section. PSIP checks each section, starting with bits 27 - 35. If there is a message code already present, it checks the next section, bits 17 - 27 and etc. When an empty section is detected, the new message code is stored in that section. If there is no empty section, an error is indicated as above.

The printing routine is activated in Monitor launch program MLUPDT whenever there are messages to be printed, i.e., when MCLMSG $\neq 0$. As the messages are printed on-line, Monitor clears the sections of MCLMSG.

3.16.3 Usage

a) Calling Sequence:

Message code must be in accumulator.

```
TSX          PSIP, 4

Error Return

Normal Return
```

b) Storage Required: 29 locations

Reference is made to one monitor communication cell: MCLMSG.

c) Timing:

Shortest: 20-7090 cycles

Longest: 41-7090 cycles

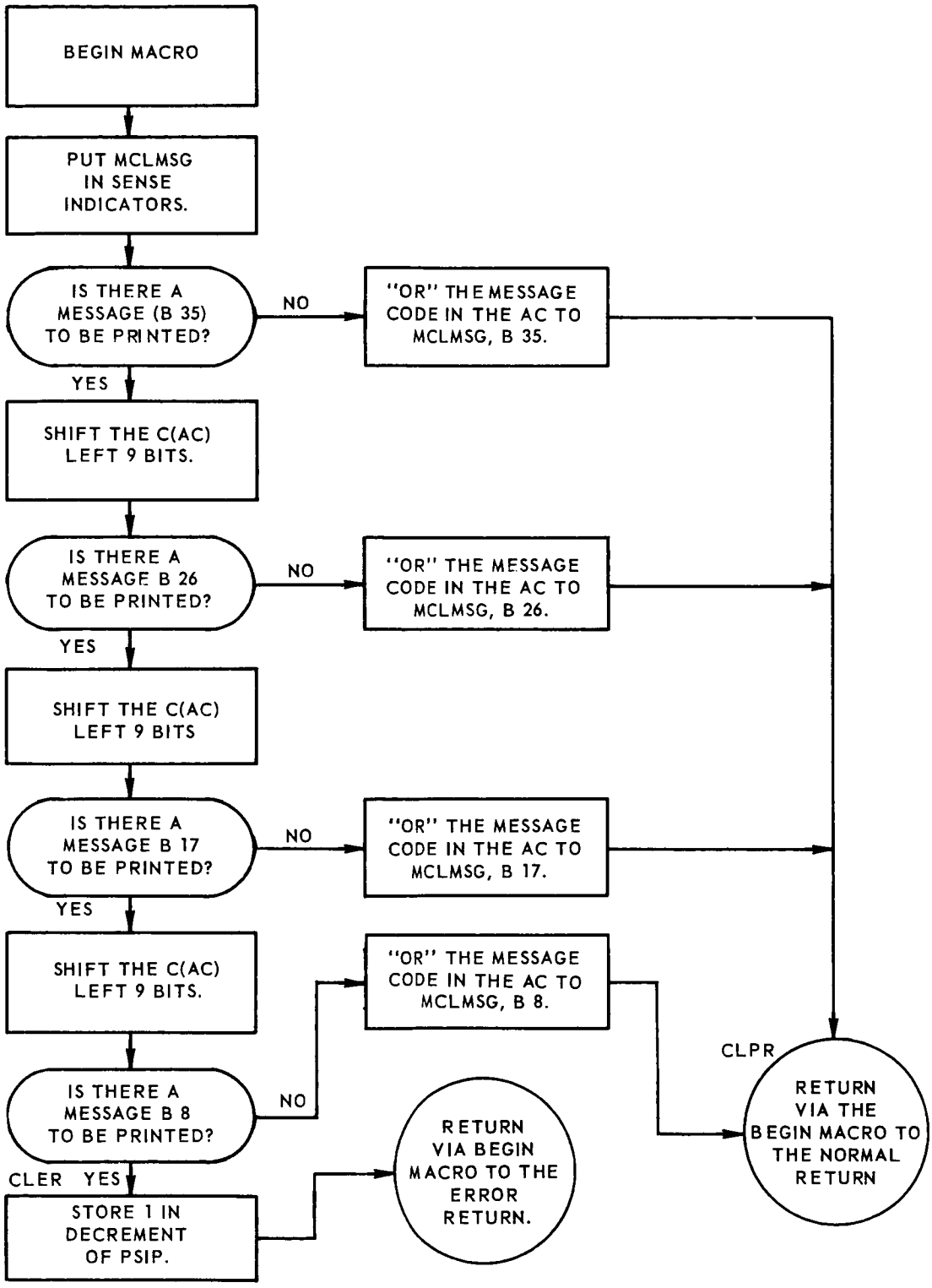


FIGURE 3-42. PSIP ROUTINE FLOW CHART

3.17 HOLD PHASE MAIN CALCULATION PROGRAM (CCHOLD)

CCHOLD computes the subcapsule present position. It then determines the existence of one of the following conditions and takes the indicated action.

a) The prior indication given by CCMAIN that powered flight has ended is still valid. The test in c) is then made.

b) It is invalid. Monitor is informed to return to the complete launch processing.

c) The prior indication given by CCMAIN about the number of posigrade rockets fired is still valid. The CCHOLD computation is then complete, or,

d) The prior indication about the posigrade is no longer current. A direct transfer into the middle of CCMAIN is made, causing recomputation of the GO, NO-GO recommendation and the Δt_r .

The CCHOLD program flow chart is shown in Figure 3-43.

3.17.1 Input Requirements

The hold phase is entered after the normal launch cycle has been completed; that is, after the GO, NO-GO calculation has been made. The program obtains control at a rate equivalent to the arrival rate of the selected source.

3.17.2 Output Requirements

Computes present position (ϕ_{pp} and λ_{pp}).

3.17.3 Method

Upon entering the program, the latitude and longitude are updated by the subroutine A3MSCP which uses the new capsule inertial position $\bar{r}$ at a given t.

Because the GO, NO-GO computation could have been made without having received CAPS or the signal may have since disappeared, a check is made to determine if the signal is present in the current telemetry. If CAPS or ABOI is indicated then the program checks for a change in posigrade telemetry and saves the new count of posigrade rockets fired.

When a change in posigrade information is determined a further check is made to determine whether B-GE or Azusa data was available during the GO,NO-GO. If so, CCHOLD then recalculates the final GO, NO-GO including the change in velocity due to posigrades. If there is no recalculation, the program will exit to the Monitor suffix (MFLHLD) which allows the launch system to cycle in the hold phase.

When both CAPS and ABOI are missing, CCHOLD then checks the discrete signal SECO. If SECO has been received an internal indication is set so that hereafter CAPS will not be assumed. The various counters and indicators used for the GO, NO-GO are then reinitialized and exit is made to the Monitor suffix (MFLNML) which will cause execution of the complete CCMAIN program.

It should be emphasized that SECO and ABOI are not treated in the same fashion. Since SECO appears as a discrete signal in the B-GE data, it cannot be manually overridden. Therefore, if we choose to believe that our GO, NO-GO calculation was valid purely on the basis of a possible erroneous SECO and an assumed CAPS, no verification of the true termination of powered flight would be possible. Thus, we have to await either of the telemetry signals CAPS or ABOI, which can be overridden by the Cape, before we can assume that powered flight has terminated.

3.17.4 Usage

Constants:

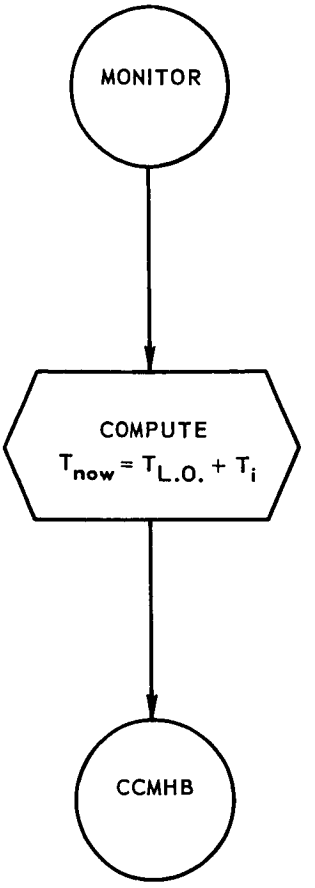
K00010	DEC	10
K00008	DEC	8
KCH233	OCT	233000000000
K003.0	DEC	3.0

Space Required:

71 instructions

Under certain conditions CCHOLD uses certain sections of CCMAIN constants:

TCCONS + 149:	DEC	20000000.
TCCONS + 90:	DEC	.92534419 E-3 24 ft/sec, Mercury units



NOTE

MONITOR WILL STORE THE $T_i \overline{R}_3 \overline{V}_3$ COMPUTED BY COORDINATE CONVERSION OF IP 7090 IN TCMANR OR CCRAWR WILL STORE $T_1 \overline{R}_3 \overline{V}_3$ IN TCMANR

FIGURE 3-43. CCHOLD PROGRAM FLOW CHART *(Sheet 1 of 3)*

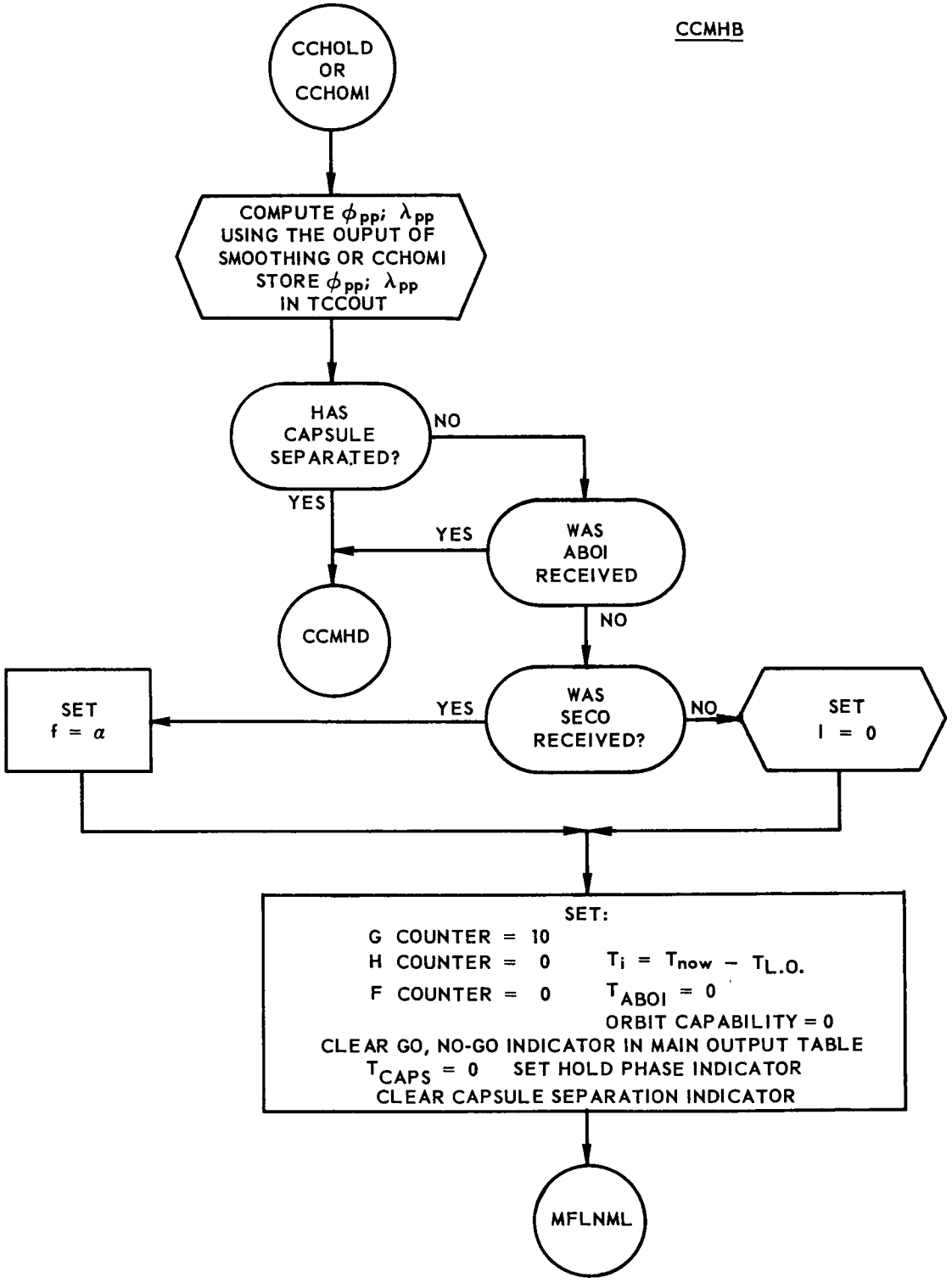


FIGURE 3-43. CCHOLD PROGRAM FLOW CHART *(Sheet 2 of 3)*

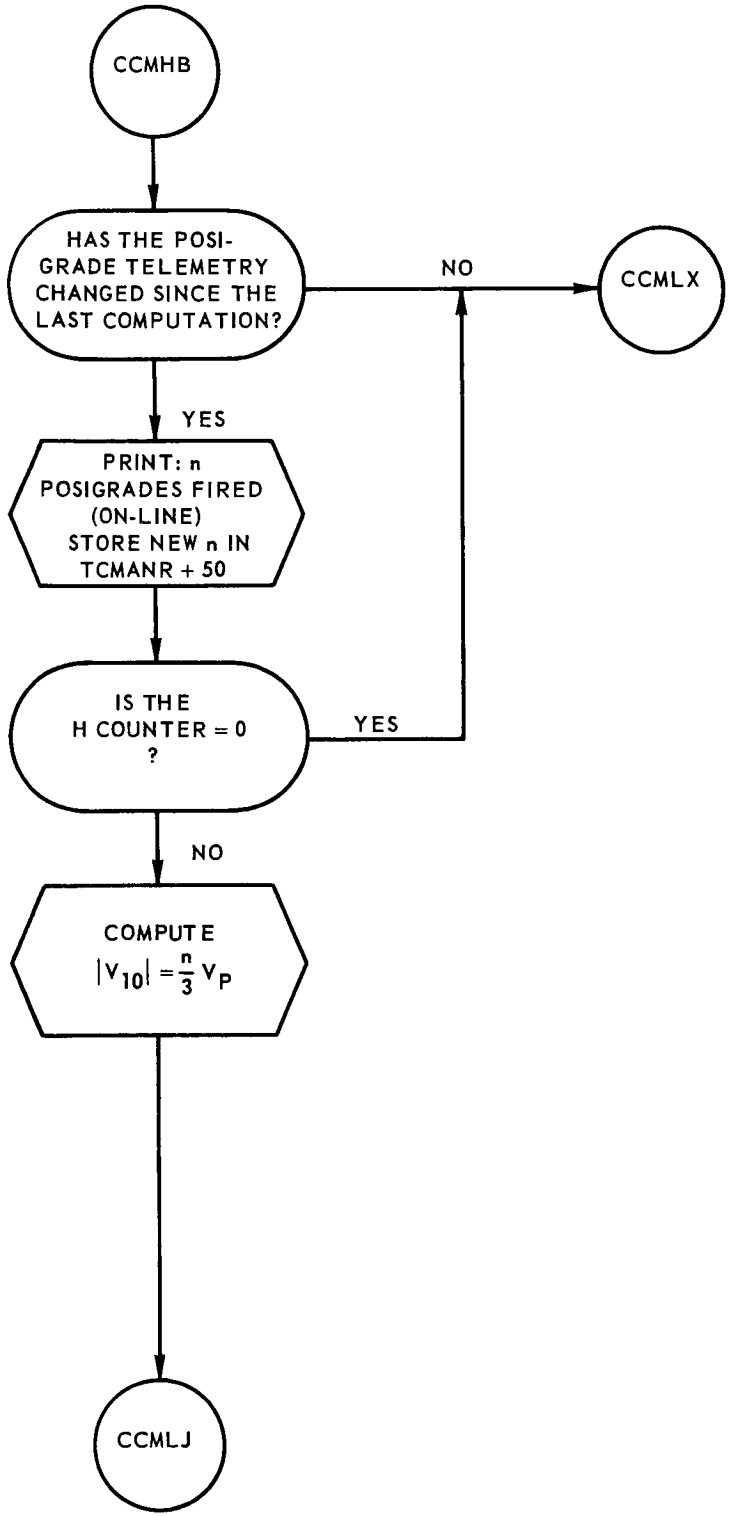


FIGURE 3-43. CCHOLD PROGRAM FLOW CHART *(Sheet 3 of 3)*

3.18 MISSING DATA COMPUTATION PROGRAM (CCMISS)

The function of this program is two-fold. Primarily, it examines the available telemetry for signals and records the time associated with the new signals and clears the indicators associated with previously detected signals which have since disappeared. Secondly, CCMISS updates the velocity ratio and the flight path angle for display on plotboard 1 and also on the Flight Dynamics Officer's Console (FDOC). The CCMISS program flow chart is shown in Figure 3-44.

3.18.1 Input Requirements

This program is entered during the normal launch cycle whenever there is no data or whenever the data recorded is of poor quality. The entry rate is therefore a function of the 1/2 second missing data check or the arrival rate of the selected source.

3.18.2 Output Requirements

Prior to ABOI (abort initiate) or ESRF (Escape Rockets Fired) the Velocity Ratio (V/V_r) and the flight path angle (γ) are updated by extrapolation in the following manner. First the nominals for each are obtained by solving polynominals (for V/V_r and for (γ) with respect to time). Then the last deviation from the nominal, which was calculated with the last good data, is taken from storage and added to the new nominal. This constitutes the extrapolation of V/V_r and γ . This updating is discontinued after ABOI or ESRF.

If tower separation (TOWS) has not occurred, the program constantly checks for ABOI. Having received this signal it now awaits the indication of ESRF. The firing of the escape rockets prior to TOWS and the receipt of the abort bit are sufficient to place the program system into one of the two abort programs below TOWS.

The low or medium abort programs are not used after TOWS. If ABOI is detected the capsule separation signal is tested (CAPS) or, after sufficient time has elapsed, the capsule separation is assumed. Exit is then made to the Monitor suffix (MFLNML) or to the block CCMLV (in CCMAIN).

3.18.3 Usage

Storage Required:

192 instructions.

CCMISS then transfers to various sections of CCMAIN.

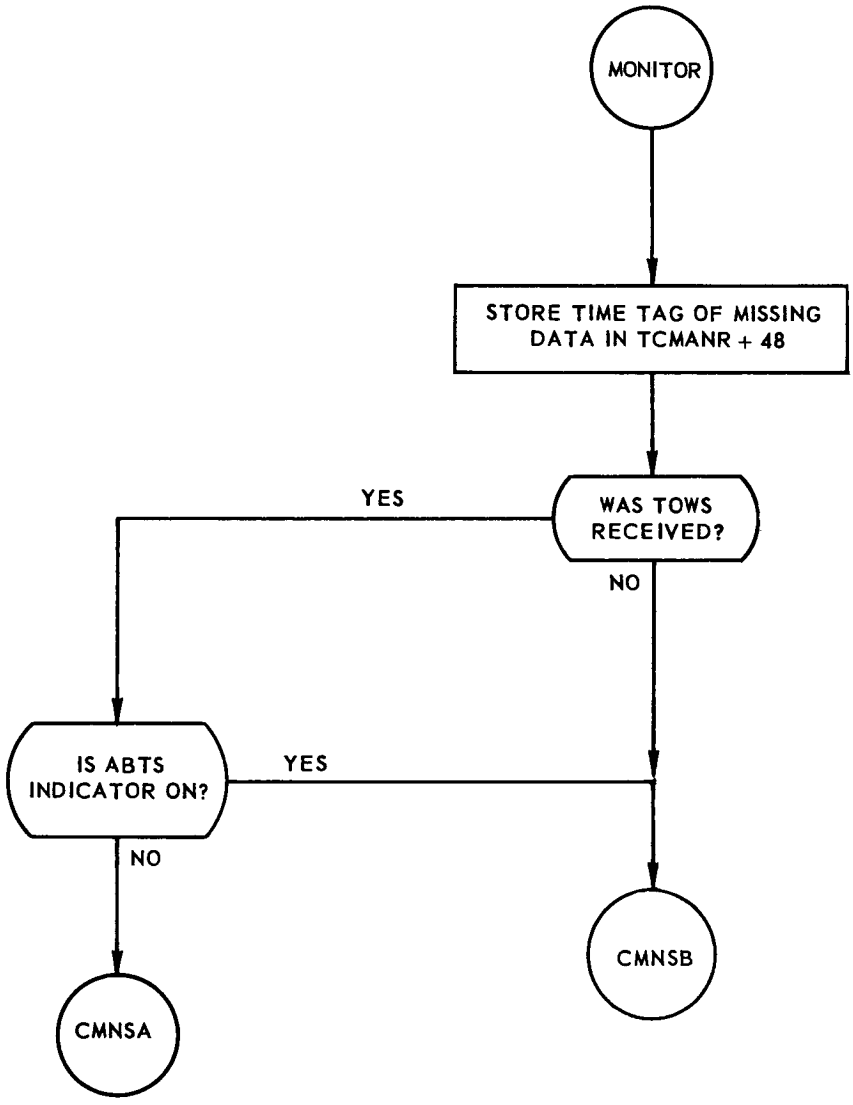


FIGURE 3-44 CCMISS PROGRAM FLOW CHART *(Sheet 1 of 6)*

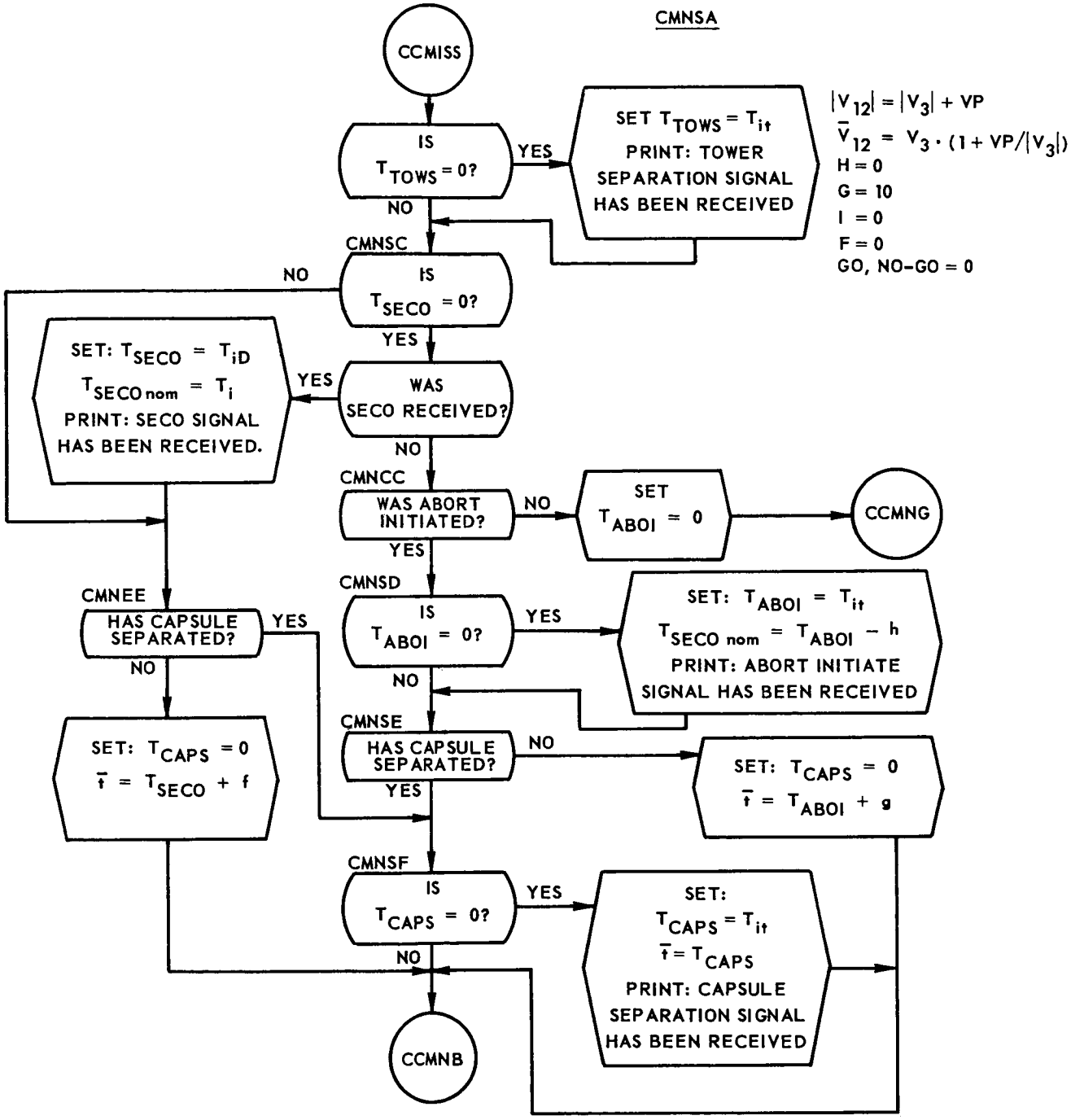


FIGURE 3-44. CCMISS PROGRAM FLOW CHART *(Sheet 2 of 6)*

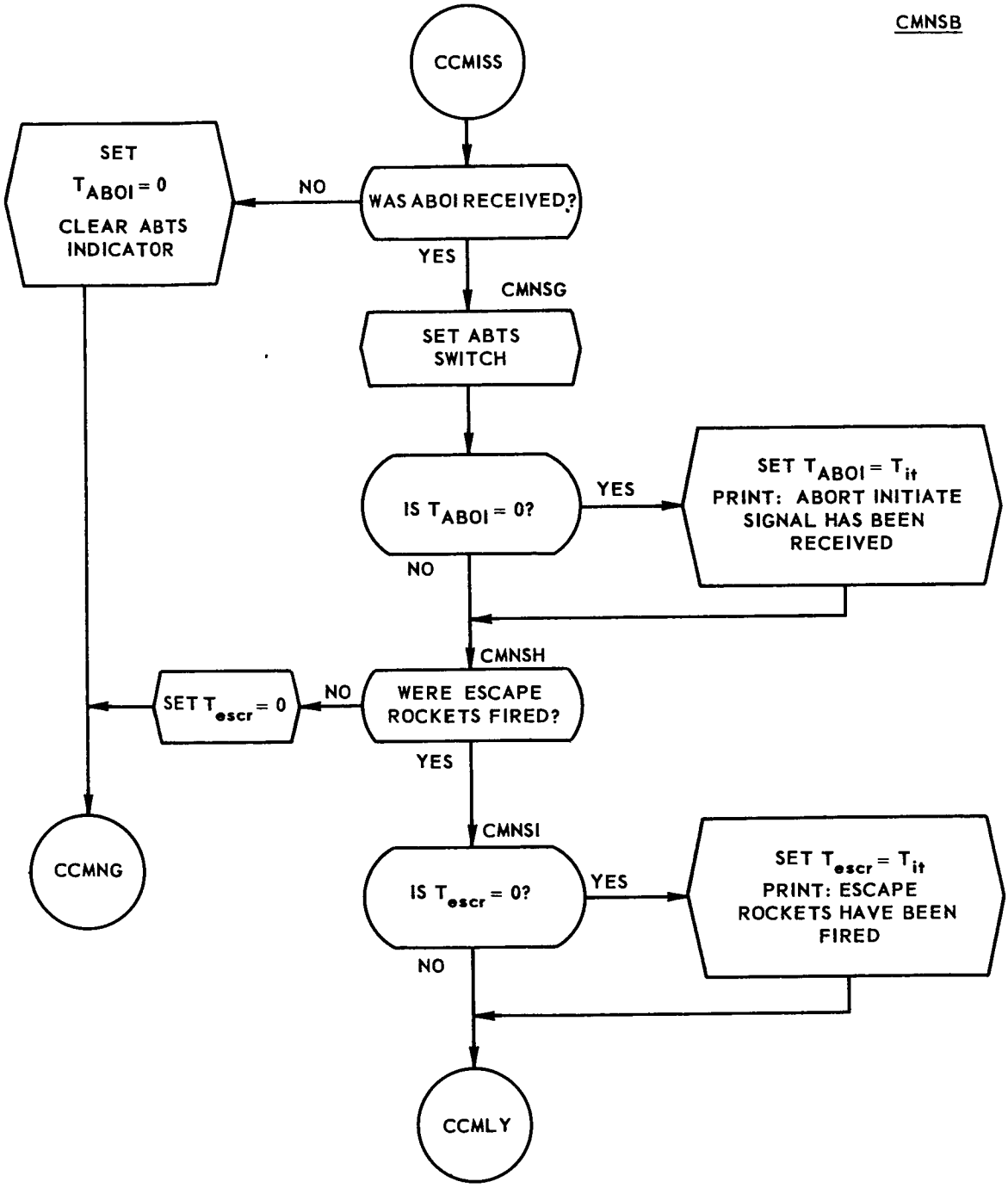


FIGURE 3-44 CCMISS PROGRAM FLOW CHART *(Sheet 3 of 6)*

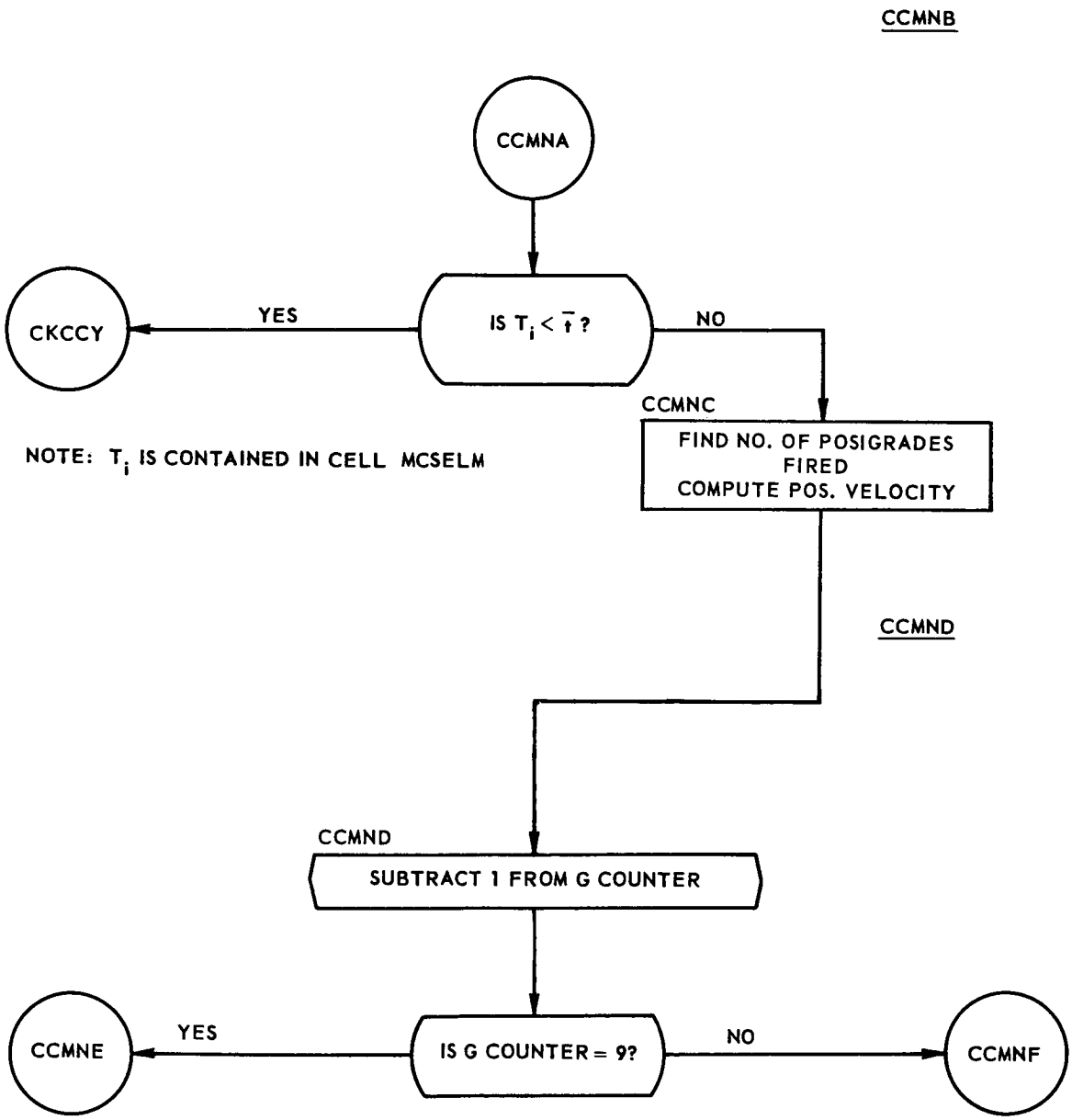
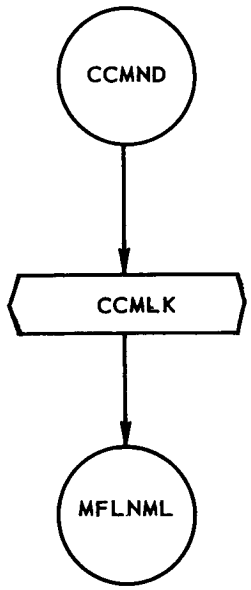


FIGURE 3-44 CCMISS PROGRAM FLOW CHART *(Sheet 4 of 6)*

CCMNE



CCMNF

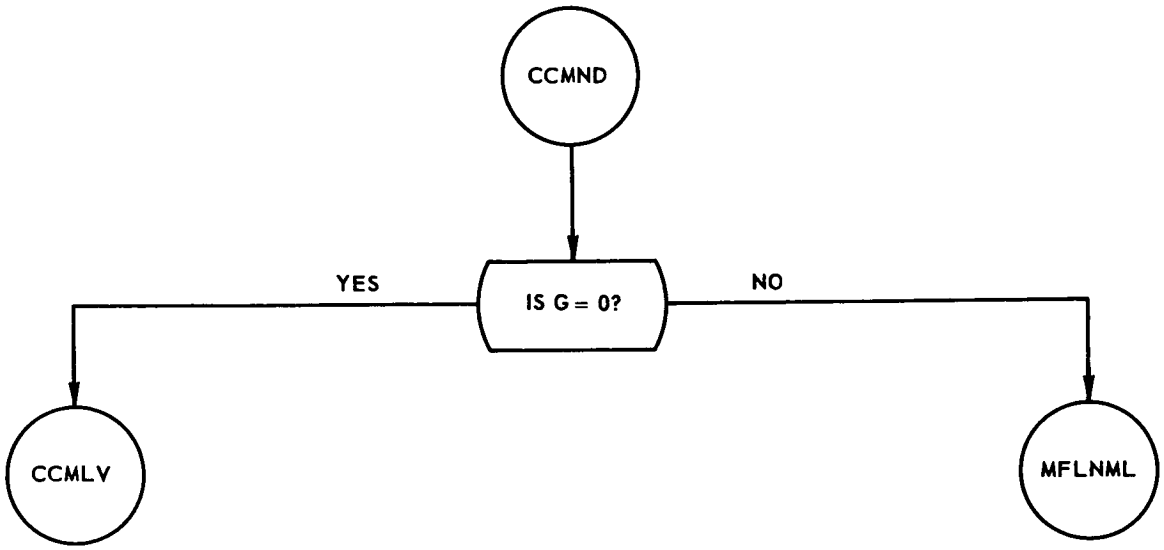


FIGURE 3-44 CCMISS PROGRAM FLOW CHART *(Sheet 5 of 6)*

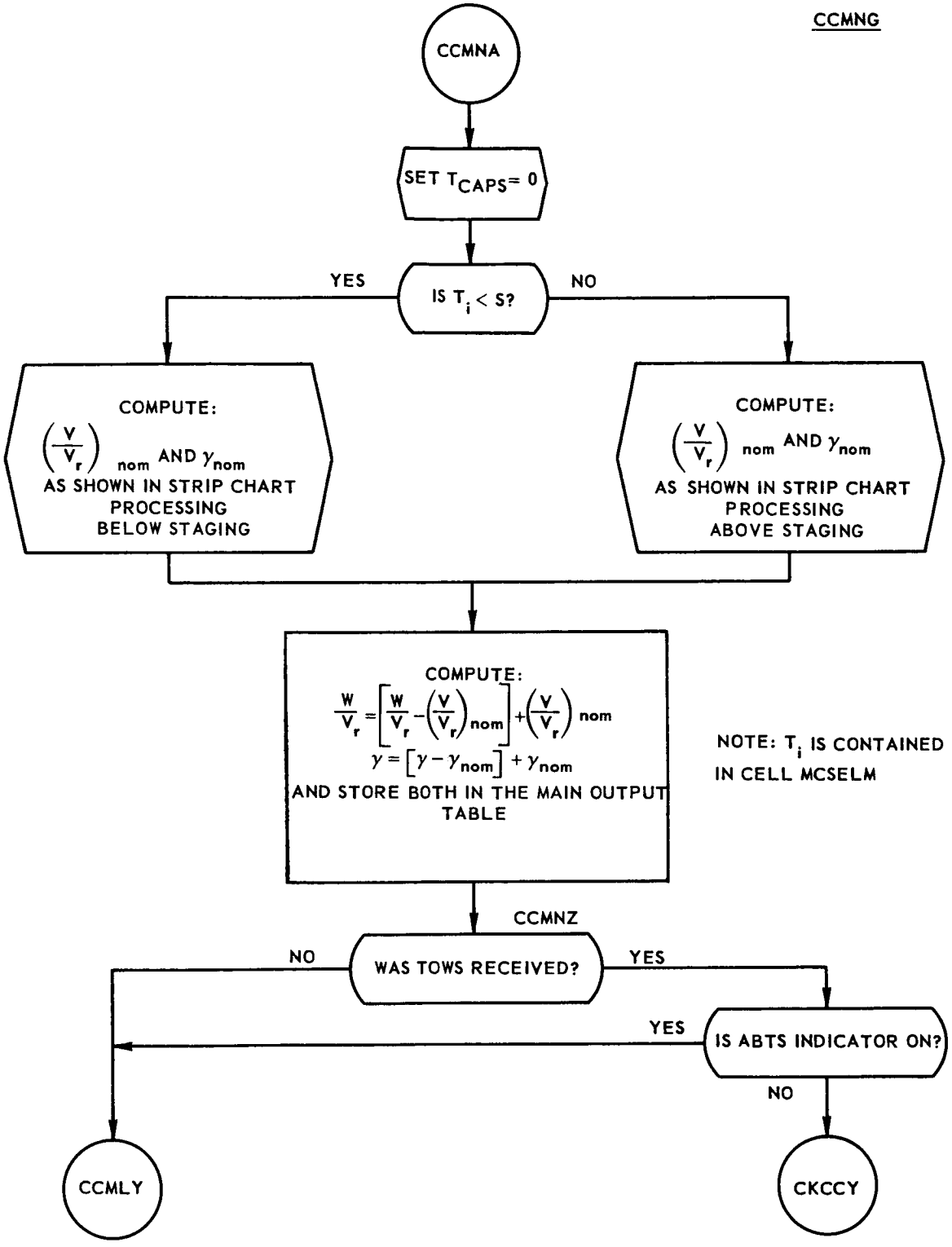
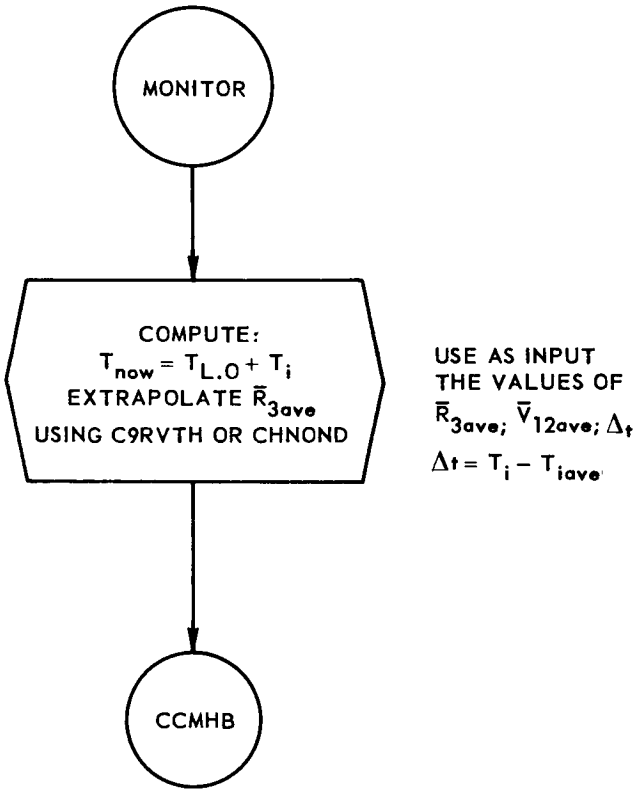


FIGURE 3-44 CCMISS PROGRAM FLOW CHART *(Sheet 6 of 6)*



NOTES:
T_i IS FOUND IN MCSELM
$T_{L.O.}$ IS FOUND IN MCLFTM
T_{iave} IS FOUND IN TCMANR + 55

FIGURE 3-45 CCHOMI PROGRAM FLOW CHART

3.19 MISSING DATA DURING HOLD (CCHOMI)

If during the hold portion of the launch phase CCRAWR or CC7091 rejects the bad AN/FPS-16 data entering line 2, or there is only Azusa data or there is no data available, then the CCHOMI program is used. This program extrapolates the average R and V vectors for use by the CCHOLD program. The flow chart for CCHOMI is shown in Figure 3-45.

3.19.1 Input Requirements

Control is given to the program after the GO, NO-GO calculation has been made and the selected source data is missing or of poor quality. The rate of entry is therefore governed by the 1/2 second missing data check or the arrival rate of the selected source.

3.19.2 Output Requirements

Extrapolates the $\bar{R}$ and $\bar{V}$ vectors for input to the Main Computation During Hold program (CCHOLD).

3.19.3 Method

This program performs the same function as CCHOLD with one exception. Because there is no current inertial vector from which to calculate the present capsule position the program, using subroutine C9RVTH, or CHNOND (in the case of a circular orbit), extrapolates to present time the average vector obtained from the CCMAIN program. This extrapolated vector is then used by the CCHOLD program. For further details see the write-up discussing CCHOLD, which operates exactly as this program after this point.

3.19.4 Usage

CCHOMI uses 36 instructions and then transfers to a section of CCHOLD.

3.20 LOW ABORT PROGRAM (CCRTYL - CCRTMI)

If a decision is made by Cape Canaveral to abort the mission before the launch vehicle has reached 100,000 ft. then the Goddard computers use the CCRTYL program. This program computes the display quantities which Goddard supplies at 6 second intervals to the displays at Cape Canaveral. If the abort decision is made prior to any data entering this program at the specified input times, then entry is made through the "missing data entry" (CCRTMI) which updates $\bar{r}$ and $\bar{v}$ parameters by elliptical extrapolation and then uses the updated quantities to compute outputs for display at Cape Canaveral. The flow chart for CCRTYL - CCRTMI is shown in Figure 3-46.

3.22.1 Input Requirements

Input to CCRTYL consists of seven locations in the input block of the program in the following order:

T - Time since life off (floating point seconds)

$\bar{R}_3$ - Positional data (Mercury units, inertial coordinate system)

$\bar{V}_3$ - Velocity data (Mercury units, inertial coordinate system)

If entry is made through the "missing data entry" CCRTMI no input is required.

The following Mercury programs and library routines are used by CCRTYL:

C9RVTH	U1ASCO
A3MSCP	U1SQRT - square root
U1ATNA	U3VMAG

3.20.2 Output Requirements

CCRTYL generates output for the following TCCOUT table locations:

TCCOUT	ϕpp (Geodetic Lat. of Present Position) Floating Point radians.
+ 1	λ pp (Geodetic Long. of Present Position) Floating Point radians.
+ 2	ϕrefined (Geodetic Lat. of impact point) Floating Point radians.
+ 3	λ (Longitude of Impact Point) Floating Point Radians.
+ 4	Ground Time Remaining Until Landing (seconds in decrement - minutes in address)

+ 11	Orbit Number (Set to 0)
+ 12	GMTLC (GMT of Landing) (seconds in decrement - minutes in address)
+ 13	ϕ_{LP} (Geodetic Lat. of Landing Point) Floating Point radians.
+ 14	λ_{LP} (Geodetic Long. of Landing Point) Floating Point radians.
+ 15	H (Altitude of Capsule) Floating Point nautical miles.
+ 16	γ (flight path angle) Floating Point radians.
+ 17	Elapsed Ground Time Since Retrofire = 0
+ 21	i (inclination angle) Floating point radians.
+ 23	V (inertial velocity) Floating point ft/sec.
+ 25	Elapsed Time Since Launch integral seconds.
+ 28	H(altitude of capsule) Floating Point nautical miles.
+ 29	Recovery Area Set for Recovery Area A.

3.20.3 Method

The following equations are used by CCTRYL:

$$T_i = T_{i-1} + \Delta T_{CONS} \quad \text{(used by CCRTMI only)}$$

$$|R_3| = \sqrt{X_3^2 + Y_3^2 + Z_3^2}$$

$$h = |R_3| - R_E$$

$$|V_8| = \sqrt{\dot{X}_8^2 + \dot{Y}_8^2 + \dot{Z}_8^2}$$

$$\gamma = \sin^{-1} \frac{X_3 \dot{X}_8 + Y_3 \dot{Y}_8 + Z_3 \dot{Z}_8}{|R_3| \, |V_8|}$$

$$T_{now} = T_{L.O.} + T_i$$

$$e = \sqrt{\frac{\dfrac{(X_3 \dot{X}_8 + Y_3 \dot{Y}_8 + Z_3 \dot{Z}_8)^2}{|R_3|}}{2 - |R_3| \; |V_8|^2} + \left[|R_3| \; |V_8|^2 - 1 \right]^2}$$

$$h_{AP} = \left(\frac{|R_3|}{2 - |R_3| \; |V_8|^2} \right) (1 + e) - R_E$$

$$GMTLC = T_{L,O,} + T_i + \Delta T_L$$

$$T_f = T_{L,O,} + T_i + \Delta T_{f_j} \quad (j = 1, 2)$$

$$\dot{X}_{8_e} = \dot{X}_8 + \omega Y_3$$

$$\dot{Y}_{8_e} = \dot{Y}_8 - \omega X_3$$

$$\dot{Z}_{8_e} = \dot{Z}_8$$

$$\Delta T_L = \Delta T_{f_j} + K_{f_j} \quad (j = 1, 2)$$

3.20.4 Usage

Constants

	Value	Function
K00.UT	806.8104	Conversion Mercury units to seconds
	.999251039	Radius of spherical earth (Mercury units)
	3443.929	Conversion Mercury units to nautical miles
KCFTSC	25936.294946 (Constant subject to change - Mercury units)	Conversion Mercury units to feet per sec. Apogee constant used to discriminate between using inertial or rotational co-ordinates to compute impact points

K.OMEG	.058833543	angular velocity of earth (radians per Mercury unit of time).
MCLFTM		contains time since lift off (sec.)
MCFINI		= 0 continue computations $\neq$ 0 instructs monitor to cease inputting to low abort program
MFLRT1		suffix to CCRTYL, CCRTMI

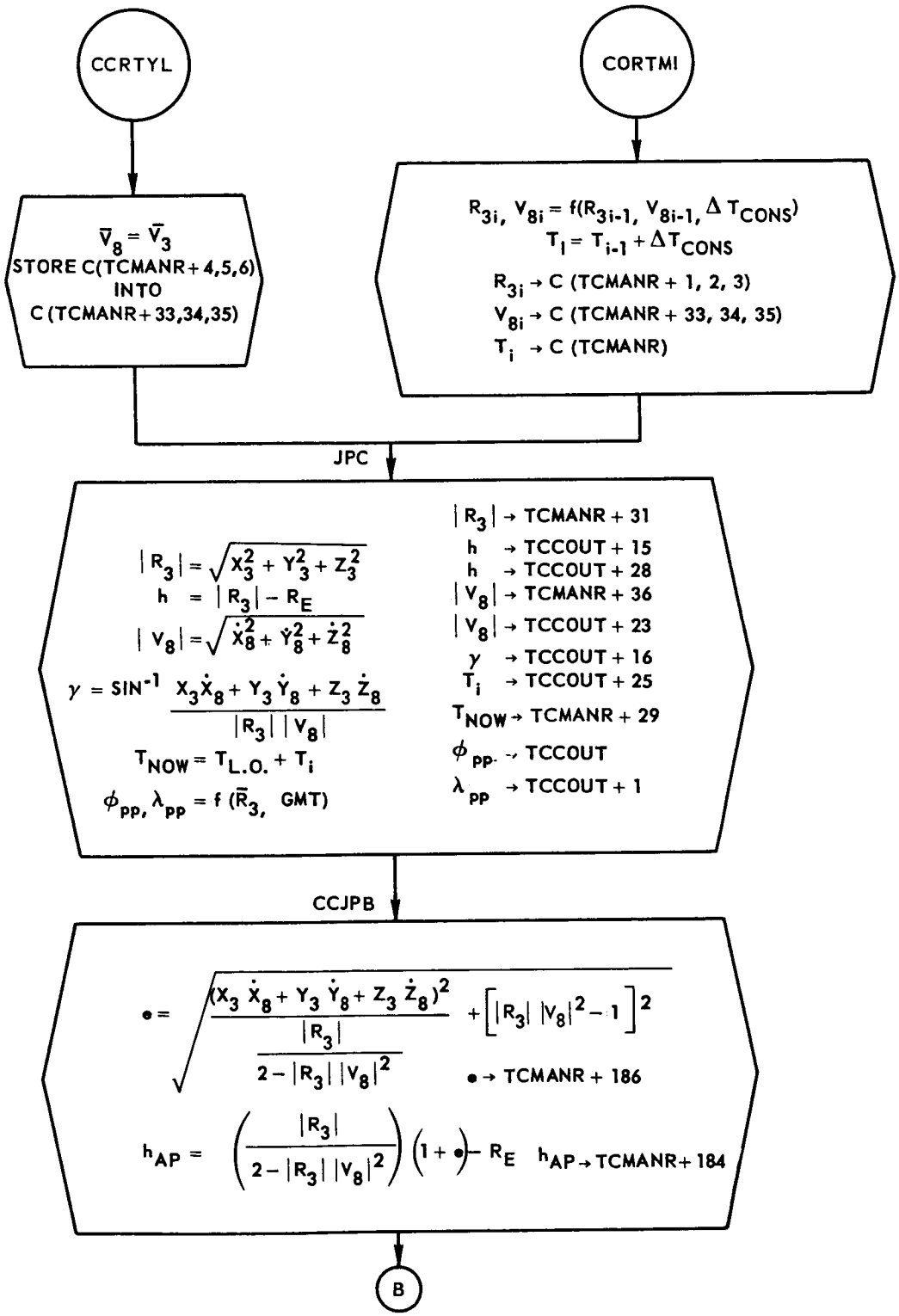


FIGURE 3-46 CCRTYL-CCTRMI PROGRAM FLOW CHART *(Sheet 1 of 2)*

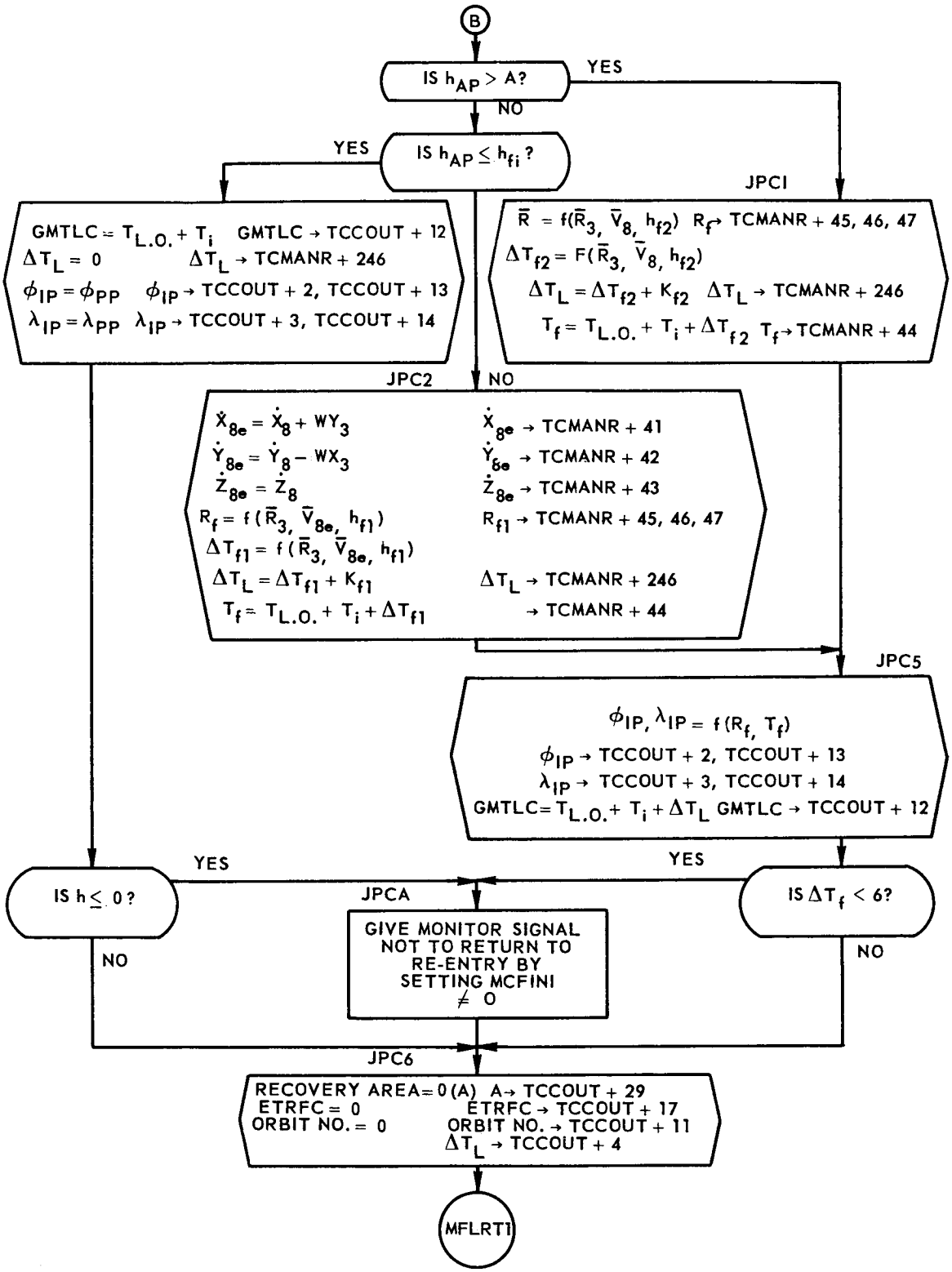


FIGURE 3-46 CCRTYL-CCTRMI PROGRAM FLOW CHART *(Sheet 2 of 2)*

3.21 MEDIUM ABORT PROGRAM (CCMEAB)

The CCMEAB program performs additional tests and computations on the raw or processed AN/FPS-16 radar data as it is received from the coordinate conversion programs. This data is used to generate re-entry tables at predetermined time intervals. The flow chart for CCMEAB is shown in Figure 3-47.

3.21.1 Input Requirements

The following library routines are used by CCMEAB:

U3VMAG - Generate magnitude from vector

U1ASCO - Arc Sine - Arc Cosine

U1SQRT - Square root

Table TCMANR provides CCMEAB with the following data in floating point format:

TCMANR	T_i (Time since lift-off)	
+ 1	X	$\bar{r}$
+ 2	Y	
+ 3	Z	
+ 4	$\dot{X}$	$\bar{v}$
+ 5	$\dot{Y}$	
+ 6	$\dot{Z}$	

The following information located in communication cell MCNOCH also serves as input to CCMEAB:

MCNOCH - equal to zero means data is available to be processed.

- not equal to zero indicates data is not available at this time.

The following input parameters are used by CCMEAB and are located in LCONS:

- $1\ell_o$: base factor by which σ_v cannot exceed σ_{v-1}.
- $2m_o$: base factor by which σ_γ cannot exceed $\sigma_{\gamma-1}$.

- 3K : length of arc (Δt) over which to collect data before determining whether or not to generate a new re-entry table (K is given in half seconds).

- 4 : maximum value of v acceptable = 14,000 ft/sec. in Mercury units.

- 5 : maximum value of γ acceptable = $\pi/3$ radians.

- 6Δl : factor by which I is increased when K time has expired and $\sigma v \not\leq \ell \sigma v-1$.

- 7Δm : factor by which m is increased when K time has passed and $\sigma_\gamma \leq m \sigma_{\gamma -1}$.

If entry is made to CCMEAB after a re-entry table has been generated, the time of completion of the table is stored by Monitor in location TCMEAB + 7 to be used as a base time upon which to determine the time span before another table is needed. If data was such that a re-entry table was not generated by NOCPNI at end of K seconds, then the base time is set internally.

3.21.2 Output Requirements

The output of CCMEAB is stored in table TMRARF. TMRARF - 1 through TMRARF - 5 contains position and velocity components averaged over K seconds and TMRARF - 6 contains averaged time. The format of the TMRARF table is shown below.

TMRARF	X		
+ 1	Y	$\bar{r}$	floating point
+ 2	Z		
+ 3	$\dot{X}$		
+ 4	$\dot{Y}$	$\bar{v}$	
+ 5	$\dot{Z}$		
+ 6	T, fixed point (B 35) seconds GMT time		

The second output of CCMEAB is contained in the following communication cell:

MCWIDR = zero if no new re-entry table is to be generated

= zero indicates to Monitor suffix MFLRT2 to queue Numerical Integration program (NOCPNI) to generate a new re-entry table.

3.21.3 Method

Computation of $|V| = \sqrt{\dot{X}^2 + \dot{Y}^2 + \dot{Z}^2}$

$$|R| = \sqrt{X^2 + Y^2 + Z^2}$$

$$\gamma = \frac{\sin^{-1} \; X\dot{X} + Y\dot{Y} + Z\dot{Z}}{|R| \qquad |V|}$$

Computation of standard deviation:

$$\sigma_V = \sqrt{\frac{\Sigma(|V|^2)}{H} - \left(\frac{\Sigma\,|V|}{H}\right)^2}$$

$$\sigma_\gamma = \sqrt{\frac{\Sigma(\gamma^2)}{H} - \left(\frac{\Sigma\,\gamma}{H}\right)^2}$$

Where H = the number of points accumulated over last K half seconds.

Computation of average vectors:

$$\frac{\Sigma \text{ components}}{\text{Number of components summed}}$$

3.21.4 Usage

The common and the internal table formats are shown below:

Common table Floating Point)

TCMEAB	T
+ 1	X
+ 2	Y
+ 3	Z
+ 4	$\dot{X}$
+ 5	$\dot{Y}$
+ 6	$\dot{Z}$
+ 7	base time
+ 8	base time - K

+ 9	$\lvert V \rvert$
+ 10	γ
+ 11	$\lvert V \rvert^2$
+ 12	γ^2
+ 13	$\lvert R \rvert$
+ 14	$\lvert R \rvert \lvert V \rvert$
+ 15	σ_V and intermediate values
+ 16	σ_γ and intermediate values
+ 17	σ_{V-1} (compiled as 0)
+ 18	$\sigma_{\gamma\ -1}$ (compiled as 0)
+ 19	l
+ 20	m

Internal Table (Floating Point)

LRVTS	:	
		$\Sigma X, \frac{\Sigma X}{H}$
1	:	$\Sigma Y, \frac{\Sigma Y}{H}$
2	:	$\Sigma Z, \frac{\Sigma Z}{H}$
3	:	$\Sigma \dot{X}, \frac{\Sigma \dot{X}}{H}$
4	:	$\Sigma \dot{Y}, \frac{\Sigma \dot{Y}}{H}$
5	:	$\Sigma \dot{Z}, \frac{\Sigma \dot{Z}}{H}$
6	:	$\Sigma T, \frac{\Sigma T}{H}$
7	:	$\Sigma \lvert V \rvert$

+ 8	:	$\Sigma \gamma$
+ 9	:	$\Sigma (\lvert V \rvert^2)$
+ 10	:	$\Sigma (\gamma)$

Entry from -- MPLCCM

Exit to -- MFLRT2

Constants

K001.0	Dec	1.0
KCH233	OCT	233000000000

Storage Required

Instructions - 136

Intermediate tables	-	19 locations
Input table	-	7 locations
Output table	-	7 locations
Common	-	21 locations
Communication cells	-	2

Timing

First (K - 1) 1/2 seconds - minimum average number of 7090 computer cycles - 14, maximum average, 996.

K^{th} 1/2 second - minimum average number of 7090 computer cycles - 64, maximum average - 1405.

Checkout

External system debugging: 50 data sentence cases ($\bar{r}$, $\bar{v}$, and t) including several combinations of indicators to debug all branches.

Complete system debugging: tested in all ways; simulated, B simulator, Ampex tape and live.

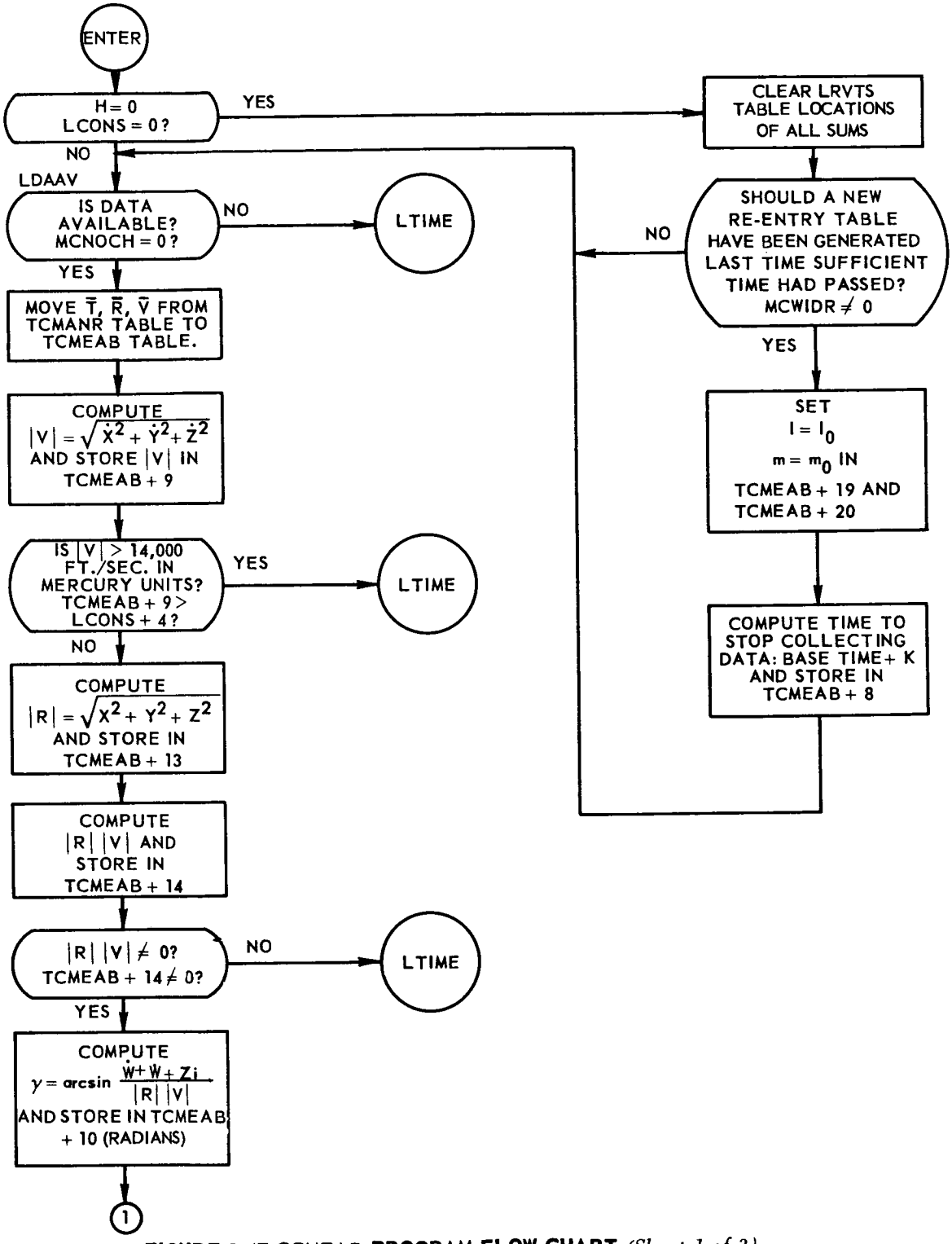


FIGURE 3-47 CCMEAB PROGRAM FLOW CHART *(Sheet 1 of 3)*

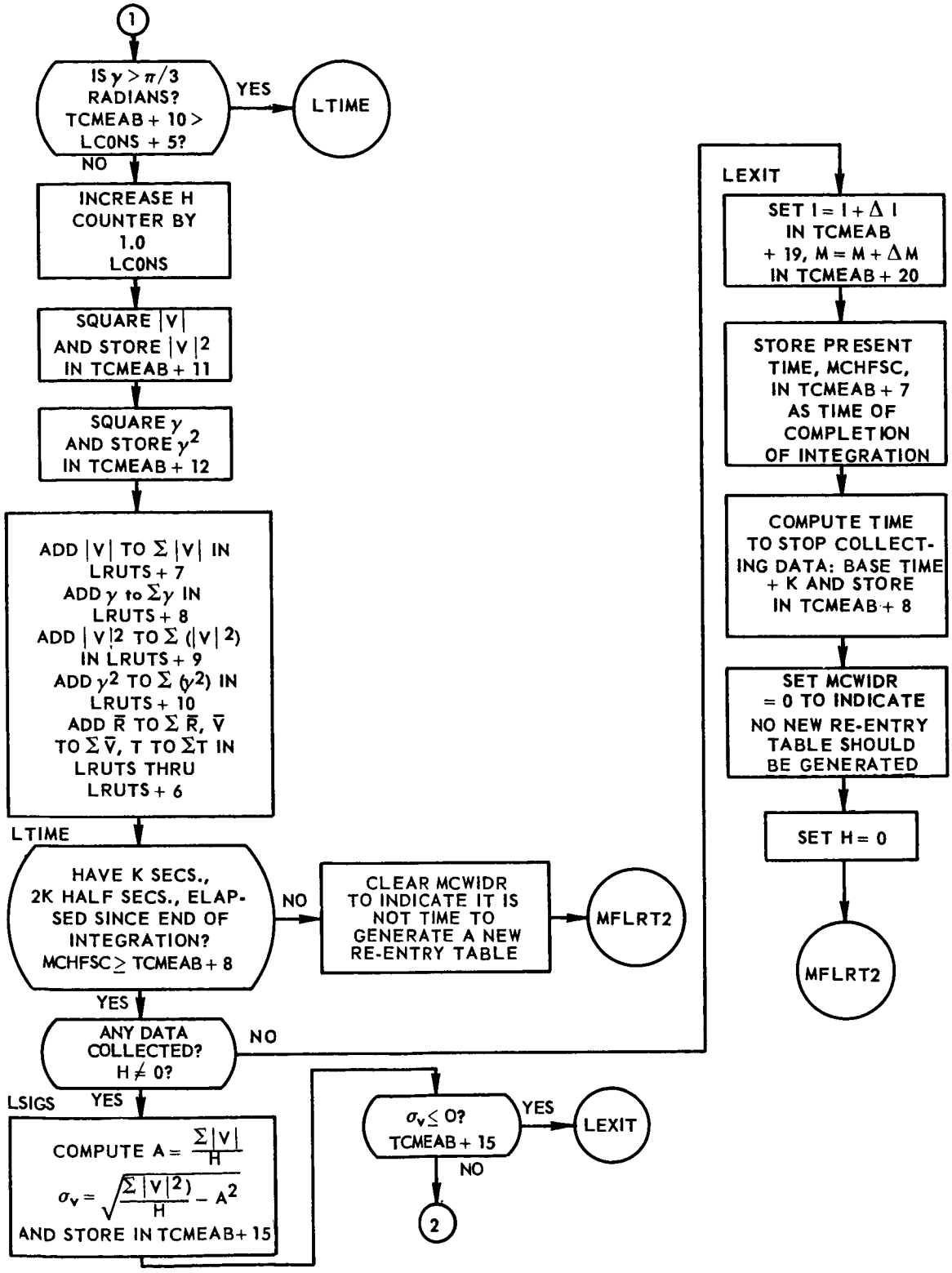


FIGURE 3-47 CCMEAB PROGRAM FLOW CHART *(Sheet 2 of 3)*

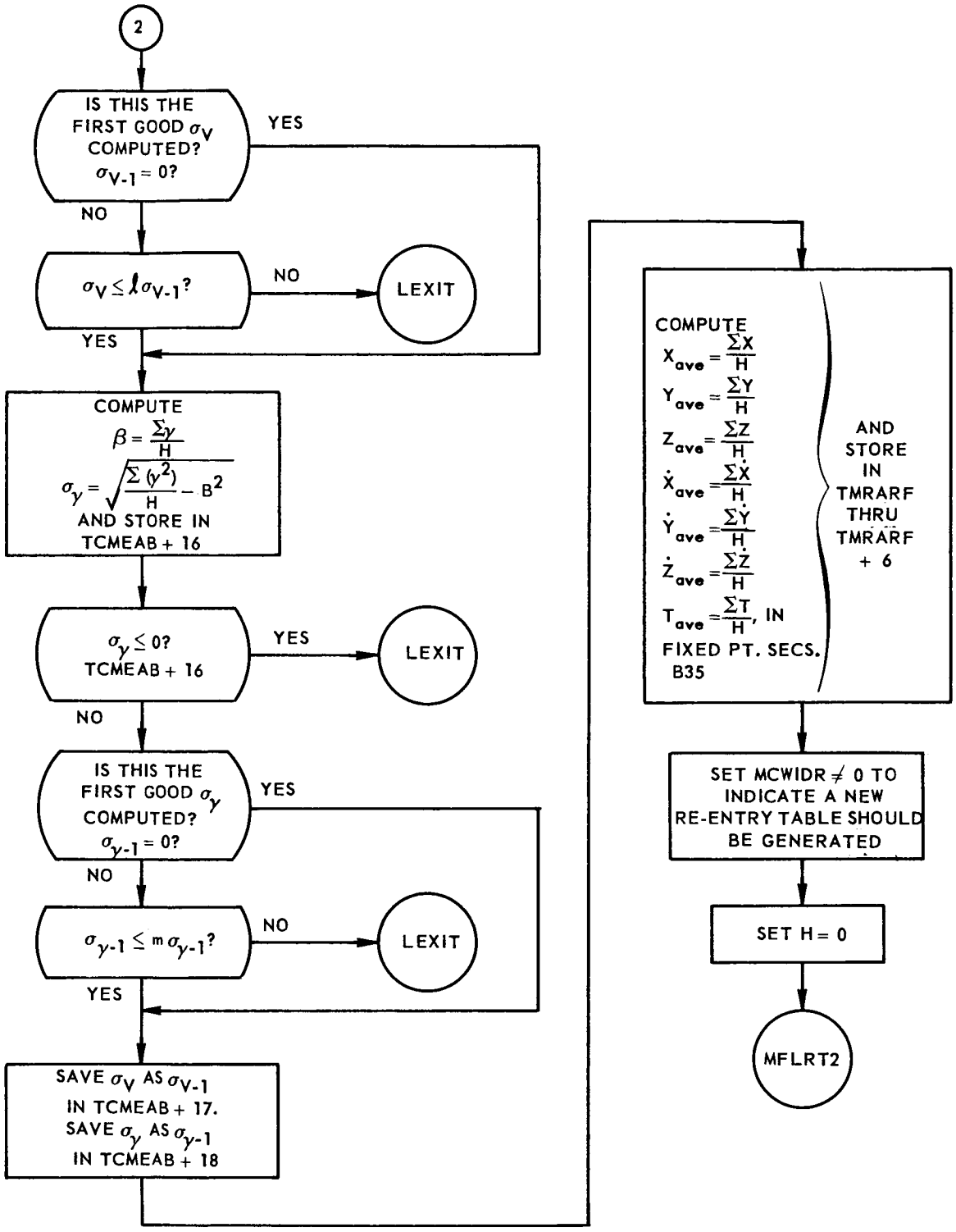


FIGURE 3-47 CCMEAB PROGRAM FLOW CHART *(Sheet 3 of 3)*

3.22 HIGH ABORT PROGRAM (CCABRT)

This program computes the quantities for display at the Mercury Control Center when an abort occurs after tower separation (TOWS). The flow chart for CCABRT is shown in Figure 3-48.

3.22.1 Input Requirements

The following programs and subroutines are used by the CCABRT program:

Mercury Programs

C9RVTH - Computation of elliptic orbit program

CHNOND - Computation of circular orbit program

A3MSCP - Compute latitude and longitude program

Library Routines

U1SQRT - square root

U1ASCO - arc sine/arc cosine

U3DOTP - "dot" product

U1ATNA - arc tangent A

U3VMAG - generate magnitude from vector

U1ATAB - arc tangent (A/B)

The following data serves as input to CCABRT:

a) MCLOCT containing address of integration table to be used (TNINT1 or TININT2).

b) MCHFSC containing GMT half second clock time

c) MCLFTM - GMT time of lift off in floating point seconds.

d) TCCOUT +13 - GMTRC, Greenwich Mean Time to fire retrorockets (minutes in address, seconds in decrement).

e) TNINT2 - Re-entry integration table, generated format specified by N0CPNI.

f) TNINT1 - Orbit integration table if MCLOCT contains address of TNINT1. Generated and format specified by N0CPNI.

3.22.2 Output Requirements

The following data is generated by CCABRT and is contained in the TCCOUT table.

a) TCCOUT + 5 - Geodetic latitude of present position (floating point radians)

b) TCCOUT + 6 - Longitude of present position (floating point radians)

c) TCCOUT + 9 - GTRS Time remaining until retro fire (minutes in address, seconds in decrement).

d) TCCOUT + 10 - GMTLC - GMT of landing computed (minutes in address, seconds in decrement)

e) TCCOUT + 11 - Geodetic latitude of refined impact point (floating point radians)

f) TCCOUT + 12 - Longitude of refined impact point (floating point radians)

g) TCCOUT +18 - GO, NO-GO (integer will always = 2)

h) TCCOUT + 19 - H. Height of capsule above spherical earth (floating point nautical miles)

i) TCCOUT +20 - Gamma. Flight path angle (floating point radians)

j) TCCOUT + 22 - V. Inertial speed (floating point feet per second).

k) TCCOUT + 24 - D. Downrange distance (floating point nautical miles)

l) TCCOUT + 28 - H. Height of capsule above spherical earth (floating point nautical miles)

3.22.3 Other Requirements

a) Both integration tables must be completed before entering CCABRT.

b) GMT clock time must be equal or greater to the first time in integration table specified.

3.22.4 Method

a) All equations in CCABRT are taken from either Strip Chart Processing or the main launch program (CCMAIN).

b) CCABRT first finds $\bar{r}$ and $\bar{v}$ in the integration table preceding the GMT. C9RVTH is then used to extrapolate for $\bar{R}_3$, $\bar{V}_3$ at GMT.

c) The following output quantities are computed:

$$|V_3| = \sqrt{\dot{X}_3^2 + \dot{Y}_3^2 + \dot{Z}_3^2}$$

$$|R_3| = \sqrt{X_3^2 + Y_3^2 + Z_3^2}$$

$$h = |R_3| - R_e$$

d is obtained using the strip chart formulations transforming $\bar{R}_3$ to $\bar{R}_2$.

$$\gamma = \text{ARCSIN} \quad \frac{X_3 \dot{X}_3 + Y_3 \dot{Y}_3 + Z_3 \dot{Z}_3}{|R_3|\,|V_3|}$$

d) ϕ_{pp} and λ_{pp} are computed using A3MSCP

e) The first time CCABRT is entered by Monitor ϕ_{LP} and λ_{LP} are computed using A3MSCP.

3.22.5 Usage

a) There is no calling sequence to the CCABRT program.

b) Space required

392_{10} locations, broken down as follows:

265 program instructions

74 erasable storage

5 non-erasable storage

c) Coding Information

Constants Used:

K00000	DEC	0
KM0001	DEC	-1

K00001	DEC	1
K00002	DEC	2
K00003	DEC	3
K00004	DEC	4
K00006	DEC	6
K00060	DEC	60
K0001.0	DEC	1.
K0002.0	DEC	2.
K0060.	DEC	00060.
KECR.T	DEC	00000.003
KFEEBD	DEC	2.0E-7
K00.SH	DEC	.0012394486
KLAMDO	BSS	16, X
K00.UT	DEC	806.8104
K.OMES	DEC	7.29211508E-5
K.ECC2	DEC	6.6934215E-3
K.INB2	DEC	1.00673852
KCNVRG	DEC	0.000017
KKK2PI	DEC	6.283185307
KCH233	OCT	233000000000
KMNMSK	OCT	000777777777

TCCONS table

Erasable storage locations

CSGMT	CSTID
CSRT	CSR01

CSR02	CSRLP
CSROM	CSRPP
CSSOM	CSRV1
CSCOM	CSROP
CSCLP	CSRR2
CSSLP	CSRT
CSCBT	CSRNV
CSSBT	CSRT1
CSLLR	

Timing required

Approximately 3004 cycles

+ C9RVTH

A3MSCP

The program was checked out in SOS, altering the data into the program, including integration tables TNINT1 and TNINT2, the time MCHFSC, and which integration table to be used (MCLOCT).

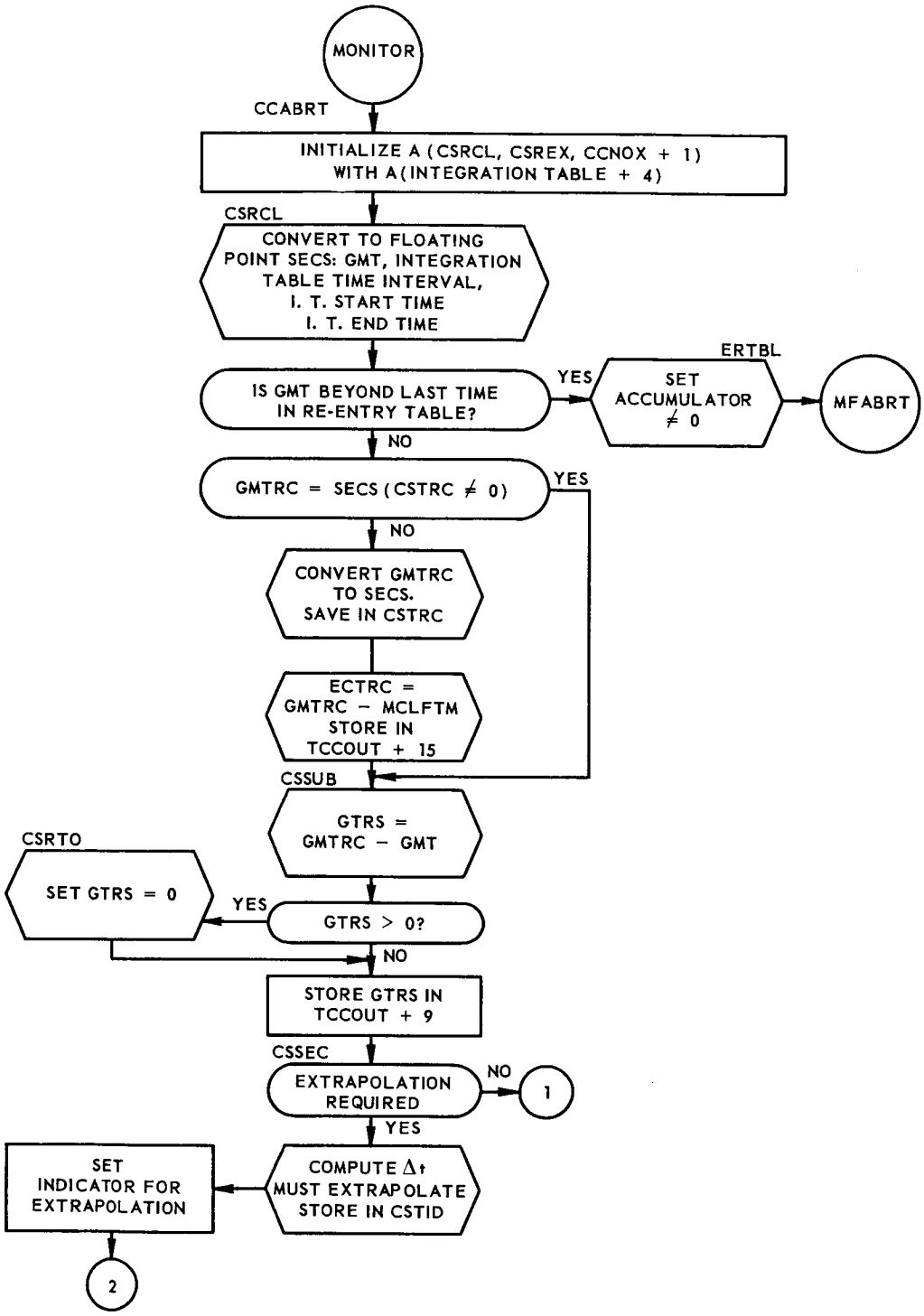


FIGURE 3-48. CCABRT PROGRAM FLOW CHART *(Sheet 1 of 3)*

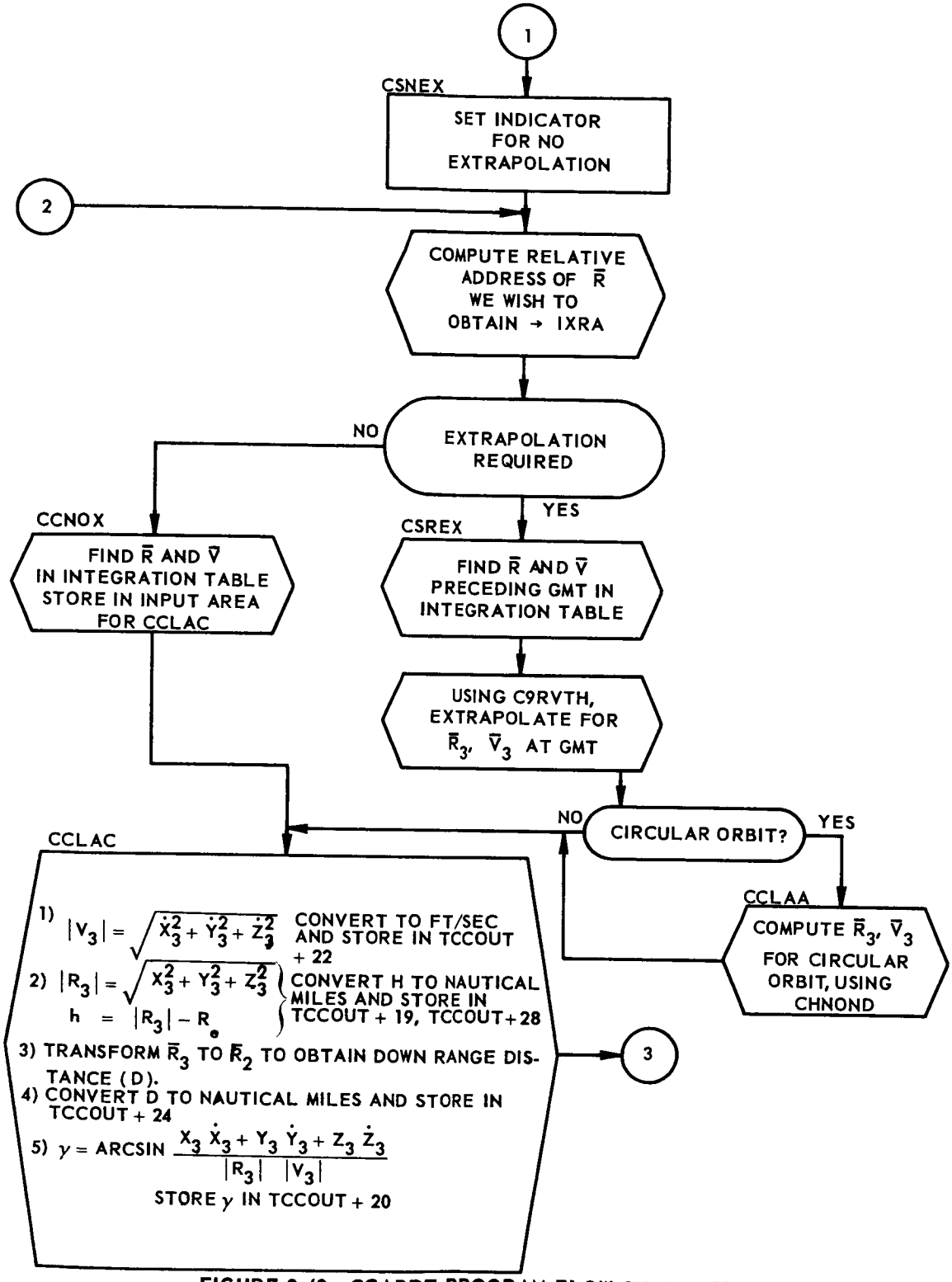


FIGURE 3-48. CCABRT PROGRAM FLOW CHART *(Sheet 2 of 3)*

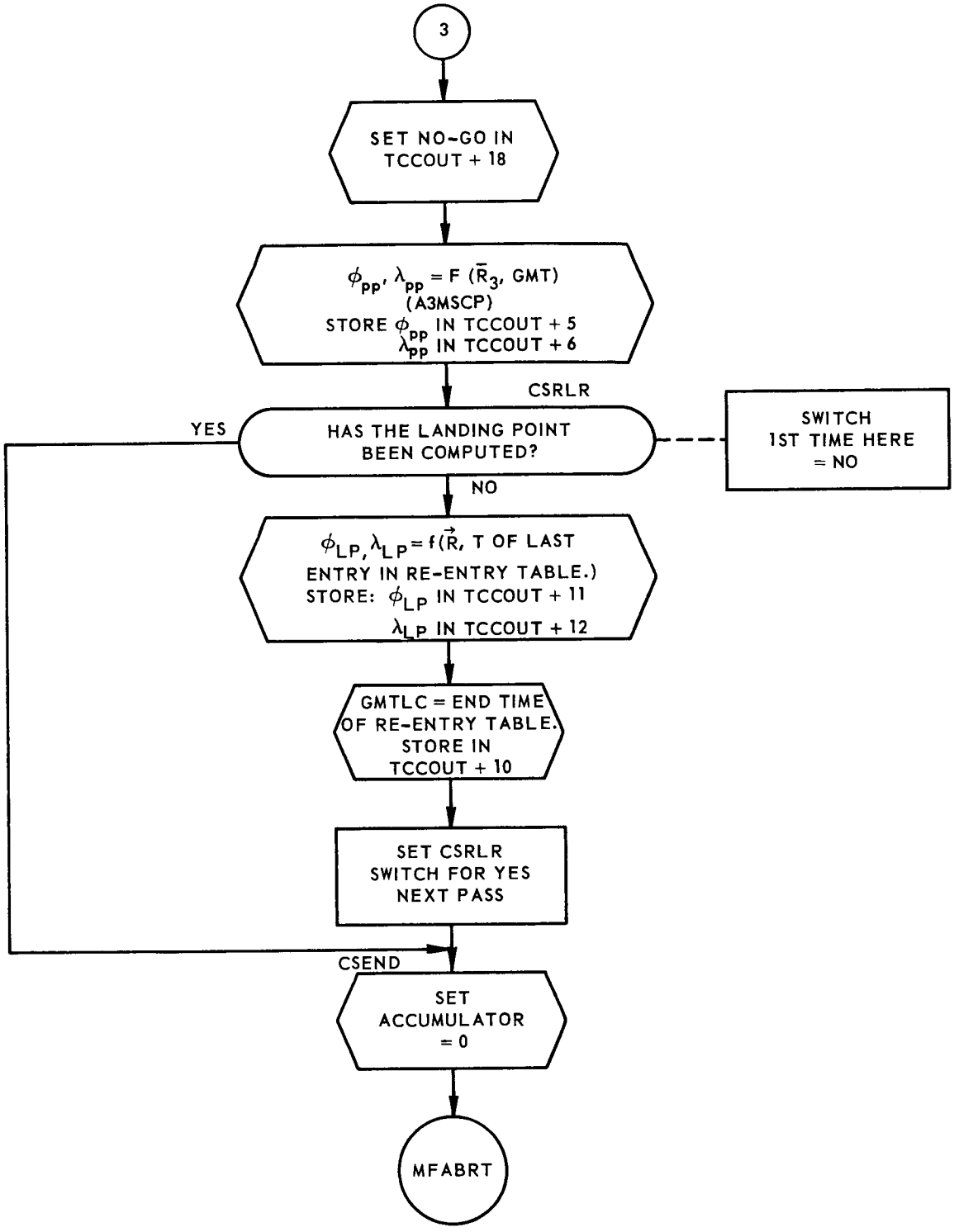


FIGURE 3-48. CCABRT PROGRAM FLOW CHART *(Sheet 3 of 3)*

Section 4

ORBIT AND RE-ENTRY PROCESSORS

When the decision to continue the mission following insertion is made by either the Mercury or Bermuda Control Centers, the IBM 7090 computers pass from launch into orbital phase computing. During this phase the Goddard computers determine the flight's orbital characteristics from data received from the tracking sites and the Mercury Control Center. In addition, input data from the sites is processed and retransmitted back to them as capsule acquisition data, and to the Mercury Control Center for plotboard presentation.

The re-entry phase begins just prior to the time of capsule retrorocket fire and continues until impact. During this time the Goddard computers accept input data from the tracking sites and from the Mercury Control Center, process it, determine the time to fire retrorockets, and retransmit this calculated time to all tracking sites. Refined impact points are also computed and sent with retrofiring times. The general flow diagram for the orbit and re-entry phases is shown in Figure 4-1.

The Goddard 7090's are called upon to perform many orbital and re-entry flight computations in addition to those mentioned above. This section contains all the processor programs used by the Goddard computers during orbit and re-entry.

4.1 EDIT PROGRAM (E0LED1)

During orbit a network of tracking sites located around the world monitors the capsule's position and the status and performance of its human occupant. Summaries of this data are sent by teletype to Goddard. The data is processed by the input program (I0TTIN) and only the positional data is retained in memory until such time as Monitor instructs the E0LED1 program to edit the information.

The E0LED1 Program:

a) Deletes radar messages having erroneous times and slant ranges.

b) Finds the time of closest approach (t_{ca}) to each site.

c) Replaces erroneous azimuth and elevation angles by the fixed point interger +1.

The flow chart for the E0LED1 Program is shown in Figure 4-2.

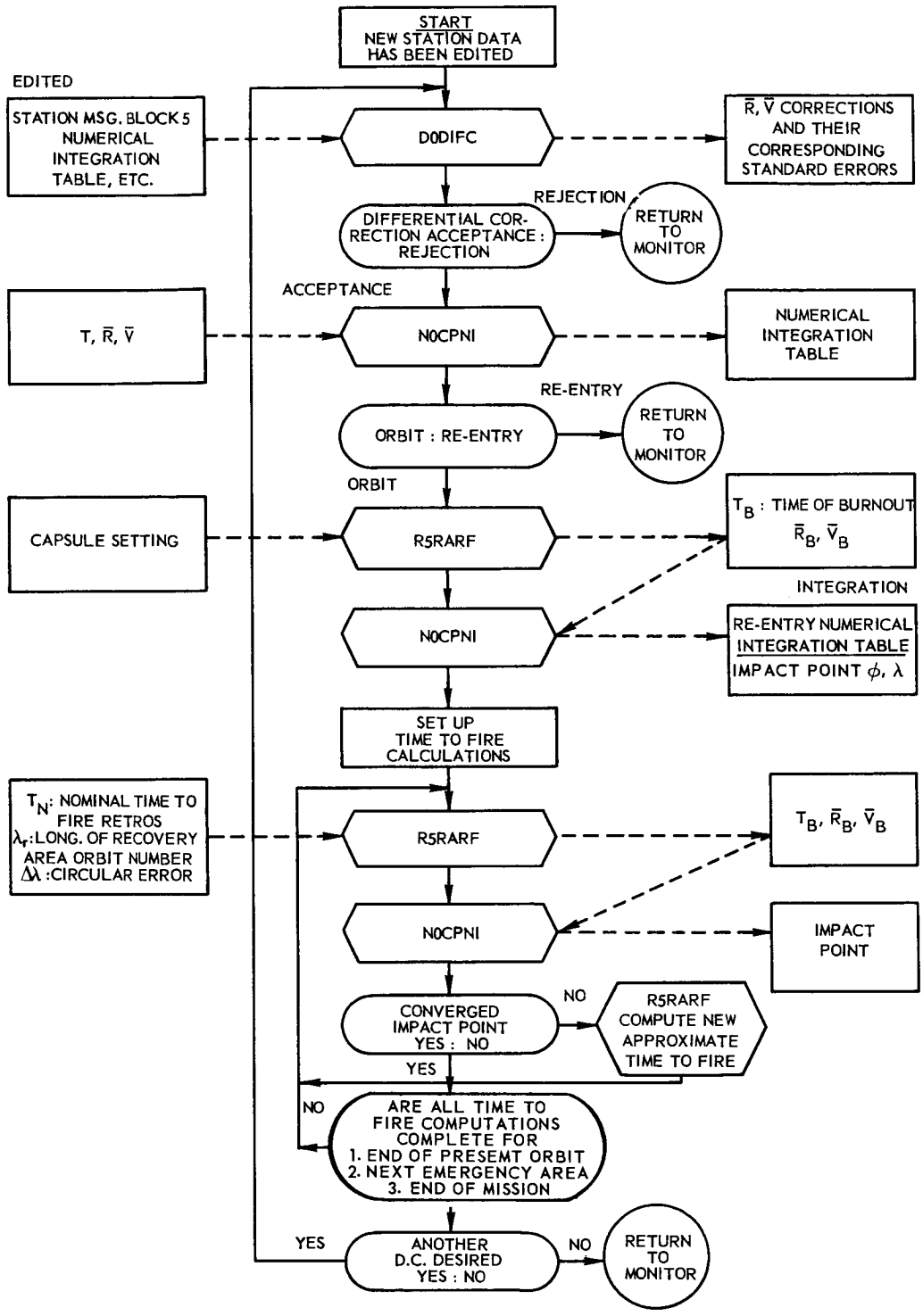


FIGURE 4-1. ORBIT AND RE-ENTRY PHASE GENERAL BLOCK DIAGRAM

4.1.1 Input Requirements

E0LED1 receives its input from:

a) Location MCLED1, whose address field contains the internal station number, j.

b) Table TMSTCH - The address field of location TMSTCH + j contains the first location of the station characteristics block of the j -th station minus 1.

c) A 19-location table, TMLSDB. The address field of location, TMLSDB +j - 1 contains the first location of the data block of station j, TMRMES +203 (j - 1), to be edited.

The data block is arranged in the following format:

TMRMES + 203 (j - 1)	Station number,, pass number (both supplied by Monitor).
+ 1	Number of radar messages in this block, N (≥ 0).
+ 2	Blank.
+ 3	Time t_1, fixed point (min,, sec).
+ 4	Range R_1, floating point, Mercury unit.
+ 5	Azimuth A_1, floating point, radians.
+ 6	Elevation E_1, floating point, radians.
+ 7	t_2 second radar message.
+ 8	R_2 second radar message.
+ 9	A_2 second radar message.
+10	E_2 second radar message.
. . . . .	etc., up to 50 messages.

4.1.2 Output Requirements

The output of the E0LED1 program is the same as the input data block and containing the edited messages in the following format:

TMRMES + 203 (j - 1)	Station number,, pass number.
+ 1	Number of radar messages retained after editing, $\bar{N}$ ($\leq$ N).
+ 2	Location, relative to TMRMES + 203 (j - 1), of t_{ca}.
+ 3	t_1, fixed point (min, , sec).
+ 4	R_1, floating point, Mercury unit.
+ 5	A_1, floating point, radians, or fixed point +1.
+ 6	E_1, floating point, radians, or fixed point + 1.
. . . . .	etc.
	An entire message is deleted if the given time and/or range is erroneous.

4.1.3 Method

Time Sequence Check (Figure 4-2, Sheets 4-7): Divide the time data into sequences of consecutive time data, each sequence satisfying the following conditions.

C-1—The sequence is increasing monotonely.

C-2—The increment between any two elements of the sequence is a nonzero multiple of six seconds.

C-3—The span of the sequence (increment between the first and the last elements) does not exceed seven minutes.

The longest of these sequences is accepted as valid time data.

Each of those elements not in the accepted sequence is then examined. If the accepted sequence is extended to include this element and the extended sequence still meets the conditions C-1, 2, and 3 (above), the element is accepted as valid. Otherwise, the radar message containing it is deleted.

If the total number of radar messages accepted by this test is less than nine, all range, azimuth and elevation (RAE) data is accepted without further tests and t_{ca} is taken to be the time associated with minimum R. Otherwise, the RAE data is subjected to the smoothness check described below.

Smoothness Check—Range Data: By applying the curve fitting method (see below) to points (t_i, R_i), find and delete those radar messages having range values not within 2.5σ of the curve(s) fitted, thus resulting in $\overline{N}$ radar messages.

Time of Closest Approach: The time of closest approach, t_{ca}, is the time of observation associated with the smallest value of the curve(s) fitted to ranges. Denote by $\overline{N}_1$ the number of points up to and including the point at t_{ca}, and by $\overline{N}_2$ the number of remaining points ($\overline{N}_2 = \overline{N} - \overline{N}_1$).

Smoothness Check—Elevation Data: Using the same curve fitting method to (t_i, E_i) $(i = 1, 2, \ldots, \overline{N}_1)$, and to (t_i, E_i) $(i = \overline{N}_1 + 1, \ldots, \overline{N})$, separately, find and tag those radar messages having elevation values not within the bound of tolerance.

Smoothness Check—Azimuth Data (Figure 4-2, sheets 8 and 9): The points $(t_i\ A_i)$ are fitted and tagged in the same manner. However, if the sequence of azimuth values increases through 2π to values less than π (when the capsule passes north of the station), then 2π is added to each of these latter values for curve fitting purposes and is substracted when the fitting is completed. If, in any radar message, either A or E is tagged, replace both by a fixed point integer +1.

General Curve Fitting Procedure (Subroutine E0LSF) (Figure 4-2, sheets 4-7): Erroneous RAE observations are determined by applying the following curve fitting procedure to the n points (X_i, Y_i), where X_i denotes the entries of the sequence of valid time data (t_i), Y_i denotes the entries of the corresponding sequence of slant range (R_i), azimuth (A_i) or elevation (E_i) observations, and $n = \overline{N}, \overline{N}_1$ or $\overline{N}_2$.

a) Using the method of least squares, a fourth-order curve is first fitted to the n points (X_i, Y_i). The points lying on this curve, Z_i, the squares of the residuals, d_i^2, and the variance, σ^2, are computed.

b) The quantities $p = \left[\frac{n}{3}\right]$ and $q=n-3p$ are next computed. If $p < 6$, i.e., $n < 18$, those values of Y_i for which $d_i^2 > (\lambda\sigma)^2$ are tagged by a minus sign, and

no additional curves are fitted to these points. If, however, $p \geq 6$, i.e., $n \geq 18$, the sums $S_k = \sum_{3(k-1)+1}^{3(k+1)} d_i^2$, $k = 1, 2, \ldots, p-2$, $S_{p-1} = \sum_{3(p-2)+1}^{n} d_i^2$ are computed and the maximum determined. The group of six consecutive points producing the largest sum determines a subdivision of the n points of the sequence into two subsequences of n_1 and n_2 points, respectively, having V consecutive points in common ($6 \leq v \leq 9$).

c) A fourth order least squares curve is next fitted to the n_1 points of the first subsequence, and the values Z_{1i}, d_{1i}^2 and σ_1^2 are computed. If $\sigma_1^2 < \sigma^2$, the values Z_{1i} and d_{1i}^2 replace the corresponding values of Z_i and d_i^2. If $\sigma_1^2 \geq \sigma^2$, the values of Z_i and d_i^2 are retained and σ_1^2 is set equal to σ^2. The method of b) is then applied to this subsequence. If $p_1 \geq 6$, a final subdivision of the n_1 points into two overlapping subsequences of n_{11} and n_{12} points, respectively, is made. Each of these subsequences is then fitted with a fourth order least square curve; as above, the values of Y_i in each span are tagged whenever the accepted value of the squared residual exceeds $(\lambda \sigma_m)^2$, where, in each case, σ_m is the smaller of σ_1 and the newly computed variance. If $p_1 < 6$, the points Y_i, $i = 1, 2 \ldots n_1$, are tagged using the value $(\lambda \sigma_1)^2$ as a threshold without further fitting.

d) The determination and tagging of error points among the n_2 points of the second major subsequence is accomplished in an entirely analogous manner. Fourth-order least squares curves are fitted to the sequence of points (X_i, Y_i) $(i = n_1 - v + 1, \ldots n)$ and, if $p_2 \geq 6$, to the n_{21} and n_{22} points of the determined subsequences. Tagging of erroneous Y_i entries is again based on the residuals and the variance of the curve displaying the best fit in each span.

e) After the maximum number of curves has been fitted according to the procedure outlined above, the final tagging of Y_i $(i = 1, 2, \ldots n)$ is

determined. The tag of any value of Y_i which has been fitted by only one curve is the tag determined by that curve. However, any value of Y_i which has been fitted by more than one curve is tagged <u>if and only if</u> it has been tagged in each of these fittings.

f) The value currently assigned to the parameter λ in each of the above tagging processes is 2.5.

<u>Fourth-Order Least Squares Fit (Subroutine LSP4E)</u>:

a) The coefficients c_j ($j = 0, 1, 2, 3, 4$) of the fourth-order least squares polynomial, $y = \sum_{j=0}^{4} c_j x^j$, best fitting the sequence of k points (X_i, Y_i) ($i = 1, 2, \ldots k$) are determined by solving the following system of simultaneous equations:

$$c_0 a_{11} + c_1 a_{12} + c_2 a_{13} + c_3 a_{14} + c_4 a_{15} = b_{11}$$

$$c_0 a_{21} + c_1 a_{22} + c_2 a_{23} + c_3 a_{24} + c_4 a_{25} = b_{21}$$

$$c_0 a_{31} + c_1 a_{32} + c_2 a_{33} + c_3 a_{34} + c_4 a_{35} = b_{31}$$

$$c_0 a_{41} + c_1 a_{42} + c_2 a_{43} + c_3 a_{44} + c_4 a_{45} = b_{41}$$

$$c_0 a_{51} + c_1 a_{52} + c_2 a_{53} + c_3 a_{54} + c_4 a_{55} = b_{51}$$

where $a_{rc} = \sum_{i=1}^{k} X_i^{r+c-2}$ and $b_{r1} = \sum_{i=1}^{k} (Y_i X_i^{r-1})$; $r, c = 1,2,3,4,5$.

b) To improve the accuracy of this solution, the X and Y coordinates are transformed as follows:

$X_i^1 = (X_i - X_t) / 6$, where X_t is that value of X_i, $i = 1, 2, \ldots k$, making $|X_i - (X_1 + X_k) / 2|$ a minimum. $Y_i^1 = Y_i - Y_t$, where Y_t is the mean of the Y_i, $i = 1, 2, \ldots k$.

c) The system of a) above, is solved by applying Cramer's Rule. Each of the determinants, $|V_{ij}|$, in this solution is evaluated by reducing it to the following form:

$$
|V_{ij}| = \begin{vmatrix} v_{11}^{(4)} & v_{12}^{(4)} & v_{13}^{(3)} & v_{14}^{(2)} & v_{15}^{(1)} \\ 0 & v_{22}^{(3)} & v_{23}^{(3)} & v_{24}^{(2)} & v_{25}^{(1)} \\ 0 & 0 & v_{33}^{(2)} & v_{34}^{(2)} & v_{35}^{(1)} \\ 0 & 0 & 0 & v_{44}^{(1)} & v_{45}^{(1)} \\ 0 & 0 & 0 & 0 & v_{55} \end{vmatrix} = v_{11}^{(4)} \, v_{22}^{(3)} \, v_{33}^{(2)} \, v_{44}^{(1)} \, v_{55}
$$

where:

$$v_{i5}^{(1)} = v_{i5} / v_{55} \quad (i = 1, 2, 3, 4)$$

$$v_{ij}^{(1)} = v_{ij} - (v_{i5} / v_{55}) \, v_{5j} \quad (i = 1, 2, 3, 4, 5; j = 1, 2, 3, 4)$$

$$v_{i4}^{(2)} = v_{i4}^{(1)} / v_{44}^{(1)} \quad (i = 1, 2, 3)$$

$$v_{ij}^{(2)} = v_{ij}^{(1)} - (v_{i4}^{(1)} / v_{44}^{(1)}) \, v_{4j}^{(1)} \quad (i = 1, 2, 3, 4; j = 1, 2, 3)$$

$$v_{i3}^{(3)} = v_{i3}^{(2)} / v_{33}^{(2)} \quad (i = 1, 2)$$

$$v_{ij}^{(3)} = v_{ij}^{(2)} - (v_{i3}^{(2)} / v_{33}^{(2)}) \, v_{3j}^{(2)} \quad (i = 1, 2, 3; j = 1, 2)$$

$$v_{12}^{(4)} = v_{12}^{(3)} / v_{22}^{(3)}$$

$$v_{i1}^{(4)} = v_{i1}^{(3)} - (v_{i2}^{(3)} / v_{22}^{(3)})\, v_{21}^{(3)} \quad (i = 1, 2)$$

d) Finally, the points lying on the curve

$$Z_i = Z_i^1 + Y_t = \sum_{j=0}^{4} C_j \left(X_i^1\right)^j + Y_t, \quad i = 1, 2, \ldots k,$$

the squares of the residuals

$$d_i^2 = (Y_i - Z_i)^2, \qquad i = 1, 2, \ldots k,$$

and

$$\sigma^2 = \left[1/(k-5)\right] \sum_{i=1}^{k} d_i^2$$

are computed.

4.1.4 Usage

a) Space Required:

Instructions	-	1244
Erasable	-	703
Constants	-	21
Total		1968 locations

b) Time Required: The computer running time required by E0LED1 is a function of the quantity and quality of the input.

4.1.5 Checkout

The E0LED1 program has been operating in the Mercury Program System since December, 1960.

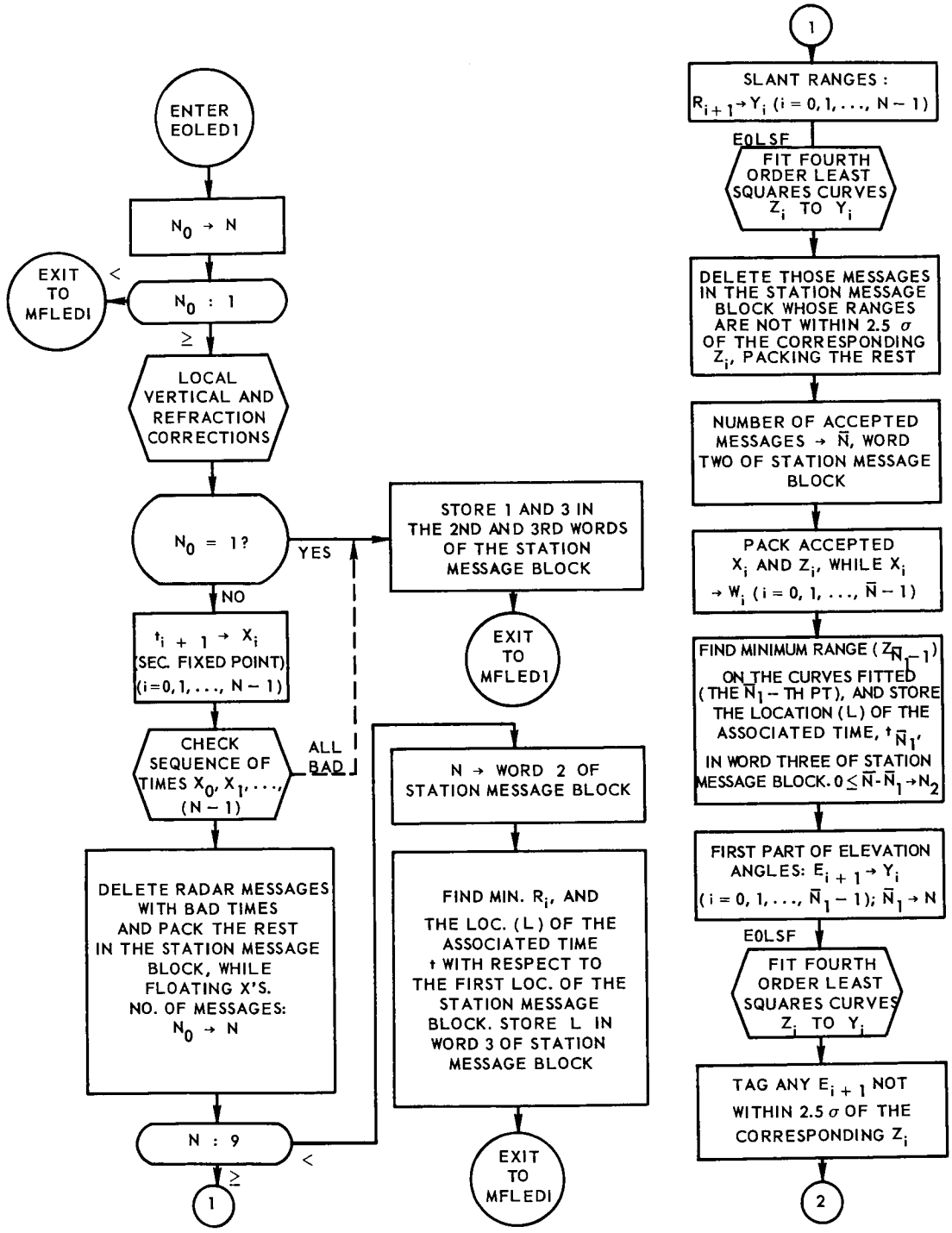


FIGURE 4-2. EOLED1 PROGRAM FLOW CHART *(Sheet 1 of 9)*

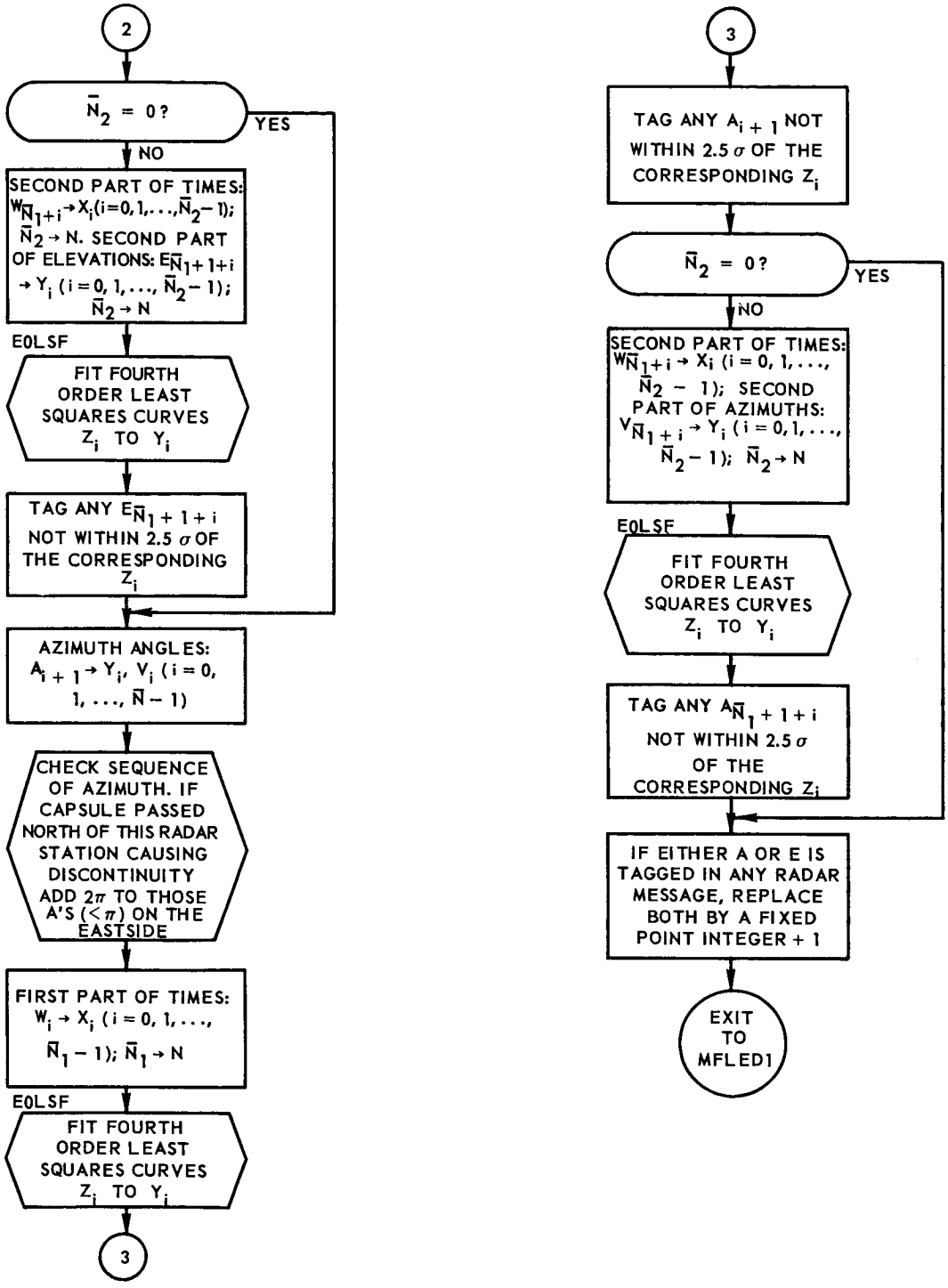


FIGURE 4-2. EOLED1 PROGRAM FLOW CHART *(Sheet 2 of 9)*

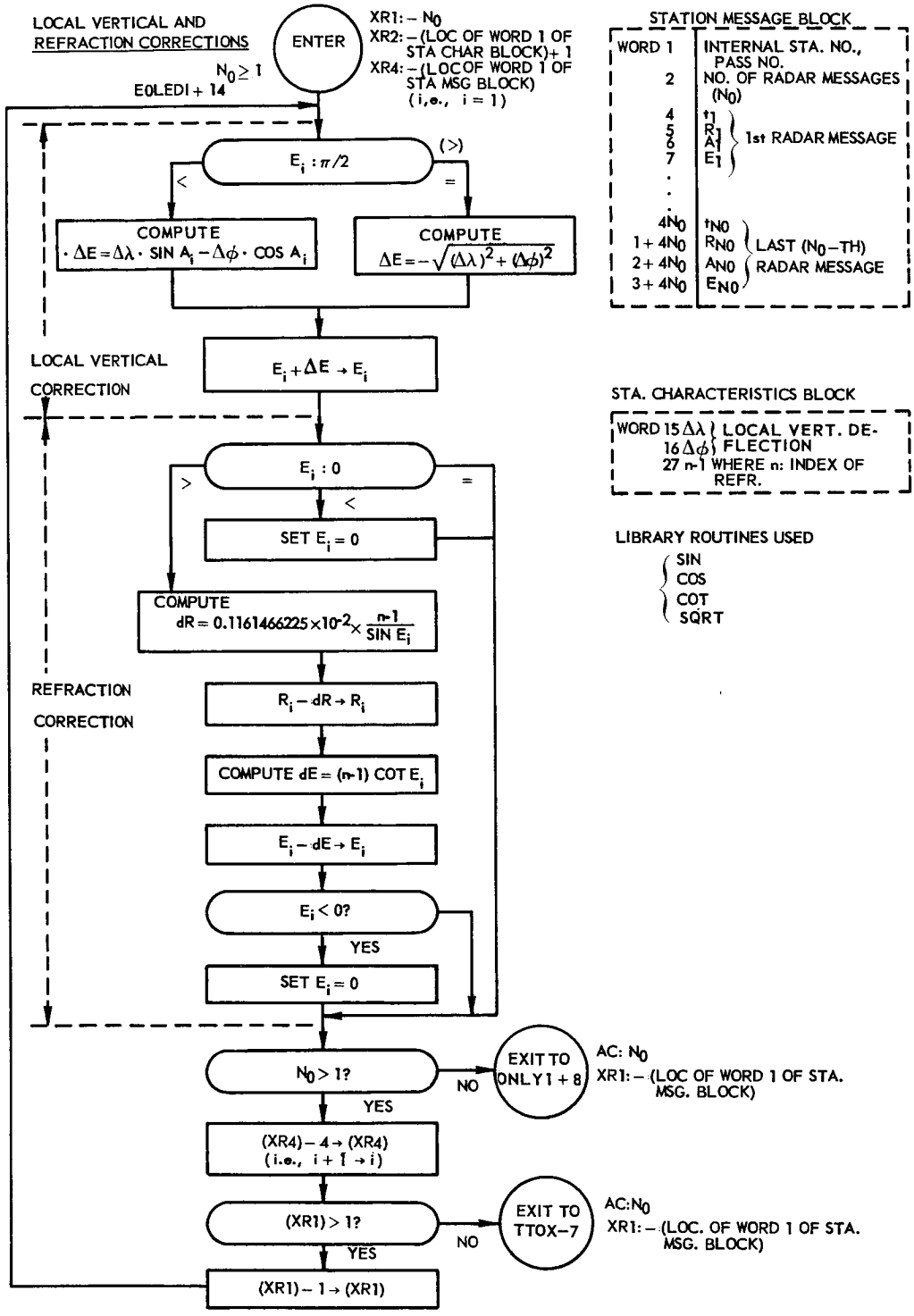


FIGURE 4-2. EOLED1 PROGRAM FLOW CHART *(Sheet 3 of 9)*

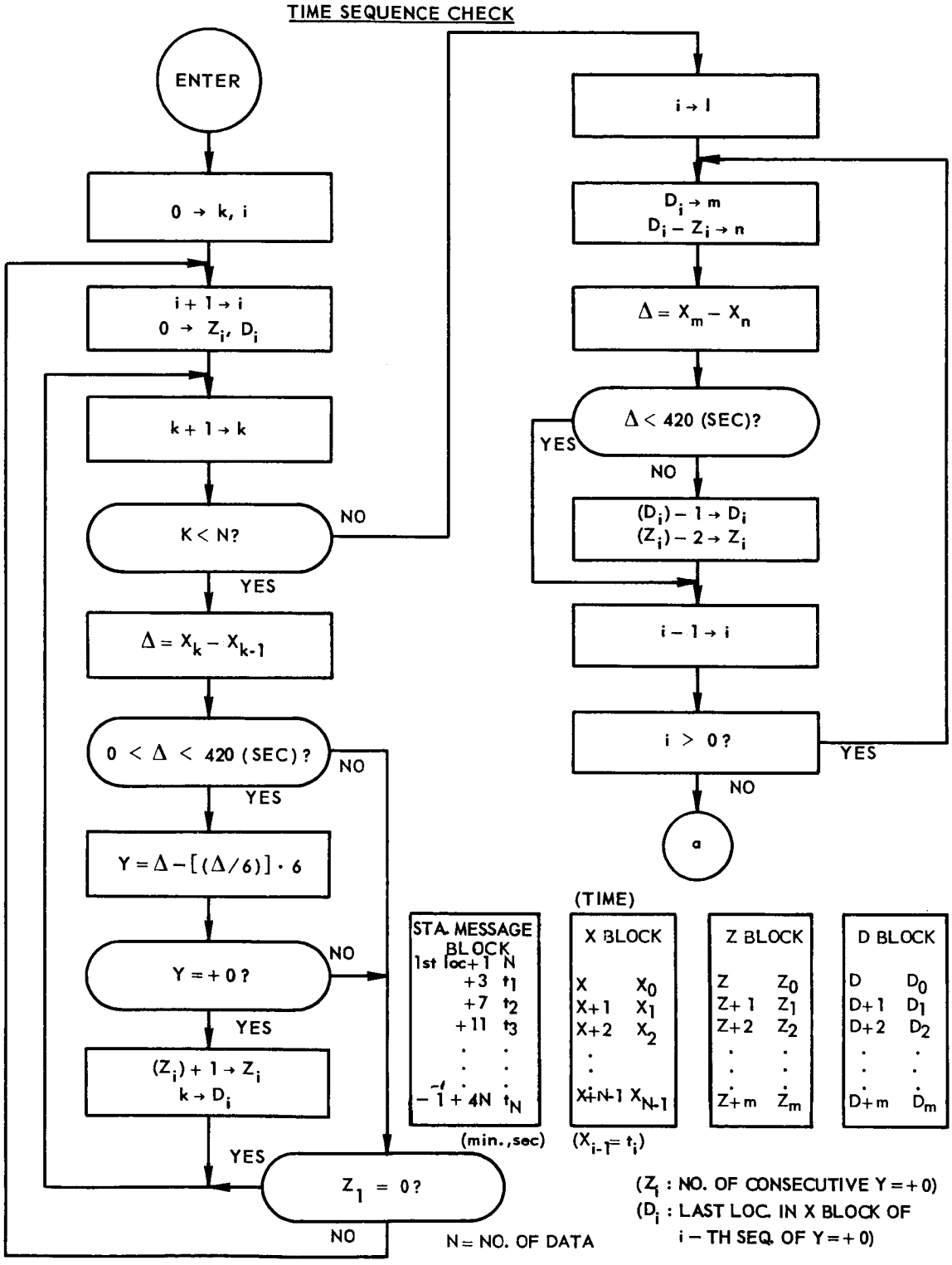


FIGURE 4-2 E0LED1 PROGRAM FLOW CHART *(Sheet 4 of 9)*

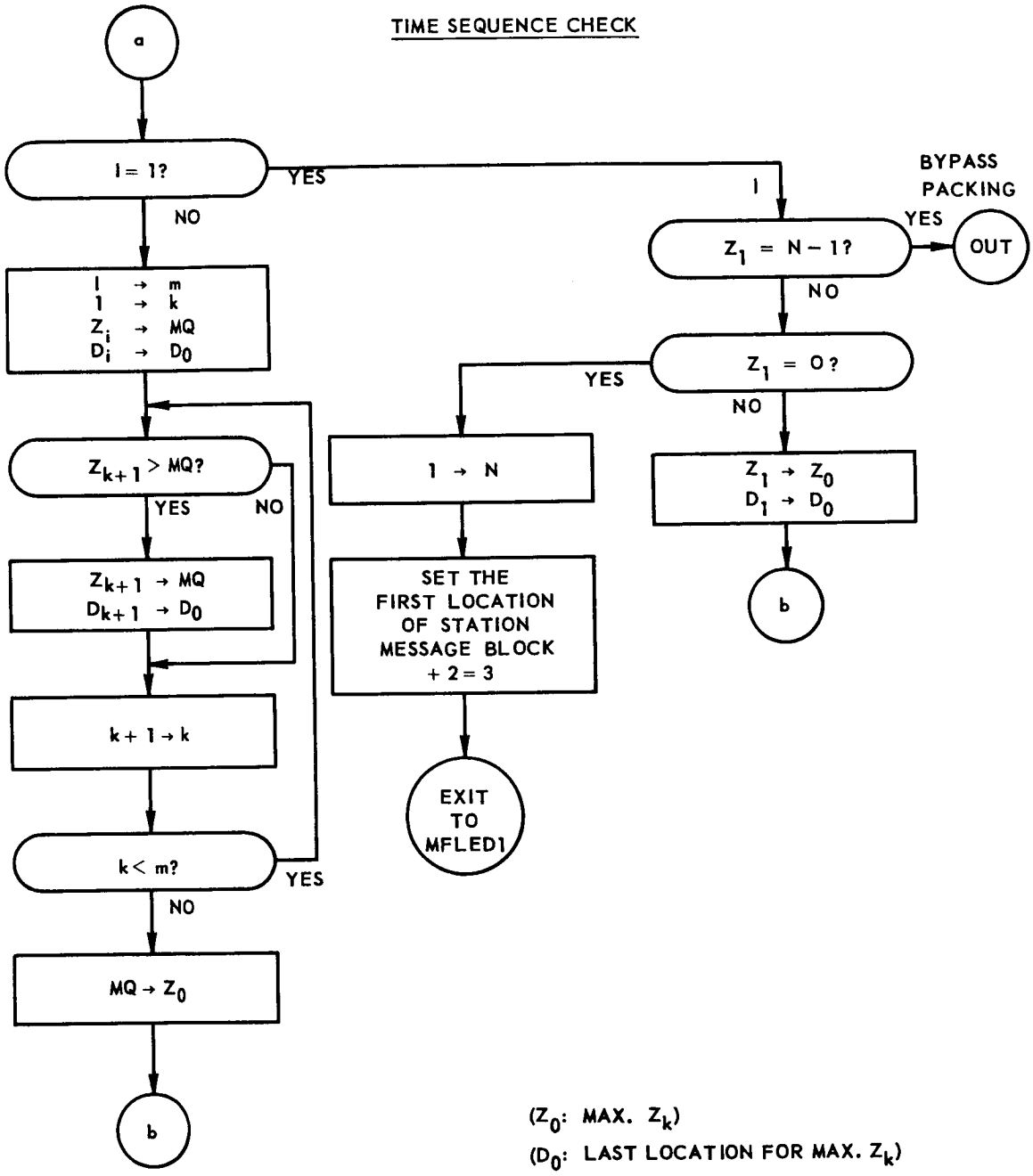


FIGURE 4-2 EOLED1 PROGRAM FLOW CHART *(Sheet 5 of 9)*

TIME SEQUENCE CHECK

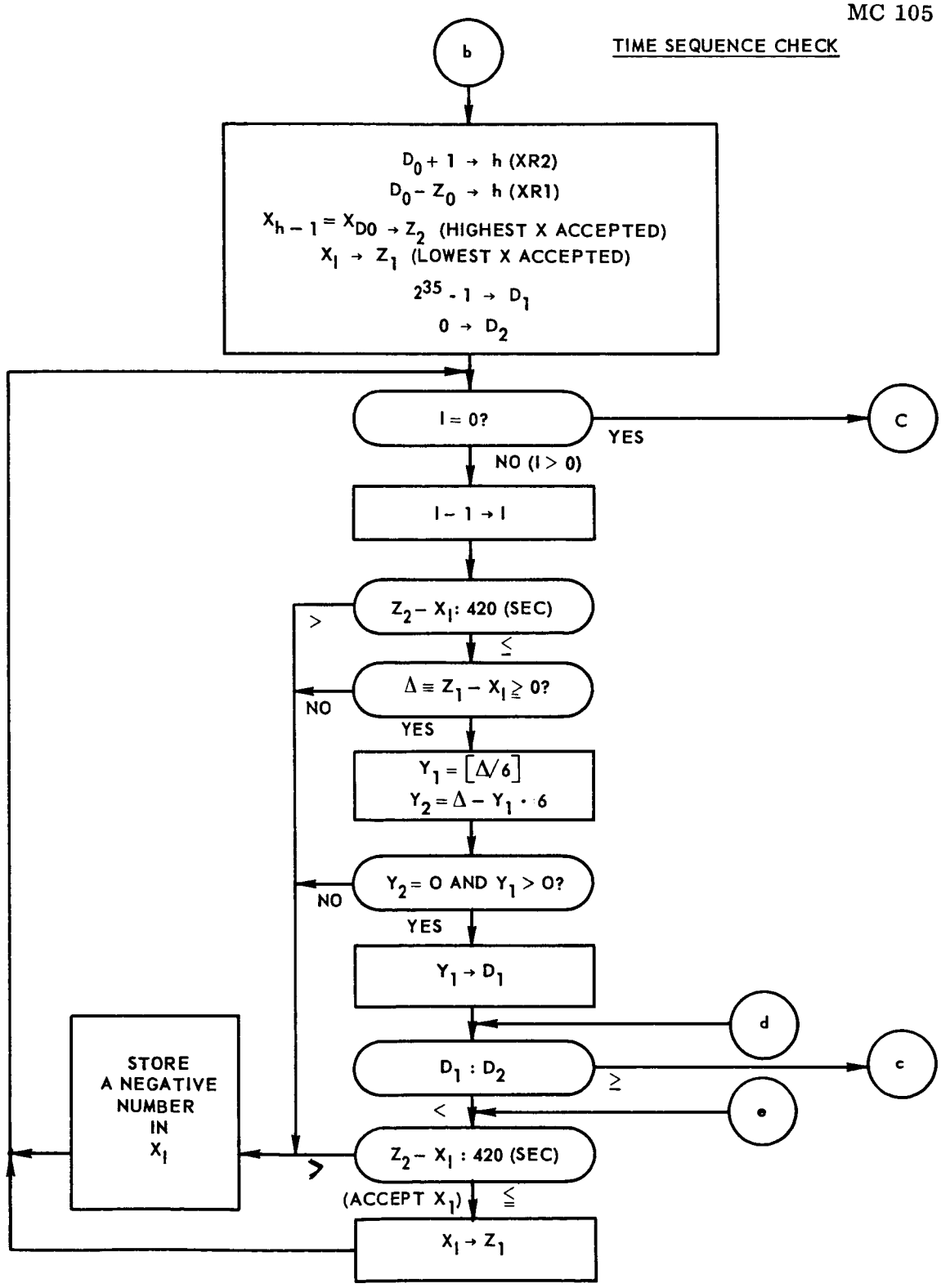


FIGURE 4-2 EOLED1 PROGRAM FLOW CHART *(Sheet 6 of 9)*

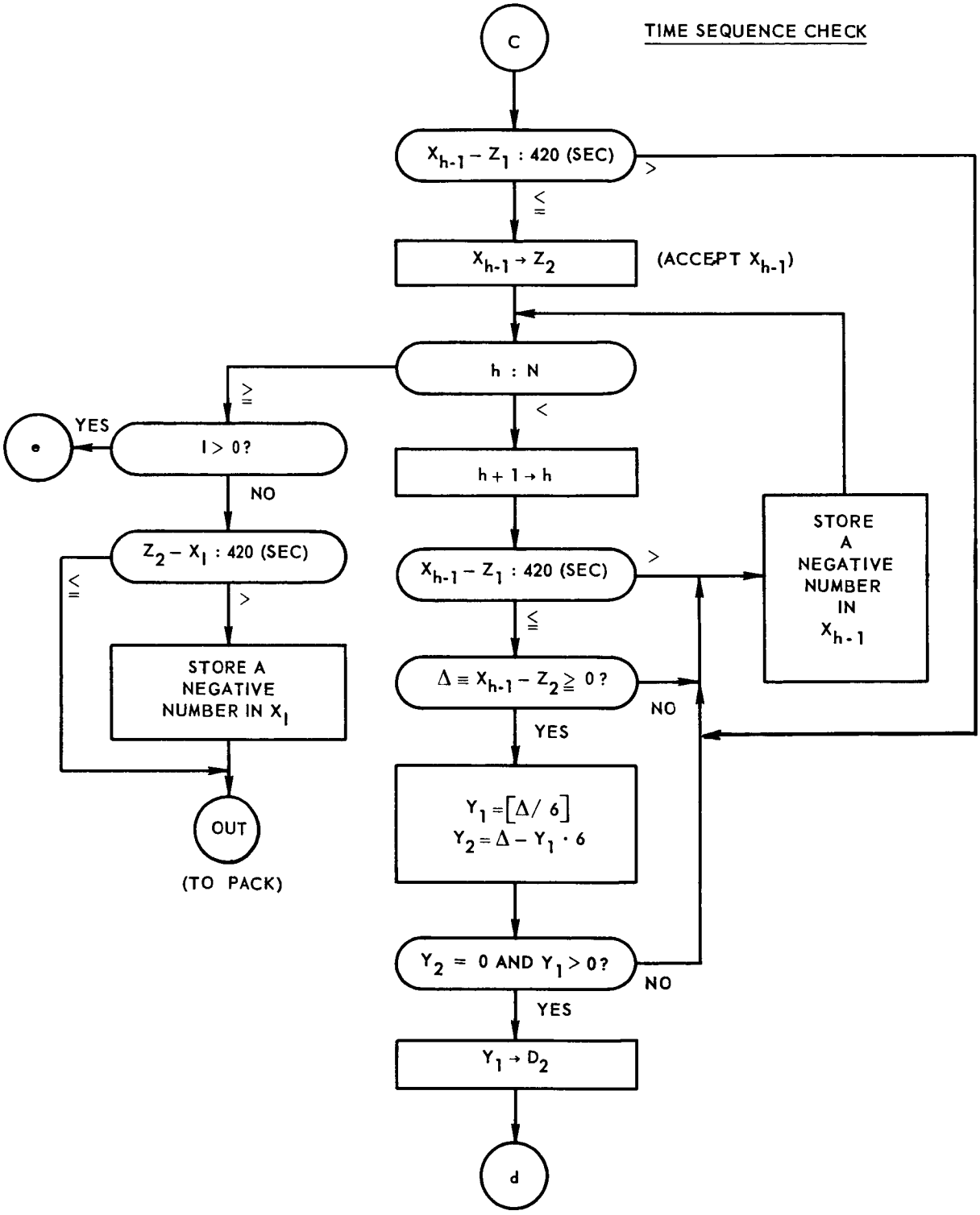


FIGURE 4-2 EOLED1 PROGRAM FLOW CHART *(Sheet 7 of 9)*

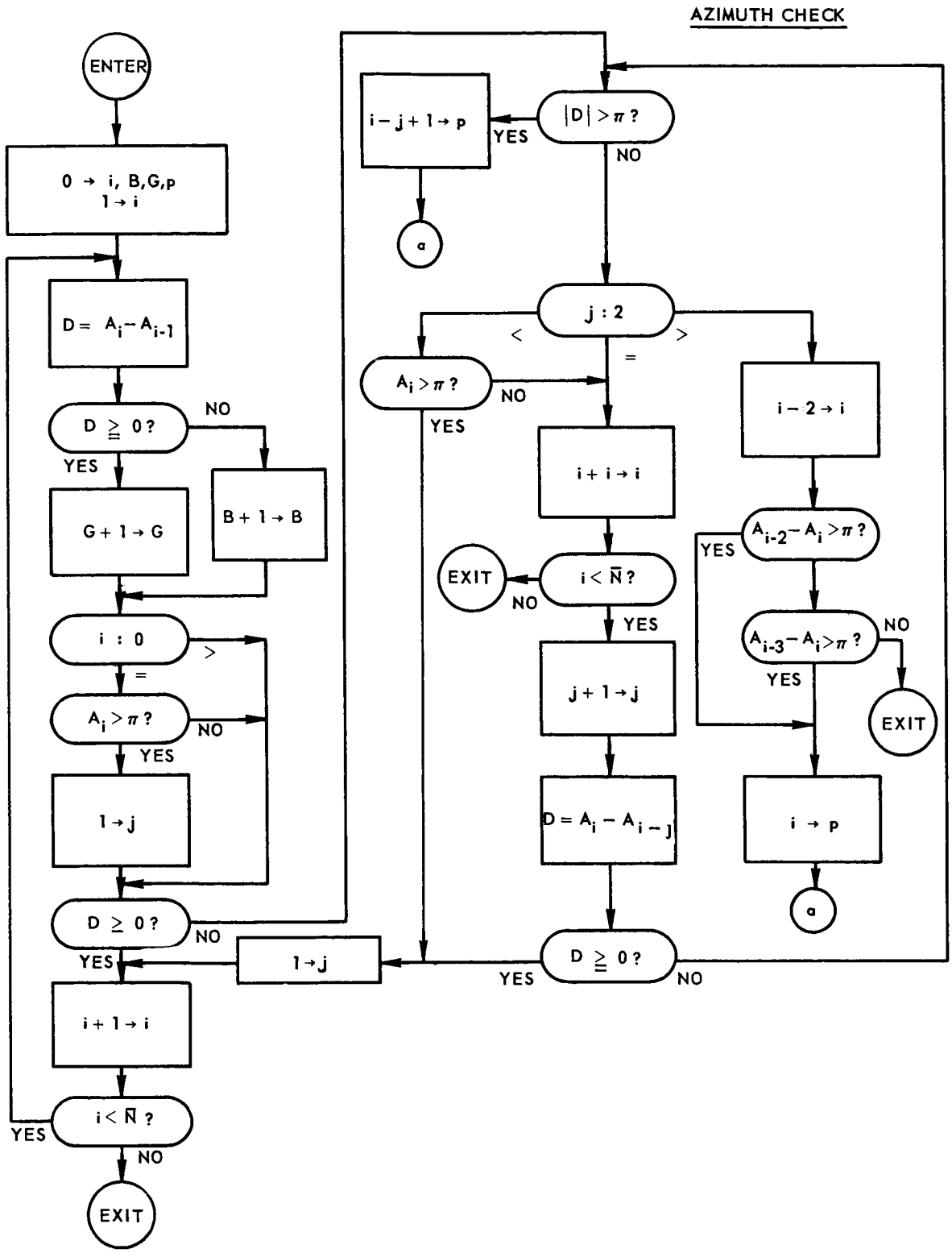


FIGURE 4-2 EOLEDI PROGRAM FLOW CHART *(Sheet 8 of 9)*

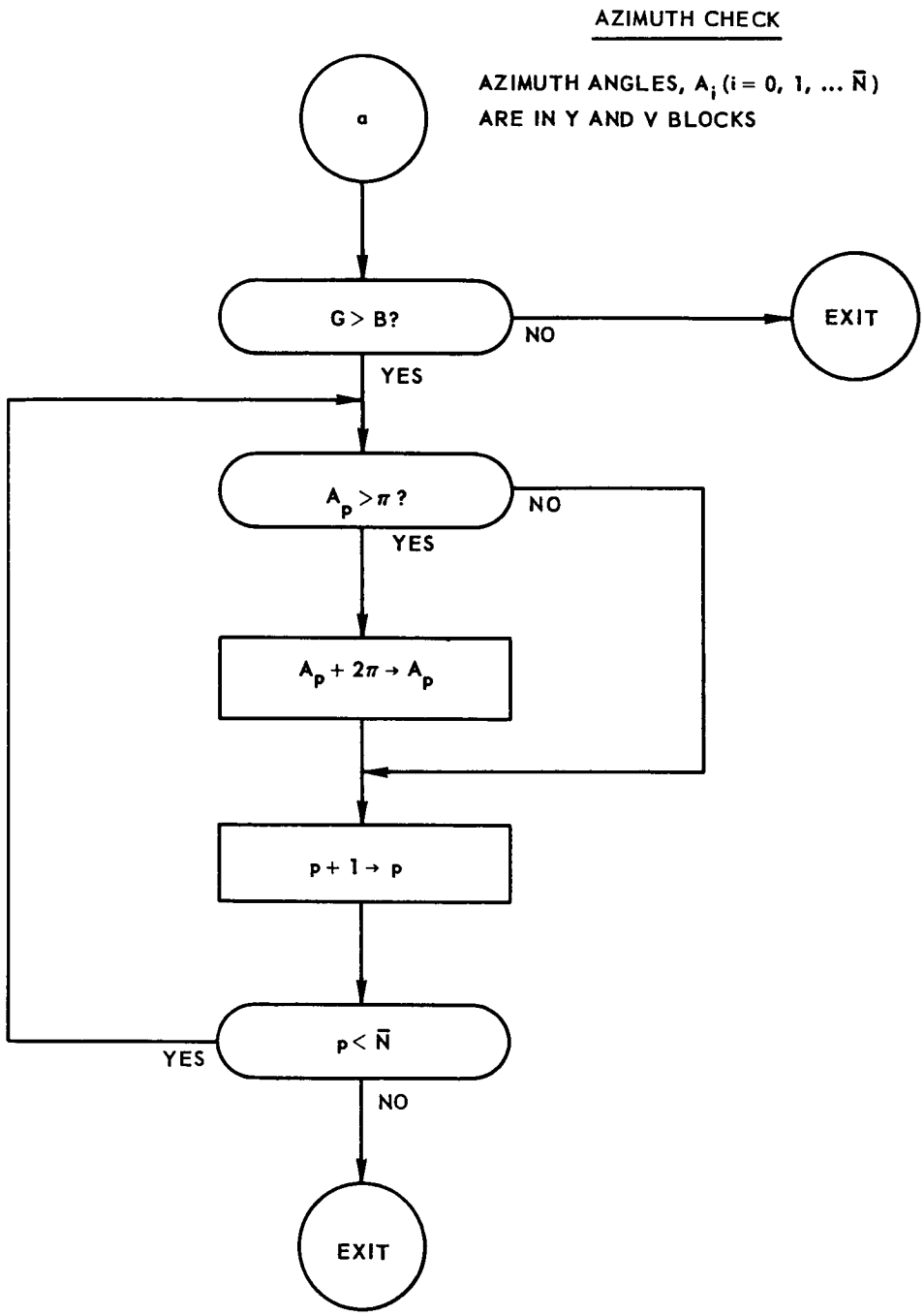


FIGURE 4-2 EOLED1 PROGRAM FLOW CHART *(Sheet 9 of 9)*

4.2 NUMERICAL INTEGRATION CONTROL PROGRAM (N0CPNI)

The purpose of the N0CPNI program is to integrate the three Newtonian equations of motion to obtain an orbit prediction table containing $\bar{r}$ and $\bar{v}$ vectors for specified times. The vector equation of motion is

$$\bar{r} = -\frac{\partial v}{\partial \vec{r}} + \text{Drag}$$

where the potential (V) is approximated by:

$$V = -\frac{1}{r}\left[1 - \frac{C^2}{3r^2}\left(\frac{3\left(\frac{Z}{r}\right)^2 - 1}{2}\right) + \cdots\right]$$

Thus:

$$\frac{\partial V}{\partial \vec{r}} = \begin{bmatrix} \frac{\partial V}{\partial X} \\ \frac{\partial V}{\partial Y} \\ \frac{\partial V}{\partial Z} \end{bmatrix} = \begin{bmatrix} -\frac{X}{r^3}\left[1 - \frac{C^2}{2r^2}\left(5\left(\frac{Z}{r}\right)^2 - 1\right)\right] \\ -\frac{Y}{r^3}\left[1 - \frac{C^2}{2r^2}\left(5\left(\frac{Z^2}{r}\right) - 1\right)\right] \\ -\frac{Z}{r^3}\left[1 - \frac{C^2}{2r^2}\left(5\left(\frac{Z^2}{r}\right) - 3\right)\right] \end{bmatrix}$$

The numerical integration technique employed in the Mercury Program System is the Cowell Method. The basic formulas used are taken from Dr. Herget's book, Computation of Orbits, Chapter 1, "The Calculus of Finite Differences." The following two essential equations are derived and appear on page 8 of Dr. Herget's book:

$$(hD)^{-2} = \Delta^{-2} + \frac{1}{12} - \frac{1}{240}\Delta^2 + \frac{31}{60480}\Delta^4 - \frac{289}{3628800}\Delta^6 + \ldots \quad (4\text{-}1)$$

$$(hD)^{-1} = \Delta^{-1} - \frac{1}{12}\Delta + \frac{11}{720}\Delta^3 - \frac{191}{60480}\Delta^5 + \ldots \quad (4\text{-}2)$$

These two operators are applied to the function $F = h\,\ddot{\bar{r}}$ (h is the integration time interval) to obtain the following expressions for the position and velocity vectors:

$$(hD)^{-2} \quad F = h^{-2} \; D^{-2} \; h^{2} \; \ddot{\vec{r}} = \overline{r}$$

$$(hD)^{-1} \quad F = h^{-1} \; D^{-1} \; h^{2} \; \ddot{\vec{r}} = h\overline{v}$$

Thus $\overline{r}$ and $h\overline{v}$ can be computed by evaluating the corresponding finite differences. In the form of the Cowell Method which is employed in the Mercury Program System it is more convenient to express each of the difference (Δ^j), appearing in the right-hand sides of Equations (4-1) and (4-2), as linear combination of F_i (both are shown in the N2EXCR writeup). The resulting extrapolation and correction formulas appear in the N2EXCR writeup.

N0CPNI serves as the basic control program for the programmed numerical integration system. N1SUFT develops the initial converged second derivative function table, using N2EXCR to extrapolate and correct each new function entry in the table. The N3SDCL, N4DRAG, and N6OPEP subroutines are used to calculate the second derivative; N5OTBL develops the resulting prediction output table, which contains the converged $\overline{r}$ and $\overline{v}$ vectors. The flow chart for N0CPNI is shown in Figure 4-3.

4.2.1 Input Requirements

The N0CPNI program uses seven program, i.e., N1SUFT, N2EXCR, N3SDCL, N4DRAG, N5OTBL, N6OPEP and N7VARS, and three library subroutines: U1EXPE, U1SQRT and U1ATAB. The writeups for these programs are presented in Subsections 4.3 through 4.6.

The following parameters are the input requirements to N0CPNI:

K000.X -- the X coordinate in floating point

K000.Y -- the Y coordinate in floating point

K000.Z -- the Z coordinate in floating point

K00.VX -- the velocity along the X axis, in floating point

K00.VY -- the velocity along the Y axis, in floating point

K00.VZ -- the velocity along the Z axis, in floating point

K000TA -- the anchor time associated with the above $\overline{r}$ and $\overline{v}$ values, expressed in fixed point in minutes or seconds, depending on K0000H. This time is measured from midnight preceding launch.

K000TL -- contains address, in fixed point, of start of TNINTI, if integration table is desired, or address of start of TNINT2, if the reentry table is desired.

K0000H -- the time step, in fixed point, at which orbit points are to be calculated.

K000TU -- if a one, K0000H represents seconds; if a zero, K0000H represents minutes (both in fixed point).

K00DTB -- length of time to integrate backward; expressed in fixed point minutes or seconds, depending on K0000H.

K00DTF -- length of time to integrate forward; expressed in fixed point minutes or seconds, depending on K0000H.

KDORND -- If zero, the N4DRAG program is used; if not zero, the N4DRAG program is not used (in fixed point).

KR0RRV -- If zero, r values only in output table; if one, r & v values in output table (in fixed point).

K1NTVN -- + n, number of integration steps, in fixed point, per output (where n can be any positive integer except zero).

NOTE: The above "K" locations are set synonomous to a 15-word input table TMBBNI in the order in which they are listed.

MCZRWX -- Monitor control word whose decrement indicates the type of table being generated.

decrement = 0, orbit table

decrement $\neq$ 0, re-entry table

4.2.2 Output Requirements

The final output is in table form, in the following format:

a) If zero, the output table represents a minutes integration; if one, the output table represents a seconds integration (both in fixed point).

b) The number of minutes or seconds interval at which entries are made in the table (in fixed point). This quantity is equal to K0000H times KINTVN.

c) The time associated with the first set of entries in the table, the starting time (in fixed point).

d) The time associated with the last set of entries in the table, the ending time (in fixed point).

e) X f) Y g) Z h) VX i) VY j) VZ k) X l) Y Etc.	$\bar{r}$ and $\bar{v}$ parameters stored at the interval defined by a and b (above), from the starting time to the ending time (in floating point).

N0CPNI has two intermediate output tables, TNFUNK and TEMNO, which are described under "Erasable Locations."

4.2.3 Other Requirements

The contents of K00DTB and the contents of K00DTF must be exactly divisible by the contents of K0000H and also by the contents of location b (above); otherwise, the output table is set up wrong.

4.2.4 Method

The equations used in this program are taken from The Computation of Orbits by Dr. P. Herget, and appear in the program descriptions of the various subroutines. The output table for N0CPNI is accurate to seven significant digits.

4.2.5 Usage

a) Calling Sequence -- N0CPNI is entered with TRA N0CPNI; the program exits with the accumulator set to zero and TRA MFCPNI. Error exit is by a TSX MSUERR, 4.

b) Space Required -- 2630 locations, plus 740 locations for subroutines (not including library routines).

c) Special Usage -- When the height for which the drag coefficient is being calculated in N4DRAG falls below 60,000 feet, the accumulator is set to a minus zero and the program exits with a TRA MFCPNI.

d) Constants (excludes constants used in N4DRAG):

K00001 Dec 1

K00010 Dec 10

K000NB Dec 30—Number of iterations to try for convergence of F_i in N2EXCR

K00039 Dec 39

K00000 Dec 0

K00003 Dec 3

K00002 Dec 2

K00006 Dec 6

K00021 Dec 21

K000JB Dec 30—Number of iterations to try for convergence of $'F_i$ in N1SUFT

KCH233 Oct 233000000000

K001.0 Dec 1.0

K002.0 Dec 2.0

K003.0 Dec 3.0

K005.0 Dec 5.0

K00.MH Dec .074366916 - minutes to Mercury units conversion factor

K00.SH DEC .0012394486 - seconds to Mercury units convergence factor

K00.C2 DEC .00324696

K0.CTB DEC 1E-15

K0.CTC DEC 1E-4

K.TCTB DEC 1E-15

e) Tables

TNSAVR	BSS	6, F	converged r, v for last time steps $T_{(i-1)}$
TN7N12	PZE		Indicator for N7VARS set 0 from N1SUFT - 1 from N2EXCR.
TN7ICT	PZE		Number of times interval had to be reduced a one entrance to N2EXCR.
TNETIM	PZE		Time of last r, v in output table (fixed point min or sec as K000TU).
TN7HSI	PZE		If time step is less than unity, time step is here in floating point units.
TNIFPT	PZE		If time step is less than unity, present time for r, v is stored here in floating point units.

f) Erasable Locations -- TEMNO through TEMNO + 99, and TEMN3 through TEMN3 + 9. TNFUNK intermediate storage. The TNFUNK and TEMNO table formats are shown on the following pages.

g) Time Required:

$$335 + B + \left[118 + H(166 + C) \right] * + \left[137 + G(166 + C) \right] ** \text{ cycles}$$

where

B = timing for N1SUFT

C = timing for N2EXCR

$$G = \frac{K00DTF}{K0000H}$$

$$H = \frac{K00DTB}{K0000H}$$

4.2.6 Checkout

The N0CPNI program was checked out by comparing its output with that of IJKNI.

---

* Drop this term if no forward integration.
**Drop this term if no backward integration.

TABLE 4.1 TNFUNK TABLE FORMAT

Location	Contents
	$\underline{X}$
TNFUNK + 0	F_i/h^2
+ 1	$F_i \pm 1$
+ 2	$F_i \pm 2$
+ 3	$F_i \pm 3$
+ 4	$F_i \pm 4$
+ 5	$F_i \pm 5$
+ 6	$F_i \pm 6$
+ 7	$''F_i \pm 1$
+ 8	$'F_i \pm 3/2$
+ 9	$'F_i \pm 1/2$
+ 10	$''F_i$
+ 11	X_i
+ 12	V_{x_i}
	$\underline{Y}$
+ 13	F_i/h^2
+ 14	$F_i \pm 1$
+ 15	$F_i \pm 2$
+ 16	$F_i \pm 3$
+ 17	$F_i \pm 4$
+ 18	$F_i \pm 5$
+ 19	$F_i + 6$
+ 20	$''F_i \pm 1$
+ 21	$'F_i \pm 3/2$
+ 22	$'F_i \pm 1/2$

TABLE 4.1 TNFUNK TABLE FORMAT (Cont'd)

Location	Contents
+ 23	$''F_i$
+ 24	X_i
+ 25	V_{x_i}
	$\underline{Z}$
+ 26	F_i/h^2
+ 27	$F_i \pm 1$
+ 28	$F_i \pm 2$
+ 29	$F_i \pm 3$
+ 30	$F_i \pm 4$
+ 31	$F_i \pm 5$
+ 32	$F_i \pm 6$
+ 33	$''F_i \pm 1$
+ 34	$'F_i \pm 3/2$
+ 35	$'F_i \pm 1/2$
+ 36	$''F_i$
+ 37	X_i
+ 38	V_{x_i}

TABLE 4.2 TEMNO TABLE FORMAT

Location	Contents	
TEMNO + 0	I. R. 4 N2EXCR	
+ 1	$'F_i + 1/2\,(X, Y, Z)$	
+ 2	$''F_i\,(X, Y, Z)$	
+ 3	$\Sigma A_K F_i \pm K$ or $\Sigma B_K F_i \pm K$	
+ 4	$\Sigma A'_K F_i \pm K$ or $B'_K F_i \pm K$	
+ 5	Work cell	
+ 6	X	F_i from N3SDCL
+ 7	Y	F_i from N3SDCL
+ 8	Z	F_i from N3SDCL
+ 9	N	
+ 10	Contents I. R. 1	N2EXCR
+ 11	Contents I. R. 2	N2EXCR
+ 12	F_a	X
+ 13	F_{a+1}	X
+ 14	F_{a+2}	X
+ 15	F_{a+3}	X
+ 16	F_{a+4}	X
+ 17	F_{a+5}	X
+ 18	F_{a+6}	X
+ 19	F_a	Y
+ 20	F_{a+1}	Y
+ 21	F_{a+2}	Y
+ 22	F_{a+3}	Y
+ 23	F_{a+4}	Y

TABLE 4.2 TEMNO TABLE FORMAT (Cont'd)

Location	Contents	
TEMNO + 24	F_{a+5}	Y
+ 25	F_{a+6}	
+ 26	F_a	Z
+ 27	F_{a+1}	
+ 28	F_{a+2}	
+ 29	F_{a+3}	
+ 30	F_{a+4}	
+ 31	F_{a+5}	
+ 32	F_{a+6}	
+ 33	F_a	X
+ 34	F_{a+1}	
+ 35	F_{a+2}	
+ 36	F_{a+3}	
+ 37	F_{a+4}	
+ 38	F_{a+5}	
+ 39	F_{a+6}	
+ 40	F_a	Y
+ 41	F_{a+1}	
+ 42	F_{a+2}	
+ 43	F_{a+3}	
+ 44	F_{a+4}	
+ 45	F_{a+5}	
+ 46	F_{a+6}	

TABLE 4.2 TEMNO TABLE FORMAT (Cont'd)

Location	Contents	
TEMNO + 47	F_a	Z
+ 48	F_{a+1}	
+ 49	F_{a+2}	
+ 50	F_{a+3}	
+ 51	F_{a+4}	
+ 52	F_{a+5}	
+ 53	F_{a+6}	
+ 54	J	
+ 55	$''F_a$ (X)	
+ 56	$'F_a - 1/2$ (X)	
+ 57	T_i	
+ 58	$F_1 + F_{-1}$ then $G_1 (F_1 + F_{-1})$ or	
	$F_1 - F_{-1}$ then $H_1 (F_1 - F_{-1})$	
+ 59	$F_2 + F_{-2}$ then $G_2 (F_2 + F_{-2})$ or	
	$F_2 - F_{-2}$ then $H_2 (F_2 - F_{-2})$	
+ 60	$F_3 + F_{-3}$ then $\Sigma_{K=1}^{3} G_K (F_K + F_{-K})$ or	
	$F_3 - F_{-3}$ then $\Sigma_{K=1}^{3} H_K (F_K - F_{-K})$	
+ 61	Integration direction indicator	
	0 = integrate forward	
	1 = integrate backward	
+ 62	$''F_a$ (Y)	
+ 63	$'F_{a-1/2}$ (Y)	
+ 64	$''F^J_6$ (X)	
+ 65	$''F^{J-1}_6$ (X)	

TABLE 4.2 TEMNO TABLE FORMAT (Cont'd)

Location	Contents	
TEMNO + 66	R_{x_a}	
+ 67	V_{x_a}	
+ 68	—	
+ 69	$''F_a$ (Z)	
+ 70	$'F_{a-1/2}$ (Z)	
+ 71	$''F_G^J$ (Y)	
+ 72	$''F_G^{J-1}$ (Y)	
+ 73	R_{y_a}	
+ 74	V_{y_a}	
+ 75	I. R. 1	EXCR
+ 76	I. R. 2	EXCR
+ 77	I. R. 4	EXCR
+ 78	$''F_G^J$ (Z)	
+ 79	$''F_G^{J-1}$ (Z)	
+ 80	R_{z_a}	
+ 81	V_{z_a}	
	<u>No Storage</u>	
+ 82	Index Register 1	N0CPNI
+ 83	Index Register 2	N0CPNI
+ 84	Index Register 4	N0CPNI
+ 85	Index Register 1	N3SDCL storage
+ 86	Index Register 2	N3SDCL storage
+ 87	Index Register 3	N3SDCL storage

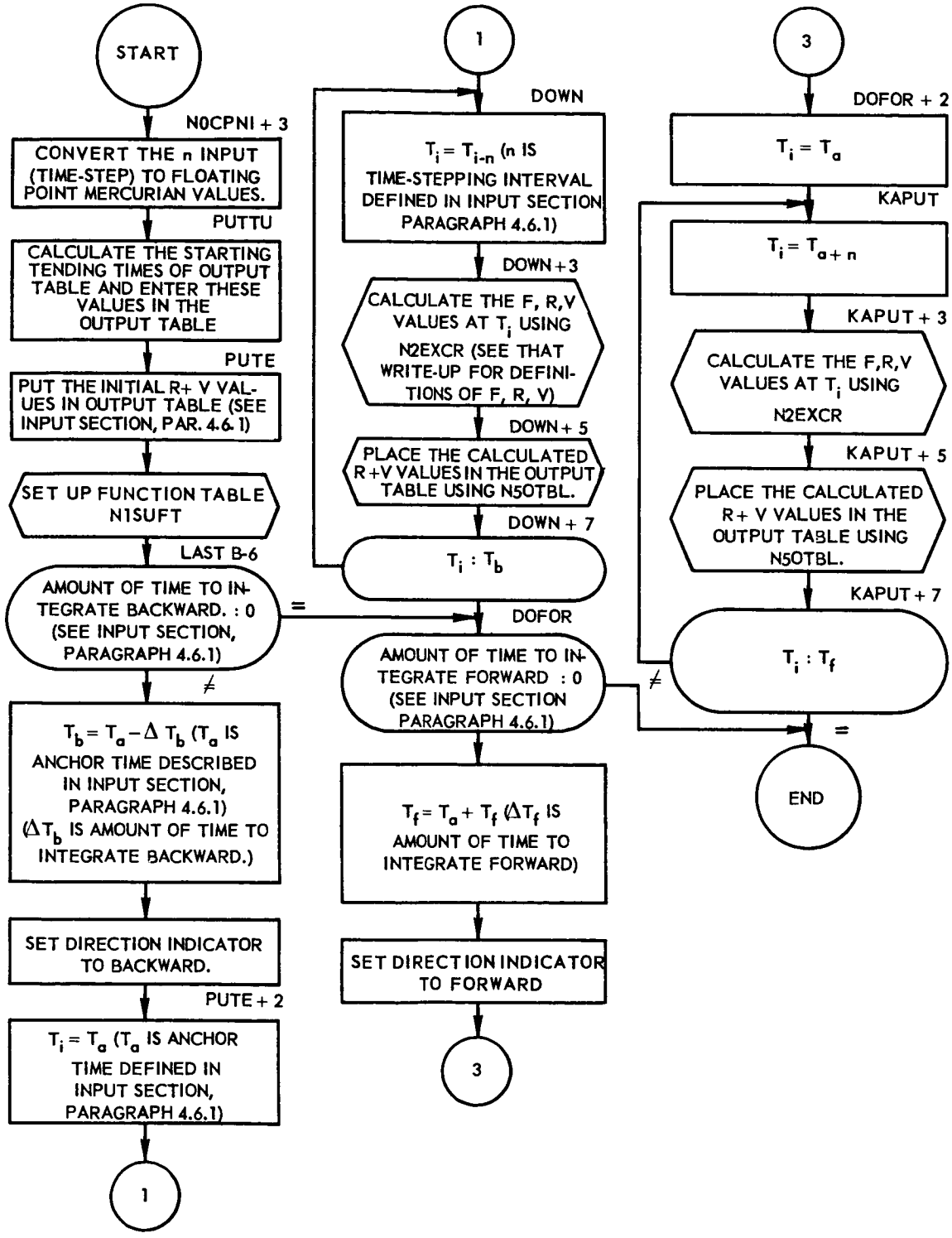


FIGURE 4-3 NOCPNI PROGRAM FLOW CHART

4.3 SET UP FUNCTION TABLE PROGRAM (N1SUFT)

This program establishes a second derivative table which is used to initiate the numerical integration function. The flow chart for N1SUFT is shown in Figure 4-4.

4.3.1 Input Requirements

Other programs used with the N1SUFT program are N2EXCR, N3SDCL, and N7VARS. The following parameters are input requirements for the N1SUFT program:

K000.X -- described in N0CPNI

K000.Y -- described in N0CPNI

K000.Z -- described in N0CPNI

K00.VX -- described in N0CPNI

K00.VY -- described in N0CPNI

K00.VZ -- described in N0CPNI

K00TA -- described in N0CPNI

K0000H -- described in N0CPNI

K000.H -- contains the floating point equivalent of the contents of K0000H

K00.2H -- contains the contents of K000.H squared.

4.3.2 Output Requirements

The output of N1SUFT is the TNFUNK table (described at the end of the N0CPNI program write-up).

4.3.3 Method

The equations used in this program, taken from The Computation of Orbits by Dr. P. Herget, are:

a) $$''F_a = R_a + G_a F_a + \sum_{k=1}^{3} G_k \left(F_k + F_{-k} \right)$$

b) $$'F_{a-1/2} = (h)\, V_a + H_a F_a + \sum_{k=1}^{3} H_k \left(F_k - F_{-k} \right)$$

c) $''F_{a-1} = F_a - 'F_{a-1/2}$

The output table (TNFUNK) is not assumed to be correct until successive calculations of the $''F_6$ value differ by less than IE - 15. The table is accurate to seven significant digits.

4.3.4 Usage

a) Calling Sequence:

α	TSX	NISUFT, 4
$\alpha + 1$		Error return
$\alpha + 2$		Normal return

b) Space Required -- 261 locations

c) Constants used:

K00039	DEC	39	
K00021	DEC	21	
K00000	DEC	0	
K00001	DEC	1	
K00.GO	DEC	-.0963348773	G_a (see 4.3.3,a)
K00.G1	DEC	.0074115416	G_r k = 1,3 (see 4.3.3,a)
K00.G2	DEC	.00099014103	
K00.G3	DEC	.0000796407	
K00.H0	DEC	-.5	H_a (see 4.3.3,b)
K00.H1	DEC	.0648396164	H_k, k = 1,3 (see 4.3.3,b)
K00.H2	DEC	-.0139550265	
K00.H3	DEC	.0015790344	
K.TCTB	DEC	IE-15 -- convergence criterion for $''F_6^{\,j} : 'F_6^{\,(j-i)}$ (see Figure 4-4, block SQRE)	

K000JB PZE 30 -- number of tries to make for convergence before decreasing interval

d) Erasable locations -- TEMNO through TEMNO + 99, and TEMN3 through TEMN3 + 9.

e) Time Required:

$$1627 + D + J\,(1609 + 12\,\left[1709 + D + N\,(1954 + D)\right] \text{ cycles}$$

where

D = timing for N4DRAG

J = number of convergence tries in NISUFT (2)

N = number of convergence tries in N2EXCR (1)

4.3.5 Checkout

The NISUFT program was checked out by comparing its output with that of IJKNI.

Sample Section Of a Difference Table Used in NISUFT

Argument	Second Sum	First Sum	Function	First Difference	Second Difference	Third Difference
T_{a-h}	$''F_{a-1}$		F_{a-1}		Δ''_{a-1}	
		$'F_{a-1/2}$		$\Delta'_{a-1/2}$		$\Delta'''_{a-1/2}$
T_a	$''F_a$		F_a		Δ''_a	
		$'F_{a+1/2}$		$\Delta'_{a+1/2}$		$\Delta'''_{a+1/2}$
T_{a+h}	$''F_{a+1}$		F_{a+1}		Δ''_{a+1}	
		$'F_{a+3/2}$		$\Delta'_{a+3/2}$		$\Delta'''_{a+3/2}$
T_a	$''F_{a+2}$		F_{a+2}			

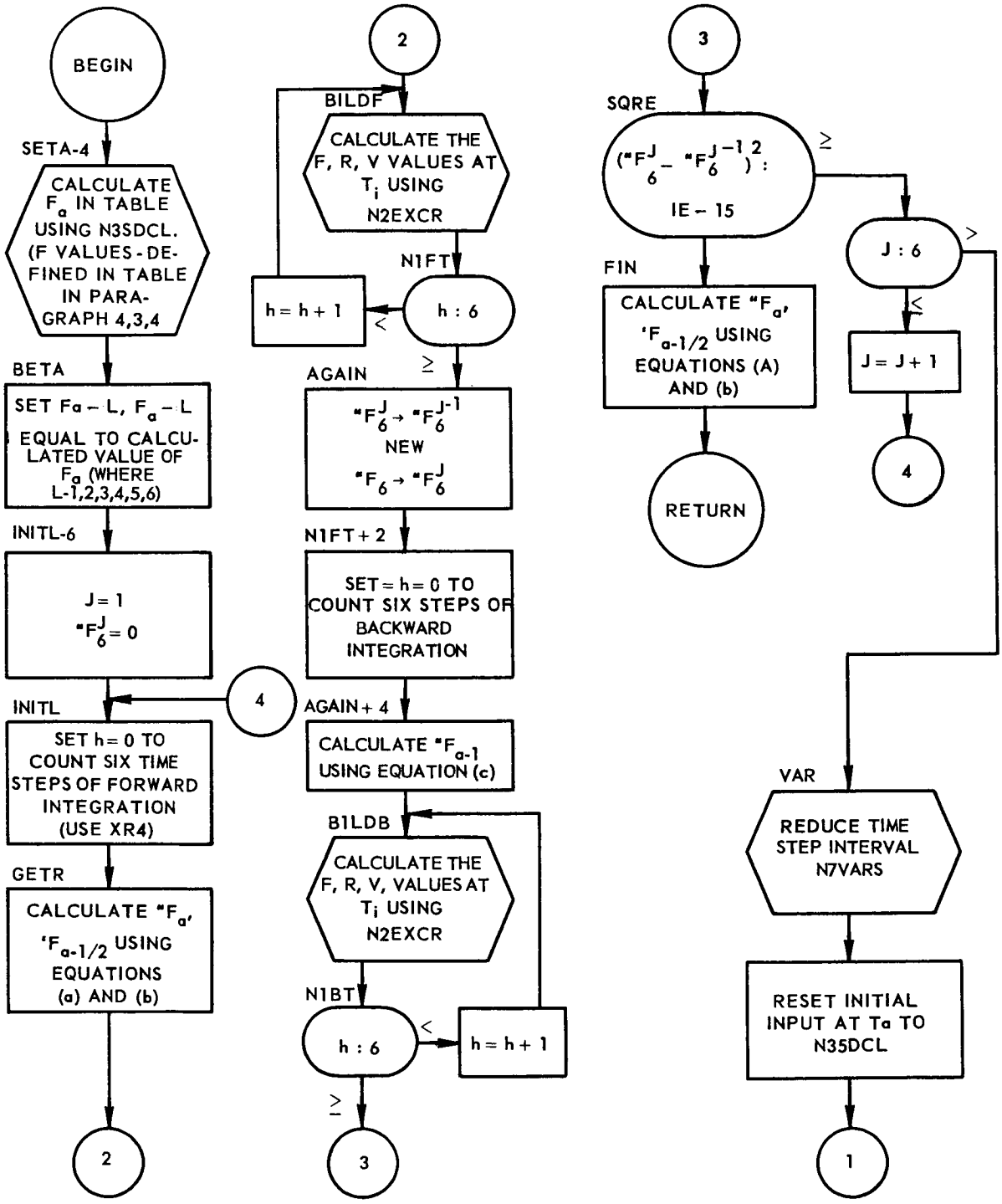


FIGURE 4-4 NISUFT PROGRAM FLOW CHART

4.4 EXTRAPOLATION AND CORRECTION EQUATION DERIVATION PROGRAM (N2EXCR)

The N2EXCR program calculates F_i, R_i and V_i for a given time T_i. The flow chart for N2EXCR is shown in Figure 4-5.

4.4.1 Input Requirements

The input to the N2EXCR program comes from the following locations:

a) The TNFUNK table (described in the N0CPNI program write-up).

b) K000.H -- contains the floating point equivalent of the contents of K0000H.

c) K00.2H -- contains the contents K000.H squared.

d) TEMNO + 51. If zero, integration is carried forward; if one, integration is carried backward (fixed point).

N2EXCR uses the following programs: N3SDCL and N7VARS.

4.4.2 Output Requirements

The updated TNFUNK table is the output of N2EXCR.

4.4.3 Method

The equations used in this program are taken from The Computation of Orbits by Dr. P. Herget.

a) $'F_{i-1/2} = 'F_{i-3/2} + F_{i-1}$ (for forward integration)

b) $'F_{i+1/2} = 'F_{i+3/2} - F_{i+1}$ (for backward integration)

c) $''F_i = ''F_{i-1} + 'F_{i-1/2}$ (for forward integration)

d) $''F_i = F_{i+1} - 'F_{i+1/2}$ (for backward integration)

$$\left.\begin{array}{l} \text{e) } R_i = ''F_i + \sum_{k=1}^{6} A_k F_{i-k} \\ \\ \text{f) (h) } (V_i) = 'F_{i-1/2} + \sum_{k=1}^{6} A'_k F_{i-k} \end{array}\right\} \text{(forward extrapolation for } R_i + V_i)$$

$$\left.\begin{array}{l} \text{g)}\ R_i = {}''F_i + \sum_{k=1}^{6} A_i\, F_{i+k} \\ \text{h)}\ (h)\ (V_i) = {}'F_{i+1/2} - \sum_{k=1}^{6} A'_k\, F_{i+k} \end{array}\right\} \text{(backward extrapolation for } R_i + V_i\text{)}$$

$$\left.\begin{array}{l} \text{i)}\ R_i = {}''F_i + \sum_{k=0}^{6} B_k\, F_{i-k} \\ \text{j)}\ (h)\ (V_i) = {}'F_{i-1/2} + \sum_{k=0}^{6} B'_k\, F_{i-k} \end{array}\right\} \text{forward correction for } R_i + V_i$$

$$\left.\begin{array}{l} \text{k)}\ R_i = {}''F_i + \sum_{k=0}^{6} B_k\, F_{i+k} \\ \text{l)}\ (h)\ (V_i) = {}'F_{i+1/2} - \sum_{k=0}^{6} B'_k\, F_{i+k} \end{array}\right\} \text{backward correction for } R_i + V_i$$

The F table values calculated are not assumed to be correct until successive calculation of the F_i value differ by less than IE-15. The table is accurate to seven significant digits.

4.4.4 Usage

a) Calling Sequence:

a	TSX	N2EXCR, 4
$a + 1$		Error Return
$a + 2$		Normal return

b) Space Required -- 234 locations.

c) Constants:

K00001	DEC	1
K00039	DEC	39
K00003	DEC	3

K000NB	PZE	30 -- Maximum number of iterations to try for convergence of F_i	
K00.A1	DEC	.46038333	A_k, k = 1, 6
K00.A2	DEC	-1.09306667	
K00.A3	DEC	1.41426667	
K00.A4	DEC	-1.0424	A_k, k = 1, 6
K00.A5	DEC	.41235	
K00.A6	DEC	-.0682	
K0.A1P	DEC	2.28673082	
K0.A2P	DEC	-5.1099041	
K0.A3P	DEC	6.55591931	
K0.A4P	DEC	-4.82397487	A'_k, k = 1, 6
K0.A5P	DEC	1.90782077	
K0.A6P	DEC	-.31559193	
K00.B0	DEC	.065495756	B_k, k = 0, 6
K00.B1	DEC	.067409063	
K00.B2	DEC	-.110635752	
K00.B3	DEC	.104370596	
K00.B4	DEC	-.05999091	
K00.B5	DEC	.019393189	
K00.B6	DEC	-.002708609	
K00.B0P	DEC	.30422454	B'_k, k = 0, 6
K0.B1P	DEC	.46038360	
K0.B2P	DEC	-.54653605	
K0.B3P	DEC	.47142857	
K0.B4P	DEC	-.26060681	
K0.B5P	DEC	.08247355	
K0.B6P	DEC	-.01136740	

d) Erasable Locations -- TEMNO through TEMNO + 99, and TEMN3 through TEMN3 + 9.

K000TU	(see 4.2.1)		
K00CTB	DEC	1E-15	Convergence criterion for F_i when generating a minutes output table.
K00CTC	DEC	1E-15	Convergence criterion for F_i when generating a seconds re-entry table.
TN7ICT	(see 4.11.1,i)		

e) Time Required:

1709 + D + N (1954 + D) cycles

where:

D is timing for N4DRAG

N is number of tries to converge (1) in N2EXCR

4.4.5 Checkout

The N2EXCR program was checked out by comparing its output with that of IJKNI.

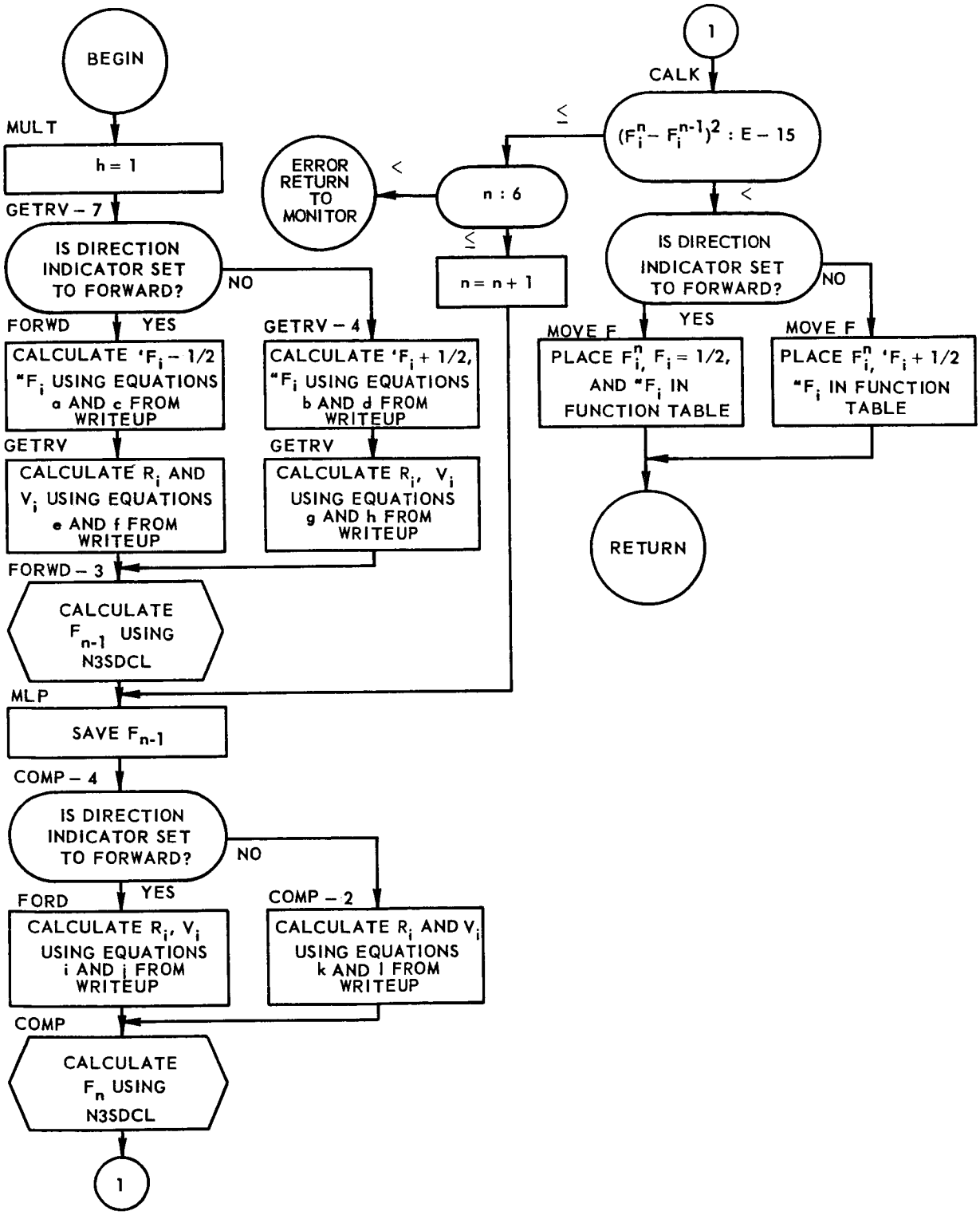


FIGURE 4-5 N2EXCR PROGRAM FLOW CHART

4.5 SECOND DERIVATIVE CALCULATION CONTROL PROGRAM (N3SDCL)

N3SDCL calculated F_i/h^2 by adding the output of subroutine N60PEP and N4DRAG. The flow chart for N3SDCL is shown in Figure 4-6.

4.5.1 Input Requirements

The N3SDCL program uses two programs: N60PEP and N4DRAG.

Inputs to the N3SDCL program are:

a) The X, Y, Z, VX, VY, and VZ values from the following TNFUNK table locations:

TNFUNK + 11 = X

TNFUNK + 24 = Y

TNFUNK + 37 = Z

TNFUNK + 12 = VX

TNFUNK + 25 = VY

TNFUNK + 38 = VZ

b) K000TI -- the time for which r and v values are currently being calculated in N0CPNI, the present integration time, expressed in fixed point minutes or seconds, depending on K0000H.

c) K000TU -- If a one, K0000H represents seconds, if zero, K0000H represents minutes (both in fixed point).

d) KDORND -- If zero, the N4DRAG program is used, if not zero, the N4DRAG program is not used (both in fixed point).

4.5.2 Output Requirements

The output of N3SDCL is F_i/h^2 and is placed in the TNFUNK table (see Table 4.1).

TNFUNK + 0 = $\ddot{X}$

TNFUNK + 13 = $\ddot{Y}$

TNFUNK + 26 = $\ddot{Z}$

4.5.3 Method

The equations used in N3SDCL are taken from The Computation of Orbits by Dr. P. Herget.

$$\ddot{X} = \frac{\partial P}{X} + D_x$$

$$\ddot{Y} = \frac{\partial P}{Y} + D_y$$

$$\ddot{Z} = \frac{\partial P}{Z} + D_z$$

The N3SDCL program is accurate to seven significant digits.

4.5.5 Usage

a) Calling Sequence

α	TSX	N3SDCL, 4
$\alpha + 1$	Error Return	
$\alpha + 2$	Normal return	

b) Space Required

57 locations

c) Erasable Locations

TEMNO + 85, 86, 87 (used for the index registers)

d) Constants

K00003	DEC	3		
KDSUBX	SYN	TDSUBX	D_x	output of N4DRAG (see 4.6.2)
KDSUBY	SYN	TDSUBY	D_y	
KDSUBZ	SYN	TDSUBZ	D_z	

K00.XP	SYN	TNIINT	$\frac{\partial P}{\partial x}$	output of N60PEP (see 4.10.2)
K00.YP	SYN	TNIINT + 1	$\frac{\partial P}{\partial y}$	
K00.ZP	SYN	TNIINT + 2	$\frac{\partial P}{\partial z}$	
TNDRAG	BSS	7	input to N4DRAG (see 4.6.1)	
TNETIM	PZE		(see 4.2.5)	

e) Time Required

539 + D + P cycles

where

D is timing for N4DRAG

P is timing for N60PEP

4.5.6 Checkout

The N3SDCL program was checked by comparing its output to that of LJKNI.

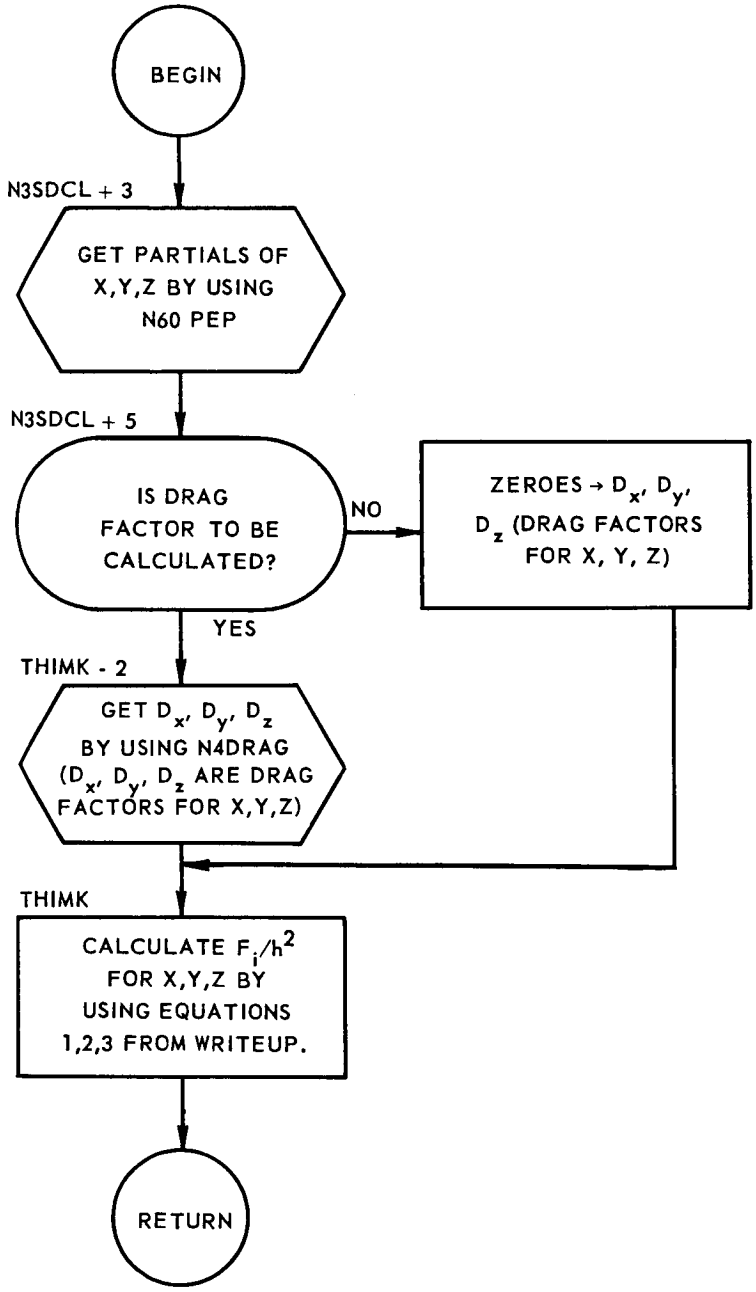


FIGURE 4-6. N3SDCL PROGRAM FLOW CHART

4.6 CAPSULE DRAG FORMULATION PROGRAM (N4DRAG)

The N4DRAG program produces the components of the drag acceleration and capsule impact data during a Mercury mission. The flow chart for N4DRAG is shown in Figure 4-7.

4.6.1 Input Requirements

N4DRAG uses the Mercury subroutines, N4PDEN, N4COEF and A3MSCP, and the library subroutine, U1SQRT. Write-ups for N4PDEN and N4COEF are contained in Subsections 4.7 and 4.8, respectively. The write-up for A3MSCP is located in Subsection 3.14.

Input to the N4DRAG program is:

TNDRAG	Time, t_0, in minutes or seconds in fixed point
TNDRAG + 1 X TNDRAG + 2 Y TNDRAG + 3 Z TNDRAG + 4 VX TNDRAG + 5 VY TNDRAG + 6 VZ	$\bar{r}$ and $\bar{v}$ vectors corresponding to time t_0, in floating point.
T0DRAG	A location set to zero or 1 by the retrofire program to determine the output of N4DRAG.
TN7N12	(See 4.11.1, g)
TN4NEX	Indicator set by R2AINT (subroutine of Differential Correction) (NOTE: Normal Status + 0) If minus: the N4DRAG program will not store any impact data; one of the six tables needed for the numerical differential correction is being generated. If minus zero: Do not exit at 60,000 feet. Thus the numerical differential correction tables (after the first one) will all have the same ending time.

TEMN0 + 61: Common location

+ 1 indicates backward integration

+ 0 indicates forward integration

4.6.2 Output Requirements

The output requirements for N4DRAG are:

TDSUBX TDSUBY TDSUBZ	X, Y, and Z components of the drag acceleration in floating point Mercury units

If T0DRAG is set equal to zero, time at 450,000 ft. is stored in T0DCHC and the re-entry impact data is stored in the TMORMC block:

TMORMC + 12	GMT of landing based on present capsule setting; min/address, sec/decrement.
TMORMC + 13	Latitude of landing point for end of mission in floating point radians.
TMORMC + 14	Longitude of landing point in floating point radians.

If T0DRAG $\neq$ 0, retrofire impact data is stored in block TMIMPP:

TMIMPP	First time, t_1, $\leq$ 60,000 ft.; min/address, sec/decrement.
TMIMPP + 1	ϕ at time t_1, in floating point.
TMIMPP + 2	λ at time t_1, in floating point.
TMIMPP + 3	last time, t_2, $\leq$ 450,000 ft.; min/address, sec/decrement.
TMIMPP + 4 X TMIMPP + 5 Y TMIMPP + 6 Z TMIMPP + 7 VX TMIMPP + 8 VY TMIMPP + 9 VZ	$\bar{r}$ and $\bar{v}$ parameters at time t_1 in floating point.

Other Requirements: The normal state of T0DRAG is zero. The retrofire program sets this indicator to a nonzero value only during retrofire calculations: it resets the indicator to zero upon completion of these calculations.

4.6.3 Method

The following equations are used by the N4DRAG program:

$$V = \sqrt{\left(V_X + Y_{\omega_Z}\right)^2 + \left(V_Y - X_{\omega_Z}\right)^2 + V_Z^2}$$

h geop. meters = 6378145.0 $h_{\text{Mercury units}} / (1 + h_{\text{Mercury units}})$ formulated from ICAO standard atmosphere.

$$D_X = -1/2\, C_d\, \rho V\, (V_X + Y\omega_Z)$$

$$D_Y = -1/2\, C_d\, \rho V\, (V_Y - X\omega_Z)$$

$$D_Z = 1/2\, C_d\, \rho V\, V_Z$$

ω_Z = 0.58833543, rotational velocity of the earth

The N4DRAG program is accurate to seven significant digits.

4.6.4 Usage

a) Calling Sequence:

α	TSX	N4DRAG, 4
$\alpha + 1$		Re-entry return
$\alpha + 2$		Normal return

b) Space Required - 130 locations, excluding constants and tables -- two of which are temporary storage locations.

c) Error Codes -- two in the decrement of location N4DRAG if height is $\leq$ 60,000 feet.

d) Special Usage -- Exit will not be made for a 60,000 feet indication under the following conditions:

1) Initial function table set up by N1SUFT is in process. (TN7N12 = 0).

2) Backward integration is in process (TEMNO + 61 ≠ 0)

3) After first of six numerical differential correction tables has given a time of a 60,000 feet exit, TN4NEX is set to zero to use this time as an exit for the remaining tables (not a height indication).

e) Constants:

K0.6E4	DEC	.0028672914	60,000 ft. in Mercury units
K045.4	DEC	.021504685	450,000 ft. in Mercury units
K.OMEG	DEC	.058833543	rotational velocity of time of earth/Mercury units
K000.5	DEC	0.5	
K001.0	DEC	1.0	
K00060	DEC	6.0	
K0.RAD	DEC	6378145.	Equatorial radius of earth in meters
K000TU			(See Subsection 4.2.1)
K0DENS			Constant within coding at calling sequence to N4PDEN.

f) Tables:

T00IND -- ^{t} 450,000 ft.: min/address, sec/decrement

TCSUBD -- coefficient of drag

TVELOC -- velocity vector magnitude (including rotation of Earth).

TN4LLH -- three-word output block for A3MSCP containing latitude (radians), longitude (radians), and height (Mercury units).

g) Erasable Locations -- AA and AB. TEMN4 PZE Temporary location.

h) Time Required:

Average Time = 4.788 msec. $+ t_{A3MSCP} + t_{N4DENS} + t_{N4COEF} + t_{U1SQRT}$

Time if h < 60,000 ft. = $2.028 + t_{A3MSCP} + t_{U1SQRT}$

1.128 msec. additional to store impact data.

4.6.5 Checkout

The N4DRAG program was checked out by comparing machine results with hand calculations, running integrations (orbit and re-entry) to test the reasonableness of program, and comparing orbit integrations with IJKNI output.

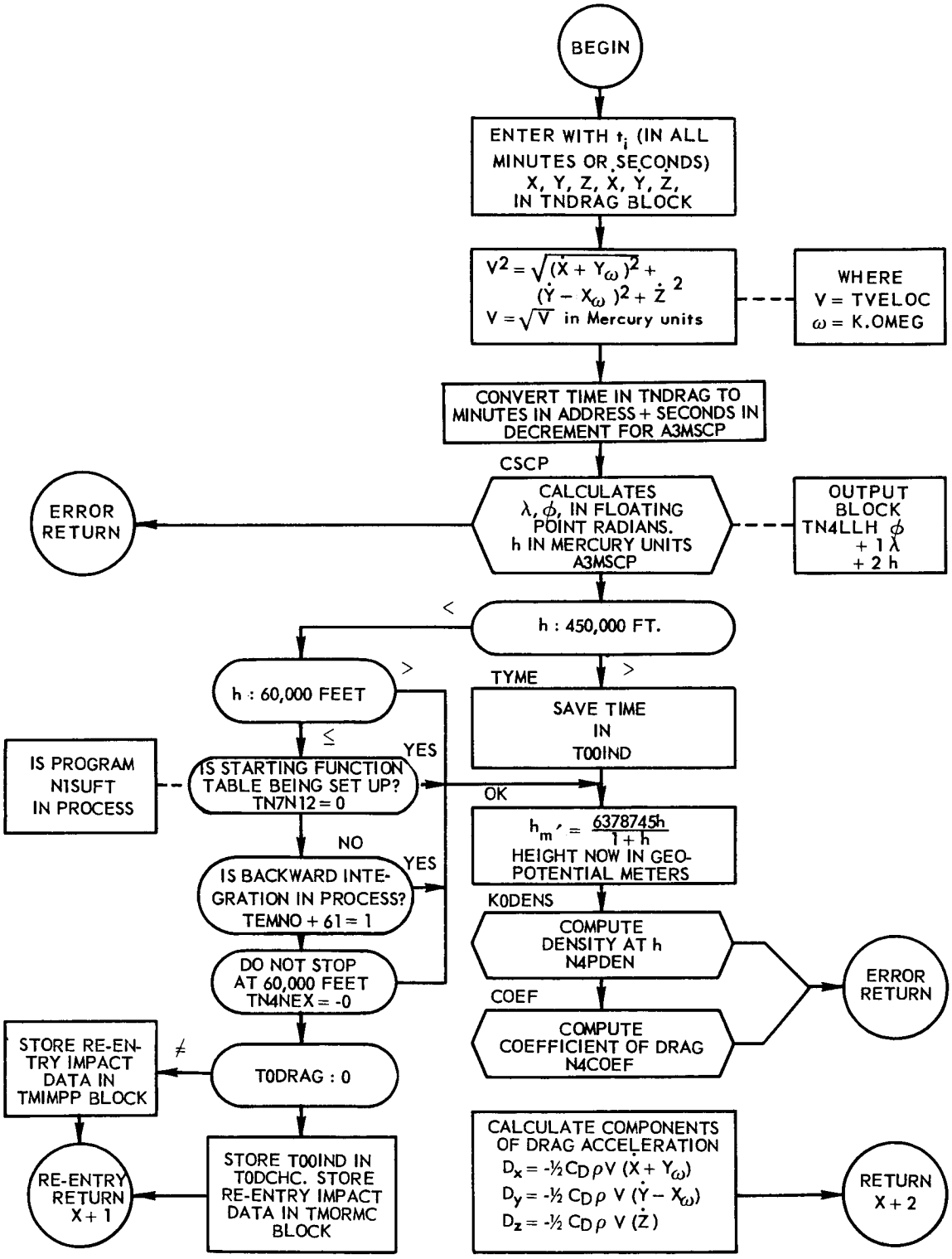


FIGURE 4-7 N4DRAG PROGRAM FLOW CHART

4.7 ATMOSPHERIC DENSITY COMPUTATION PROGRAM (N4PDEN)

This program produces, in Mercury units, the atmospheric density corresponding to a given altitude and stores it in the calling sequence at $\alpha + 2$ (K0DENS + 2). The flow chart for the N4PDEN program is shown in Figure 4-8.

4.7.1 Input Requirements

The exponential library subroutine, U1EXPE, is used with N4PDEN. Capsule altitude is supplied to N4PDEN in geopotential meters in the calling sequence at $\alpha + 1$ (K0DENS + 1).

4.7.2 Output Requirements

Density in Mercury units corresponding to the given altitude is stored in the calling sequence at $\alpha + 2$ (K0DENS + 2).

4.7.3 Method

Given altitude, h, in geopotential meters, then for ρ (density) in kg/m^3.

For low altitudes where $100 < h < h_c$:

$$\begin{aligned} \text{Ln}\,\rho = {} & .027890921 - 2.0913592\left(h/10^5\right) - 85.812859\left(h/10^5\right)^2 \\ & + 203.13606\left(h/10^5\right)^3 - 143.35070\left(h/10^5\right)^4 \\ & - 83.952568\left(h/10^5\right)^5 + 141.51559\left(h/10^5\right)^6 \\ & - 44.56152\left(h/10^5\right)^7 \end{aligned}$$

For high altitudes where $h_c \leqq h < 278260$:

$$\begin{aligned} \text{Ln}\,\rho = {} & 19.748648 - 56.044534\left(h/10^5\right) + 15.551762\left(h/10^5\right)^2 \\ & + 11.168017\left(h/10^5\right)^3 - 6.8256342\left(h/10^5\right)^4 \\ & + .19714784\left(h/10^5\right)^5 + .50902268\left(h/10^5\right)^6 \\ & - .083568426\left(h/10^5\right)^7 \end{aligned}$$

$\rho = \exp(\text{Ln}\,\rho)$ in kg/m^3

$\rho = 4.3414636 \times 10^{-5}\,\rho$ in Mercury units

If mutilation is called for, $\rho = K\rho$ where K = C(K.MUTE)

4.7.4 Usage

a) Calling Sequence:

TSX	N4PDEN, 4
PZE	Altitude in m^1 (supplied)
PZE	Density in Mercury units (output)
Error Return	(Not used)
Normal Return	

b) Space Required -- 70 locations, excluding subroutines and K locations.

c) Special Usage:

If mutilation is desired:

1) Set the c (indicator K001S1) = 1

2) Store mutilation coefficient in location K.MUTE

d) Constants:

KOMASK			Sense indicator mask
K.N4HC	DEC	136025.0	Critical altitude
K001S1			If one, mutilate density, constant value once set.
K100.0	DEC	100.0	
K10.ES	DEC	100000.0	
K0.DE1	DEC	.000044545482	Density at 100 miles in Mercury units
K.MUTE			Density mutilation coefficient
K04.M5			Converts metric density to Mercury units
MSUERR			Location for error exit

Index registers 1 and 4 and the sense indicators are used with constants restored on normal exit.

e) Erasable Locations -- INSVE and N4HGT

4.7.5 Checkout

Density tables were reproduced according to acceptable standard deviation of fit to logarithms. Coefficients were obtained using LSQ1 with data taken from ARDC Model atmosphere 1956, AFCRC-TR-59 267, Air Force Surveys in geophysics No. - 115, pages 36 - 47.

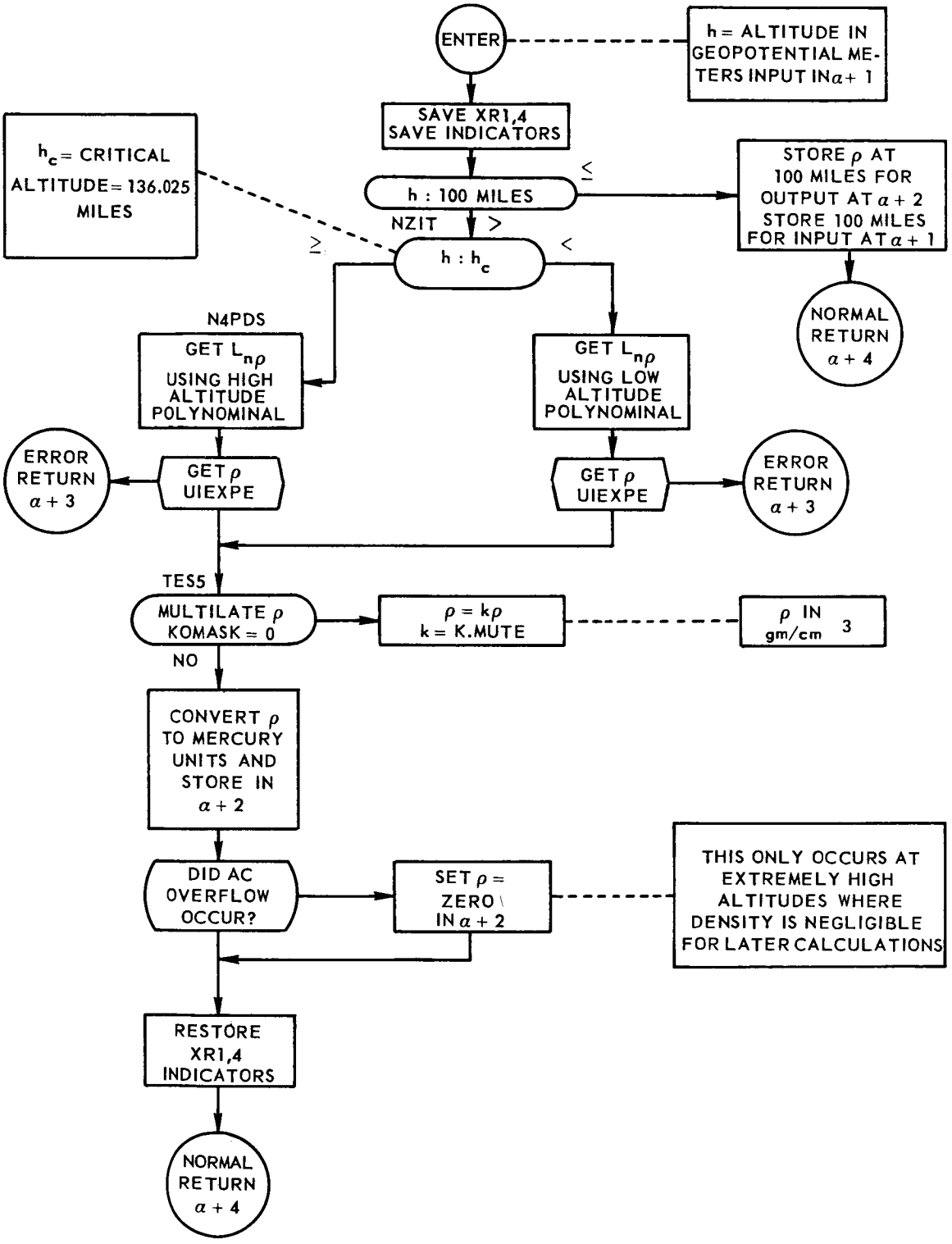


FIGURE 4-8 N4PDEN PROGRAM FLOW CHART

4.8 COEFFICIENT OF DRAG CALCULATION PROGRAM (N4COEF)

The N4COEF program calculates the coefficient of atmospheric drag on the capsule. The flow chart for this program is shown in Figure 4-9.

4.8.1 Input Requirements

Two library routines are used as input to the N4COEF program:

a) U1SQRT - square root routine.

b) U1SICO - sine/cosine routine.

The following information also serves as input to N4COEF:

a) K0DENS - 1, containing altitude in geopotential meters, floating point.

b) TVELOC, containing velocity vector magnitude in Mercury units, floating point.

c) MCZRWX, Monitor control word whose decrement indicates the type being generated.

decrement = 0, orbit table

decrement ≠ 0, re-entry table

4.8.2 Output Requirements

Location TCSUBD, which contains the coefficient of drag in Mercury units (area/mass of capsule), is the output from N4COEF.

4.8.3 Method

N4COEF uses the following equations:

$$(T_m)_b + T_{ob} - 288.16 = (T_m)_b \tag{1}$$

$$(T_m)_b + L_m (h = h_b) = T_m \tag{2}$$

$$C_s = 20.046333 \sqrt{T_m} \qquad h \leqq 90{,}000 \text{ M} \tag{3}$$

$$(C_D')0 = 1.75 + .25 \cos \left[\frac{172205.0 - h}{32000.0} \right] \qquad 145 \text{ KM}' \leq h \leq 177\text{KM}' \tag{4}$$

$C_D = (C_D') \; 410203811.0$

The N4COEF program is accurate to seven significant digits.

4.8.4 Usage

a) Calling Sequence

α	TSX	N4COEF, 4
$\alpha + 1$	Error return (not used)	
$\alpha + 2$	Normal return	

b) Space Required

115 locations including internal tables.

c) Constants

KCD.CV	DEC		Area/mass in Mercury units (floating point), where weight factor represents orbit weight of capsule. Variable because of weight change for various launchings.
KCD.CW	DEC		Same as above constant with re-entry weight as a factor.
K.MACH	DEC	7905.3827	Mercury units of velocity.
KS.UND	DEC	20.0463333	Proportionality constants, $C_s - T_m$ equation.
K00.25	DEC	.25	
K001.5	DEC	1.5	
K1.525	DEC	1.525	
K01.75	DEC	1.75	
K002.0	DEC	2.0	
K00.P1	DEC	3.14159265	

K010.0	DEC	10.0
K32.E3	DEC	32000.0
K0.9E4	DEC	90000.0
KL.HGT	DEC	141766.0
KT.GHT	DEC	172205.0

d) Tables:

TCSUBD	Coefficient of drag (Mercury units)
TVELOC	Velocity vector magnitude (Mercury units)

e) Common Locations -- CA SYN STRGE

f) Time Required:

Normal time = .672 milliseconds

Maximum time = $(3.312 + .06_i + .108_j)$ milliseconds

where:

i = number of lookups in HSUBB table

j = number of lookups in MACH # table

4.8.5 Checkout

All possibilities were tested against check calculations. N4COEF was run with numerical integration and compared with N0CPNI runs, using a constant coefficient of drag.

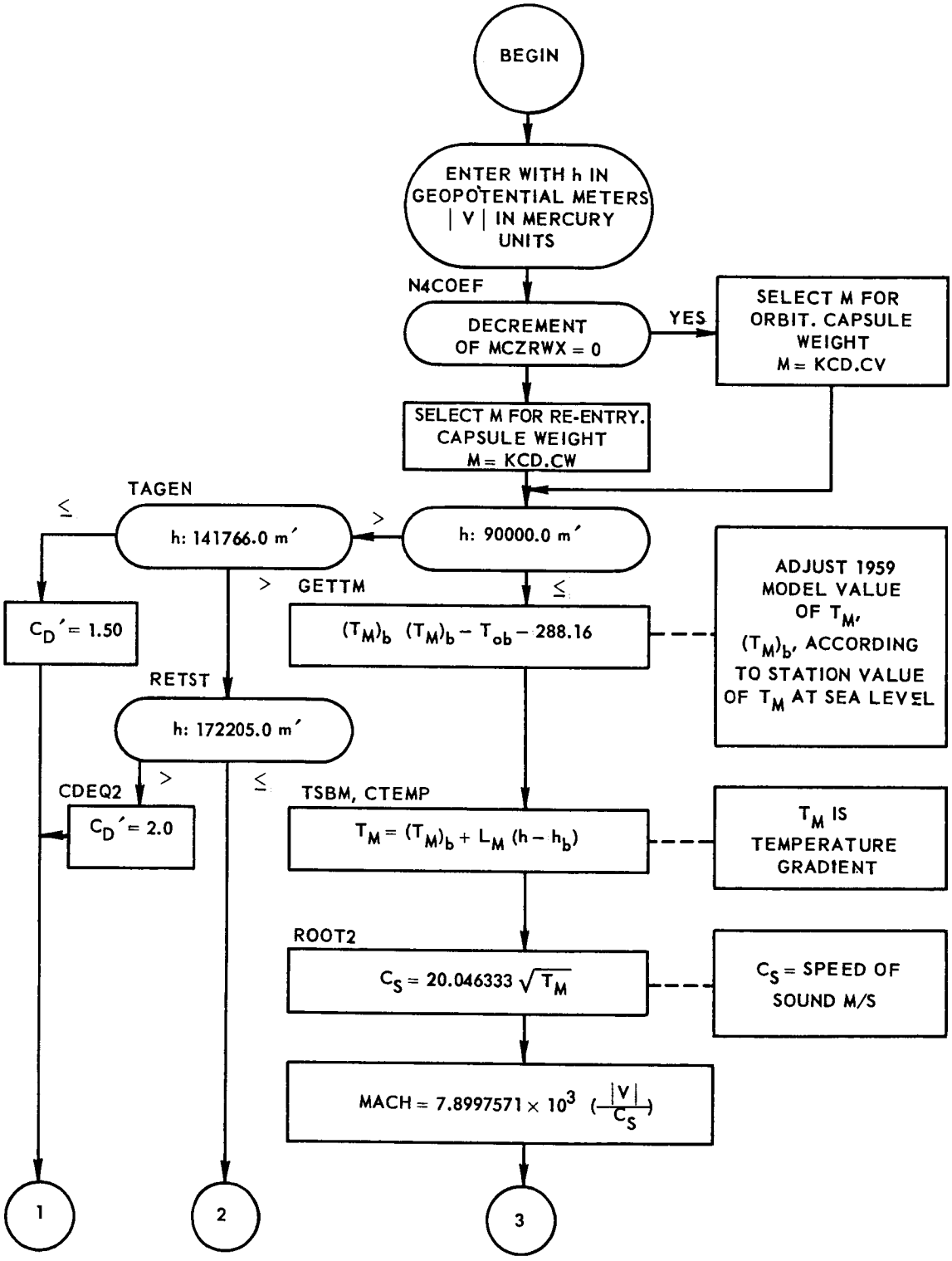


FIGURE 4-9 N4COEF PROGRAM FLOW CHART *(Sheet 1 of 2)*

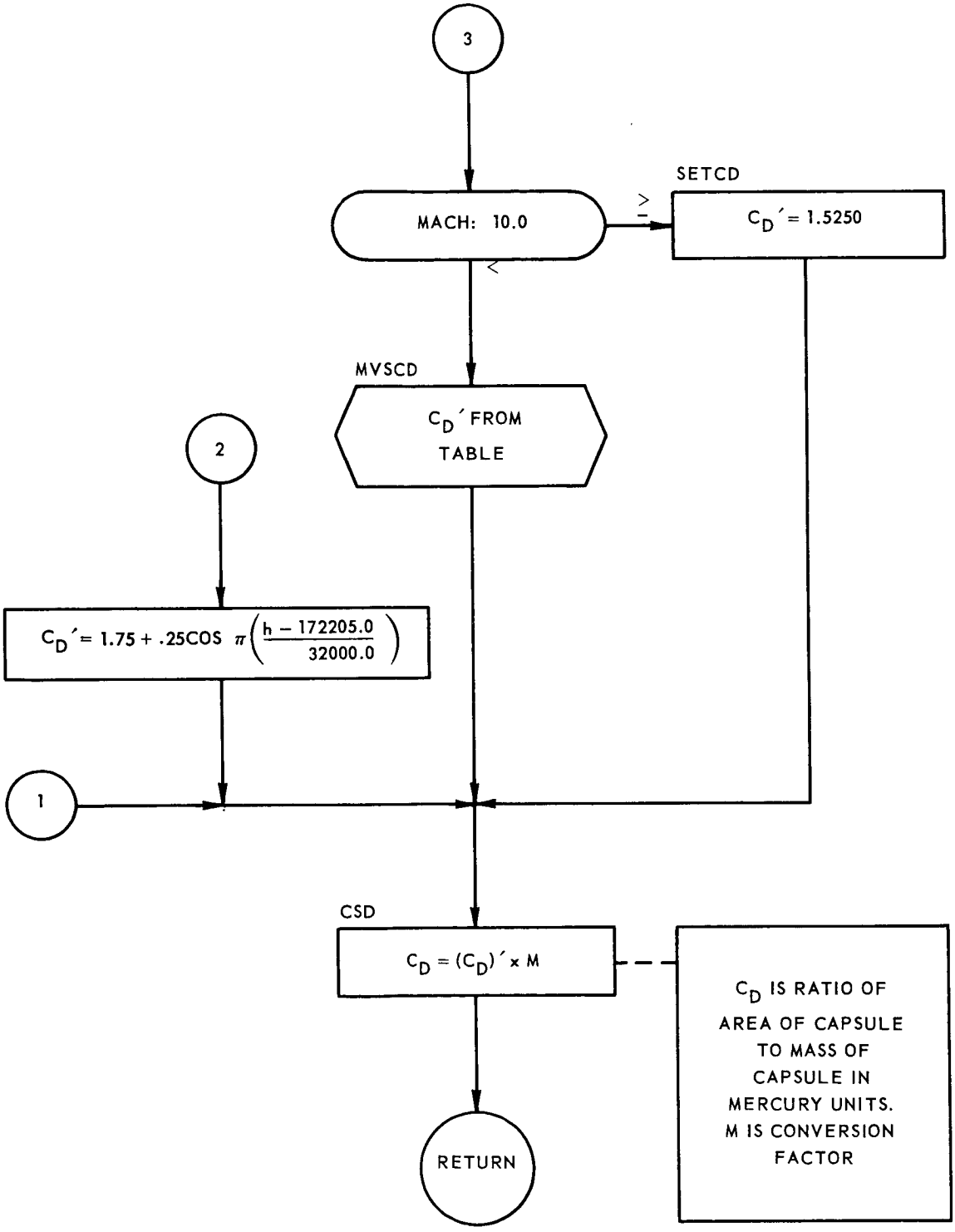


FIGURE 4-9. N4COEF PROGRAM FLOW CHART *(Sheet 2 of 2)*

4.9 NUMERICAL INTEGRATION TABLE COMPILER PROGRAM (N5OTBL)

The purpose of N5OTBL is to build the integration output table in the required format using $\bar{r}$ or $\bar{r}$ and $\bar{v}$ values produced by the N2EXCR program. The flow chart for the N5OTBL program is shown in Figure 4-10.

4.9.1 Input Requirements

Inputs to the N5OTBL program are:

TNFUNK + 11	X	$\bar{r}$ vector
TNFUNK + 24	Y	
TNFUNK + 37	Z	
TNFUNK + 12	v_x	$\bar{v}$ vector
TNFUNK + 25	v_y	
TNFUNK + 38	v_z	
(K000TI)	Present time	
(K000ST)	Starting time	
(K000TL)	Table location	
(K000TL) I)	Integration output interval	
(KR0RRV)	Indication of $\bar{r}$ or $\bar{r}$ and $\bar{v}$ numerical integration table. 0 indicates $\bar{r}$ table I indicates $\bar{r}$ and $\bar{v}$ table	

4.9.2 Output Requirements

The output table of the N5OTBL program is in the following format:

(K000TL)	0 min I sec	Indication of minutes or seconds	
	1	Fixed point minutes or seconds as indicated by first location	Integration output interval
	2		Starting time
	3		Ending time

For $\bar{r}$ Numerical Integration Table	For $\bar{r}$ and $\bar{v}$ Numerical Integration Table
r_x^1	r_x^1
r_y^1	r_y^1
r_z^1	r_z^1
r_x^2	v_x^1
r_y^2	v_y^1
r_z^2	v_z^1
	r_x^2
	r_y^2
	r_z^2
	v_x^2
	v_y^2
	v_z^2

4.9.3 Method

The following two equations are used in the N5OTBL program:

$\bar{r}$ numerical integration table

$$\left\{(K000TL) + 4 + 3 \left[(K000TI) - (K000ST)\right]\right\} \Big/ \left[C\,(K000TL) + 1)\right]$$

$\bar{r} + \bar{v}$ numerical integration table

$$\left\{(K000TL) + 4 + \left[6\ (K000TI) - (K000ST)\right]\right\} \Big/ \left[C\,(K000TL + 1)\right]$$

Use these equations to search for correct location to store $\bar{r}$ and $\bar{v}$ values.

4.9.4 Usage

a) Calling Sequence:

α	TSX	N5OTBL, 4
$\alpha + 1$		Normal return

b) Space Required -- 42 locations for the main program; 83 additional locations (K locations and temporary storage).

c) Constants:

K000TI	Present time
K000ST	Start time
K000TL	Table location
KRORRV	Indication of $\bar{r}$ or $\bar{r}$ and $\bar{v}$
K00001	1
K00003	3
K00006	6
K00010	10

d) Common Locations -- TNFUNK (39 locations)

e) Time Required -- 1.8 milliseconds

Syn Cards for N5OTBL		
K000TI	SYN	TNIINT+6
K000TL	SYN	TMBBNI+7
KRORRV	SYN	TMBBNI+13
K000ST	SYN	TNIINT+7

4.9.5 Checkout

The N5OTBL program has been operating in the Mercury Program System since December, 1960.

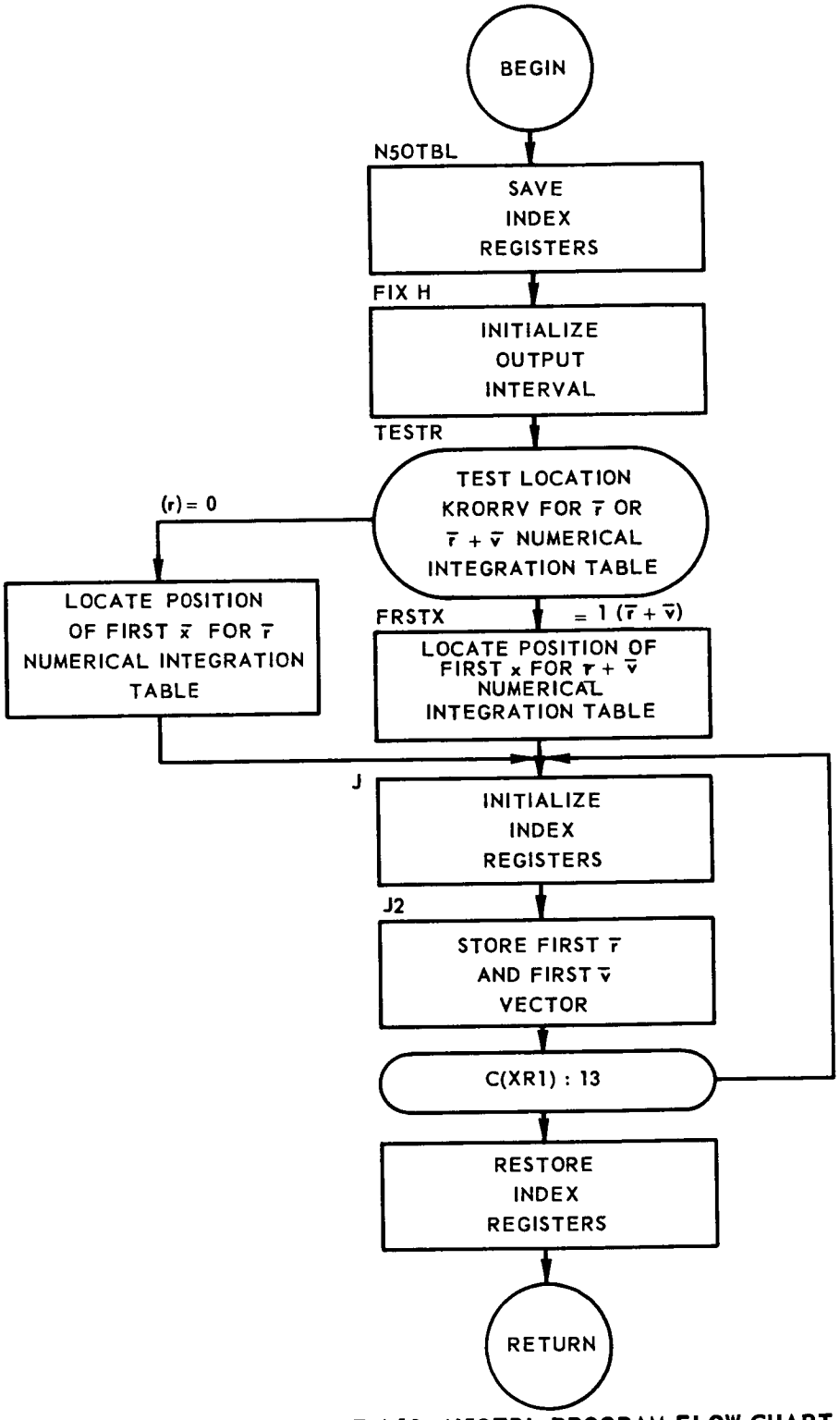


FIGURE 4-10. N5OTBL PROGRAM FLOW CHART

4.10 EARTH'S EFFECT ON CAPSULE CALCULATION PROGRAM (N60PEP)

The N60PEP program calculates the force on the capsule due to earth's potential. The flow chart for this program is shown in Figure 4-11.

4.10.1 Input Requirements

The square root library subroutine, U1SQRT, is used with this program. the following information from the TNFUNK table serves as input to N60PEP:

Input	Symbolic Location	
X	TNFUNK + 11	Floating Point
Y	TNFUNK + 24	
Z	TNFUNK + 37	

4.10.2 Output Requirements

Output from the N60PEP consists of the following data:

$\frac{\partial P}{\partial X}$	K00.XP	Floating Point
$\frac{\partial P}{\partial Y}$	K00.YP	
$\frac{\partial P}{\partial Z}$	K00.ZP	

4.10.3 Method

The following equations are used by the N60PEP program:

$$-\frac{\partial P}{\partial X} = -\frac{x}{r^3}\left\{1 - \frac{C2}{2r^2}\left[5\left(\frac{Z}{r}\right)^2 - 1\right]\right\} = XM \tag{1}$$

$$-\frac{\partial P}{\partial Y} = -\frac{y}{r^3}\left\{1 - \frac{C2}{2r^2}\left[5\left(\frac{Z}{r}\right)^2 - 1\right]\right\} = YM \tag{2}$$

$$-\frac{\partial P}{\partial Z} = -\frac{z}{r^3}\left\{1 - \frac{C2}{2r^2}\left[5\left(\frac{Z}{r}\right)^2 - 3\right]\right\} = Z\left(M + \frac{C2}{r^5}\right) \tag{3}$$

$$r = \sqrt{X^2 + Y^2 + Z^2} \tag{4}$$

4.10.4 Usage

a) Calling Sequence

α	TSX	N60PEP,4
$\alpha+1$	Normal return	

(Error return in U1SQRT library subroutine is handled by Monitor. Return is to a location in Monitor.)

b) Space Required

64 locations (Main Program) plus U1SQRT and 10 temporary locations designated by TEMN3.

c) Constants

Symbols	Value
K00.C2	.00324696
K00000	0
K001.0	1.0
K002.0	2.0
K003.0	3.0
K005.0	5.0

d) Tables Used:

TNFUNK - 38 locations

TEMN 3 - 10 locations

e) Timing

14.424 milliseconds

SYN CARDS FOR N60PEP

K00.XP	SYN	TNIINT
K00.YP	SYN	TNIINT + 1
K00.ZP	SYN	TNIINT + 2

4.10.5 Checkout

The N60PEP program has been operating in the Mercury Program System since March, 1961.

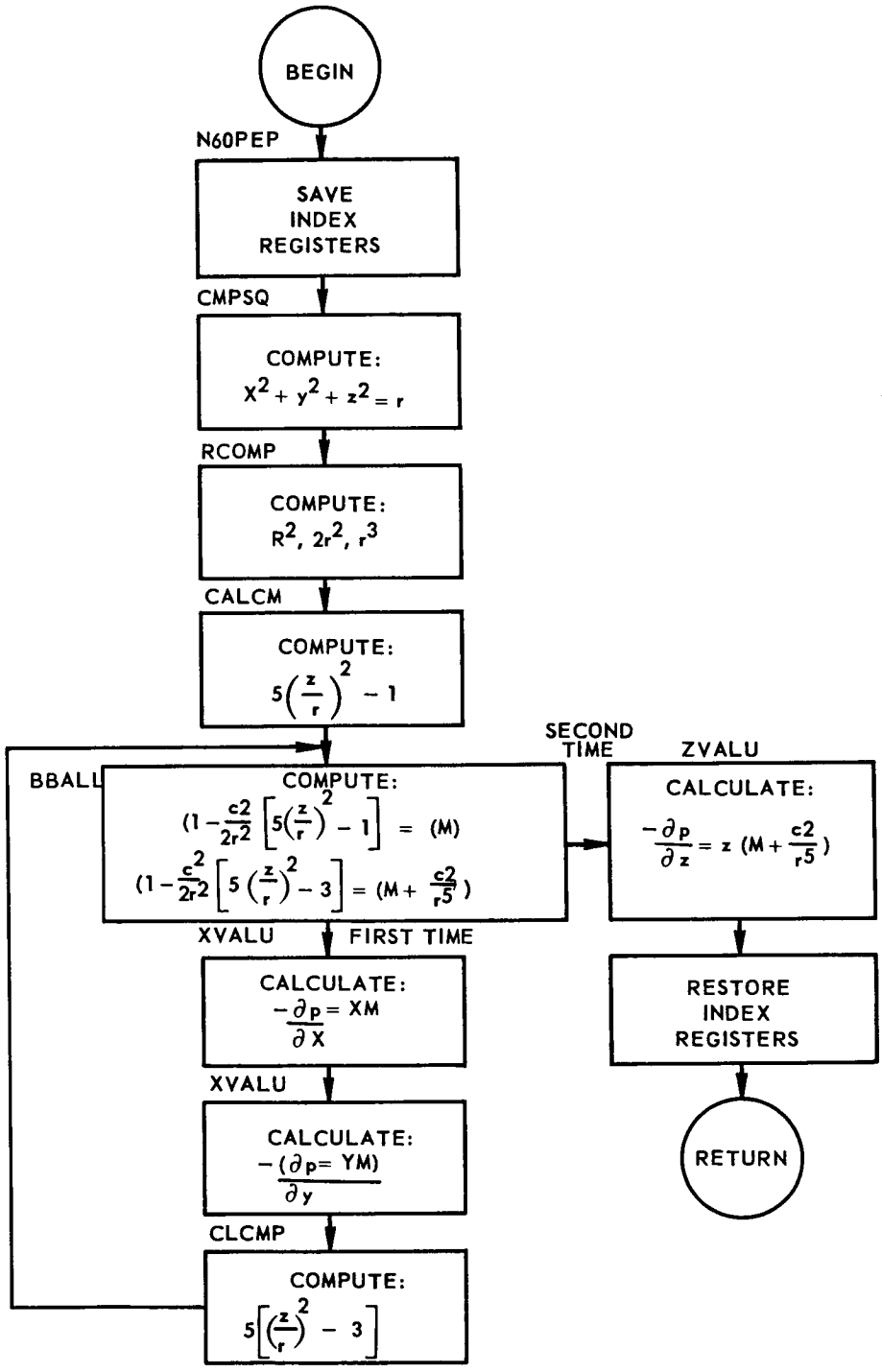


FIGURE 4-11 N60PEP PROGRAM FLOW CHART

4.11 VARIABLE STEP INTEGRATION EQUATIONS PROGRAM (N7VARS)

The purpose of this program is to decrease the step time of integration by one-half and to calculate F values for half step. The flow chart for N7VARS is shown in Figure 4-12.

4.11.1 Input Requirements

The following serve as input to the N7VARS program:

a) K000TI: Present time of integration; if TN7HSI is $\neq 0$, time is nearest unit.

b) K0000H: Integration time step; if TN7HSI is $\neq 0$, time step is to nearest unit.

c) K000.H: Time step in Mercury units.

d) K00.2H: $(K000.H)^2$ in Mercury units.

e) KINTVN: Number of steps per output.

f) TNFUNK: Function table (see Table 4.1).

g) TN7N12: Indicator = 0 - entrance from N1SUFT therefore do not calculate half step F values.

Indicator $\neq 0$ - entrance from N2EXCR.

h) TNSAVR: Last $\overline{r}$ and $\overline{v}$ were saved in these locations (f).

i) TN7ICT: Counter to determine the number of times the time step was decreased for convergence in a particular interval.

j) TN7HSI: Indicator = 0 - time step is a unit number.

Indicator $\neq 0$ - time step is less than unity and is expressed in floating point units.

k) TNIFPT: Present time of integration in floating point units.

l) TEMN0 + 61: Indicator = 0 - forward integration in process.

Indicator $\neq 0$ - backward integration in process.

4.11.2 Output Requirements

The output of N7VARS consists of the following information:

a) K000TI: Former T_i adjusted for decrease in time step.

b) K0000H
c) K000.H
d) K00.2H

All values associated with time step divided by 2.

e) KINTVN: Number of steps per output increased.

f) TNFUNK: Only if entrance is made from N2EXCR re-evaluated function table with half-line values calculated.

4.11.3 Method

The following equations are used by the N7VARS program:

$$F_{i \pm 1/2} = \sum_{k=0}^{5} T_R F_{i-k} \quad \text{specialized deviation} \quad (1)$$

$$F_{i \pm 3/2} = \sum_{k=0}^{5} T'_R F_{i-k} \quad \text{of Lagrangian six point} \quad (2)$$

$$F_{i \pm 5/2} = \sum_{k=0}^{5} T''_R F_{i-k} \quad \text{interpolation.} \quad (3)$$

$$''F_i = R_i - \sum_{k=1}^{6} A_k F_{i \pm k} \quad \text{(forward or backward)} \quad (4)$$

$$'F_{i-1/2} = (h) V_i - \sum_{k=1}^{6} A'_i F_{i-k} \quad \text{(forward)} \quad (5)$$

$$'F_{i \pm 1/2} = (h) V_i + \sum_{k=1}^{6} A'_k F_{i+k} \quad \text{(backward)} \quad (6)$$

From Subsection 4.4.3

4.11.4 Usage

a) Calling Sequence

TSX N7VARS, 4

Normal Return

b) Error Codes

Program transfers to error return if interval is reduced too small (TN7ISI < K00HSI) or if the number of steps per output (KINTVN) exceeds decrement storage (15 bits).

c) Space Required

150 locations + 10 temporary locations designated by TEMN7.

d) Special Usage

The following are three possible cases of non-convergence where N7VARS would be used:

a) N1SUFT while setting up function table 1) TN7N12 = 0 as input to N7VARS, 2) C(AC) = 0 on return from N7VARS.

b) N2EXCR while being used to set up function table (N1SUFT) 1) TN7N12 = 0 as input, 2) C(AC) = 0 on return from N7VARS.

c) N2EXCR while being used to update already established function table 1) TN7N12 = 0 as input, 2) C(AC) $\neq$ 0.

In cases (a) and (b), N7VARS will only change interval and initialize constants depending on the time step.

e) Constants

K002.0	DEC	2.0	
K004.0	DEC	4.0	
K000.5	DEC	0.5	
K00.A1...6		see 4.4.4, c	
K0.A.1P...6P		see 4.4.4, c	
K00HSI	DEC	1/16 (smallest time step desired)	
TNVS.C	DEC	.24609375	T_R, R = 0, 5 (in 4.11.3, Eq. 1)
TNVS.C + 1	DEC	-.02734375	T_R, R = 0, 5 (in 4.11.3, Eq. 1)
TNVS.C + 2	DEC	.01171875	T_R, R = 0, 5 (in 4.11.3, Eq. 1)
TNVS.C + 3	DEC	1.23046875	T_R, R = 0, 5 (in 4.11.3, Eq. 1)
TNVS.C + 4	DEC	.41015625	T_R, R = 0, 5 (in 4.11.3, Eq. 1)
TNVS.C + 5	DEC	-.09765625	T_R, R = 0, 5 (in 4.11.3, Eq. 1)

TNVS.C + 6 DEC	-.8203125	T'_R, R = 0, 5 (in 4.11.3, Eq. 2.)
TNVS.C + 7 DEC	.8203125	
TNVS.C + 8 DEC	.5859375	
TNVS.C + 9 DEC	.4921875	
TNVS.C + 10 DEC	-.2734375	T'_R, R = 0, 5 (in 4.11.3, Eq. 2)
TNVS.C + 11 DEC	.5859375	
TNVS.C + 12 DEC	-.17578125	T''_R R = 0, 5 (in 4.11.3, Eq. 3)
TNVS.C + 13 DEC	.08203125	
TNVS.C + 14 DEC	-.09765625	
TNVS.C + 15 DEC	.02734375	
TNVS.C + 16 DEC	-.01171875	
TNVS.C + 17 DEC	.01171875	

f) Timing

Entrance from N1SUFT

K0000H $\geq$ 1	59 cycles	(7090)
K0000H < 1	79 cycles	(7090)

Entrance from N2EXCR

4.11.5 Checkout

The N7VARS program coding was checked by hand calculation using the equations in 4.11.3. The actual formulation of Eq. 1, 2, and 3 was checked by: 1) generating an integration table with data such that it would be necessary to use a smaller interval than designated thus using N7VARS, 2) reproducing this integration table but using the reduced interval all the way, 3) comparing r and v values in both tables showed six place accuracy at t_i immediately after reduction of the interval.

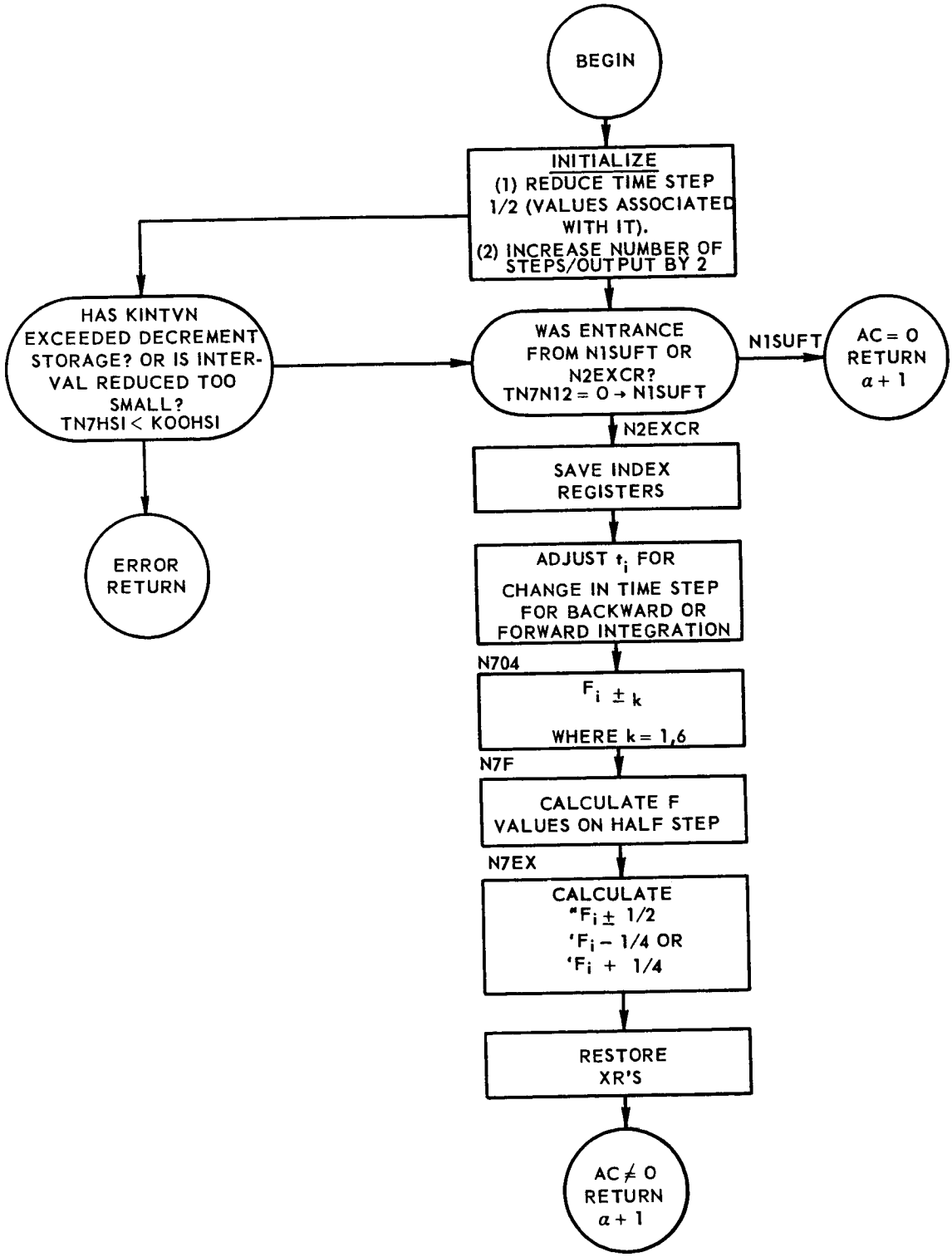


FIGURE 4-12. N7VARS PROGRAM FLOW CHART

4.12 DIFFERENTIAL CORRECTION

During a Mercury mission the Goddard 7090 computers accept radar tracking data from the worldwide tracking network. The computers use this data to establish the present position of the capsule to improve the orbit determination so successive radar stations can acquire the capsule, and to supervise re-entry. Differential correction uses site radar data to correct orbital position and velocity parameters. A brief discussion of the mathematical techniques employed in differential correction follows.

If the position and velocity vectors of a capsule in orbit are known, its coordinates at any future time can be predicted. This computation is accomplished by numerically integrating the three second-order differential equations of motion using Cowell's method. However, the accuracy of this prediction is dependent upon the accuracy of the initial vectors and on the precision with which the perturbations due to the earth's shape and atmosphere are represented. The accuracy of the initial vectors, $\overline{r}_s$ and $\overline{v}_s$, is affected by measurement errors in radar equipment. The perturbations are approximated using models of the earth's shape and atmosphere based on the latest available information. Hence, to account for these deviations to the elliptical orbit, the predictions must be continually improved or updated using radar observations. The method which accomplishes the updating task is called differential correction.

Assume that a numerical integration prediction table has been generated by N0CPNI using Cowell's method based on the position and velocity vectors, $\overline{r}_s$ and $\overline{v}_s$, at time t_s. The six components of base vectors are referenced to an inertial coordinate system. Hence, a set of values t_i, x_{1i}, x_{2i}, x_{3i}, x_{4i}, x_{5i}, x_{6i} and i ranges over the orbit lifetime are obtained. In addition, assume that a set of observations t_{jk}, R_{jk}, A_{jk}, E_{jk} (or, in vector notation; t_{jk}, F_{jk}) in local coordinates are available where:

R_{jk} = k^{th} observation of range at the jth radar station

A_{jk} = k^{th} observation of azimuth at the jth radar station

E_{jk} = k^{th} observation of elevation at the jth radar station

t_{jk} = time of the kth observation at the jth radar station

At any t_{jk}, we can interpolate into the numerical integration table and obtain a corresponding $\overline{r}$. Knowing the position of the jth station at t_{jk}, then R'_{jk}, A'_{jk},

E'_{jk} is obtained which is the predicted k^{th} observation at the jth station based on $\bar{r}_s$ and $\bar{v}_s$.

Now $\Delta F = F_{jk} - F'_{jk}$ represents, except for measurement errors, the difference between the actual position of the capsule at t_{jk} and its predicted position. Since F is a function of the six independent variables x_{ms} $(m = 1, \ldots, 6)$, first order differential effects are considered by use of the total differential (called an Equation of Condition):

$$dF_{jk} = \sum_{m=1}^{6} \frac{\partial F_{jk}}{\partial x_{ms}} \Delta x_{ms} = \Delta F_{jk}$$

Where each Δx_{ms} is an incremental change in the corresponding component of the position and velocity vectors $\bar{r}$ and $\bar{v}$. (Note: these vectors are not constant through an orbit, but are shifted at each of the radar stations.) In this equation, the partial differential coefficients and ΔF_{jk}, are known at each t_{jk}. Hence, a system of equations with six unknowns is obtained. These equations are weighted to account for measurement errors which vary with the radar stations and the individual components of F_{jk}. A solution is then obtained by the weighted least squares technique. After establishing that this solution, is, in fact, an improvement, a numerical integration table is generated about the new $\bar{r}_s$ and $\bar{v}_s$ and the process is repeated until the Δx_{ms} are insignificant.

The D0DIFC program writeup designed to direct and control the differential control function together with its subprograms are included in subsections 4.13 to 4.24.

4.13 DIFFERENTIAL CORRECTION CONTROL PROGRAM (D0DIFC)

D0DIFC is the control program for 14 programs which perform the differential correction function in the Mercury Program System. The flow chart for the D0DIFC program is shown in Figure 4-13.

4.13.1 Input Requirements

To accomplish its function D0DIFC requires the following Mercury programs and library routines:

Mercury Programs

D1PCDC	D2CSFE	D2SLNE
D2SSMT	D2SUNE	D2WEQT
D3CDRM		D3WTLU
D4CFRV		
RINDCP	R2AINT	

Library Subroutines

U7INTP, U1SQRT, UA1LSC

The following tables serve as input to the D0DIFC program:

a) TMSTMS - a table of up to 15 words of the following format:

Sign: (+) if additional words follow

(-) if this is the last word

Address: address of the first location of a block of edited radar data from one station.

The control words are arranged in descending order of capsule crossing; that is, the control word pertaining to the last station crossed is listed first. Stations are dropped when t_a is greater than a preset time interval from t_a for the last station reporting. Only one station may be added. The position of the added control word, relative to TMSTMS, is placed in the A (AC). Thus a zero in the A(AC) means that the added control word is in TMSTMS, a 1 indicates

that it is in TMSTMS +1, etc. If no control word is added, that is, if a second or subsequent differential correction is to be computed with the original data, the S(AC) will be minus and plus otherwise. No more than 15 edited data blocks are available, so TMSTMS contains no more than 15 control words. The locations of edited blocks of messages from all stations crossed in the last full orbit are covered by this technique. No messages are made available to the orbit computing program unless the station has stopped transmitting for that pass, and the data has been edited.

b) TNINT1 - A block containing an orbit integration table in the following format:

TNINT1		Form of table: 0 = minutes, 1 = seconds.
+1	Fixed point	Interval at which entries are made in the table.
+2		The starting time of the table.
+3		The ending time of the table.
+4	Floating point	X
+5		Y
+6		Z
+7		$\dot{X}$
+8		$\dot{Y}$
+9		$\dot{Z}$

etc.

c) TNINT2 - Re-entry table, same format as above.

d) The $\bar{r}$, $\bar{v}$ and t from the last differential correction for the present station are available to D0DIFC in locations TDRANV to TDRANV + 6.

e) Monitor supplies the station characteristics block in the format shown in Table 4-3.

f) TMSTCH - A 37-word reference table with the address -1 of the station characteristic block for each station. Each word in the block contains the location of the word preceding the first word in the desired station block. For example:

TABLE 4-3. STATION CHARACTERISTICS BLOCK CONTENTS

Word No.	
1	Prefix: Site Type (0 = AN/FPS-16; 1 = Verlort; 2 = telemetry) Decrement: Earth Sector Number Address: DCC Input Subchannel Number
2	Prefix: Other radars at this site (7 = AN/FPS-16; 1 = Verlort; 0 = none) Address: Internal Station Number
3	DCC Output Subchannel Mask (Subchannel 10=PZE 0,0,16; Subchannel 11=PZE 0,0,8)
4, 5	Station Name (in BCD)

	Item	Source	Source Format	Source Unit	Converted Format	Converted Unit
6	Geodetic Longitude (λ)	Given	Dec.	DMS*	Flt. Pt.	Radians
7	Geodetic Latitude (ϕ)	Given	Dec.	DMS*	Flt. Pt.	Radians
8	Geocentric Latitude (ϕ')	Computed			Flt. Pt.	Radians
9	sin ϕ	Computed			Flt. Pt.	Radians
10	cos ϕ	Computed			Flt. Pt.	Radians
11	R_s sin ($\phi-\phi'$)	Computed			Flt. Pt.	Earth Radii
12	R_s cos ($\phi-\phi'$)	Computed			Flt. Pt.	Earth Radii
13	Radius from Earth Center to Sta. (R_s)	Computed			Flt. Pt.	Earth Radii
14	Altitude above Ellipsoid (H)	Given	Dec.	Meters	Flt. Pt.	Earth Radii
15	Local Vertical Deflection Longitude ($\Delta\lambda$)	Given	Dec.	DMS*	Flt. Pt.	Radians
16	Local Vertical Deflection Latitude ($\Delta\phi$)	Given	Dec.	DMS*	Flt. Pt.	Radians
17	Calibrated Boresight Elevation	Given	Dec.	DMS*	Flt. Pt.	Radians
18	Calibrated Boresight Azimuth	Given	Dec.	DMS*	Flt. Pt.	Radians
19	Inertial Longitude at Reference Time	Computed			Flt. Pt.	Radians
20	Air Pressure at Ground Level (P_q)	Given	Dec.	Millibars	Flt. Pt.	Millibars
21	Temperature at Ground Level (T_g)	Given	Dec.	°C	Flt. Pt.	°K
22	Water Vapor Content at Ground Level (e_g)	Given	Dec.	Millibars	Flt. Pt.	Millibars
23	Azimuth Deviation from True North	Given	Dec.	DMS*	Flt. Pt.	Radians
24	Square Root of Weight 1 (Range)	Given	Dec.		Flt. Pt.	–
25	Square Root of Weight 2 (Azimuth)	Given	Dec.		Flt. Pt.	–
26	Square Root of Weight 3 (Elevation)	Given	Dec.		Flt. Pt.	–
27	Modulus (n-1)	Computed			Flt. Pt.	–
28	R sin ($\phi-\phi'$)	Computed			Flt. Pt.	Earth Radii
29	R cos ($\phi-\phi'$)	Computed			Flt. Pt.	Earth Radii
30	sin ϕ'	Computed			Flt. Pt.	Radians
31	cos ϕ'	Computed			Flt. Pt.	Radians
32	Boresight Elevation Correction (C_3)**	Computed			Flt. Pt.	Radians
33	Boresight Azimuth Correction (C_2)**	Computed			Flt. Pt.	Radians
34	Open for Expansion					

*DMS—degrees, minutes and seconds
**Original Input—raw radar reading

TMSTCH	blank
+ 1	Loc - 1 of Station Char. Block No. 1
+ 2	Loc - 1 of Station Char. Block No. 2
+ 3	Loc - 1 of Station Char. Block No. 3
etc.	

g) Station message blocks in the following format:

Word - 1. Station identification number in address, pass number in decrement.

Word - 2. Number of radar messages in this block.

Word - 3. Location, relative to word 1, of time for minimum range in this block. (Address field)

Word - 4. Time (T_1) fixed point, minutes in address, seconds in decrement.

Word - 5. Range (R_1) floating point in Mercury units.

Word - 6. Azimuth (A_1) floating point in radians.

Word - 7. Elevation (E_1) floating point in radians.

Word - 8. T_2
Word - 9. R_2
Word - 10. A_2
Word - 11. E_2
} 2nd radar message

An erroneous message is indicated by + 1.

h) MCNRRF

= 0 during orbit

≠ 0 during re-entry

i) TFESAB

The first even second after burnout in fixed point seconds B35.

4.13.2 Output Requirements

The following are the output requirements of D0DIFC:

a) At the completion of D0DIFC, a transfer is made to MFDIFC with the accummulator in the following format:

AC = 0	perform numerical integration
AC ≠ 0	do not perform numerical integration
S(AC) = +	do not perform another differential correction until new data has been collected.
S(AC) = -	perform another differential correction before supplying new data.
1 in P(AC)	input for numerical integration is in table TNNDCI (for 6 NDC tables).
0 in P(AC)	input for numerical integration is in table TDRANV (normal setting).

b) TDRANV - if indication has been made to perform an integration, the 15-location table shown in Table 4-4 is set as input.

c) TDPRNT - 100 word output block used for printing on-line messages of differential correction results (see TDPRNT table format shown in Table 4-5).

TABLE 4-4 TDRANV TABLE FORMAT

Location	Contents	
TDRANV	X Coordinate	Floating point
+ 1	Y Coordinate	
+ 2	Z Coordinate	
+ 3	Velocity in X direction	
+ 4	Velocity in Y direction	
+ 5	Velocity in Z direction	
+ 6	Time; Fixed point units refer to + 9	
+ 7	1) TNINT1 if integration table desired 2) TNINT2 if re-entry table desired	
+ 8	Time stepping interval (integer); units refer to + 9	
+ 9	1) If bit position 35 = 1, interval is seconds 2) If bit position 35 = 0, interval is minutes	
+ 10	1) length of time to integrate backwards (integer) 2) zeros if no backward integration desired	
+ 11	1) length of time to integrate forward 2) zeros if no forward integration desired	
+ 12	1) zeros; use N4DRAG program 2) $\neq$ zero; do not use N4DRAG program	
+ 13	1) + 1, $\bar{r}$ and $\bar{v}$ in output table 2) zero; $\bar{r}$ values only in output table	
+ 14	n number of integration steps per output	

TABLE 4-5. TDPRNT TABLE FORMAT (Sheet 1 of 2)

Location	Contents
Word #1 Table	000 000 000 0XX (where XX is the number of stations giving data)
TDPRNT +1	ΔR
+2	
+3	
+4	ΔV
+5	
+6	
+7	Sigma: standard error of fit
+8	K_{-1}
+9	K_0
+11	$\sqrt{\sum_{i=1}^{3} \left(S_e \sqrt{Q_i}\right)^2}$
+12	$\sqrt{\sum_{i=4}^{6} \left(S_e \sqrt{Q_i}\right)^2}$
+13	Internal Station Numbers
.	
.	
.	
+26	
+27	Equations per station 1st Edit (in the same order as station numbers)
.	
.	
.	
+41	
+42	2nd Edit
.	
.	
.	
+56	

TABLE 4-5. TDPRNT TABLE FORMAT (Sheet 2 of 2)

Location	Contents
TDPRNT +57 . . . +71	3rd Edit
+72 . . . +86	Residual/M
+87 . . . +89	A,E,I
+90 . . . +92	R
+93 . . . +95	V
+96	GMT associated with above R,V vectors in fixed point seconds.

Note: If the first word in the above table is 0, no information will be printed.

4.13.3 Method

The following discussion pertains to the comparison and acceptance of a new solution (position and velocity vectors).

Assume that a solution exists at the last "anchor point" and that a new solution has been obtained at the new "anchor point". The question is: Should the new solution be rejected because it is less reliable than the former one? Each solution has a "standard deviation" or "probable error of unit weight" and each unknown has a weight, Q_i, defined by

$$\text{probable error of } W_i = \left[W_i\right] = \left[\text{unit weight}\right] \sqrt{Q_i} = (\text{Standard deviation}) \sqrt{Q_i}$$

Read: $\left[X_i\right]$ as "the probable error of X_i".

Now the accuracy with which any position can be predicted is

$$\left[\bar{\rho}\right]^2 = \sum_{i=1}^{6} \left(\frac{\partial \bar{\rho}}{\partial W_i}\right)^2 \left[W_i\right]^2$$

In the present case, the independent variables, W_i, are the components of $\bar{r}_o$ and $\bar{v}_o$. Since, approximately,

$$\bar{r}(t_i) = f_i r_o + g_i \bar{v}_o \quad \text{and } \bar{\rho} = \bar{r} - \bar{R}$$

then, at the new "anchor point" $[R] = 0$

$$\left[\bar{\rho}\right]^2 = \left[\bar{r}\right]^2 = (Q_1 + Q_2 + Q_3) \quad (\text{S. E.})^2$$

However, this is independent of $\left[\bar{v}_o\right]$, because $g_o = 0$. The following criterion is used which implies that $f = 1 = g$, namely

$$K_o^2 = (\text{S. E.})_o^2 \sum_1^6 Q_i = \sum_1^6 \left(S_e \sqrt{Q_i}\right)^2$$

During orbit, this is a figure of merit associated with the most recent solution which characterizes the quality of the predictions at a time which is approximately fifteen minutes later, than the most recent "anchor point", i.e., $t_i = t_o + 15$ min. In the re-entry phase, we chose $t_i = t_o$, $g = 0$. This gives the form $K_o^2 = \sum_1^3 \left(S_e \sqrt{Q_i}\right)^2$

Using the same t_i, a figure of merit is now derived from the solution at the last "anchor point" to be compared with K_o above. For this purpose assume that an observation has been made at t_i, and we would then be obliged to compute the Equations of Condition at this point. In the course of such a calculation, the following would be derived:

$$\partial \bar{r} = \left[A_{ij} \right] \text{, a (3 x 6) matrix.}$$

Then let

$$K_{-1}^2 = (S.E.)_{-1}^2 \sum_{j=1}^{6} \left(A_{1j}^2 + A_{2j}^2 + A_{3j}^2 \right) Q_j$$

This represents $\left[\bar{\rho} \right]^2$ at t_i, based upon the previous solution. Therefore, if $K_o > b\ K_{-1}$, the new solution is rejected where "b" is some factor for determining convergence.

4.13.4 Usage

a) Calling Sequence

α TRA D0DIFC

D0DIFC returns control to Monitor at location MFDIFC.

b) Space Required

526_{10}

c) Error Returns

If an error occurred while processing a single observation at t_i, the observation t_i is neglected and the program attempts to continue. In all other cases, D0DIFC exits with an indication of a rejected correction but no printing will result on-line (TDPRNT 0).

d) System Constants

K0ZERO	DEC	0	Fixed point 0
KM0000	MZE		Fixed point minus 0
K00001	DEC	1	Fixed point 1
K00005	DEC	5	Fixed point 5

K00018	DEC	18	Fixed point 18
K00030	DEC	30	Fixed point 30
K00060	DEC	60	Fixed point 60
K00090	DEC	90	Fixed point 90
K000.5	DEC	0.5	Floating point .5
K0060.	DEC	60.	Floating point 60.0
K01.E3	DEC	1000.	Floating point 1000.
K00.SH	DEC	.12394486E-2	
KCH233	OCT	233 000 000 000	

e) Tables:

TAWTPC	BSS	88	Weighted partials maximum number of locations = C(K0000N) * C(K000MB).
TCORBP	BSS	16	Near circular orbit parameter functions.
TD2SSM	BSS	15	Sum (DF**2)/M for each station.
TDEQCT	BSS	24	The partial coefficients.
TDFNDC	PZE	- = No K-1 calc. + = calc. K-1	Indicator to stop K-1 calculation and comparing during numerical differential correction method.
TDNIET	PZE		C(TNETIM - [2(TNINT2 + 1)] in re-entry.
TDNOBR	BSS	10	Number of observations processed for each station in differential correction interval in re-entry.
TDRANV	SYN	KLAMDO + 1	15 locations of numerical integration input block.

TDTIMT	PZE		Indicator to calculate value for end of numerical integration table.
TDTOBS	PZE		Time of observation fixed point seconds; re-entry.
TEQCIN	BSS	27	Intermediate input or output block.
TFESAB	PZE		First second after burnout in fixed point.
TLSSSE	BSS	17	Solutions to least squares, SE, SE of elements.
TMRARF	BSS	15	Input table for numerical integration for first time in re-entry.
TMSTCH	BSS	37	Reference table for station characteristics table.
TMSTMS	BSS	15	Reference to radar message blocks.
TN4NEX	PON		Not zero = stop numerical integration at 60,000 ft. Zero = continue numerical integration.
TNDCIN	PZE		Zero = D1PCDC, not zero = numerical differential correction method for partials.
TNEARM	PZE		Anchor point for re-entry in fixed point seconds.
TNETIM	PZE		Time of last entry in numerical integration table.
TNINT1	BSS	2200	Orbit integration table.
TNINT2	BSS	1800	Re-entry integration table.
TNOMFS	BSS	15	Number of observations desired from each station reporting.

TNSCOR	DEC	1, 1, 1, 2, 2, 3, 4, 5, 6, 6, 7, 7, 8, 9, 10, 11, 11, 2, 2	
TNUMNI	PZE	0	Test for all 6 tables set up for numerical differential correction (0 = not use numerical differential correction, 6 = all set up).
T0DCHC	PZE		Time at 450,000 ft. used by differential correction with minutes in the address, seconds in decrement.
TPWTDF	BSS	11	W (DF) maximum number of locations (K000MB).
TRVTAP	BSS	13	Anchor point block (see SYN cards for format).
TSTWTS	BSS	33	Weight word table.
TSVSEE	BSS	6	Se_j - the six standard errors of the elements saved from the previous correction.

f) Constants

		TEQCIN BSS	27
K0SWTR	SYN	TEQCIN	Square root of station weight for R.
K0SWTA	SYN	K0SWTR + 1	Square root of station weight for A.
K0SWTE	SYN	K0SWTR + 2	Square root of station weight for E.
K00TOB	SYN	K0SWTR + 3	Time observed.
KRVECX	SYN	K00TOB +1	R_X
KRVECY	SYN	K00TOB +2	R_Y
KRVECZ	SYN	K00TOB +3	R_Z

KSTCHL	SYN	K00TOB 4	Location of the station characteristics block for station processed.
KTSUBI	SYN	K0SWTR 8	t_i in Mercury units (floating point)
K00ROB	SYN	K0SWTR + 9	Range observed.
K00AOB	SYN	K0SWTR + 10	Azimuth observed.
K00EOB	SYN	K0SWTR + 11	Elevation angle observed.
KVVECX	SYN	K0SWTR + 12	V_X
KVVECY	SYN	K0SWTR +13	V_Y
KVVECZ	SYN	K0SWTR + 14	V_Z
K00DRC	SYN	K0SWTR + 15	Calculated range.
K00DAC	SYN	K0SWTR + 16	Calculated azimuth.
K00DEC	SYN	K0SWTR + 17	Calculated elevation angle.
KMATRX	SYN	K0SWTR + 18	M matrix (uses 9 locations).
		TDCCNT	BSS 24
KCCNTR	SYN	TDCCNT	Counts number of times in large loop.
KSRIND	SYN	TDCCNT + 1	Minus sign-no new station came in.
KRESID	SYN	TDCCNT + 2	Indicates type of residual check.
KD2CNT	SYN	TDCCNT + 3	Indicates 1st time in D2SSMT.
KD2SUT	SYN	TDCCNT + 4	Indicates 1st time in D2SUNE.
KDCSTT	SYN	TDCCNT + 5	Indicates 1st time in D2WEQT - set = 1.
KCDRMI	SYN	TDCCNT + 6	Indicator for D3CDRM - set = 1 for each station processed in interval.
K0LSMB	SYN	TDCCNT + 7	Contains location of station message block.

K0NDRM	SYN	TDCCNT + 8	Number of observations to be taken from each station.
K0000N	SYN	TDCCNT + 9	Number of partials, 6.
K00MCT	SYN	TDCCNT + 10	Number of equations bound for D2SUNE.
K0SUMM	SYN	TDCCNT + 11	Total number of equations for least squares solution.
KSUMWT	SYN	TDCCNT + 12	Sum of the weights.
KDFSQU	SYN	TDCCNT + 13	Sum (DF) squared.
KSAVSE	SYN	TDCCNT + 14	Saved from previous correction standard error.
KSEMPY	SYN	TDCCNT + 15	SE times residual check factor.
KINTIM	SYN	TDCCNT + 16	
KRETBL	SYN	TDCCNT + 17	Location to index TMORES table for D2WEQT.
KTOTLM	SYN	TDCCNT + 18	All equations from stations in interval using all observations not just desired.
KTDFSQ	SYN	TDCCNT + 19	Sum W (DF) squared for station in interval using all observations not just desired.
KINSDF	SYN	TDCCNT + 20	Indicator to D2WEQT to calculate residuals.
KSUBM1	SYN	TDCCNT + 21	K_{-1}
KSAVK0	SYN	TDCCNT + 22	K_0 from last correction.
KNEWK0	SYN	TDCCNT + 23	K_0
		TSETCT BSS 11	
KLASTS	SYN	TSETCT	Factor for K_{-1} comparison.
KSAMES	SYN	TSETCT +1	Factor for K_0 comparison.

KWTIND	SYN	TSETCT + 2	Zero - use D3WTLU (weight table).
KNOBBD	SYN	TSETCT + 3	Total number of observations desired from station.
KCPRVI	SYN	TSETCT + 4	Zero = RV
K000MB	SYN	TSETCT + 5	Upper bound of TPWTDF block.
KCCNBD	SYN	TSETCT + 6	Desired number of times for differential correction edit loops.
KSEFTR	SYN	TSETCT + 7	Residual check factor in floating point.
KTIMAH	SYN	TSETCT + 8	Number ahead to calculate K_{-1}.
KPERST	SYN	TSETCT + 9	Number of integration steps output interval.
KRDELT	SYN	TSETCT + 10	Output interval for numerical integration.
KDCBND	SYN	TSETCT + 11	Bound for number of differential corrections per station.

4.13.5 Checkout

The D0DIFC program has been operating in the Mercury Program System since February, 1961.

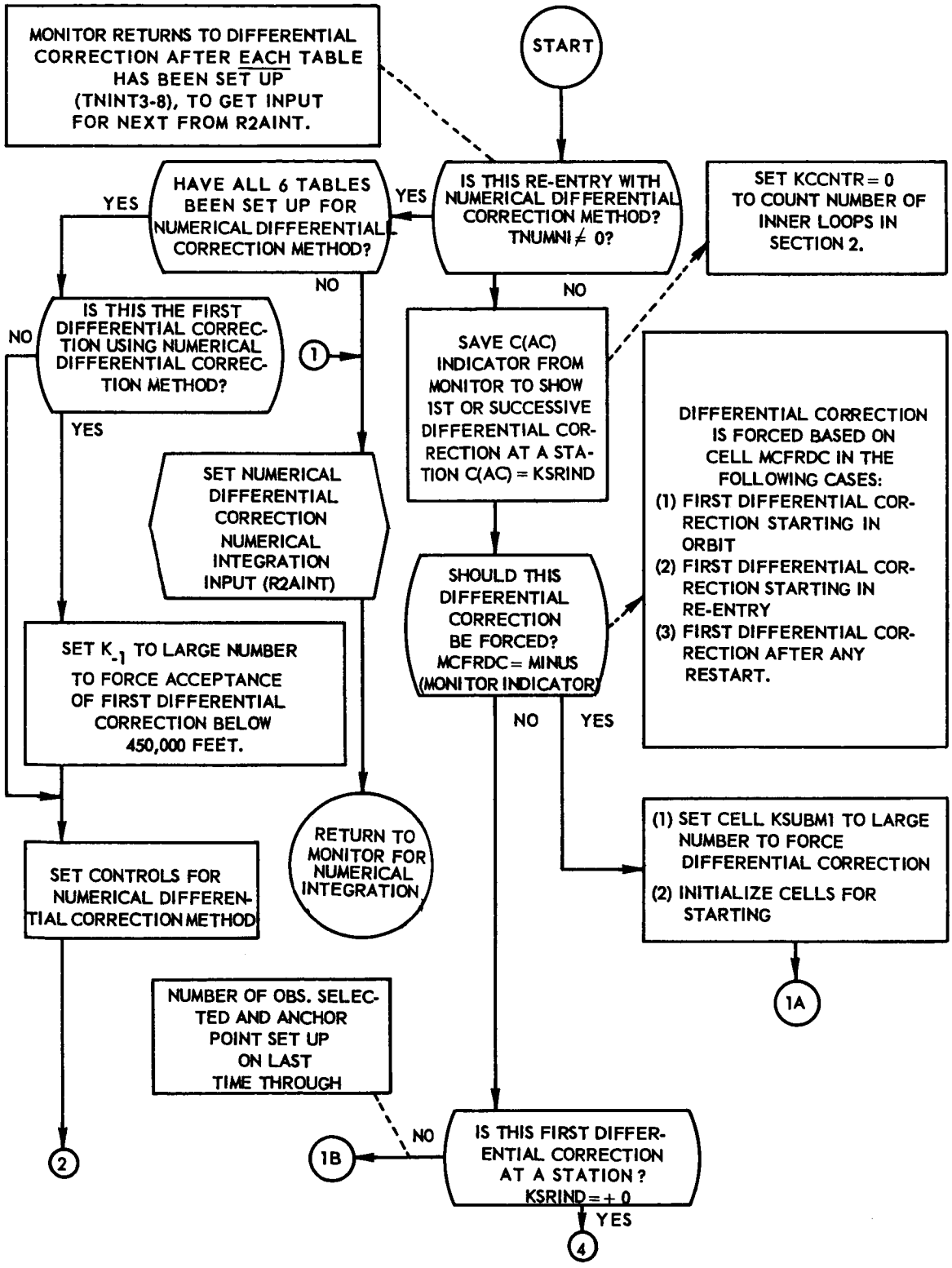


FIGURE 4-13 DODIFC PROGRAM FLOW CHART *(Sheet 1 of 4)*

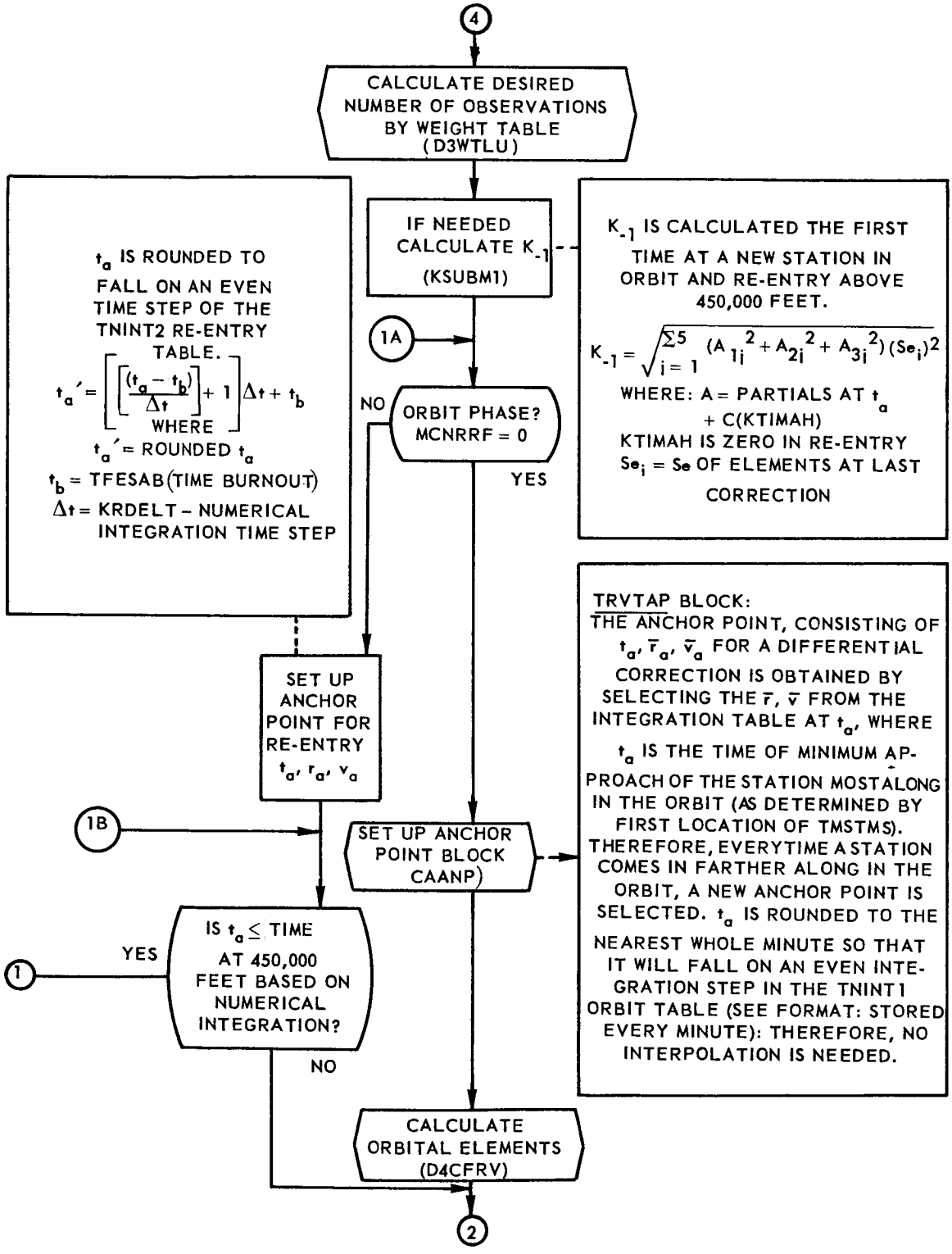


FIGURE 4-13 D0DIFC PROGRAM FLOW CHART *(Sheet 2 of 4)*

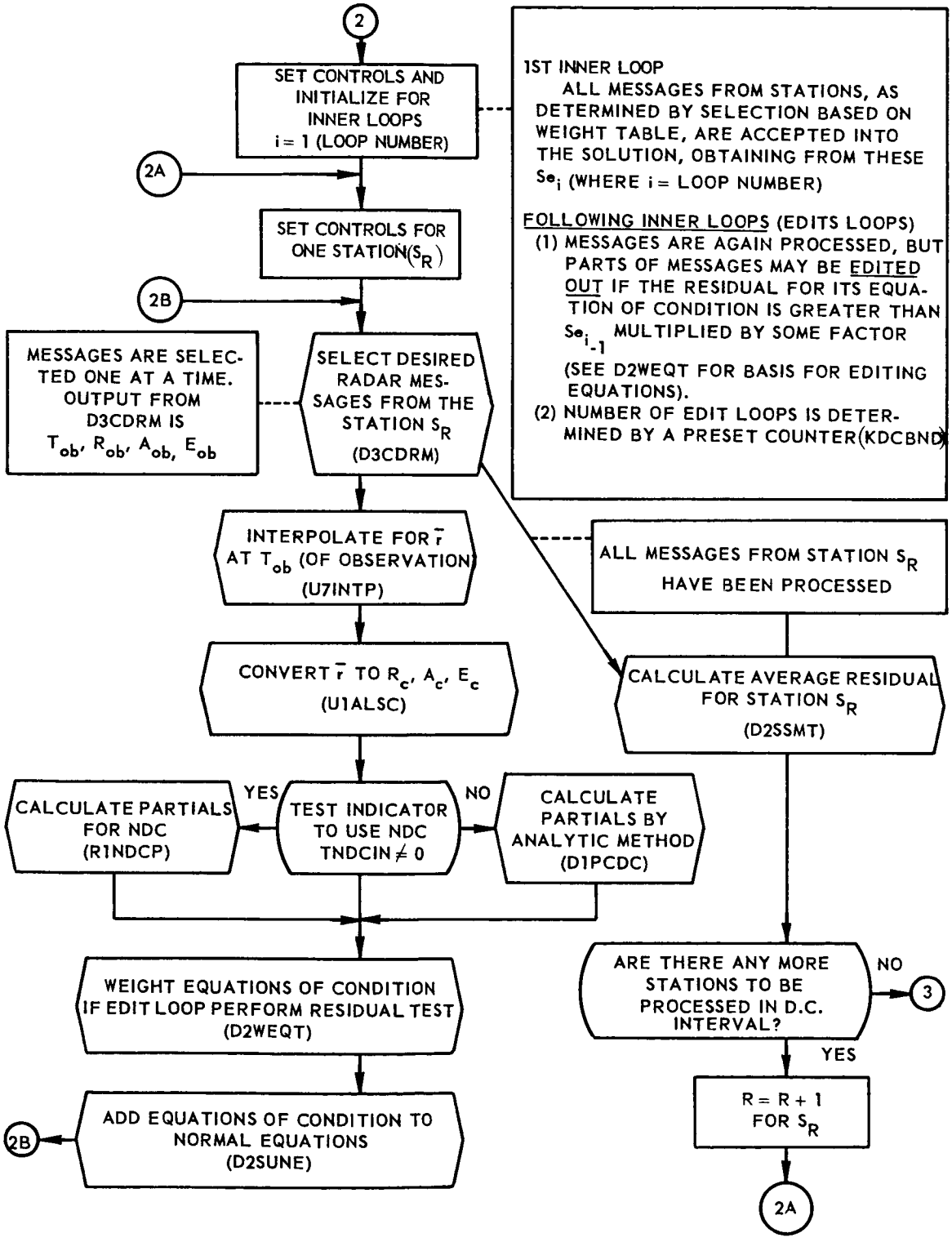


FIGURE 4-13 D0DIFC PROGRAM FLOW CHART *(Sheet 3 of 4)*

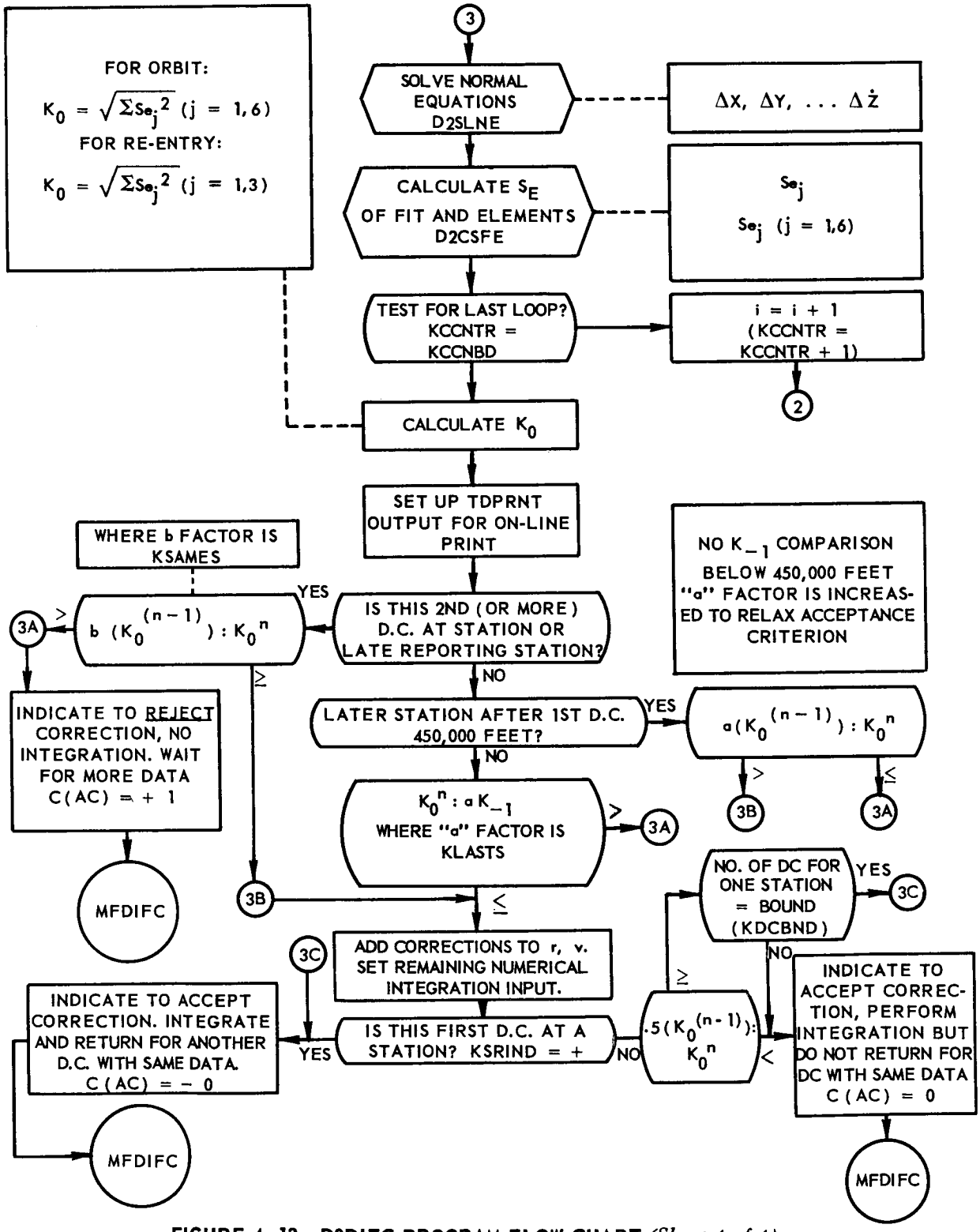


FIGURE 4-13. D0DIFC PROGRAM FLOW CHART *(Sheet 4 of 4)*

4.14 CALCULATE FUNCTIONS OF $\bar{r}$ AND $\bar{v}$ PROGRAM (D4CFRV)

The purpose of the D4CFRV program is to calculate the quantities (functions of $\bar{r}$, $\bar{v}$) from a given set of capsule position and velocity values. The flow chart for D4CFRV is shown in Figure 4-14.

4.14.1 Input Requirements

This program uses three library subroutines U1SQRT, U3XPR0, and U3D0TP. The following inputs requirements to the D4CFRV program are in floating point.

KRAVC1	The X coordinate
KRAVC2	The Y coordinate
KRAVC3	The Z coordinate
KVAVC1	The velocity along the X axis
KVAVC2	The velocity along the Y axis
KVAVC3	The velocity along the Z axis

4.14.2 Output Requirements

The output of the D4CFRV program consists of the following (in floating point):

KASUBO	a_0
KKMESQ	$1 - e^2$
KRABSL	$\sqrt{X^2 + Y^2 + Z^2}$
KVABSL	$\sqrt{VX^2 + VY^2 + VZ^2}$
KECCEN	e
KNSUBO	$a_0^{-3/2}$
KRVEC1	Large R vector
KREVC2	Large R vector

KREVC3 Large R vector

K.RTPO $\sqrt{P}$

4.14.3 Method

The following equations are used by the D4CFRV program:

$$\left|\bar{r}_0\right| = \left(X^2 + Y^2 + Z^2\right)^{1/2} \tag{1}$$

$$\left|\bar{v}_0\right| = \left(VX^2 + VY^2 + VZ^2\right)^{1/2} \tag{2}$$

$$a_0 = \frac{r_0}{2 - \bar{r}_0 \bar{v}_0^{\,2}} \tag{3}$$

$$e^2 = \left(\frac{a_0 - r_0}{a_0}\right)^2 + \frac{\left(\bar{r}_0 \cdot \bar{v}_0\right)^2}{a_0} \tag{4}$$

$$N_0 = a_0^{-3/2} \tag{5}$$

$$\sqrt{P} = \left|\bar{r}_0 \times \bar{v}_0\right| \tag{6}$$

$$\bar{R} = \frac{\bar{r}_0 \times \bar{v}_0}{\sqrt{P}} \tag{7}$$

D4CFRV is accurate to seven significant digits.

4.14.4 Usage

a) Calling Sequence

α	TSX	D4CFRV, 4
$\alpha + 1$		Error return
$\alpha + 2$		Normal return

b) Space Required

147 locations excluding locations for library subroutines

c) Constants Used

K001.0	DEC	1.0
K002.0	DEC	2.0

d) Eraseable Locations

TEMD4 through TEMD4 + 9

e) Timing Required

12.9 milliseconds

4.14.5 Checkout

The D4CFRV program was checked out by comparing its output with results of hand computations and with the output from the COG program.

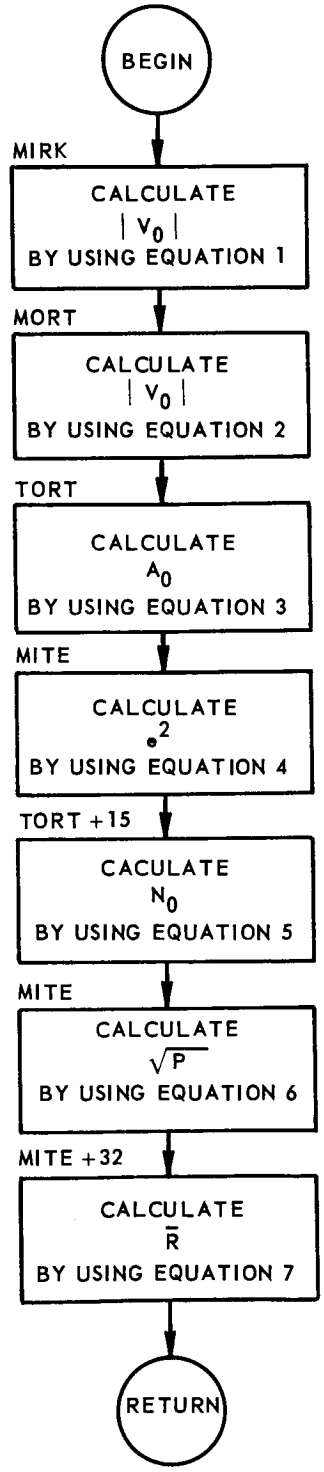


FIGURE 4-14. D4CFRV FROGRAM FLOW CHART

4.15 PARTIAL COEFFICIENT CALCULATION PROGRAM (D1PCDC)

The purpose of D1PCDC is to compute the partial coefficients for the equations of condition that arise from one observation. The flow chart for D1PCDC is shown in Figure 4-15.

4.15.1 Input Requirements

D1PCDC uses five library subroutines:

a) U1SQRT—square root

b) U3DOTP—vector dot product

c) U3MATM—matrix multiplication

d) U3XPRO—vector cross product

e) U1SICO—sine/cosine

The following parameters serve as input to the D1PCDC program.

KRAVC1 KRAVC2 KRAVC3	$\bar{r}_a$ (Anchor point r vector)
KVAVC1 KVAVC2 KVAVC3	$\bar{v}_a$ (Anchor point v vector)
KRVECX KRVECY KRVECZ	$\bar{r}_c$ (r vector computed)

KMATRX	M Matrix	(3 x 3, 9 locations)
KASUBO	a_0	semi-major axis
KKMESQ	$1 - e_0^2$	1 - eccentricity squared

KNSUBO	n_0	$a_0^{-3/2}$
KRVEC3	$\cos i_0$	i_0 = inclination
KTSUBO	T_a	Anchor point time (floating point, Mercury units)
KTSUB1	T_i	Time observed (floating point, Mercury units)
KRABSL	r_a	absolute value of $\bar{r}$
AAA	A	Azimuth (K00DAC)
HHH	E	Elevation (K00DEC)
ROW	R	Range (K00DRC)
JAAA	J = .00162348	constant

4.15.2 Output Requirements

The following data is outputted by the D1PCDC program:

TDEQCT	18 partial for differential correction.

4.15.3 Method

The following equations (from the Computation of Orbits by Dr. P. Herget) are used with the D1PCDC program:

$$\bar{r}_a \times \bar{v}_a \tag{1}$$

$$f = \frac{(\bar{r}_c \times \bar{v}_a) \cdot (\bar{r}_a \times \bar{v}_a)}{(\bar{r}_a \times \bar{v}_a) \cdot (\bar{r}_a \times \bar{v}_a)} \tag{2}$$

$$g = \frac{r_a^2 (\bar{r}_c \cdot \bar{v}_a) - (\bar{r}_a \cdot \bar{v}_a)(\bar{r}_c \cdot \bar{r}_a)}{(\bar{r}_a \times \bar{v}_a) \cdot (\bar{r}_a \times \bar{v}_a)} \tag{3}$$

$$F = r_a (1 - f) \tag{4}$$

$$G = \frac{g - F\left(\bar{r}_a \cdot \bar{v}_a\right)}{r_a} \tag{5}$$

$$T = t_i - t_o \tag{6}$$

$$L = \left(3T - g - G_{ra}\right) / r_i \tag{7}$$

$$P_3 = \left(a/r_a\right)\left(GL - 2F\right) \tag{8}$$

$$P_2 = \left[\left(F/r_i\right) G\right] / r_0$$

$$P_1 = \left\{\left[\left(G r_a\right) / r_i\right] G + P_3 + F\right\} / r_a^{\,3}$$

$$\bar{P} = \left(P_1, P_2, P_3\right)$$

$$q_3 = a\,(3g - 3T + FL) \tag{9}$$

$$q_2 = F^2/r_i$$

$$q_1 = q_3/r_0^{\,3} + P_2$$

$$\bar{q} = \left(q_1, q_2, q_3\right)$$

$$df = \bar{P}\,Y,\ dg = \bar{q}\,Y \tag{10}$$

$$\text{where } Y = \begin{pmatrix} \bar{r}_0 & O_0 \\ \bar{v}_0 & \bar{r}_0 \\ O_0 & \bar{v}_0 \end{pmatrix} \quad ;\ \text{a (3 x 6 Matrix)} \quad O = (0, 0, 0)$$

$$dr = \begin{pmatrix} f\,0 & 0\,g & 0\,0 \\ 0\,f & 0\,0 & g\,0 \\ 0\,0 & f\,0 & 0\,g \end{pmatrix} + \begin{pmatrix} \bar{r}_a \\ \bar{v}_0 \end{pmatrix}^T \begin{pmatrix} df \\ dg \end{pmatrix} \tag{11}$$

$$RAD = \begin{pmatrix} \text{Cos E Sin A}, & \text{Cos E Cos A}, & \text{Sin E} \\ \frac{\text{Cos A}}{R}, & \frac{-\text{Sin A}}{R} & O \\ \frac{-\text{Sin E Sin A}}{R}, & \frac{-\text{Sin E Cos A}}{R}, & \frac{\text{Cos E}}{R} \end{pmatrix} \tag{12}$$

$$\xi = J \cdot n \left\{ \left(\cos i_0\right) / \left[a\left(1-e^2\right)\right]^{2} \right\} T \tag{13}$$

$$B = \begin{pmatrix} \cos\xi & \sin\xi & 0 \\ -\sin\xi & \cos\xi & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{14}$$

$$A = \left[(RAD)\, M \right] B \tag{15}$$

Calculate A (dr) (16)

4.15.4 Usage

a) Calling Sequence

α	TSX	DIPCDC, 4
$\alpha + 1$		Error in U1SICO
$\alpha + 2$		Normal Return

b) Space Required

292 locations
+ 100 Temporary Locations
+ 22 K Locations
+ 18 Location table
+ Subroutines

K Locations needed

KTIMTO	PZE	$(T_i - T_a)$ in floating point, Mercury Units.
K001.0	DEC	1.0

```
K002.0      DEC     2.0

K003.0      DEC     3.0
```

plus the constants used in input and output.

c) Timing

141.3 milliseconds

4.15.5 Checkout

Checks on this program were made to 8 decimal places. Accuracy was exact except in a few instances where the value was off in the eighth place, probably due to rounding.

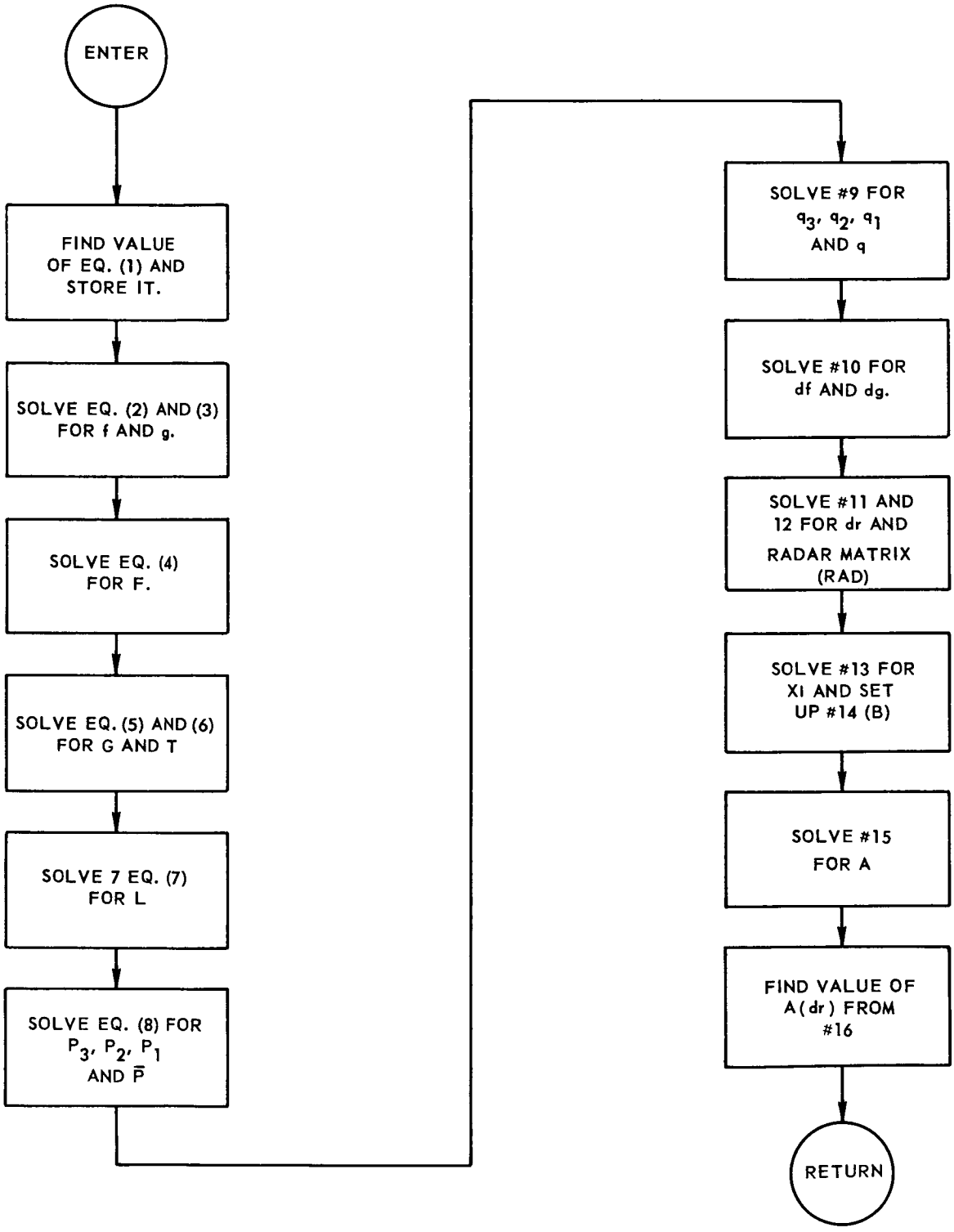


FIGURE 4-15 D1PCDC PROGRAM FLOW CHART

4.16 DEVELOPMENT OF SIX NUMERICAL INTEGRATION OUTPUT TABLES PROGRAM (R2AINT)

Given $\overline{r}_a$ and $\overline{v}_a$ at a time t_a during the re-entry of the capsule, set up six new tables of predicted position vectors by perturbing the components of $\overline{r}_a$ and $\overline{v}_a$, one at a time, by amounts ΔX_a, ΔY_a, ΔZ_a, $\Delta \dot{X}_a$, $\Delta \dot{Y}_a$, $\Delta \dot{Z}_a$, and integrating backward from time t_a to time t_i preceding the first second following burnout of the retrorockets and forward to 60,000 feet for the first table. The length of the first table is determined and the remaining five tables are set up to integrate forward the same amount of time as the first table. The flow chart for R2AINT is shown in Figure 4-16.

4.16.1 Input Requirements

The input to the R2AINT program is from the following locations:

a) A six word block, TRVTAP + 1, in the following format.

TRVTAP + 1	X_a	in floating point Mercury units.
+ 2	Y_a	in floating point Mercury units.
+ 3	Z_a	in floating point Mercury units.
+ 4	$\dot{X}_a$	in floating point Mercury units.
+ 5	$\dot{Y}_a$	in floating point Mercury units.
+ 6	$\dot{Z}_a$	in floating point Mercury units.

b) A location, TNEARM, which contains the anchor point time, t_a, in fixed point seconds and corresponds to the $\overline{r}_a$ and $\overline{v}_a$ found in the TRVTAP + 1 table. R2AINT assumes that the anchor point has been chosen at an integration step in the re-entry integration table.

c) A location, TN4NEX, is set equal to non-zero for the first table so as to integrate down to 60,000 ft. For the remaining 5 tables, TN4NEX is set equal to zero so as to assure integrating forward the amount of time specified in location TNNDCI + 11. (Time to integrate forward.) For either case TN4NEX must be set equal to minus to avoid storing over output information in TMORES. The above indicators are set for the N4DRAG, a subroutine of N0CPNI.

d) A location, TNUMNI, set to zero for the first entry to R2AINT. R2AINT steps TNUMNI by one each time a perturbed re-entry table is set up until six such tables have been set up. The differential correction program resets TNUMNI to zero.

e) A location, TFESAB, containing the time at the first second after burnout in fixed point seconds.

4.16.2 Output Requirements

The output of R2AINT is a 15 word block, TNNDCI, which is used as input to the numerical integration program N0CPNI to set up six new tables TNINT3, TNINT4, ... TNINIT8. Each table is a block of at least 112 locations as follows:

TNINT3	Indication of time units	
+ 1	Integration step	Fixed Point
+ 2	Starting time of table	
+ 3	Ending time of table	
+ 4	R_x	Floating Point Mercury Units
+ 5	R_y	
+ 6	R_z	
+ 7	$\dot{R}_x$	
+ 8	$\dot{R}_y$	
+ 9	$\dot{R}_z$	
Etc.		

4.16.3 Method

R2AINT calculates the time interval for backward integration as follows:

$$\frac{t_a - t_{ob}}{60} = N + \text{remainder}$$

where

t_a = anchor point time

t_{ob} = first second after burnout

N = number of minutes to integrate backward

then

(N + 3) 60 = number of seconds to integrate backward

The integration tables extend beyond the first second following burnout to insure that six point Lagrangian interpolation can be performed to calculate $\bar{r}$ and $\bar{v}$ for any observation time occurring after the first second following burnout.

The six integration tables are generated using as starting values:

$X_a + \Delta X,\ Y_a,\ Z_a,\ \dot{X}_z,\ \dot{Y}_a,\ \dot{Z}_a$ at time t_a

$X_a,\ Y_a + \Delta Y,\ Z_a,\ \dot{X}_a,\ \dot{Y}_a,\ \dot{Z}_a$ at time t_a

$X_a,\ Y_a,\ Z_a + \Delta Z,\ \dot{X}_a,\ \dot{Y}_a,\ \dot{Z}_a$ at time t_a

$X_a,\ Y_a,\ Z_a,\ \dot{X}_a + \Delta\dot{X},\ \dot{Y}_a,\ \dot{Z}_a$ at time t_a

$X_a,\ Y_a,\ Z_a,\ \dot{X}_a,\ \dot{Y}_a + \Delta\dot{Y},\ \dot{Z}_a$ at time t_a

$X_a,\ Y_a,\ Z_a,\ \dot{X}_a,\ \dot{Y}_a,\ \dot{Z}_a + \Delta\dot{Z}$ at time t_a

4.16.4 Usage

a) Calling Sequence

```
TRA        R2AINT
```

When the 15 word block has been set-up, R2AINT sets the accumulator = -0 and puts a 1 in the P- bit and transfers to Monitor location MFDIFC which calls in the integration program (N0CPNI).

b) Space Required

a) 6 tables 672_{10}

b) Program less constants 74_{10}

c) Constants Required

```
K00000     DEC     0
```

K00001	DEC	0	
K00001	DEC	1	
K00003	DEC	3	
K00060	DEC	60	
KM0000	DEC	-0	
KPERST	EQU	TSETCT + 10	Number of integration steps per output
KRDELT	EQU	TSETCT + 9	Output interval for numerical integration

d) Tables Required

TN4NEX	PON		Indicator for drag program (N4DRAG)
TFESAB	PZE		First second after burnout in fixed point seconds
TNEARM	PZE		Anchor point in fixed point seconds
TNNDCI	BSS	15	15 word input block to numerical integration program (N0CPNI)
TNUMNI	PZE		D0DIFC tests for all six tables
TRVTAP	BSS	13	Input block
TNINT3	BSS	112	Perturbed X table
TNINT4	BSS	112	Perturbed Y table
TNINT5	BSS	112	Perturbed Z table
TNINT6	BSS	112	Perturbed $\dot{X}$ table
TNINT7	BSS	112	Perturbed $\dot{Y}$ table
TNINT8	BSS	112	Perturbed $\dot{Z}$ table

TPA.RX	DEC	.00001	This value can vary
+1	DEC		
+2	DEC		
+3	DEC		
+4	DEC		
+5	DEC		

e) Monitor Communication Cell Used

MFDIFC Exit of R2AINT to perform numerical integration

f) Timing

Computer running time for the R2AINT program is 327 microseconds.

4.16.5 Checkout

R2AINT was checked out with the R1NDCP program using actual test data supplied by NASA.

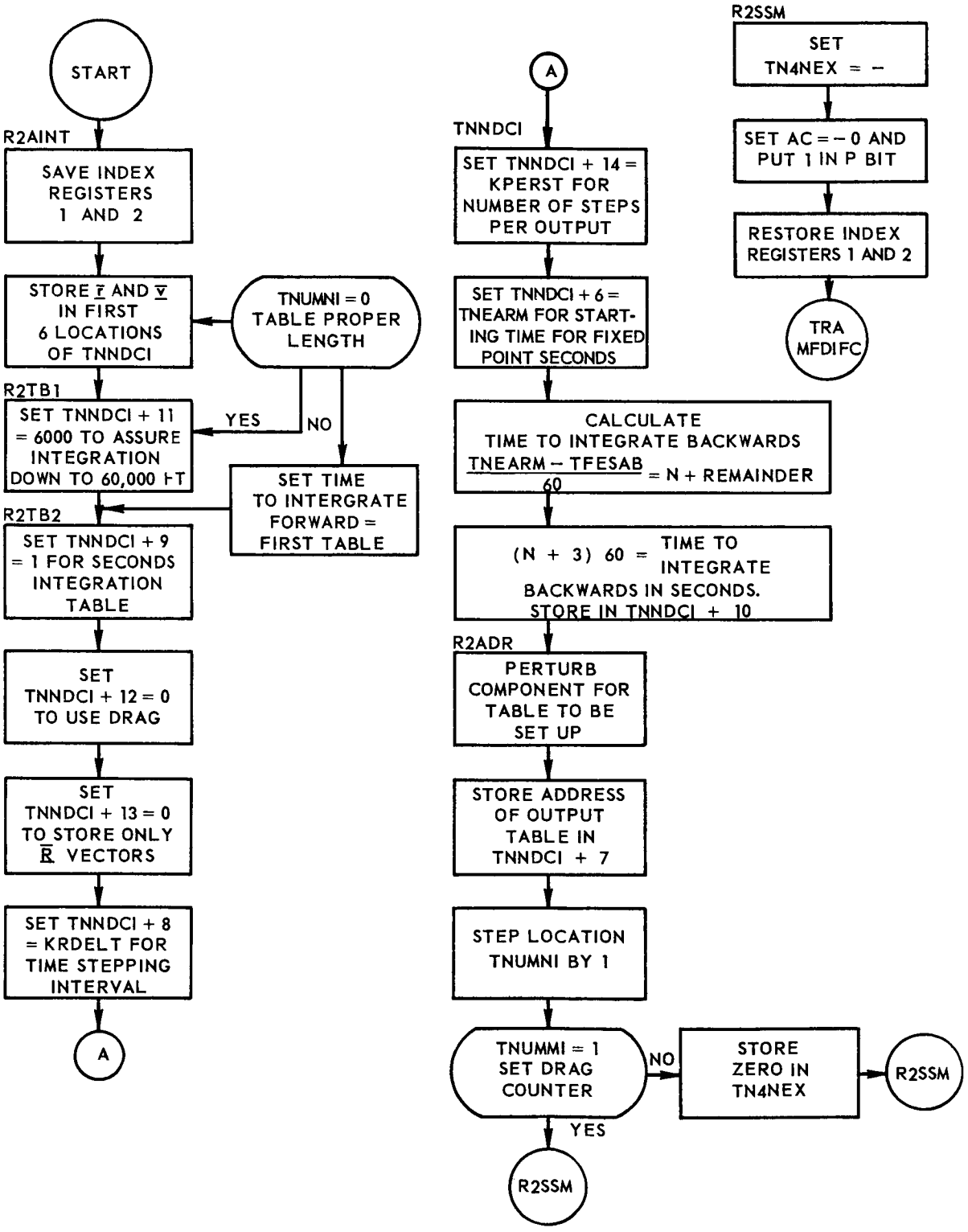


FIGURE 4-16. R2AINT PROGRAM FLOW CHART

4.17 NUMERICAL DIFFERENTIAL CORRECTION PROGRAM (R1NDCP)

This program computes by numerical means the eighteen partial derivatives appearing in the three equations of conditions arising from an observation. R1NDCP utilizes the six integration tables set up by the program R2AINT. The flow chart for R1NDCP is shown in Figure 4-17.

4.17.1 Input Requirements

R1NDCP uses two library subroutines: UA1LSC, U7INTP and Vanguard program INTP3 adapted for use as a closed subroutine within R1NDCP. The input to R1NDCP includes the following:

a) A five word block beginning at location K00TOB.

K00TOB	SYN	TEQCIN + 3	Time of the observation minutes in the address, seconds in the decrement.
+1, +2, +3	Used by R1NDCP for temporary storage.		
+4	station identification		

b) The computed range, azimuth and elevation of the capsule relative to the specified station at the given time of the observation based on the latest re-entry table.

K00DRC	SYN	TEQCIN + 15	R_c
K00DAC	SYN	TEQCIN + 16	A_c
K00DEC	SYN	TEQCIN + 17	E_c

c) The six perturbed integration tables generated in the program R2AINT.

d) TDTOBS	BSS	1	Time of the observation in seconds
e) TNETIM	BSS	1	Time in fixed point seconds of the last entry in the last of the six re-entry tables of predicted position.

4.17.2 Output Requirements

Eighteen partial derivatives are computed and stored beginning in location TDEQCT in the following order.

$$\frac{\partial R}{\partial X}, \frac{\partial R}{\partial Y}, \frac{\partial R}{\partial Z}, \frac{\partial R}{\partial \dot{X}}, \frac{\partial R}{\partial \dot{Y}}, \frac{\partial R}{\partial \dot{Z}}, \frac{\partial A}{\partial X} \cdots \frac{\partial A}{\partial \dot{Z}}, \frac{\partial E}{\partial X} \cdots \frac{\partial E}{\partial \dot{Z}}$$

4.17.3 Method

In addition to the latest re-entry trajectory, there are available to the R1NDCP program six integration tables which have been generated by perturbing the position and velocity vectors at the latest anchor point $\bar{r}_a$ and $\bar{v}_a$ by amounts ΔX_a, ΔY_a . . . . . $\Delta \dot{Z}_a$, respectively. Let I represent any one of the variables X, Y, Z, $\dot{X}$, $\dot{Y}$, $\dot{Z}$. The partials $\frac{\partial R}{\partial_I}, \frac{\partial A}{\partial_I}, \frac{\partial E}{\partial_I}$, are calculated as follows. From the integration table generated by perturbing the I component of r_a or v_a by the amount ΔI_a, compute the range R_I, azimuth A_I, and elevation E_I of the capsule relative to the given station at the specified observation time. Then

$$\frac{\partial R}{\partial I} \frac{R_I - R_C}{\Delta I_a}, \frac{\partial A}{\partial I} \frac{A_I - A_C}{\Delta I_a}, \frac{\partial E}{\partial I} \frac{E_I - E_C}{\Delta I_a}$$

where R_c, A_c, E_c, are the computed range, azimuth, and elevation, respectively, based on the latest predicted re-entry trajectory.

To calculate by 6 point Lagrangian interpolation for an observation time which occurs within the time interval $t_L - 2\Delta t$ to t_L where Δt is the time interval at which entries are made in the 6 NDC integration tables and t_L is the time of the last entry in the numerical differential correction table. R1NDCP ignores any observation at a time greater than t_L. If an observation time falls between $t_L - \Delta t$ to $t_L - 2\Delta t$ or between t_L to $t_L - \Delta t$ the Vanguard program INTP3 is used.

The following tests for discontinuity in azimuth near 360° are made prior to computing $\frac{A_I - A_c}{\Delta I_a}$.

$A_I < 90^o$, $A_c > 270^o$ Compute $\frac{A_I + 2\pi - A_c}{\Delta I_a}$

$A_I < 90^o$, $A_c \leq 270^o$ Compute $\frac{A_I - A_c}{\Delta I_a}$

$A_I \leq 270^o$ and $\geq 90^o$ Compute $\frac{A_I - A_c}{\Delta I_a}$

$A_I > 270^o$, $A_c \geq 90^o$ Compute $\frac{A_I - A_c}{\Delta I_a}$

$A_I > 270^o$, $A_c < 90^o$ Compute $\frac{A_I - (A_c + 2\pi)}{\Delta I_a}$

R1NDCP can only be used with integration tables that store only $\bar{r}$ vectors.

4.17.4 Usage

a) Calling Sequence

α	TSX	RINDCP, 4
$\alpha + 1$		Error Return
$\alpha + 2$		Normal Return

b) Space Required

194_{10}

c) Error Codes

Error returns return to $\alpha + 1$.

If the time is below t_L program returns to $\alpha + 1$.

d) Constants used

K00TOB	SYN	TEQCIN + 3
K00DRC	SYN	TEQCIN + 15
K00DAC	SYN	TEQCIN + 16
K00DEC	SYN	TEQCIN + 17
K004.0	DEC	4.0
K005.0	DEC	5.0
K001.0	DEC	1.0
K00001	DEC	1

KCH233	OCT	233000000000
K00003	DEC	3
K00007	DEC	7
K0ZERO	DEC	0

e) Tables Used

TPA.RX	BSS	6	amount to perturb the vectors by.
TDEQCT	BSS	24	Output table
TNINT3	BSS	112	X perturbed table
TNINT4	BSS	112	Y perturbed table
TNINT5	BSS	112	Z perturbed table
TNINT6	BSS	112	$\dot{X}$ perturbed table
TNINT7	BSS	112	$\dot{Y}$ perturbed table
TNINT8	BSS	112	$\dot{Z}$ perturbed table
TNETIM	PZE		Time of last observation of last reentry table.
TDTOBS	PZE		Time of observation in seconds.

f) Timing

Timing for R1NDCP is 458 microseconds.

4.17.5 Checkout

R1NDCP was checked out in the Mercury Program System using actual test data supplied by NASA.

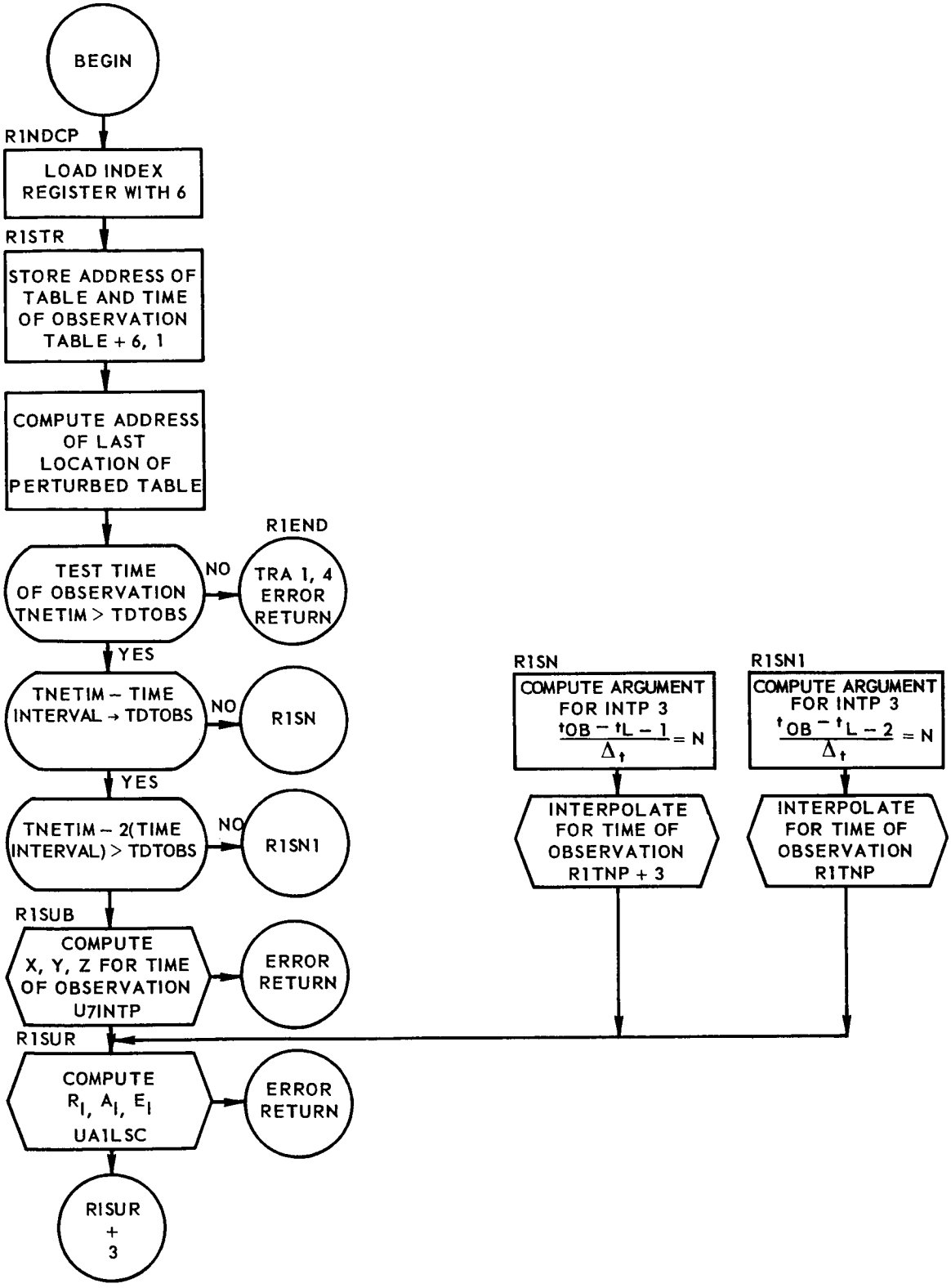


FIGURE 4-17. RINDCP PROGRAM FLOW CHART *(Sheet 1 of 2)*

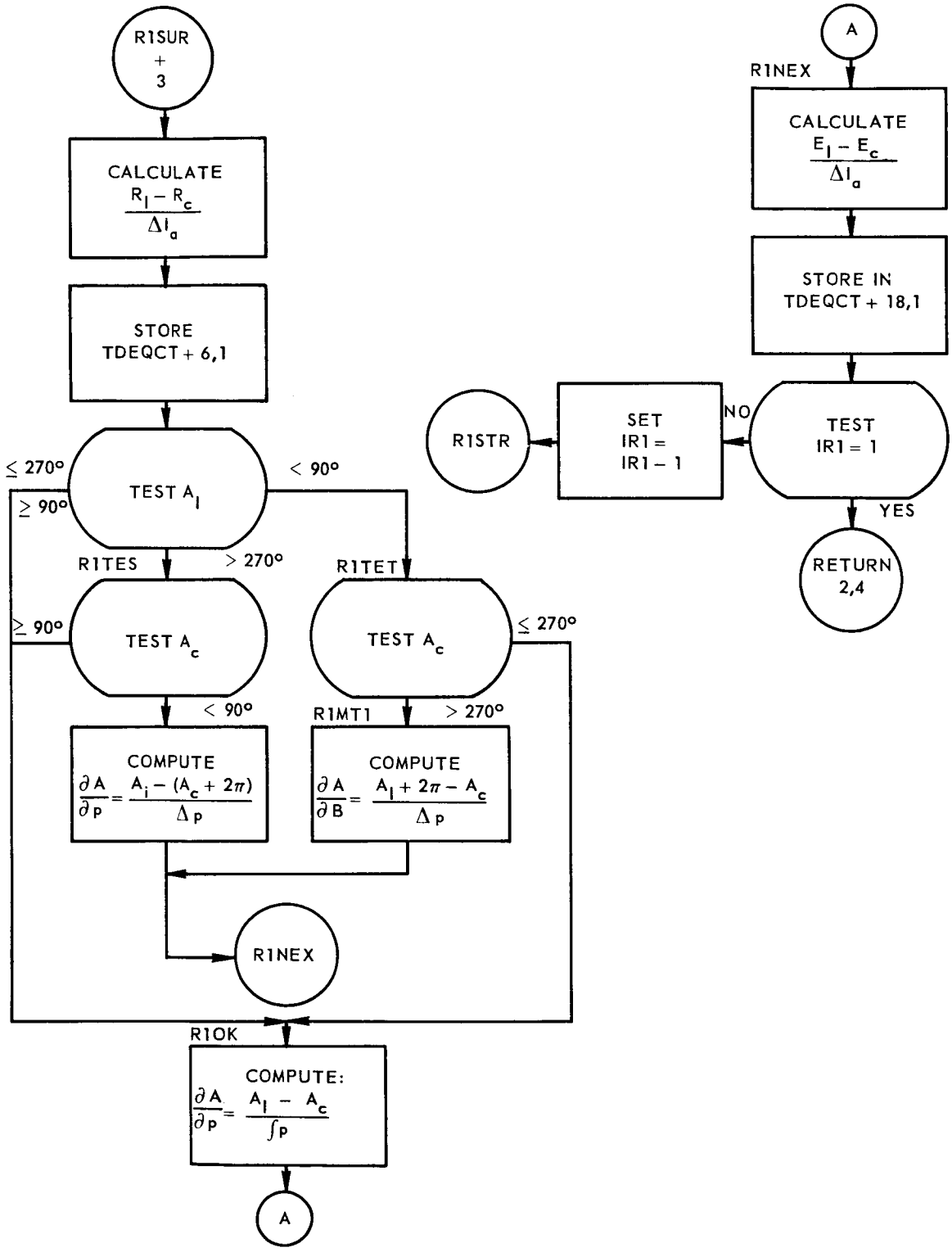


FIGURE 4-17 RINDCP PROGRAM FLOW CHART *(Sheet 2 of 2)*

4.18 WEIGHT TABLE LOOKUP PROGRAM (D3WTLU)

This program sets up a table containing the desired number of observations relative to pre-set weights from each tracking station reporting for a given interval. The flow chart for the D3WTLU program is shown in Figure 4-18.

4.18.1 Input Requirements

Input to the D3WTLU program is received from the following locations:

Symbolic Location	Definitions
a) TMSTMS	Table containing locations of station message blocks in order of longitudinal crossing (see 4.13.1, a).
b) TNSCOR	Table to correlate station identification numbers with station numbers for weight table. The table is arranged in order of station numbers with numbers for weight table in address portion. (See constants for actual values.)

c) Station Message Blocks. (See 4.13.1, g)

d) KNOBBD - Total number of observations desired for a differential correction interval.

e) TSTWTS	Table containing weights of each station prior to last station reporting (3/4 orbit), Weights are in 1/16 with 9 (4 bits each) in a word. Therefore, only 9 stations plus the last one reporting can be processed with this table. Each 4 bits from left to right represents the weight for each station prior to the last station reporting. These are in descending order of longitudinal crossing, (0101 = 5/16). Weights are included for 3 passes, with weight words for 11 separate stations on each pass.

4.18.2 Output Requirements

The output of the D3WTLU program is TNOMFS, a table containing number of observations desired from each station reporting corresponding to the TMSTMS table.

4.18.3 Method

a) Use first location of TMSTMS to determine present station.

b) Use a) above and TNSCOR to obtain weight word location to TSTWTS. Determine pass number.

c) KNOBBD minus the number of observations from the last station reporting equals the remaining number of observations desired.

d) Number of observations for station j

$$= \frac{W_j \text{ (KNOBBD - number of observations } S_1)}{16} \left(1 + \frac{\Sigma W_m}{\Sigma W_r}\right)$$

$$= \frac{W_j \text{ (KNOBBD - number of observations } S_1)}{16} \quad \frac{\Sigma W_r + \Sigma W_m}{\Sigma W_r}$$

since $(\Sigma W_r + \Sigma W_m) = 16$

$= W_j$ (KNOBBD - number of observations S_1)$/\Sigma W_r$

Where W_r = Weights for reporting stations; W_m = Weights for missing stations.

4.18.4 Usage

a) Calling Sequence

TSX	D3WTLU, 4
$a + 1$	Normal Return

b) Space Required

26 locations

c) Constants Used

K0ZERO	PZE	
K00001	DEC	1
K00011	DEC	11
TDCLLT	PZE	Address of weight word used

TDCLTB	PZE	TSTWTS-1 Base address for section of weight table for each pass, initially set for pass 1. It will be updated by the program.

d) Erasable Locations

TEMD3 - TEMD3 + 5

e) Timing

7.749 milliseconds (maximum)

5.688 + 240 T milliseconds, where T = number of reporting stations -1

4.18.5 Checkout

The D3WTLU was checked out by presenting different combinations of input and checking output.

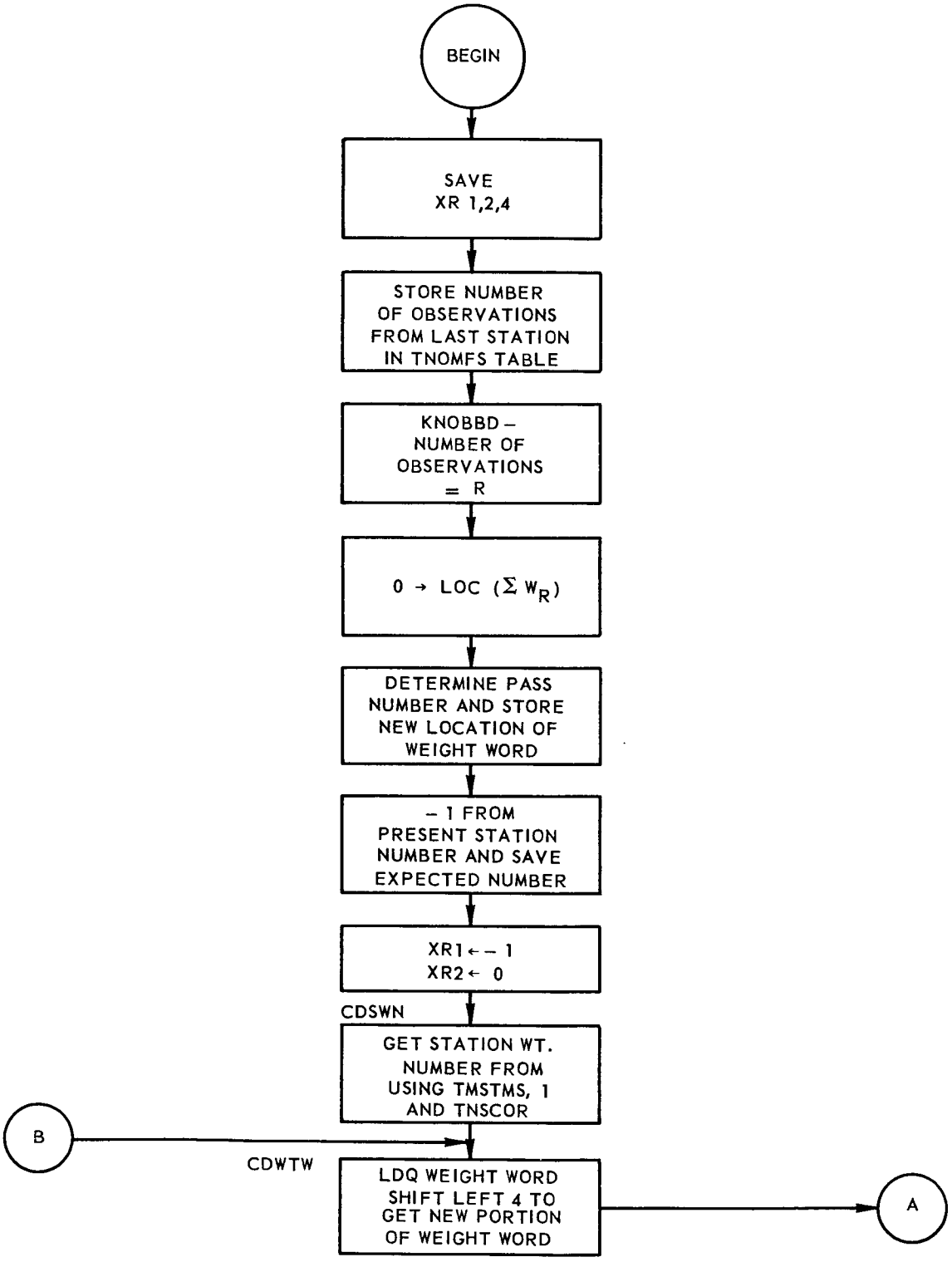


FIGURE 4-18 D3WTLU PROGRAM FLOW CHART *(Sheet 1 of 2)*

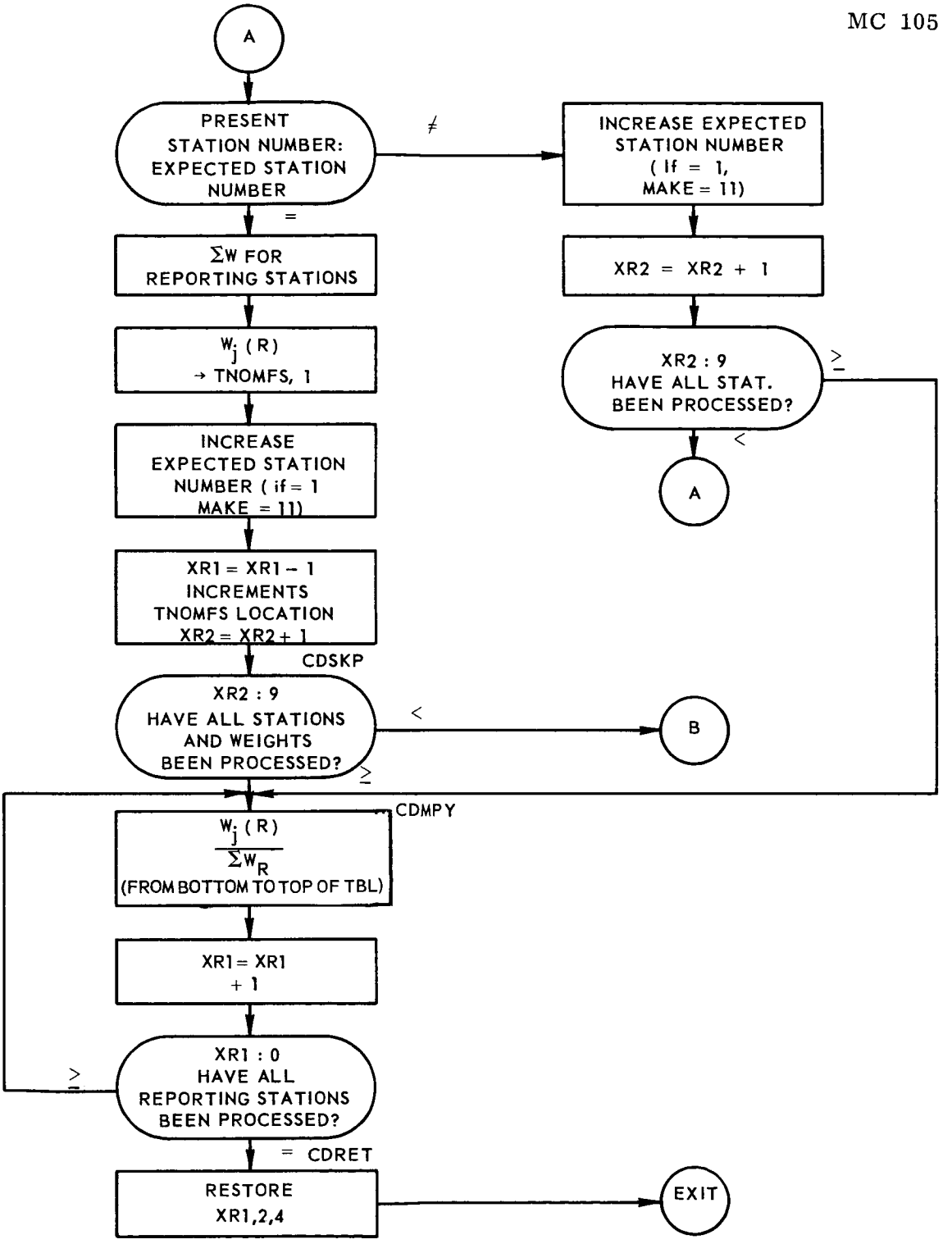


FIGURE 4-18 D3WTLU PROGRAM FLOW CHART *(Sheet 2 of 2)*

4.19 WEIGHTING THE EQUATIONS OF CONDITION PROGRAM (D2WEQT)

The purpose of this program is to weight partial coefficients, compute and weight right sides of equations of condition, and to test residual by comparison with some check factor times standard error. The flow chart for D2WEQT is shown in Figure 4-19.

4.19.1 Input Requirements

The D2WEQT program uses the sine/cosine library subroutine UISICO. The following parameters are inputted to the D2WEQT program:

a) KDCSTT: set = 1 to initialize on first entrance

b) KRESID = -0 weight partials and right sides

+0 perform residual check and weight partials and right sides.

+1 compute and weight (df); perform W(df) check

+2 weight partials according to results of W(df) check.

c) K0000N: Number of coefficients (fixed point)

d) K0SWTR: Square root of station weight for range

e) K0SWTA: Square root of station weight for azimuth

f) K0SWTE: Square root of station weight for elevation

g) KSEMPY: Standard error multiplied by residual check factor

h) K00ROB: Observed range

i) K00AOB: Observed azimuth

j) K00EOB: Observed elevation

k) K00DRC: Computed range

l) K00DAC: Computed azimuth

m) K00DEC: Computed elevation

n) K000MB: maximum number of equations that can go to normal equations block not including two extra locations for overflow.

o) TDEQCT; Table with three sets of partial coefficients corresponding to R, A, and E respectively.

p) TLSSSE: Solution of normal equations obtained from previous differential correction.

q) KRETBL: Location used in setting up intermediate TMORES table with W (df). Used only when KRESID = -0 or +1.

r) KINSDF: = 0: do not perform any checks or weight partials; only compute $\Sigma W\ (df)^2$, ΣM

$\neq 0$: Perform normal calculations.

s) KTOTLM: Total number of equations; only when KRESID = -0 or +1.

t) KTDFSQ: $\Sigma W(df)^2$

4.19.2 Output Requirements

The following parameters are outputted from the D2WEQT program:

a) K00MCT: number of equations to be used in D2SUNE (setting up normal equations)

b) KSUMWT: Sum of the weights that were accepted

c) TPWTDF: Block containing weighted (df); number in the block = c(K00MCT)

d) TAWTPC: Block containing weighted partials; number in the block = c(K00MCT) [c(K0000N)]

e) KTOTLM: Total number of equations of condition

f) KTDFSQ: $\Sigma W(df)^2$ for all observations, even those not selected for interval.

4.19.3 Method

a) KRESID = 1 Go to (d)

Test R_{ob}, A_{ob}, and E_{ob} as meaningful data

+0 ----0 1 indicates erroneous data

b) $\sqrt{W_1}\ (R_{ob} - Rc) = W_r(df)$

$\sqrt{W_2}\ (\cos E_c)\ (A_{ob} - A_c) = W_a(dF)$

$\sqrt{W_3}\ (E_{ob} - E_c) = W_e(dF)$

KRESID = +0 Go to (c)

KRESID = -0 Go to (d)

<u>W (dF) Check</u>

$W_r(dF)$: KSEMPY (also for $W_a(dF)$, $W_e(dF)$)

If greater, indicate by +1 in DWDF test location and reject data

Exit from routine for W (dF) check

$\alpha + 1$: all data rejected

$\alpha + 2$: at least one W(dF) passed the test

c) Residual Test: (Residuals: KSEMPY)

If $R_1 = \sqrt{W_1}\left[\sum_1^6 \left(a_i x_i\right) - dF\right]$ KSEMPY, R_2 and R_3 as above, data is accepted.

where

a_i = partials ; X_i = corrections

d) Weight and store partials.

If any tests have been made, partials are rejected according to results of tests.

e) Exit from routine is made as follows (except when W(df) check was made, as shown before):

$\alpha + 1$ indicates that more weighted equations can be stored before adding contributions to normal equations, therefore, at this point more partials should be computed.

$\alpha + 2$ indicates that blocks storing weighted equations are filled; therefore, go to set up normal equations.

f) If entrance was made on TSX D2WEQT + 1, locations are set as if return had been made to set up normal equations. Return is at $\alpha + 1$. No computation is performed.

g) If KINSDF = 0, only ΣW(df) 2 and ΣM are calculated and stored in KTDFSQ and KTOTLM respectively. Return is made to $\alpha + 1$ (only one return).

h) When KRESID = -0 (when no residual checks are made), KRETBL is used in incrementing locations of TMORES table to set up an intermediate table of W(df) for all selected observations.

4.19.4 Usage

a) Calling Sequence

TSX D2WEQT, 4	If performing [W(df) check] returns in calling sequence indicate:
$\alpha + 1$ Return for partials	$\alpha + 1$ get another observation
$\alpha + 2$ Set up normal equations	$\alpha + 2$ compute partials

b) Space Required

182 locations U1SICO

c) Special Usage

TSX D2WEQT + 1 (re-initializes locations)

d) Constants

K0ZERO	PZE
K00001	DEC 1
K00003	DEC 3

e) Erasable Storage

TMPD2, TMPD2 + 1, D2WDT, D2WDT + 2.

f) Timing (in milliseconds)

where p = number of equations that passed any type test (max. 3)

n = number of partials (6 or 8)

No checks

a) .188 + p (1.248 + .324 n) + 2.3 for U1SICO

b) maximum 14.108 msec including U1SICO

W (df) check

a) .24 + .624p ± 2.3 for U1SICO

b) maximum 4.532 msec.

Weight partials after W (df) check

a) .432 + p (.72 + .324 n)

b) maximum 10.378 msec

Residual check

a) 188 + p (1.428 + .66 n) + 2.3 for U1SICO

b) maximum 22.712

TSX D2WEQT + 1, 4

a) .12 msec.

4.19.5 Checkout

D2WEQT has been successfully checked out using all cases listed in 4.15.3, Method.

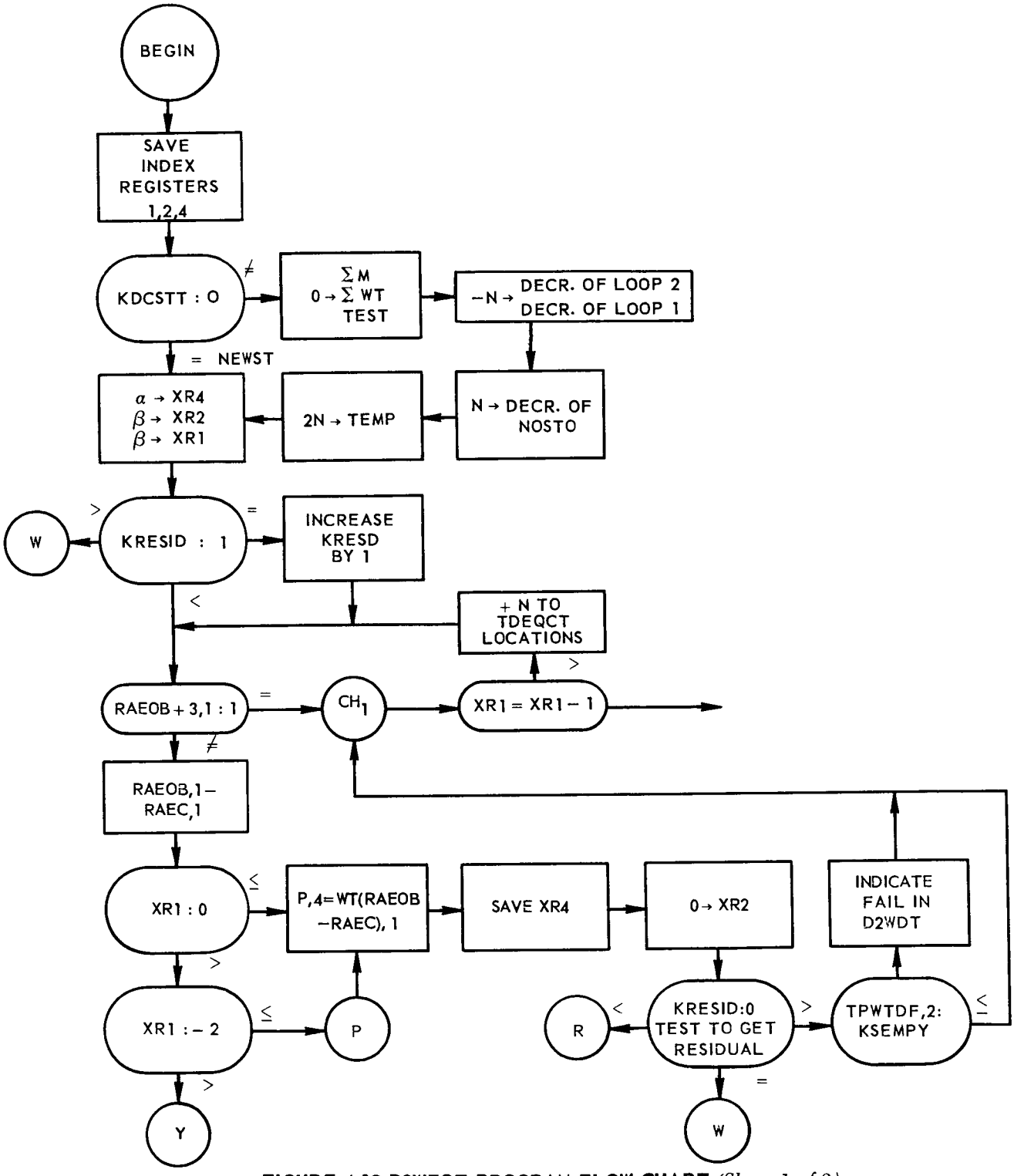


FIGURE 4-19 D2WEQT PROGRAM FLOW CHART *(Sheet 1 of 2)*

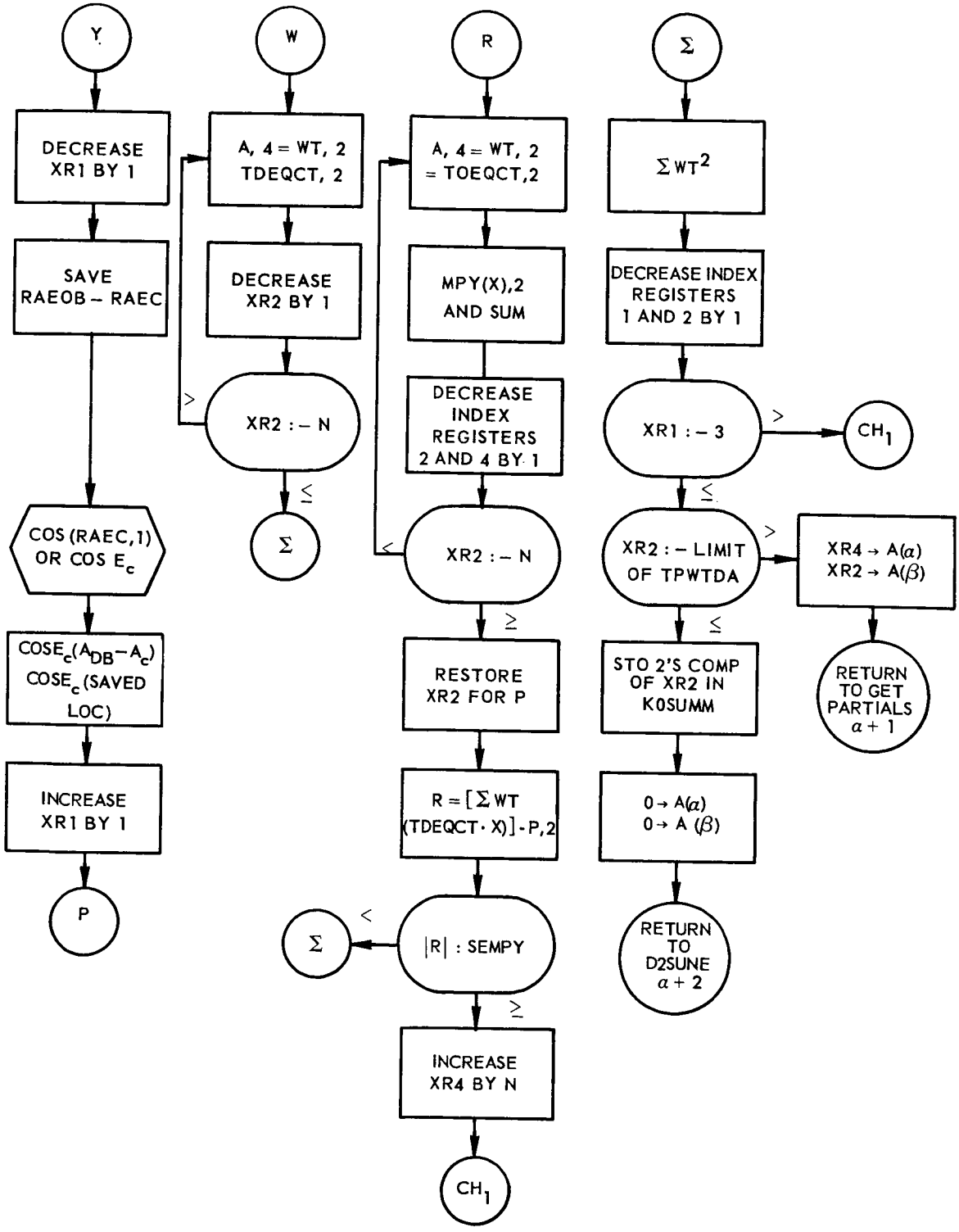


FIGURE 4-19 D2WEQT PROGRAM FLOW CHART *(Sheet 2 of 2)*

4.20 CHOOSE DESIRED RADAR MESSAGE PROGRAM (D3CDRM)

This program chooses the desired number of radar messages from the station block and stores time, range, azimuth, and elevation for each message starting with the earliest message. The flow chart for D3CDRM is shown in Figure 4-20.

4.20.1 Input Requirements

The following parameters serve as input to the D3CDRM program:

Symbolic Location	Definition
K0LSMB SYN	TDCCNT + 7 Address contains location of first word of station message block.
K0NDRM SYN	TDCCNT + 8 Desired number of radar messages.
KCDRMI SYN	TDCCNT + 6 s, 1-34 bits all zeros. Bit 35 = 1 first time through D3CDRM, = 0 all subsequent times through D3CDRM.
Station Message Block	
Y	First word of station message. Address contains station identification number, decrement contains pass number.
Y + 1	Number of radar messages in this block.
Y + 2	Address contins the location, relative to Y, of the time of minimum range, in this block. (T_m).
Y + 3	T_1 (time) fixed point, minutes in address, seconds in decrement.
Y + 4	R_1 (range) floating point, in Mercury units.
Y + 5	A_1 (azimuth) floating point, in radians.
Y + 6	E_1 (elevation) floating point, in radians.
Y + 7---Y + 10	Second radar message.

Y + 4N - 1----Y + 4N + 2	Nth radar message.
TMSTCH	Address contains the location-1 of the station characteristics block to be used. (see 4.13.1, f).

4.20.2 Output Requirements

The following parameters are the outputs from the D3CDRM program:

Symbolic Location			Definition
KOSWTR	SYN	TEQCIN	Square root of station weight for range.
KOSWTA	SYN	TEQCIN + 1	Square root of station weight for azimuth.
KOSWTE	SYN	TEQCIN + 2	Square root of station weight for elevation.
K00TOB	SYN	TEQCIN + 3	Time of radar message just processed.
K00ROB	SYN	TEQCIN + 9	Range of radar message just processed.
K00AOB	SYN	TEQCIN + 10	Azimuth of radar message just processed.
K00EOB	SYN	TEQCIN + 11	Elevation of radar message just processed.
KSTCHL	SYN	TEQCIN + 7	Address contains the number identifying which station characteristics block was used.
KTSUBI	SYN	TEQCIN + 8	Time of observation in Mercury units, floating point.

4.20.3 Method

The message (T_m) at time of minimum range and maximum elevation, and the messages closest in time to it, totaling the desired number of messages, are selected and stored in order of increasing time and starting with the earliest.

If, in testing messages to be selected on the basis of time closest to T_m, the first message of the station message block is accepted, the program names it

"the first desired message," and automatically selects enough sequential messages after T_m to reach the number desired.

If, in testing messages to be selected on the basis of time closest to T_m, the last message of the station message block is accepted, the first message is automatically: T-4(number of desired messages-1), where T is the number of the last message. All messages between it and T_m are thus selected.

After selecting the desired number of messages, the first one is stored and control is returned to 2, 4. On each subsequent TSX to D3CDRM another of the selected messages is stored, until all desired messages have been stored. Then a final return is made to 1, 4.

4.20.4 Usage

a) Calling Sequence

α	TSX	D3CDRM,4
$\alpha + 1$	all desired messages have been stored; final return.	
$\alpha + 2$	another selected message has been stored, but not all of them.	

b) Space Required

150 locations used (the last 14 of which are erasable) excluding constants, input, and output.

c) Constants Used

K00001	DEC	1
K00002	DEC	2
K00003	DEC	3
K00004	DEC	4
K00005	DEC	6
K00.SH	DEC	.00123857
KCH233	OCT	233000000000

d) Timing

The D3CDRM program requires 2.3 + N(1.2) msec for the first time through plus (1.3)N msec for the remaining times through (total), where

N = the desired number of radar messages and the number of times control is transferred to D3CDRM. Total time: 2.3 + N(2.5)msec.

4.20.5 Checkout

Table 4-6 shows times for input messages used to test D3CDRM. A block of ten messages was set up with time equal to 1342 octal minutes and seconds varying from 0.62 octal. Four tests were made.

a) The normal case is that only part of the block is needed and the messages are in the center of the block.

b) All of the messages are needed.

c) Part of the messages are needed, and they run off the end of the block.

d) Part of the messages are needed, and they run off the beginning of the block.

TABLE 4-6. TIMES OF INPUT MESSAGES IN TESTING D3CDRM

OCTAL SECONDS	*DECREMENT SECONDS	MESSAGE	TEST NO. 1		TEST NO. 2	TEST NO. 3		TEST NO. 4	
			Desire 5 of the 10 messages. Mim. range and Max. elev. is No. 7		Ten messages are are desired from the block of 10. All are excepted, no tests are made.	6 of 10 are desired. Mim. range & max. elev. is no. 9 When message # 10 is hit, the program takes the last 6 messages in the block		Same as #3 except min. range & max. elev. is No. 3	
0	0	1						14	4
6	6	2	time difference in sec.					8	9
16	14	3	22					0	1
23	19	4	17	The excepted 5				5	2
35	29	5	7	4			6	15	5
37	31	6	5	3			5		6
44	36	7	0	1			4	When message #1 was hit the program took the first 6 messages in the block.	
51	41	8	5	2		6	3		
57	47	9	11	5		0	1,		
62	50	10	16			3	2		

* Time = 1342 min + x seconds

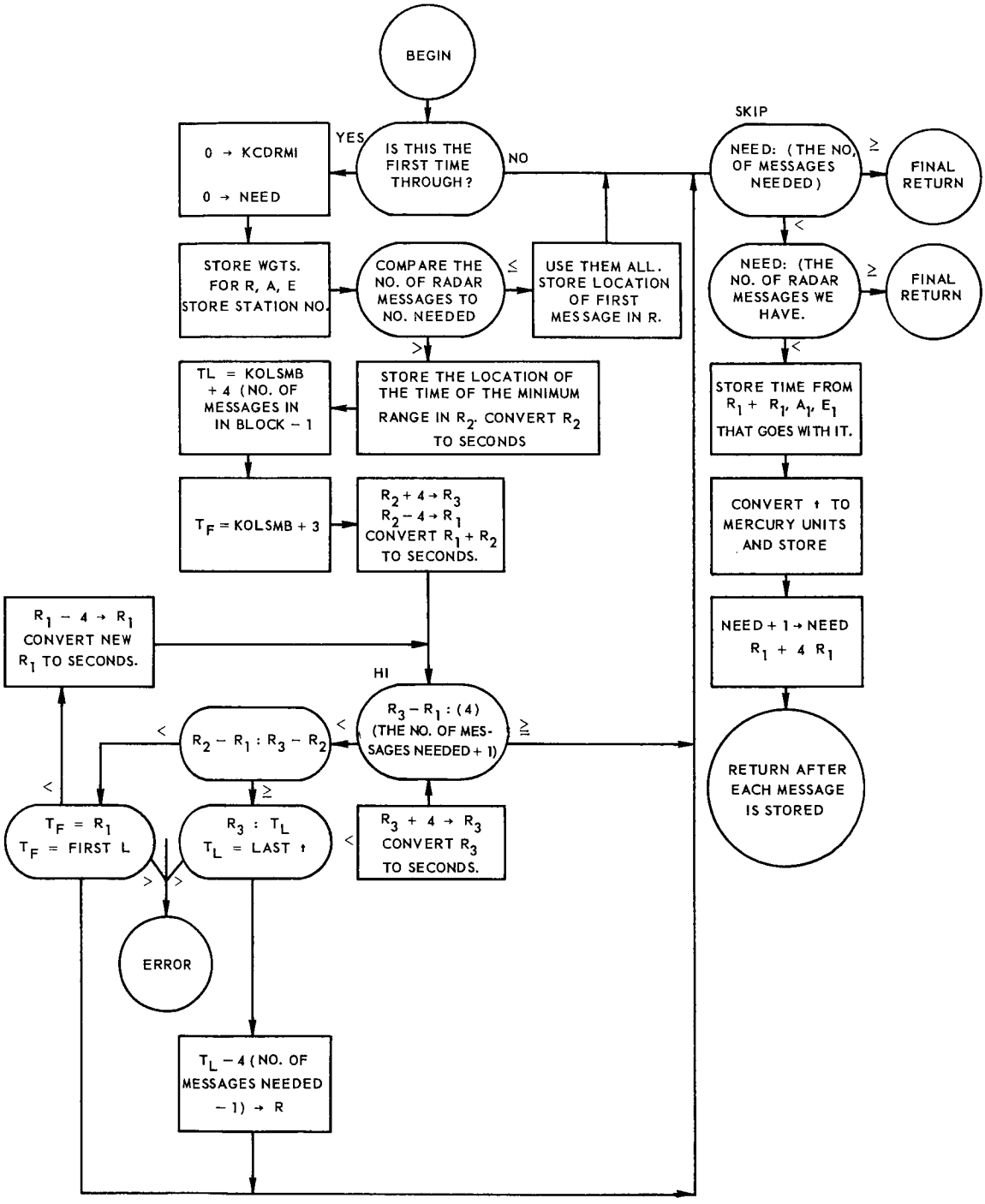


FIGURE 4-20. D3CDRM PROGRAM FLOW CHART

4.21 SET UP NORMAL EQUATIONS PROGRAM (D2SUNE)

This program obtains normal equations from a group of m equations in n unknowns. The flow chart for D2SUNE is shown in Figure 4-21.

4.21.1 Input Requirements

Symbolic Locations		Definitions
TAWTPC	=	Core location of independent variable matrix.
TPWTDF	=	Core location of first of m values of the dependent variable.
C(K00MCT)	=	Number of equations.
C(K0000N)	=	Number of unknown coefficients.

The above inputs are set in the calling sequence (D0DIFC).

KD2SUT	Indicator which is set by control program to avoid repeated initialization of cells dependent on n (number of unknowns). non zero: initialize zero: skip initialization

4.21.2 Output Requirements

a) K0SUMM: total number of equations added to normal equations block thus far.

b) Normal equations block set up to be solved by D2SLNE starting at location TEMD2 + 20.

c) KDFSQU $\sum_{i=1}^{m} (dF_i)^2, \ (dF_i = p_i)$

d) Status of TEMD2 storage after D2SUNE (see next page)

TEMD2		Σm
+1	Decrement	n
	Address	m
+2	Decrement	n + 1
	Address	m + 1
+3	Decrement	2n + 1
	Address	n(2n + 1) + 1
+4		n
+5		
+6	Address	TEMD2+20 + n(2n + 1)+1
+7	Decrement	m(n + 1)
	Address	p
+8		
+9		
+10		$\Sigma a_{i1} p_i$
+11		$\Sigma a_{i2} p_i$
+12		$\Sigma a_{i3} p_i$
.		.
.		.
.		.
+19*		$\Sigma a_{in} p_i$
+20**		$\Sigma (a_{i1})^2$
+21		$\Sigma a_{i1} a_{i2}$
+22		$\Sigma a_{i1} a_{i3}$
+23		$\Sigma a_{i1} a_{i4}$
+24		$\Sigma a_{i1} p_i$
+25		1.0
+26		
+27		
+28		
+29		
+30		$\Sigma (a_{i2})^2$
+31		$\Sigma a_{i2} a_{i3}$

TEMD2	
+32	$\Sigma a_{i2} a_{i4}$
+33	$\Sigma a_{i2} p_i$
+34	
+35	1.0
+36	
+37	
+38	
+39	
+40	$\Sigma (a_{i3})^2$
+41	$\Sigma a_{i3} a_{i4}$
+42	$\Sigma a_{i3} p_i$
+43	
+44	
+45	1.0
+46	
+47	
+48	
+49	
+50	$\Sigma (a_{i4})^2$
+51	$\Sigma a_{i4} p_i$
+52	
+53	
+54	
+55	1.0
+56	Σp_i^2

*Note: Maximum n = 10; can be changed by making T + 20 = T + 10 + n.
**For clarity in illustration of storage assume N = 4.

Since the above matrix is symetric only the diagonals and above need be stored in the normal equations block.

TEMD2+20	TEMD2+20+1	TEMD2+20+(n-1)	TEMD2+20+n
$\Sigma(a_{i1})^2$	$\Sigma a_{i1}\, a_{i2}$	$\ldots\ \Sigma a_{i1}\, a_{in}$	$\Sigma a_{i1}\, p_i$
	$\Sigma(a_{i2})^2$	$\ldots\ \Sigma(a_{i2}\, a_{in})$	$\Sigma a_{i2}\, p_i$
		$\vdots$	$\vdots$
		TEMD2+20+n(2n+1)-2	TEMD2+20+n(2n+1)-1
		$\Sigma(a_{in})^2$	$\Sigma a_{in}\, p_i$

The normal equations are stored row wise beginning in TEMD2 + 20 for the first row. Each of the following rows is stored n locations following a floating point one in locations TEMD2 + 20 + n + 1, TEMD2 + 20 + n + 1 + 2n + 2 . . . . . TEMD2 + 20 + 20 + n(2n + 1)

4.21.3 Method

Start with the fundamental equations

$$a_{11}\, x_1 + a_{12}\, x_2 + \ldots + a_{1n}\, x_n = p_1$$

$$a_{21}\, x_1 + a_{22}\, x_2 + \ldots + a_{2n}\, x_n = p_2$$

$$\vdots$$

$$a_{m1}\, x_1 + a_{m2}\, x_2 + \ldots + a_{mn}\, x_n = p_m$$

where a = Independent variables (partial coefficients)

p = dependent variables

m > n

Set equations in matrix forms:

$$\begin{pmatrix} a_{i1} & a_{i2} & \ldots & a_{in} \\ a_{21} & a_{22} & \ldots & a_{2n} \\ \vdots & \vdots & & \\ a_{m1} & a_{m2} & \ldots & a_{mn} \end{pmatrix} \begin{pmatrix} X_1 \\ x_2 \\ \vdots \\ x_{m2} \end{pmatrix} = \begin{pmatrix} p_1 \\ p_2 \\ \vdots \\ p_m \end{pmatrix}$$

or

$$[A] \{X\} = \{P\}$$

multiply both sides of the above equation by the transpose,

$[A^T]$ of $[A]$ to obtain

$$[A^T] [A] \{X\} = [A^T] \{P\}$$

or

$$[B] \{X\} = [C]$$

written as:

$$\begin{pmatrix} \sum_{i=1}^{m} (a_{i1})^2 & \sum_{i=1}^{m} a_{11}a_{12} & \cdots & \sum_{i=1}^{m} a_{i1}a_{in} \\ \sum_{i=1}^{m} a_{12}a_{11} & \sum_{i=1}^{m} (a_{12})^2 & \cdots & \sum_{i=1}^{m} a_{i2}a_{in} \\ \vdots & \vdots & & \vdots \\ \sum_{i=1}^{m} a_{in}\, a_{i1} & \sum_{i=1}^{m} a_{in}\, a_{i2} & \cdots & \sum_{i=1}^{m} (a_{in})^2 \end{pmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{matrix} \sum_{i=1}^{m} a_{i1}p_i \\ \sum_{i=1}^{m} a_{12}p_i \\ \vdots \\ \sum_{i=1}^{m} a_{in}p_i \end{matrix}$$

4.21.4 Usage

a) Calling Sequence

		Address	Tag	Decrement
α	TSX	D2SUNE	4	
$\alpha + 1$		TAWTPC	, 0 ,	TPWTDT
$\alpha + 2$		C(K00MCT)	, 0 ,	C(K0000N)
$\alpha + 3$		Normal Return		

b) Space Required

127 locations

c) Special Usage

D2SUNE must be used to set up normal equations for solution using D2SUNE

d) Constants Used

K0ZERO	PZE	
K00001	DEC	1
KD0001	PZE	0, 0, 1
K001.0	DEC	1.
KD2SUT	PZE	
K0SUMM	SYN	TDCCNT + 11
KDFSQU	SYN	TDCCNT + 13

e) Erasable Locations

TEMD2, TEMD2 + 20 + n(2n + 1)

This temporary storage contains the major output of the program (normal equations). It must be maintained until solution by D2SLNE is completed and if desired, computation of standard error of fit and elements by D2CSFE.

4.21.5 Checkout

D2SUNE was checked out with D2SLNE by comparing results from LSQ2. (see SHARE library).

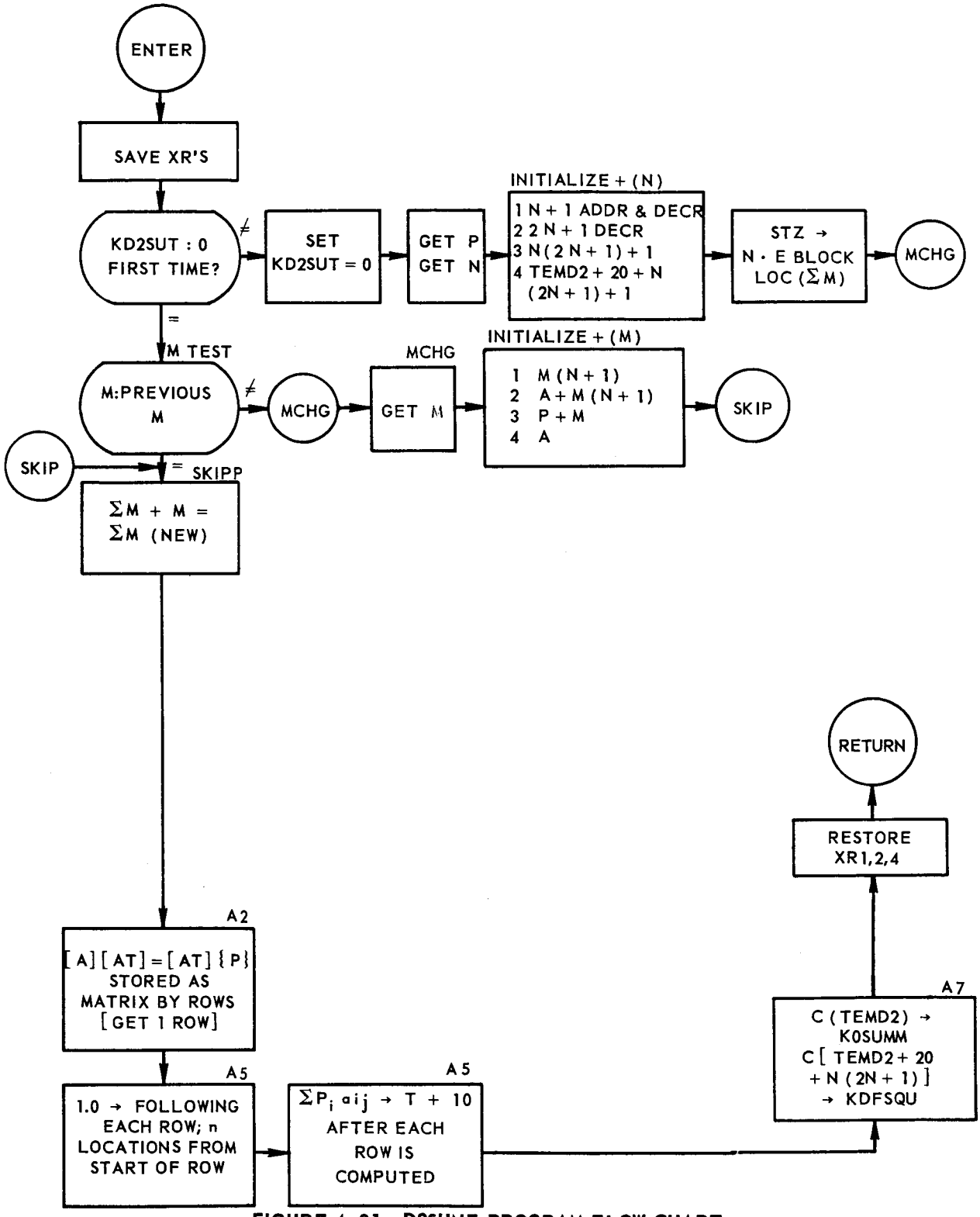


FIGURE 4-21. D2SUNE PROGRAM FLOW CHART

4.22 SOLVE NORMAL EQUATIONS PROGRAM (D2SLNE)

The purpose of this program is to solve M normal equations and obtain diagonals of the inverse matrix. The general and detailed flow charts for this program are shown in Figures 4-22 and 4-23 respectively.

4.22.1 Input Requirements

The following parameters serve as input to the D2SLNE program:

Values	Symbolic Location
N = Number of unknowns	D(TEMD2 + 1)
M = Number of equations	A(TEMD2 + 1)

Normal equations matrix stored row wise beginning in location TEMD2 + 20.

The size of TEMD2 is dependent on the size on n which cannot be greater than 10. (Refer to writeup of D2SUNE for format of TEMD2).

4.22.2 Output Requirements

The following items are outputted from the D2SLNE program:

a) $\Delta X_1, \Delta X_2 \ldots, \Delta X_n$: Stored consecutively beginning in location TLSSSE.

b) B_{ij}: Terms on the principle diagonal of the inverse of the matrix (B) (to be used later in D2CSFE).

4.22.3 Other Requirements

D2SLNE must be used after D2SUNE which sets up the normal equations. D2CSFE is then run to computed standard error of fit and elements.

4.22.4 Method

The normal equations are solved by means of elimination. The diagonals of the inverse matrix are then computed. This program must be run with D2SUNE and D2CSFE.

4.22.5 Usage

a) Calling Sequence

	Operation	Address	Tag	Decrement
α	TSX	D2SLNE,4		
$\alpha + 1$	PZE	TLSSE		
$\alpha + 2$	ERROR RETURN (Result of zero in pivot element)			
$\alpha + 3$	NORMAL RETURN			

b) Space Required

135 locations (main program)

156 locations (main program, K loc, TLSSE)

Total = 156 + TEMD2 + 20 + N(2N + 1) + 1 (Non-Erasable) where N is the number of equations.

c) Error Code

Error will come as a result of zero in pivot element.

d) Constants Used

K0ZERO	DEC	0,
K00001	DEC	1
KD0001	PZE	0, 0, 1
KD0002	PZE	0, 0, 2

e) Common Locations

[TEMD2 + 20 + N(2N + 1) + 1] non-erasable storage.

4.22.6 Checkout

D2SLNE was checked out with D2SUNE by comparing results with those of LSQ2 (see SHARE library).

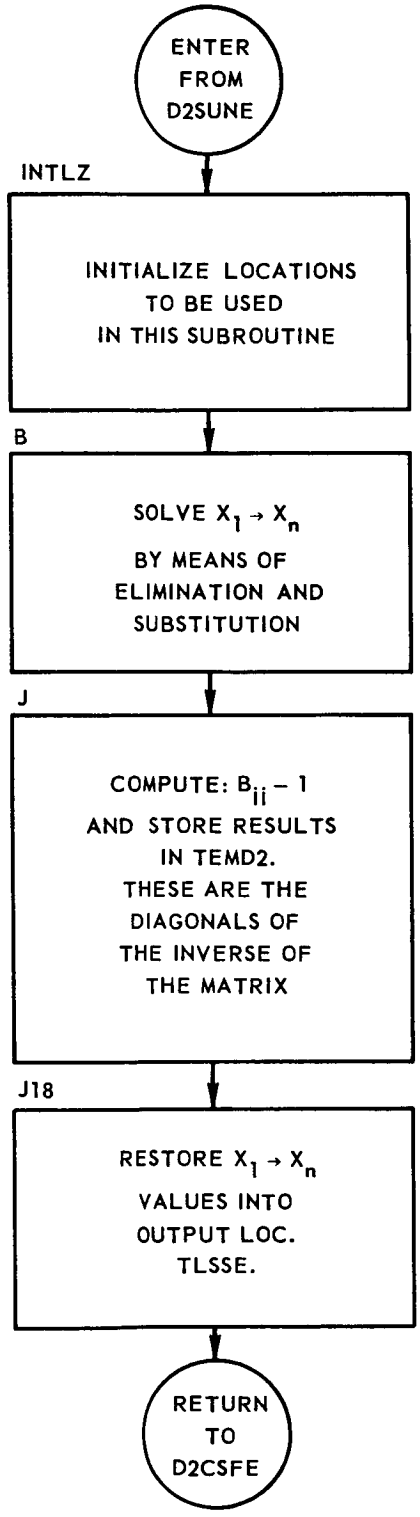


FIGURE 4-22. D2SLNE PROGRAM GENERAL FLOW CHART

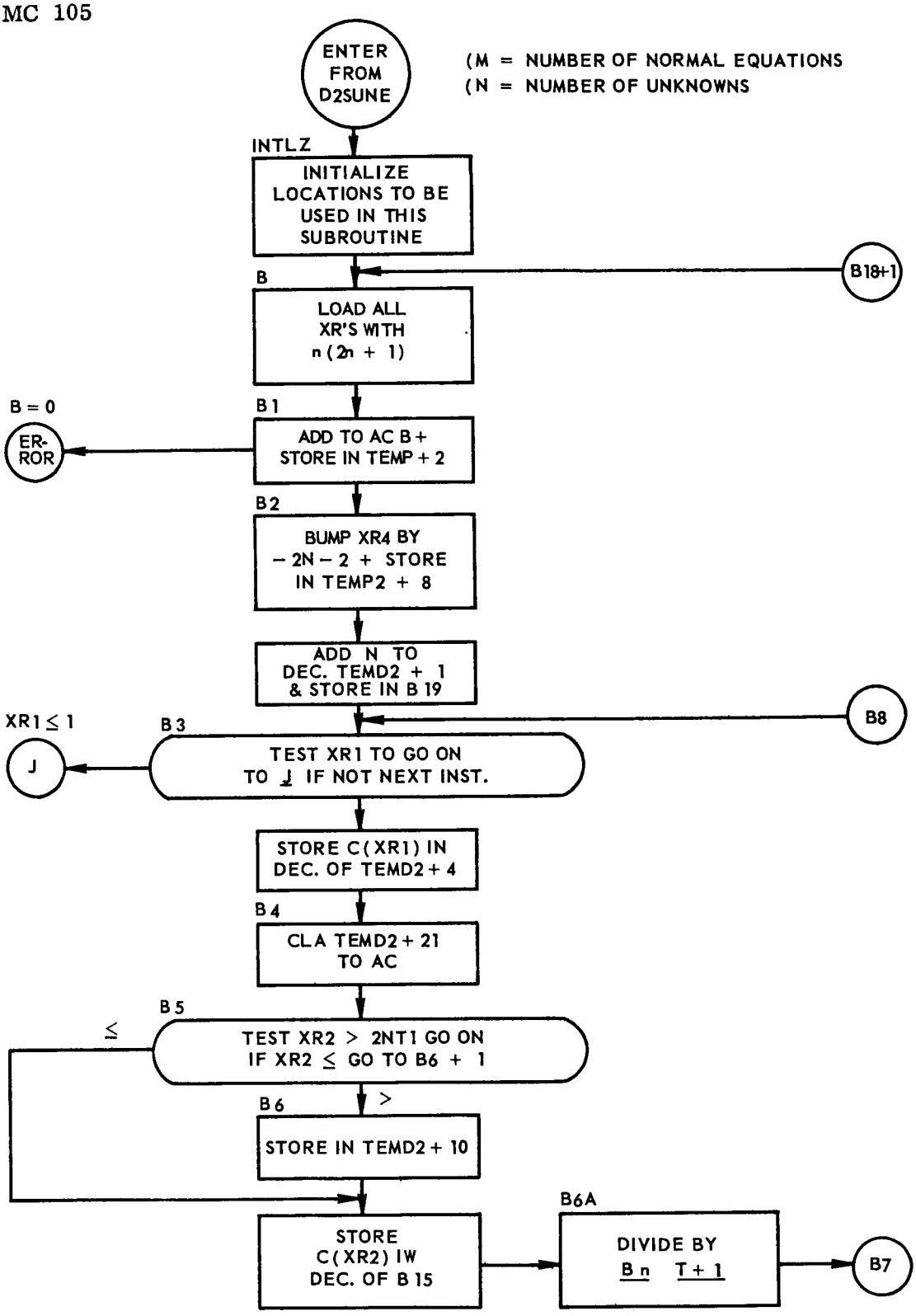


FIGURE 4-23. D2SLNE PROGRAM DETAILED FLOW CHART *(Sheet 1 of 3)*

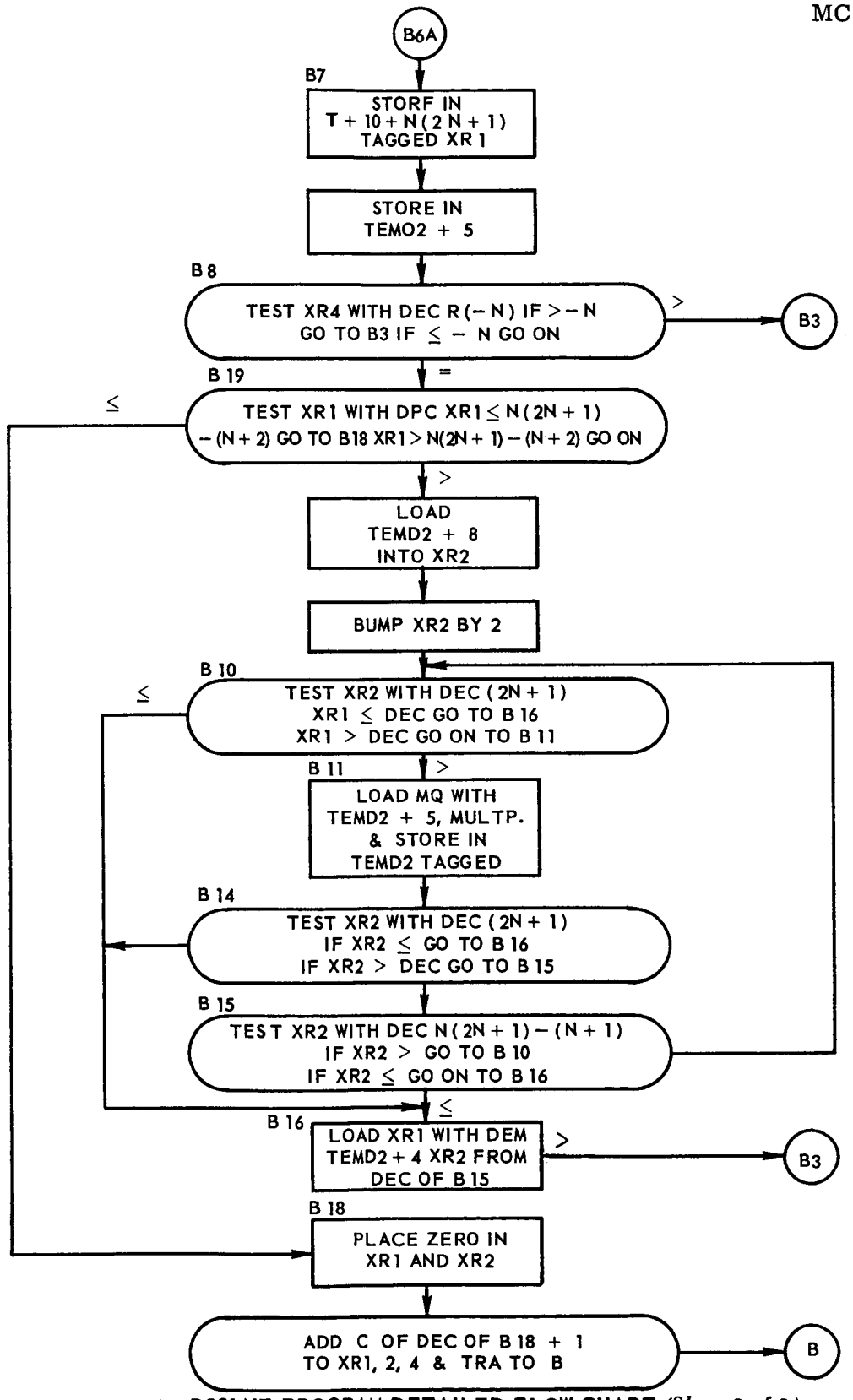


FIGURE 4-23. D2SLNE PROGRAM DETAILED FLOW CHART *(Sheet 2 of 3)*

Figure 4-24 (3 of 3)

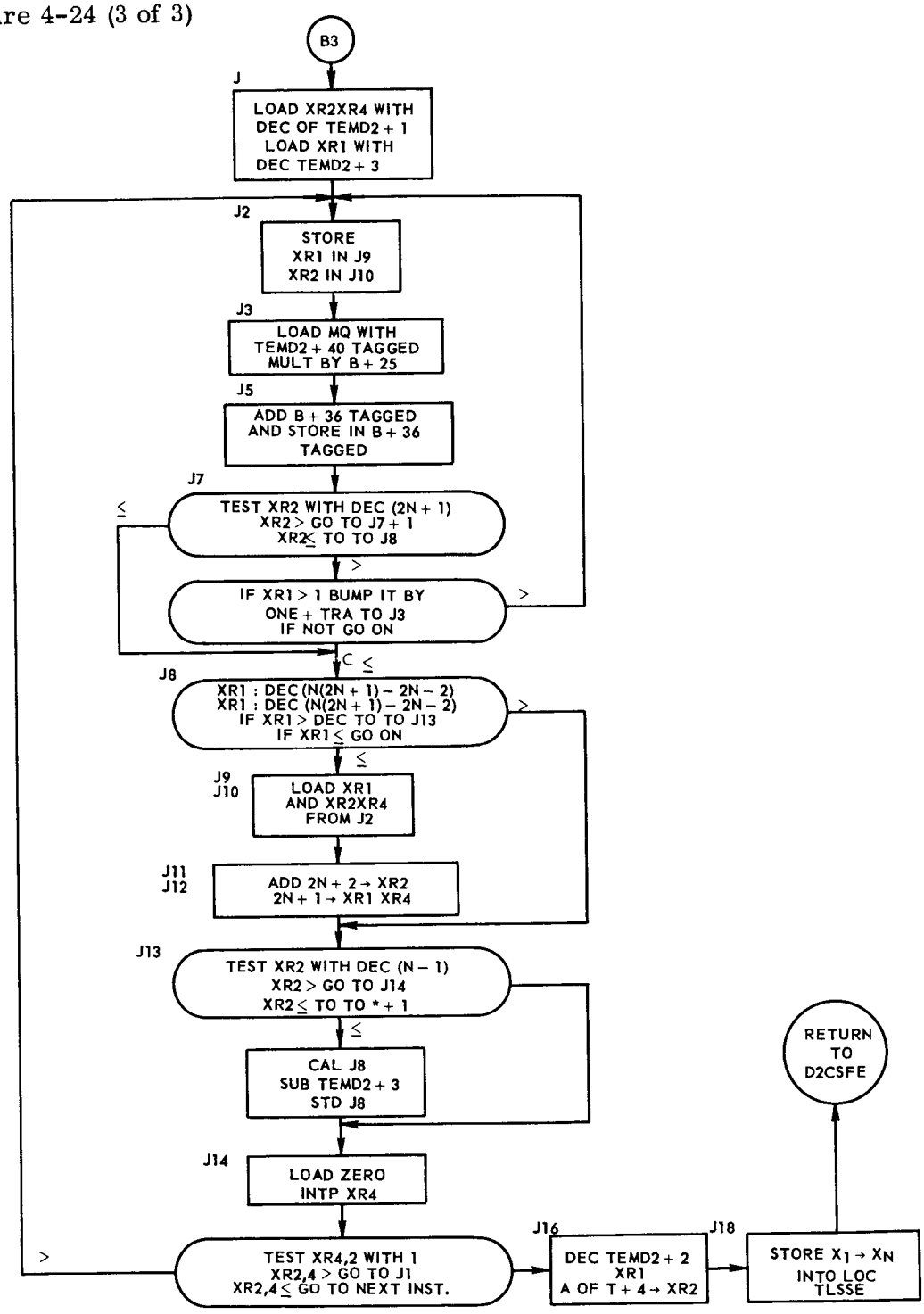


FIGURE 4-23. D2SLNE PROGRAM DETAILED FLOW CHART *(Sheet 3 of 3)*

4.23 CALCULATE STANDARD ERROR FIT PROGRAM (D2CSFE)

This program computes the standard error of fit and elements. The flow chart for C2CSFE is shown in Figure 4-24.

4.23.1 Input Requirements:

D2CSFE uses the square root library subroutine UISQRT.

Symbolic Locations	Definitions
K0000N	Number of coefficients (n)
K0SUMM	Number of equations of condition
KDFSQU	$\sum_{i=1}^{m} (dF_i)^2$
TEMD2	Block of temporary storage generated by D2SLNE and D2CSFE containing $\sum_{i=1}^{m} (dF)_i$ aij and B_{ij}^{-1} (diagonal elements of the inverse matrix).

TLSSSE, TLSSSE + (N - 1): (n) coefficients.

4.23.2 Output Requirements

TLSSSE, TLSSSE + 2N: (n) coefficients immediately followed by standard error of fit and standard error of elements.

4.23.3 Method

$$S_e = \sqrt{\frac{1}{(m-n)} \sum_{i=1}^{m} (dF_i)^2 - \sum_{j=1}^{n} X_j \left[\sum_{i=1}^{m} (dF)_i \, a_{ij} \right]}$$

Solving the above equation for the fit, $S_{(x)j}$ is obtained by

$$S_{x_j} = S_e \sqrt{B_{ij}^{-1}}$$

where B_{ij}^{-1} is the term of the principle diagonal of the inverse of the matrix representing the normal equations.

4.23.4 Usage

a) Calling Sequence

α	TSX	D2CSFE, 4
$\alpha + 1$		Error Return (error in U1SQRT)
$\alpha + 2$		Normal Return

b) Space Required

60 locations + U1SQRT

c) Error Codes

$\alpha + 1$ Error in U1SQRT

d) Special Usage

D2SUNE and D2SLNE must be executed prior to D2CSFE to compute standard error.

e) Constants Used

K0ZERO	PZE	
K00001	DEC	1
KCH233	OCT	233000000000

f) Erasable locations

TEMD2, TEMD2 + 20 ± N(2N + 1) + 2 is input; erasable after completion of D2CSFE.

g) Timing

14.928 msecs for 6 unknowns

4.23.5 Checkout

C2CSFE was checked out using output from D2SUNE and D2SLNE and hand computing the S_e and S_{ej}.

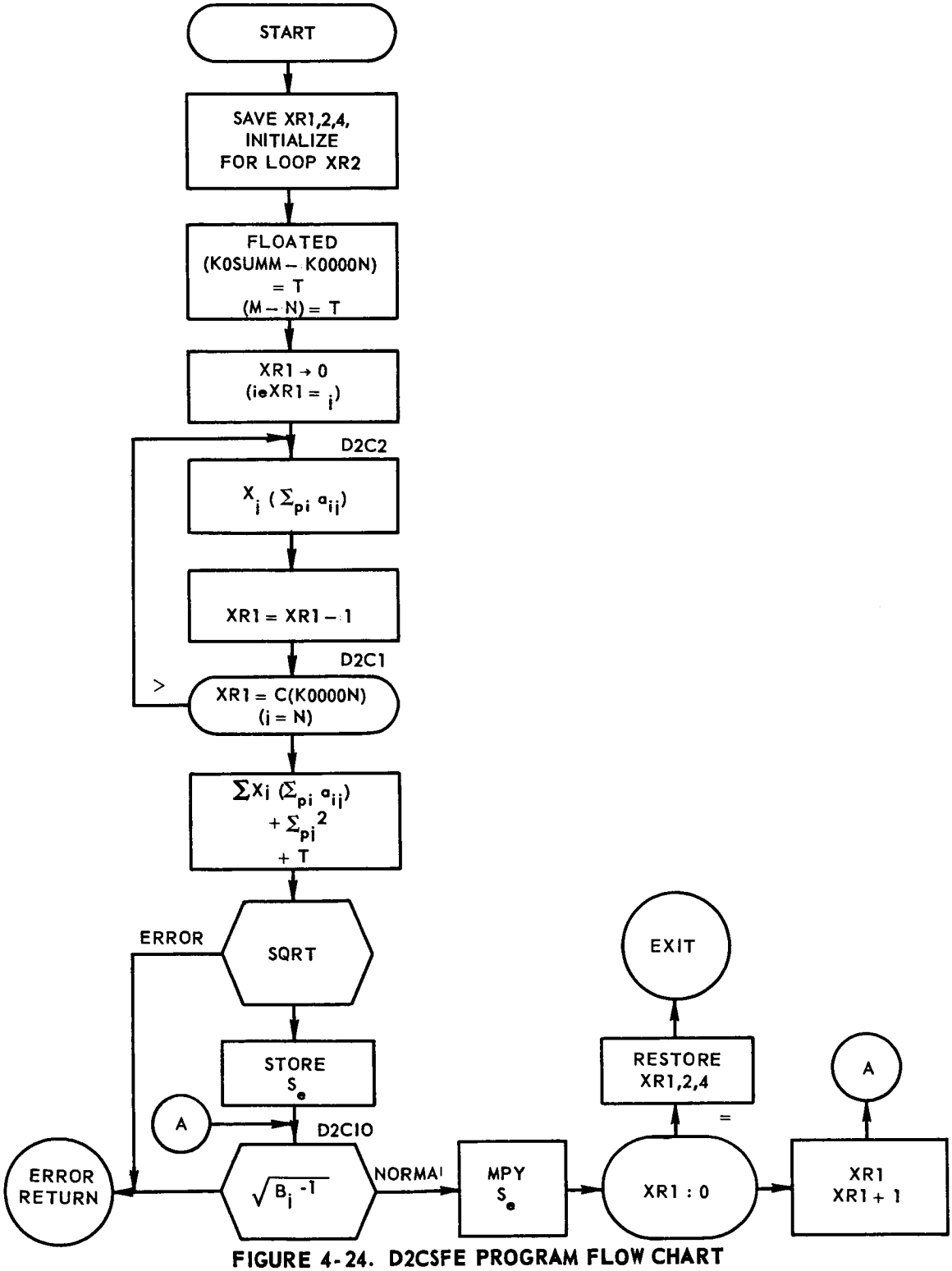


FIGURE 4-24. D2CSFE PROGRAM FLOW CHART

4.24 STATION SUM OF SQUARES OVER M PROGRAM (D2SSMT)

This program computes $\frac{\Sigma(dF)^2}{\text{Number of Equation}}$ for each station in a differential correction interval and sets the results in table form. The flow chart for D2SSMT is shown in Figure 4-26.

4.24.1 Input Requirements

The following data serves as input to the D2SSMT program:

Symbolic Location	Definition
KTOTLM	Number of equations of condition in fixed point
KTDFSQ	$\Sigma(dF)^2$, sum of the square of the right hand sides to the equations of condition in floating point.
KD2CNT	Zero for each series of stations processed. Non-zero after initialization for the series has taken place.

4.24.2 Output Requirements

TD2SSM	Table containing $\Sigma \frac{(dF)^2}{m}$ for each reporting station in the order that the stations were processed for the overall differential correction in floating point.

4.24.3 Method

ΣdF^2 and ΣM for individual stations are obtained as follows:

$$(\Sigma dF^2)_S - (\Sigma dF^2)_{S-1} = (\Sigma dF^2)_1 \text{ station*}$$

This program is entered and $\frac{\Sigma(dF)^2}{m}$ is computed and stored in its place in the table TD2SSM after each station in the interval has been processed.

4.24.4 Usage

a) Calling Sequence

α	TSX	D2SSMT, 4
$\alpha + 1$	Normal Return	

b) Space Required

28 locations

c) Constants Used

K0ZERO = 0

K00001 = 1

KCH233 = 2330000000000_8

TD2SSM BSS 17

d) Timing

First station requires 1.02 milliseconds and .78 milliseconds each time thereafter.

4.24.5 Checkout

The program was entered five times, each time storing $\frac{dF^2}{m}$ in its appropriate location. Results showed correct answers.

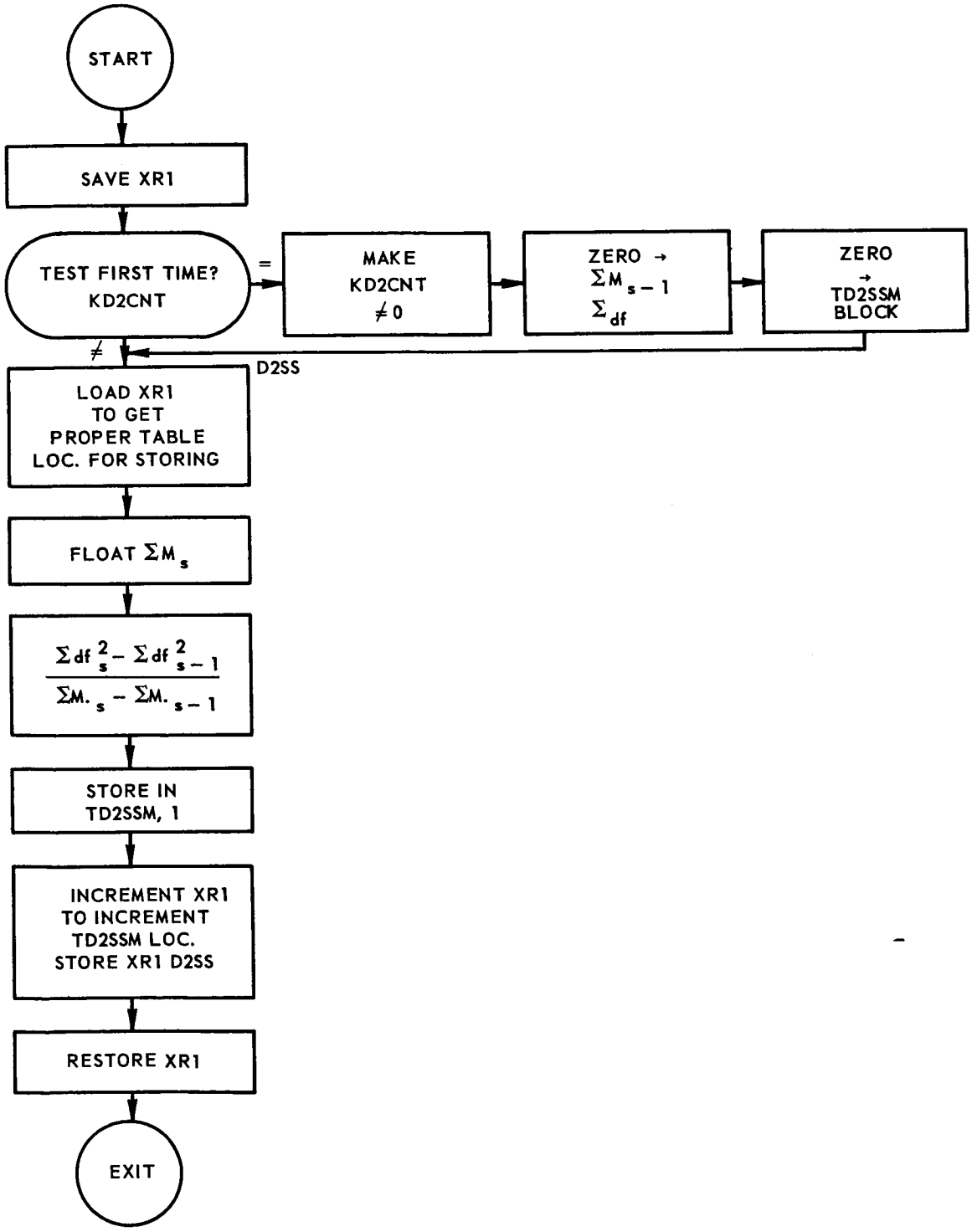


FIGURE 4-25. D2SSMT PROGRAM FLOW CHART

4.25 RETROFIRE AND RE-ENTRY CALCULATIONS PROGRAM (R5RARF)

This program introduces the re-entry phase of the mission. The purpose of this program is twofold, 1) it performs the calculations in preparation for the integration program N0CPNI which generates the re-entry table and impact data for a specified time of retrorocket firing, and 2) R5RARF calculates the time to fire retrorockets for a specified recovery area.

Whenever a new integration table is produced, a new re-entry table is generated based on these latest orbit predictions. This is accomplished by the re-entry program (R5RARF) generating a nominal r, v, and t at burn-out, based on the present capsule setting. The Numerical Integration program (N0CPNI) then generates the re-entry table using this nominal r, v, and t at burn-out. (This same procedure is used whenever a new capsule setting arrives in the computer. R5RARF computes an r, v, and t at burn-out based on this most recent setting. A new re-entry table is now generated according to the current integration table in the machine. Control is then returned to point of interruption when the new capsule setting entered the computer.)

When the re-entry table has been generated, R5RARF is entered and the times to retrofire for the present orbit, and of the mission and for the next emergency area are computed. This is accomplished by Monitor supplying a nominal time to retro-fire for end of this orbit, the longitude of the recovery area, the orbit number, and the location of a $\Delta\lambda$ corresponding to 5 miles. R5RARF computes an r, v, and t at burn-out based on the given time to fire. N0CPNI computes an impact point corresponding to this r, v, and t at burn-out. R5RARF compares this impact point to the longitude of the recovery area. If it does not lie within the prescribed 5 miles, N0CPNI is given another r, v, and t at burn-out based on the time to fire being adjusted. The impact point is generated and the comparison made as above. This cycle continues until the impact point lies within 5 miles of the recovery area. The time to fire is then computed for the end of the mission in the same manner. The next emergency area is determined (Monitor) and a nominal time supplied and the above procedure is used to determine time to fire for next emergency area. The flow chart for the R5RARF program is shown in Figure 4-26.

4.25.1 Input Requirements

R5RARF uses the following programs:

Mercury Programs

C9DTRF

R6BOTH

O5RARF

Library Routines

U1FXPT

U7INTP

U3VMAG

If N0CPNI is to generate a re-entry table and impact data for a specified time of firing, then the following situations must exist:

a) TMSSEC must contain either the present setting of the capsule clock plus 30 seconds or the actual time of firing of the first retrorocket, GMT in fixed point seconds. (There is a 30 second deadlock in the capsule clock between the setting of the block and the time that the retrorockets fire; therefore, Monitor updates the present capsule setting by 30 seconds.)

b) Entry is made at R5RARF with the MQ = +0.

c) If the retrorockets have fired, MCNRRF must contain the number of retros fired as a fixed point integer. If none have fired, then MCNRRF is assumed to be zero.

Upon the first entry to R5RARF to calculate the time to fire retros to impact in a specified area,

a) Address (MQ) should contain the location of a nominal time (GMT of a nominal clock setting) in fixed point seconds of firing of the retrorockets;

b) Decrement (MQ) should contain the location of the longitude (in floating point radians) of the recovery area;

c) Address (AC) should contain the location of the tolerance* (in floating point radians) in longitude for the given recovery area;

d) Decrement (AC) should contain an integer describing the type of recovery area:

D(AC) = 1, if end of 1st orbit recovery area

*Note: Associated with each recovery area there is a quantity $\Delta\lambda$ + tolerance which specifies how close the computed impact longitude must come to the desired impact longitude before the iterative procedure is terminated for calculating the time-to-fire retrorockets. The specified allowance error is 5 miles, which corresponds to different $\Delta\lambda$ at different latitudes. Therefore, each recovery area will have a $\Delta\lambda$ computed on the basis of the latitude of the station.

D(AC) = 2, if end of 2nd orbit recovery area

D(AC) = 3, if end of 3rd orbit recovery area

D(AC) = 0, if next emergency area

D(AC) = 3, with S(AC) minus, if end of mission recovery area

Upon subsequent entries to R5RARF in calculating a refined time to fire retrorockets for the specified area, entry is made to R5RARF with S(MQ) minus. This entry presupposes that the first entry has been made for the first phase of the calculation of the time correction and the output generated in this phase is available.

4.25.2 Output Requirements

If N0CPNI is to generate a re-entry table and impact data, or if this was the initial entry to R5RARF for the time-to-fire calculation, R5RARF constructs TMRARF, a fifteen location table containing the input data for N0CPNI in the following format:

TMRARF	X coordinate	(floating point)
+1	Y coordinate	(floating point)
+2	Z coordinate	(floating point)
+3	Velocity along x axis	(floating point)
+4	Velocity along y axis	(floating point)
+5	Velocity along z axis	(floating point)
+6	Anchor time in seconds associated with $\bar{r}$ and $\bar{v}$ values. This time is measured from midnight preceding launch.	(fixed point)
+7	Address of start of integration table desired, TNINT2 during generation of re-entry table, TNINT9 during retrofire calculations.	
+8	Set to 8, indicating the time stop at which orbit points are to be calculated.	(fixed point)
+9	One, indicating that time is expressed in seconds.	(fixed point)

+10	Length of time in seconds to integrate backwards. Set to zero during retrofire calculations; set to 16 during generation of re-entry table.	(fixed point)
+11	Length of time in seconds to integrate forwards.	(fixed point)
+12	Zero, indicating that the drag routine will be used.	(fixed point)
+13	Set to 1, indicating that $\bar{r}$ and $\bar{v}$ values appear in output table.	(fixed point)
+14	Number of integration steps per output. Set to 1 during generation of re-entry table, set to 2000 during retrofire calculations, indicating not to store re-entry table.	(fixed point)

If N0CPNI is to generate a re-entry and impact data,

a) R5RARF also stores the time of the first entry in the re-entry table in TFESAB in fixed point seconds.

b) R5RARF turns to MFRARF with the accumulator = +0.

c) N0CPNI, upon completion of the calculation of the re-entry table, stores the longitude and latitude of the impact point and the time of impact in the appropriate locations of its output table TMORMC.

If this was the initial entry to R5RARF for the time-to-fire calculation,

a) In generating the TMRARF table, R5RARF indicates that N0CPNI is not to store the re-entry table, but to store the longitude and latitude of the impact point and the time of impact in the table TMIMPP, to be examined by R5RARF.

b) R5RARF returns to MFRARF with the accumulator = +0.

On subsequent entries to R5RARF in computing a refined time to fire retrorockets, R5RARF checks whether the last computed impact point is sufficiently close to the desired impact point. If so, R5RARF computes the following output and stores it in the appropriate locations of the TMORMC block:

a) GMT of retrofire computed.

b) Elapsed capsule time of retrofire computed (ECTRC).

Also, if the designated area is the end of mission area, R5RARF computes the incremental capsule time to retrofire computed (ICTRC). R5RARF then returns to Monitor with a positive non-zero integer in the accumulator. If, however, the computed impact point is not sufficiently close to the desired impact point, R5RARF constructs TMRARF as above, indicating that N0CPNI is to generate impact data based on a corrected time of retrofire. R5RARF in this case returns to MFRARF with the accumulator = +0.

4.25.3 Method

If the time of retrofire given is a clock setting, R5RARF updates this time by 30 seconds to obtain the actual time of firing. The position and velocity vectors of the capsule at the time of firing are calculated by 6 point Lagrangian interpolation, using the library subroutine U7INTP.

If burnout does not occur on an even second, the position vector $\bar{r}_o$ and the velocity vector $\bar{v}_o$ at the first whole second after burnout are computed by the following relations:

$$\bar{r}_o = \bar{r}_{Bo} + \bar{v}_{Bo}\Delta t$$

$$\bar{v}_o = \bar{v}_{Bo} - \bar{r}_{Bo}\,\Delta t/|\bar{r}_{Bo}|^3$$

where $\bar{r}_{Bo}$ and $\bar{v}_{Bo}$ are the position and velocity vectors at burnout, and Δt is the time interval in Mercury units between burnout and the next whole second. $\bar{r}_B$ and $\bar{v}_B$ are calculated by the subroutine R6BOTH.

When R5RARF produces the re-entry table for the present capsule setting or the actual time of firing, the table includes two time steps prior to the first whole second following burnout. These entries are provided so that 6 point Lagrangian interpolation is possible for any observation time occurring after the first second following burnout.

During time-to-fire calculations, R5RARF sets the indicator T0DRAG to a non-zero value to instruct N0CPNI to store the impact data in the block TMIMPP. At the completion of these calculations, the indicator is reset to zero. When the R5RARF produces the re-entry table and impact data for the present capsule setting or the actual time of firing, the indicator T0DRAG is zero to instruct N0CPNI to store the impact data in the output block TMORMC.

During time-to-fire calculations, the re-entry table generated to obtain impact data is not stored. Since N0CPNI always stores values in the first ten locations of the re-entry table, TNINT2 is used as the normal re-entry table and TNINT9 is the re-entry table (10 locations) for the time-to-fire calculations.

The correction to the time of firing is calculated by the C9DTRF program.

The R5RARF program is accurate to 7 significant digits.

4.25.4 Usage

a) Necessary Transfer Information

See Subsection 5.1.1 for input required and 5.1.2 for conditions upon return to Monitor.

b) Space Required

169_{10} locations, excluding constants, tables and subroutines.

c) Error Codes

1) If location MCNOIP is non-zero, no impact point was obtained from numerical integration and hence no time-to-fire calculations can be done. In this case, R5RARF exits to MFRARF with a minus three in the accumulator.

2) If the $\bar{r}$ and $\bar{v}$ at retrofire time are not within the limits of the integration table (causing an error return from U7INTP), R5RARF exits to MFRARF with a minus zero.

d) Constants and Tables Used

K00001	DEC	1	
K00003	DEC	3	
K00012	DEC	12	
K00030	DEC	30	
K00060	DEC	60	
K013.4	DEC	13.44684	Minutes per Mercury unit of times
KINTV2	DEC	2000	store every 2000th integ. step
T.DLTA	DEC	0.0	Mercury time between burnout and next whole second.
TMIMPP	BSS	10	Impact data for retrofiring
TRVIBO	BSS	7	Output from U7INTP, input to R6BOTH

T0DRAG	PZE		Zero - re-entry, nonzero for retrofire
TMRARF	DEC	0,0,0,0,0,0,0,0,8,1,0,2368,0,1,0	
TNINT1	BSS	2000	
TFESAB	BSS	1	First second after burnout
TMSSEC	BSS	1	Present capsule setting, seconds, B35
TNINT2	BSS	2000	
TNINT9	BSS	10	Re-entry table for time to fire calculation

e) Monitor Locations Referred to

MCNOIP

MFRARF

f) Erasable Locations

Locations TEXIT, C, JOIN (10) are erasable

g) Timing

Let 1 = timing for U7INTP = 34.092 msec maximum

R = timing for R6BOTH = 1226 msec maximum

V = timing for U3VMAG = 2.88 msec

C = timing for C9DTRF = 2.772 msec maximum

O = timing for O5RARF = 2.79 msec

1) 5.244 msec + 1 + R + V to prepare for generation of a re-entry table = 164.82 msec

2) 5.508 msec + 1 + R + V mean time to request generation of impact data = 16508 msec

3) 1.056 msec + C + X to compute the time correction = 6.62 msec

4) .372 msec + C + O when no further time correction is necessary = 5.94 msec

4.25.5 Checkout

R5RARF was checked out by hand calculations as far as possible. Then it was tested with simulated Monitor controls for its sequential and iterative processor.

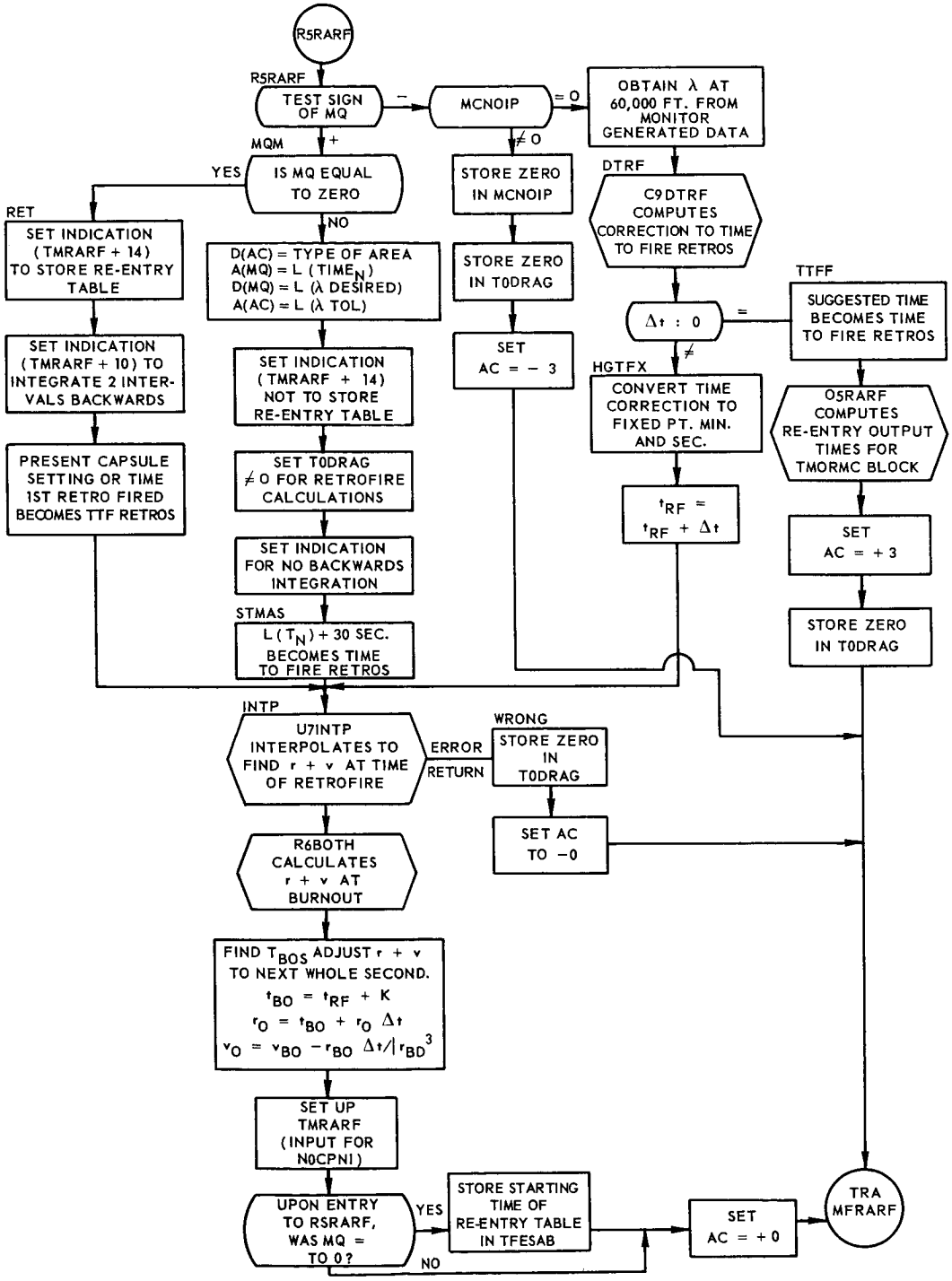


FIGURE 4-26. R5RARF PROGRAM FLOW CHART

4.26 DELTA TIME FOR RETROFIRE PROGRAM (C9DTRF)

This program computes the correction to the suggested time for firing the retrorockets to land the capsule in a designated recovery area. The flow chart for C9DTRF is shown in Figure 4-27.

4.26.1 Input Requirements

Entry to C9DTRF is made with the longitude of the midpoint of the recovery area in floating point radians ($\phi \leq \lambda \leq 2\pi$) in the accumulator and the computed longitude of the impact point in the MQ. Additional input required is described under 5.2.5.

4.26.2 Output Requirements

Upon exiting from C9DTRF, the correction to the time of firing in Mercury time units is in the accumulator. If the computed longitude is within range, the time correction is zero.

4.26.3 Additional Requirements

When first entering C9DTRF, location T1STN0 is zero, indicating a first iteration. For the next iterations until the computed longitude has converged to the desired longitude (within the limits defined by the tolerance in longitude), C9DTRF sets location T1STN0 unequal to zero. After convergence, i.e., when the time correction is computed to be zero, location T1STNO is reset to zero within C9DTRF. When using C9DTRF, if any limitation is put on the number of iterations and convergence is not obtained, location T1STNO must be set equal to zero outside C9DTRF.

4.26.4 Method

The C9DTRF uses the following equations:

a) To take care of the discontinuity in longitude, the minimum distance, $\Delta\lambda_{min}$ ($|\Delta\lambda_{min}| \leq 180^{o}$), from $\lambda_{computed}$ to $\lambda_{desired}$ is computed. See Sheet 2 of Figure 5-2 for details of this calculation.

b) $\Delta t = \Delta\lambda_{min}/(\omega_c - \omega_e)$

ω_c = component of velocity eastward

ω_e = rotational velocity of the earth

c) $\omega_c = (x\dot{y} - y\dot{x})/(x^2 + y^2)$

d) new time correction, $\Delta t = (\Delta t_{previous} - \Delta\lambda_{min})/\Delta\lambda_p$

For details on calculation of $\Delta\lambda_p$, the actual change in longitude achieved by changing the firing time by the amount Δt_p, see Sheet 2 of Figure 4-28.

The C9DTRF program is accurate to 7 significant digits.

4.26.5 Usage

a) Calling Sequence

α	TSX	C9DTRF, 4
$\alpha+1$	PZE	Location (1st component of r vector)
$\alpha+2$	PZE	Location (1st component of v vector)
$\alpha+3$		Tolerance in longitude
$\alpha+4$		Normal return

Units: longitude in floating point radians; $\overline{r}$ and $\overline{v}$ vectors in Mercury units.

It is assumed that the components of each vector are in consecutive locations. Tolerance in longitude is 5 nautical miles. The same value in radians varies with the latitude.

b) Space Required—92_{10} locations excluding constants.

c) Constants Used

K00.PI	DEC	3.14159265
KKK2PI	DEC	6.283185307
K.OMEG	DEC	.058833543
K00060	DEC	60
K00001	DEC	1

d) Table Used

T1STNO	PZE

e) Erasable Locations

12_{10}

ENS (2), RVEC(4), HGDT, LMDA2, LCOM, LDES, DLMIN, DLPRE.

f) Timing

1) .504 msec. if value is within range

2) 2.772 msec. for an initial refinement

3) 1.524 msec. for any other refinement

4.26.6 Checkout

First C9DTRF was checked out by itself testing every variation in the program. The numerical calculations proper to C9DTRF were compared with hand calculations. Then C9DTRF was checked out together with R5RARF to test how quickly its equations refined the time to fire sufficiently. For this phase of the testing, Monitor controls for the R5RARF program were simulated and the output of N0PCNI was used as input for C9DTRF.

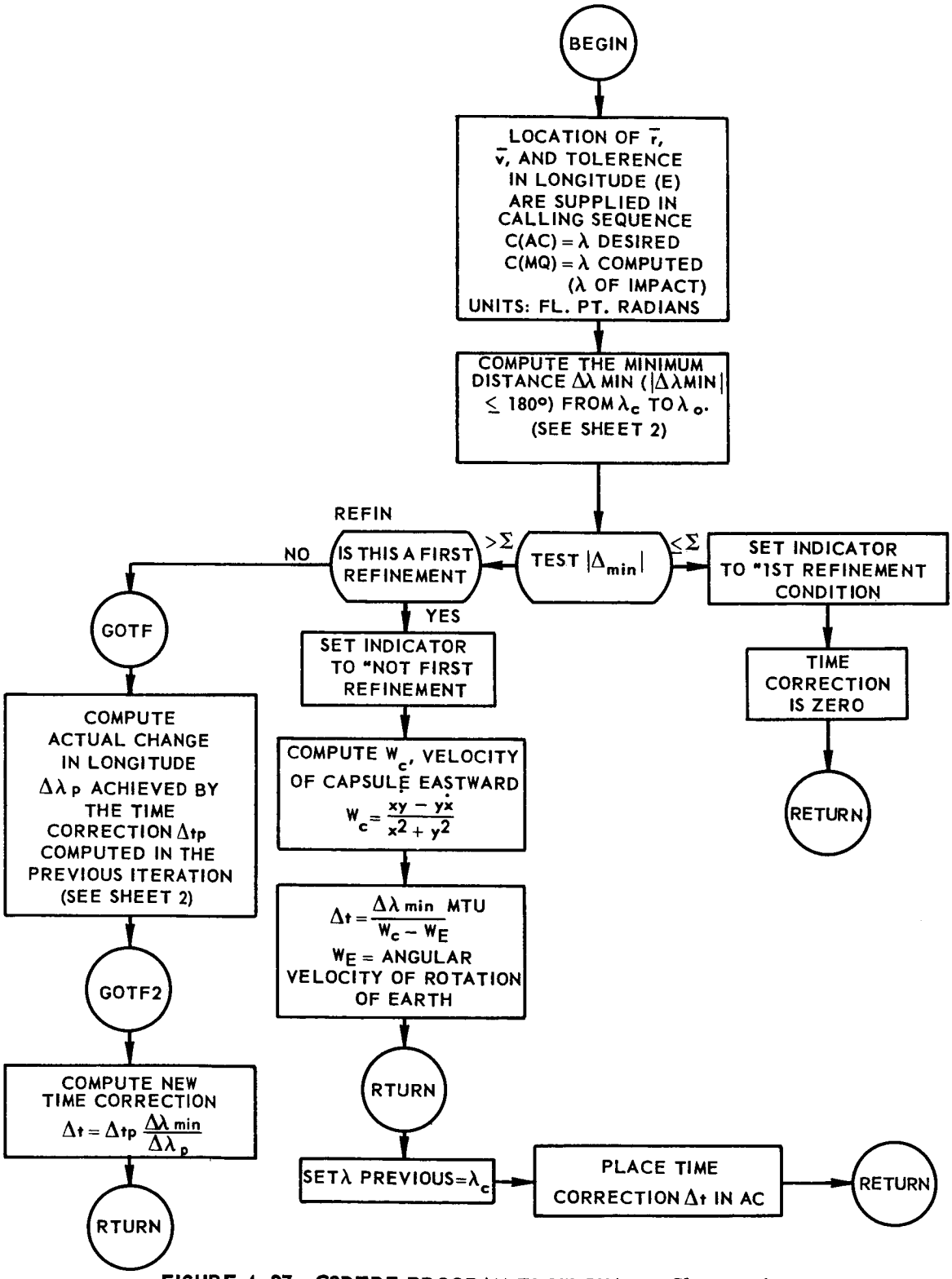
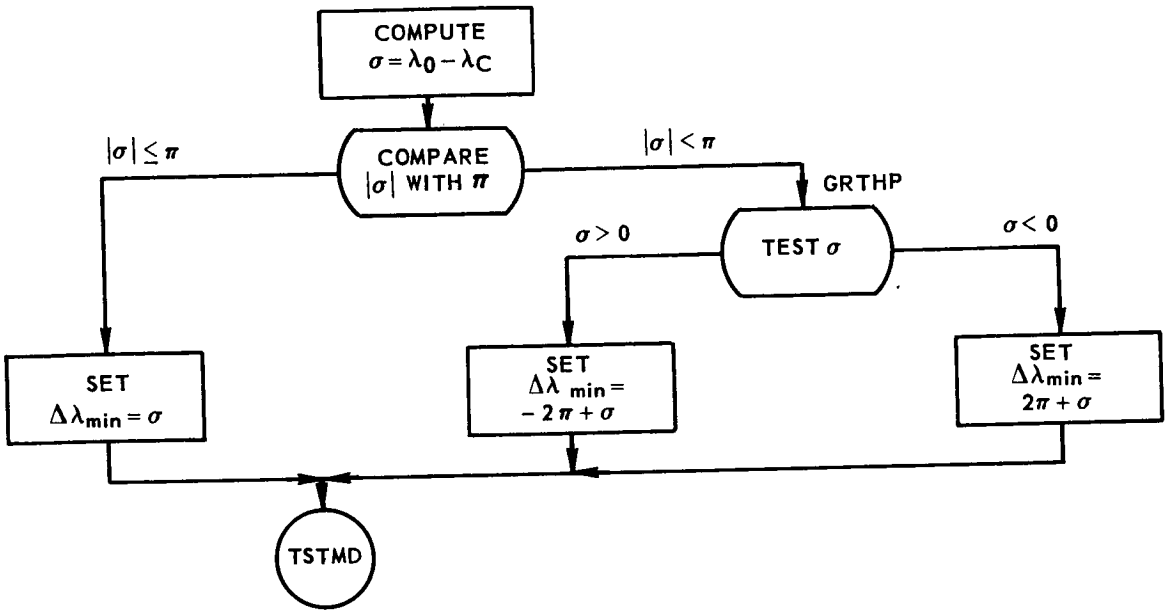


FIGURE 4-27. C9DTRF PROGRAM FLOW CHART *(Sheet 1 of 2)*

GIVEN λ_C, COMPUTED LANDING POINT LONGITUDE
λ_D, DESIRED LANDING POINT LONGITUDE
TO COMPUTE THE MINIMUM DISTANCE $\Delta\lambda_{min}$
(POSITIVE OR NEGATIVE) FROM λ_C TO λ_D.



GIVEN λ PREV. LONGITUDE OF PREVIOUS LANDING POINT
λ_C, LONGITUDE OF PRESENT COMPUTED LANDING POINT.
Δt_p, TIME CORRECTION COMPUTED IN THE PREVIOUS ITERATION
TO COMPUTE THE ACTUAL CHANGE IN LONGITUDE,
$\Delta\lambda_p$ ACHIEVED BY CHANGING THE FIRING TIME BY THE AMOUNT Δt_p.

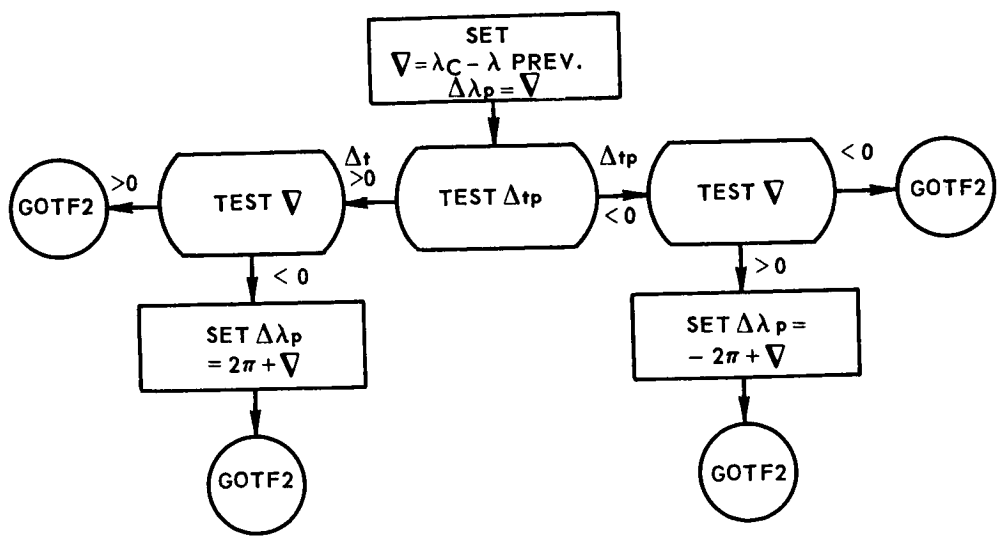


FIGURE 4-27. C9DTRF PROGRAM FLOW CHART *(Sheet 2 of 2)*

4.27 COMPUTE $\bar{r}$ AND $\bar{v}$ AT BURNOUT PROGRAM (R6BOTH)

Given position, velocity, and the time associated with these quantities at the start of retrorocket fire, R6BOTH produces the position, velocity, and the associated time at the termination of retrorocket fire. The flow chart for R6BOTH is shown in Figure 4-28.

4.27.1 Input Requirements

R6BOTH uses the following three library subroutines:

U3UNTV

U3VPRO

U1SQRT

The input to R6BOTH consists of seven words in consecutive ascending core locations: time (associated with the given $\bar{R}$, $\bar{V}$) in seconds, floating point; $\bar{R}$ components (X, Y, Z), $\bar{V}$ components (V_x, V_y, V_z) all in Mercury units, floating point.

The number of retrorockets fired, N, must be given as an integer in the accumulator when entering the program.

a) 34° orientation is assumed for the thrust vector. 1080 lbs (at surface) is the assumed (constant) thrust of rockets during retrofire. Vehicle weight is taken to be 2700 lbs.

b) Firing intervals are 5, 5, 5 seconds; duration of 12.0 second for each burst.

4.27.2 Output Requirements

The output of R6BOTH is identical to the input block in requirements and format.

4.27.3 Method

The R6BOTH program uses the following equations:

$$\bar{r} = \bar{r}_o + v_o (\Delta t)^2 \bar{r}_o / 2r_o^3 + (\Delta t)^2 \bar{A}/2$$

$$\bar{v} = \bar{v}_o - (\Delta t) \bar{r}_o / r_o^3 + (\Delta t) \bar{A}$$

This program has been tested using 4, 3, 2, 1, 0 rockets; 5 iterations of a single retrorocket, with adjusted thrust and timing; and a single pass using average thrust.

4.27.4 Usage

a) Calling Sequence

Place integral N in the AC,

α	TSX	R6BOTH, 4
$\alpha + 1$	PZE	L(I), 0, L(O)
$\alpha + 2$	Error Return	
$\alpha + 3$	Normal Return	

where

N = number of retrorockets fired

L(I) = Location of first entry in input block

L(O) = Location of first entry in output block.

Program uses 200 locations of which the last 33 are erasable.

Error return is made if and only if $N \neq 0, 1, 2, 3$.

b) Constants Used

K00001	DEC	1
K0003	DEC	3
K002.0	DEC	2.0

TTHRUS—A location table of thrust factors.

R6TMP is a block of 33 locations used for temporary storage.

c) Timing

For N = 1, 2, 3, rockets fired

Time = 25.2 + 2* (N-1)*(24.352) msec (709), i.e., 122.6 msec for N = 3.

4.27.5 Checkout

Program has been checked inter se and by hand computation for N = 3.

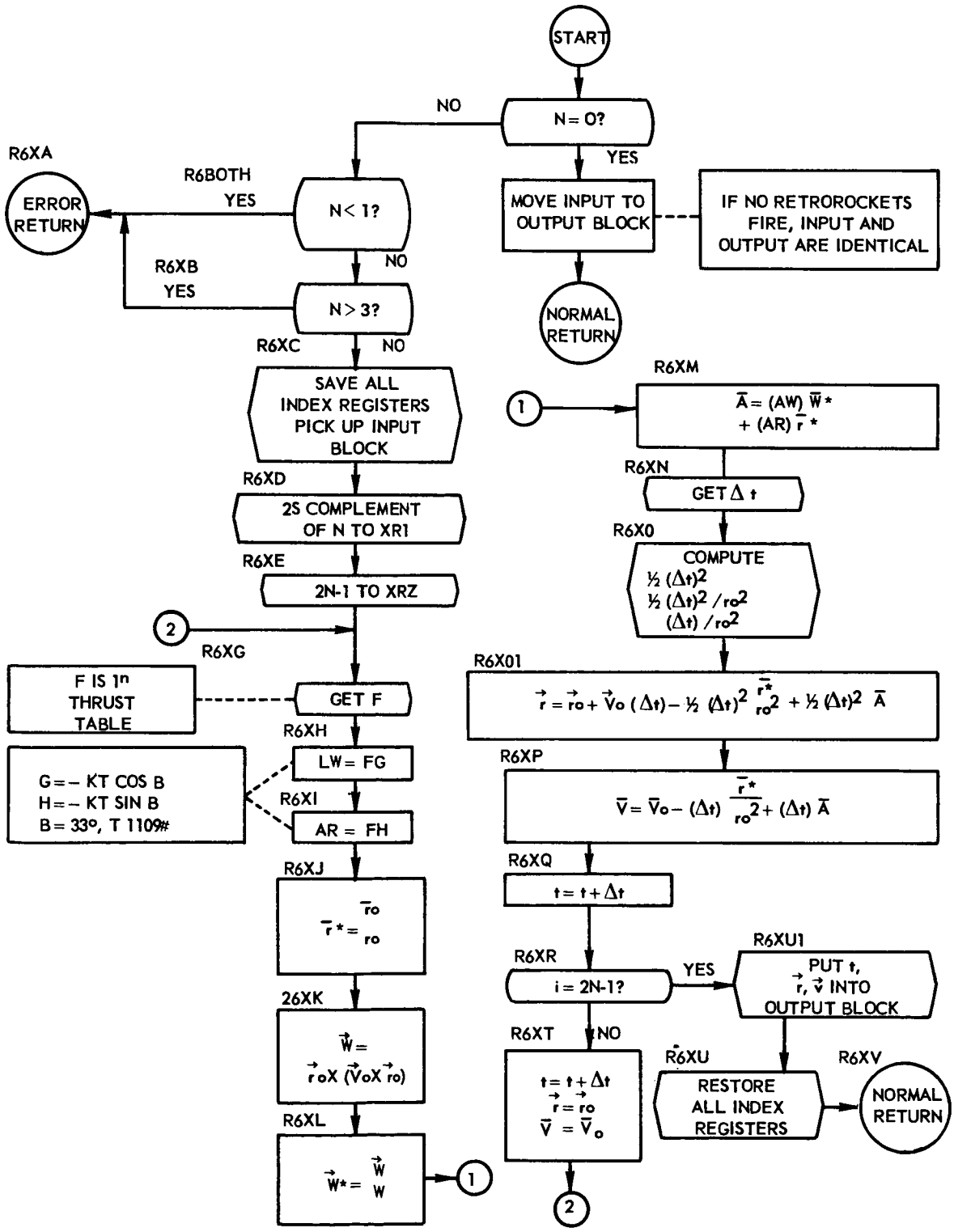


FIGURE 4-28. R6BOTH PROGRAM FLOW CHART

4.28 ELAPSED AND INCREMENTED CAPSULE TIME PROGRAM (O5RARF)

Given the computed GMT for retrofire and the type of recovery area for which this time was computed, O5RARF computes the elapsed capsule time (ECTRC) and the incremental capsule time (ICTRC) (if the designated area is the end of mission area) and places the computed times in the proper locations of the output table. The flow chart for the O5RARF program is shown in Figure 4-29.

4.28.1 Input Requirements:

The computed GMT for retrofire is fixed point seconds and should be placed in the MQ upon entry.

The proper code for the type of recovery area should be placed in the decrement portion of the accumulator as follows:

0 for an emergency abort recovery area

1 for the end of first orbit

2 for the end of second orbit

3 for the end of third orbit

4 for the end of mission

4.28.2 Output Requirements

O5RARF uses the GMT to fire retrorockets to compute the corresponding ECTRC and, if required, the ICTRC. This program places these values in the locations of table TMORMC (see Table 4-7) depending on the type of recovery area for which the times were computed.

4.28.3 Method

Equations Used

Calculation of elapsed capsule time to retrofire computed (ECTRC) from Greenwich mean time to retrofire (GMTRC)

$$(GMTRC - x)\, k + y = ECTRC$$

x = GMT of the last reading of the capsule clock.

y = the reading of the capsule clock (ECT) at time x.

k = change in capsule clock per unit change in ground time.

The ECTRS, the present setting of elapsed capsule time to retrofire and the ECTRC are converted from seconds to hours, minutes and seconds. The increments (positive or negative) will be computed for the minimum digital changes in seconds, minutes, and hours, in that order, to "advance" the clock from the ECTRS to the ECTRC. The calculation adjusts for the carry over from seconds to minutes and minutes to hours.

All calculations of time are accurate to the nearest second.

4.28.4 Usage

a) Calling Sequence

α	TSX	O5RARF, 4
$\alpha + 1$	Normal Return	

b) Space Required

168_{10} locations excluding constants and tables.

c) Constants Used

K00001	DEC	1
K00030	DEC	30
K00060	DEC	60
KD0002	PZE	,,2
KD0003	PZE	,,3
KD0004	PZE	,,4

d) Tables Used

TMETRS	BSS	1	ECT of retrofire set
TMGMT2	BSS	1	GMT of last reading
TMECT2	BSS	1	ECT at time of last reading
TMSLOP	DEC	1B2	Relation between speed of capsule clock and GMT
TMORMC	BSS	36	Output Block

TABLE 4-7. TMORMC TABLE FORMAT

Location	Description	Format
TMORMC + 5	GMT of retrofire computed for emergency abort recovery area (GMTRC)	Minutes in address Seconds in decrement
TMORMC + 6	Elapsed capsule time to retrofire computed for emergency abort recovery area (ECTRC)	Same
TMORMC + 7	GMTRC for end of present orbit	Same
TMORMC + 8	ECTRC for end of present orbit	Same
TMORMC + 9	GMTRC for end of mission	Same
TMORMC + 10	ECTRC for end of mission	Same
TMORMC + 18	Incremental capsule time to retrofire computed for end of mission (ICTRC) ± hours	Integer in address
TMORMC + 19	Same as above ± minutes	
TMORMC + 20	Same as above ± seconds	

Note: An end-of-third-orbit is treated as both end of present orbit and end of mission. Values are stored in TMORMC + 7 through TMORMC + 10 and TMORMC + 18 through TMORMC + 20.

e) Erasable Locations

Locations TAREA through DELTA are 18_{10} erasable locations.

f) Timing

For an end of mission, 4.89 msecs.

For other than an end of mission, 2.79 msecs.

4.28.5 Checkout

All variations of the program were tested and program results compared favorably with hand calculations.

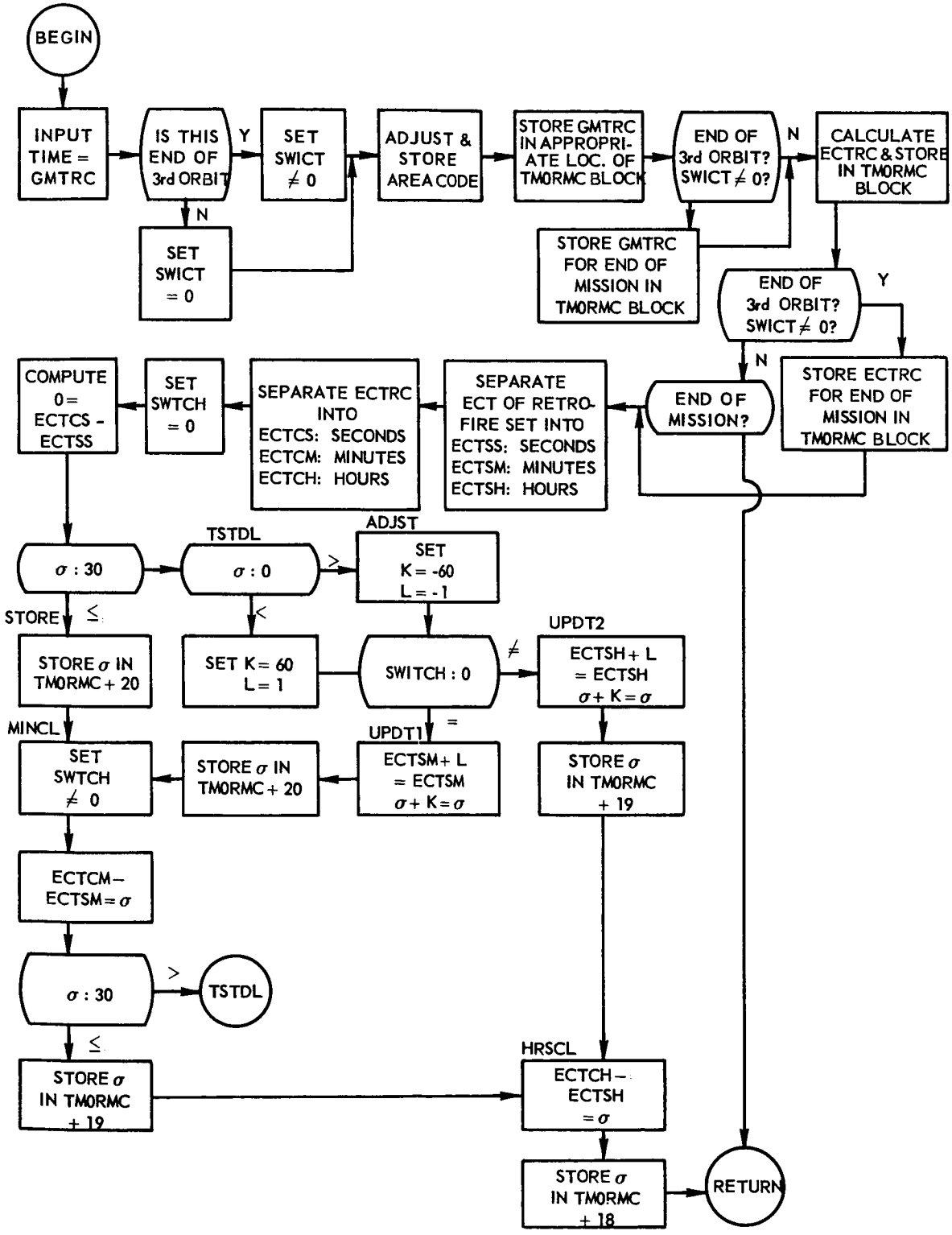


FIGURE 4-29. 05RARF PROGRAM FLOW CHART

4.29 ACQUISITION DATA GENERATION PROGRAM (A0STAD)

Given a station number and estimated time to start testing for horizon crossing, A0STAD generates acquisition data (range, azimuth, elevation) at various points of the passage of the capsule over the station. An error return is provided to indicate that no acquisition data is to be sent because the capsule will not come at least 1° in elevation above the horizon. The flow chart for A0STAD is shown in Figure 4-30.

4.29.1 Input Requirements

The following library routines are used by the A0STAD program.

U1SICO, U1SQRT, U1ATAB, U1ASCO, U1ATNA, U3DOTP, U7INTP, U3VMAG, and UAILSC.

Two control words, MCACQ1 and MCACQ2, are provided by Monitor:

D(MCACQ1) contains the internal station number, used to reference the station characteristics block.

A(MCACQ2) contains the estimated time of horizon crossing in whole minutes.

4.29.2 Output Requirements

If acquisition data is to be sent, and if the height of the capsule at the time it is approximately 1° above the horizon is greater than 450,000 ft. then A0STAD generates a 16-word output block as shown in Table 4-8.

If acquisition data is to be sent and if the height of the capsule at the time when the capsule is approximately 1° above the horizon is less than or equal to 450,000 feet the 16-word output block generated consists of the following four times with the associated R, A, E. The format is identical with TMSTAD.

Time	Description
1st time = t_1	time at which the capsule is approx 1° above the horizon
2nd time = t_2	$t_2 = t_1 + \Delta t'$
3rd time = t_3	$t_3 = t_2 + \Delta t'$
4th time = t_4	$t_4 = t_3 + \Delta t'$

Note: As of 3/24/61, $\Delta t'$ 30 seconds.

TABLE 4-8. A0STAD 16-WORD OUTPUT BLOCK FORMAT (Sheet 1 of 2)

Location	Contents	Description
TMSTAD	Time (t_1)(A = min, D = sec)	Time at which capsule is approx. 1° above the horizon.
+1	Range (floating point yards)	Range of capsule relative to the station at t_1.
+2	Azimuth (floating point radians)	Azimuth of capsule relative to station at t_1. $\phi \leq A < 2\pi$
+3	Elevation (floating point radians)	Elevation of capsule relative to station at t_1. $-\pi/2 \leq E \leq \pi/2$
+4	Time (t_{10})(A = min, D = sec)	Time at which capsule is approximately 10° along the capsule path from the horizon.
+5	Range (floating point yards)	Range of capsule relative to the station at t_{10}.
+6	Azimuth (floating point radians)	Azimuth of capsule relative to station at t_{10}. $\phi \leq A < 2\pi$
+7	Elevation (floating point radians)	Elevation of capsule relative to station at t_{10}. $-\pi/2 \leq E \leq \pi/2$
+8	Time (t_{30})(A = min, D = sec)	Time at which capsule is approx. 30° along the capsule trajectory from the horizon.
+9	Range (floating point yards)	Range of capsule relative to the station at t_{30}.
+10	Azimuth (floating point radians)	Azimuth of capsule relative to the station at t_{30}. $\phi \leq A < 2\pi$
+11	Elevation (floating point radians)	Elevation of capsule relative to the station at t_{30}. $-\pi/2 \leq E \leq \pi/2$
+12	Time (t_x)(A = min, D = sec)	$t_x = t_{ca} + \Delta t$, where t_{ca} is the time of closest approach and Δt = 10 sec.
+13	Range (floating point yards)	Range of capsule relative to station at t_x.

TABLE 4-8. A0STAD 16-WORD OUTPUT BLOCK FORMAT (Sheet 2 of 2)

Location	Contents	Description
+14	Azimuth (floating point radians)	Azimuth of capsule relative to station at t_x. $\phi \leq A < 2\pi$
+15	Elevation (floating point radians)	Elevation of capsule relative to station at t_x. $\pi/2 \leq E \leq \pi/2$

Note: If the elevation at t_{10} or t_{30} is negative, the corresponding data is not sent. This is indicated by zero in the corresponding four words of the output block.

4.29.3 Other Requirements

It is assumed that the station characteristics block is in core storage available to the routine, containing in particular:

Position	Description
8	sin ϕ (floating point radians), where ϕ is the station geodetic latitude.
9	cos ϕ (floating point radians).
10	R_s sin $(\phi - \phi')$ (floating point Mercury units) where R_s = radius of the earth at the station, ϕ = station geodetic latitude, and ϕ' = station geocentric latitude.
11	R_s cos $(\phi - \phi')$ (floating point Mercury units)
12	R_s, radius from earth center to station, (floating point Mercury units).
18	Inertial longitude of station at reference time (floating point radians). $\phi \leq \lambda < 2\pi$
29	sin ϕ' (floating point radians) (where ϕ' is the station geocentric latitude).
30	cos ϕ' (floating point radians).

The locations of the above quantities are defined (by EQU pseudo-ops) relative to the first location of the station characteristics block, which, therefore, cannot be rearranged.

It is assumed that the ending time of the re-entry table (TNINT2 + 3) and the time interval (TNINT2 + 1) are in seconds.

4.29.4 Method

A0STAD searches for initial horizon crossing (t_0) within t_s + 15 minutes and t_s - 15 minutes, where t_s = starting time. It computes t_0 by linear interpolation, if necessary.

If the initial t_s is below the horizon, A0STAD searches forward in 1 minute intervals for 15 minutes and if t_0 has not been reached, it searches backwards

from t_s for 15 minutes. If the capsule is not yet above the horizon, A0STAD proceeds to the error return.

If the initial t_s is above the horizon, or when the capsule crosses above the horizon during a backwards search, A0STAD searches backwards until the initial horizon crossing time is reached.

If the initial t_s is off the end of the re-entry table, a backward search is initiated with t_s set at the last time in the re-entry table.

After calculating t_0, A0STAD searches for the time at which the capsule is 1° above the horizon, computing t_1 by linear interpolation.

In using the re-entry table in the search for t_0 and t_1, if any intermediate calculated time is earlier than the start of the re-entry table, we switch to use the orbit table. If the time is still out of the orbit table, A0STAD proceeds to the error return.

Having established that the capsule will come in view of the station, A0STAD tests the height of the capsule at the time of 1° (t_1).

$$r_1 = \sqrt{x_1^2 + y_1^2 + Z_1^2}$$

$$h_1 = r_1 - 1 + .0033874\left(\frac{Z_1}{r_1}\right)^2$$

If $h_1 > 450{,}000$ feet, the following times of acquisition data are calculated:

$$t_2 = t_1 + \Delta' t$$

$$t_3 = t_2 + \Delta' t$$

$$t_4 = t_3 + \Delta' t$$

If $h_1 > 450{,}000$ feet, the following calculations are made to get t_{10}, t_{30}, and t_x. A0STAD finds three values of range, ρ_{-1}, ρ_1, and ρ_2 at consecutive minute intervals, satisfying the relations $\rho_{-1} \geq \rho_1 \rho_1 < \rho_2$. Assuming a parabolic fit of ρ as a function of t, A0STAD computes the time of closest approach, t_{ca}, i.e., the time at which range is a minimum.

$$t_{ca} = t_{-1} - \frac{1}{2}\left(\frac{4\rho_1 - \rho_2 - 3\rho_{-1}}{\rho_2 - 2\rho_1 + \rho_{-1}}\right)$$

where t_{-1} is the time at which range is ρ_{-1}.

The minimum horizontal distance M from the station to the capsule is given by $M = \rho_{ca} \cos E_{ca}$ where ρ_{ca} and E_{ca} are the values of range and elevation, respectively, at time t_{ca}.

t_{10}, t_{30}, t_x are computed by the following formulas:

$t_{10} = t_0 + (101.6 - .092 \text{ millisecs})$

$t_{30} = t_0 + (167.1 - .09 \text{ millisecs})$

$t_x = t_{ca} + \Delta t$ where $\Delta t = 10$ secs.

<u>Note</u>: While searching for ρ_{-1}, ρ_1, and ρ_2, as of March, 1961, in the event that the time exceeds the ending time of TNINT2, A0STAD sets t_x equal to the last time in TNINT2. Now that the recent modification of testing for $h_1 \leq 450{,}000$ feet at t_1 has been added, this provision is unnecessary and may be eliminated from A0STAD if the modification proves successful.

Calculation of range P, and sin E

a) λ', inertial longitude of station at time $t = \lambda'_o + .00437526905t$

b) Inertial position of station at time t,

$R_x = (R_s \cos \phi') \cos \lambda$

$R_y = (R_s \cos \phi') \sin \lambda$

$R_z = (R_s \sin \phi')$

where ϕ' = station geocentric latitude

R_s = radius of earth at station

c) $N_X = (\cos \phi) \cos \lambda$

$N_Y = (\cos \phi) \sin \lambda$

$N_Z = (\sin \phi)$

where ϕ = station geodetic latitude

d) $P = \bar{r} - R$ where

$\bar{r}$ = capsule inertial position

e) $P = |P|$

f) $\sin E = N \cdot P / P$

Note: Although UA1LSC computes R, A, E, and is included as a subroutine, an alternate calculation of R and sin E is included in order to save computation time.

To determine whether to use the orbit integration table or the re-entry table to obtain the inertial position vector of the capsule at some time t_{acq}, the time is compared with the time of the first even second after burnout (in location MCESAB). If $t_{acq} < t_{BO}$ the orbit table is used. If $t_{acq} \geq t_{BO}$, the re-entry table is used.

4.29.5 Usage

a) Calling Sequence

α	TSX	A0STAD,4
$\alpha + 1$	Error Return	
$\alpha + 2$	Normal return	

b) Space Required—488_{10} locations excluding subroutines, constants, and tables.

c) Error Codes—See Method, 4.31.4

Other error provisions are as follows:

1) If the range keeps increasing, i.e., if the capsule is getting further away from the horizon, an error return to 1,4 is made.

2) If at any time an error return is received from one of the library subroutines, A0STAD checks to see if there is a good value stored for either t_1, t_{10}, t_{30}, t_x. If there is a least one good time, this is outputted and the remaining locations of TMSTAD are set to zero. If not, then an error return due to bad data is made to 1,4.

d) Tables: TMSTCH, TNINTI, TNINT2

e) Constants

K00000	DEC	0
K00001	DEC	1
K00002	DEC	2
K00003	DEC	3
K00004	DEC	4
K00006	DEC	6
K00007	DEC	7
K00010	DEC	10
K00060	DEC	60
K001.0	DEC	1.0
K002.0	DEC	2.0
K003.0	DEC	3.0
K004.0	DEC	4.0
K000.5	DEC	.5
K0060.	DEC	60.
K045.4	DEC	.021504685 - 450,000 feet in Mercury units
KCH233	OCT	23000000000
K0.1DG	DEC	.0174532925 - 1 degree in radians
K0.SIG	DEC	.01745241 - sin (1 degree)
KMUYDS	DEC	6975224.19 - conversion factor - Mercury units to yds.
K.MORB	DEC	.0033874 - difference in radius of earth at equator and at pole.
K.RTER	DEC	.00437526905 - earth rotational factor

KDLTSC	DEC	30 - Δ't
KD0060	PZE	,, 60

c) Erasable locations: The last locations are erasable

d) Communication cells used: MCESAB

e) Suggested Refinements:

Either new curve fits should be made to send the proper acquisition data when the height of the capsule is $\leq$ 450,000 feet at t, or there should be a special indication to distinguish this kind of acquisition data (30 seconds apart) from the normal way of calculating the data.

4.29.6 Checkout

All variations of the program were successfully tested. For details and hand calculations an A0STAD notebook is available at the Goddard Space Flight Center.

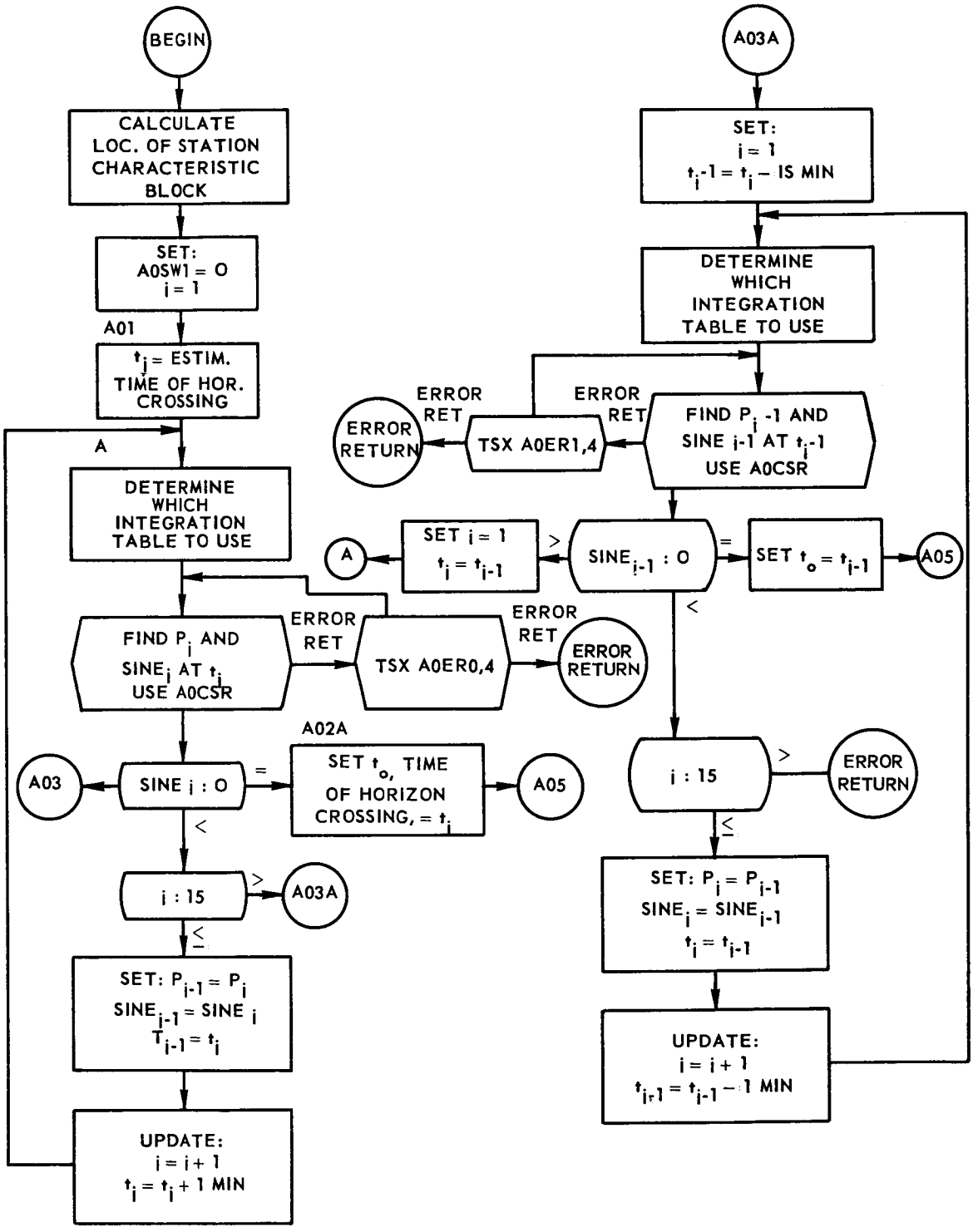


FIGURE 4-30. AOSTAD PROGRAM FLOW CHART *(Sheet 1 of 6)*

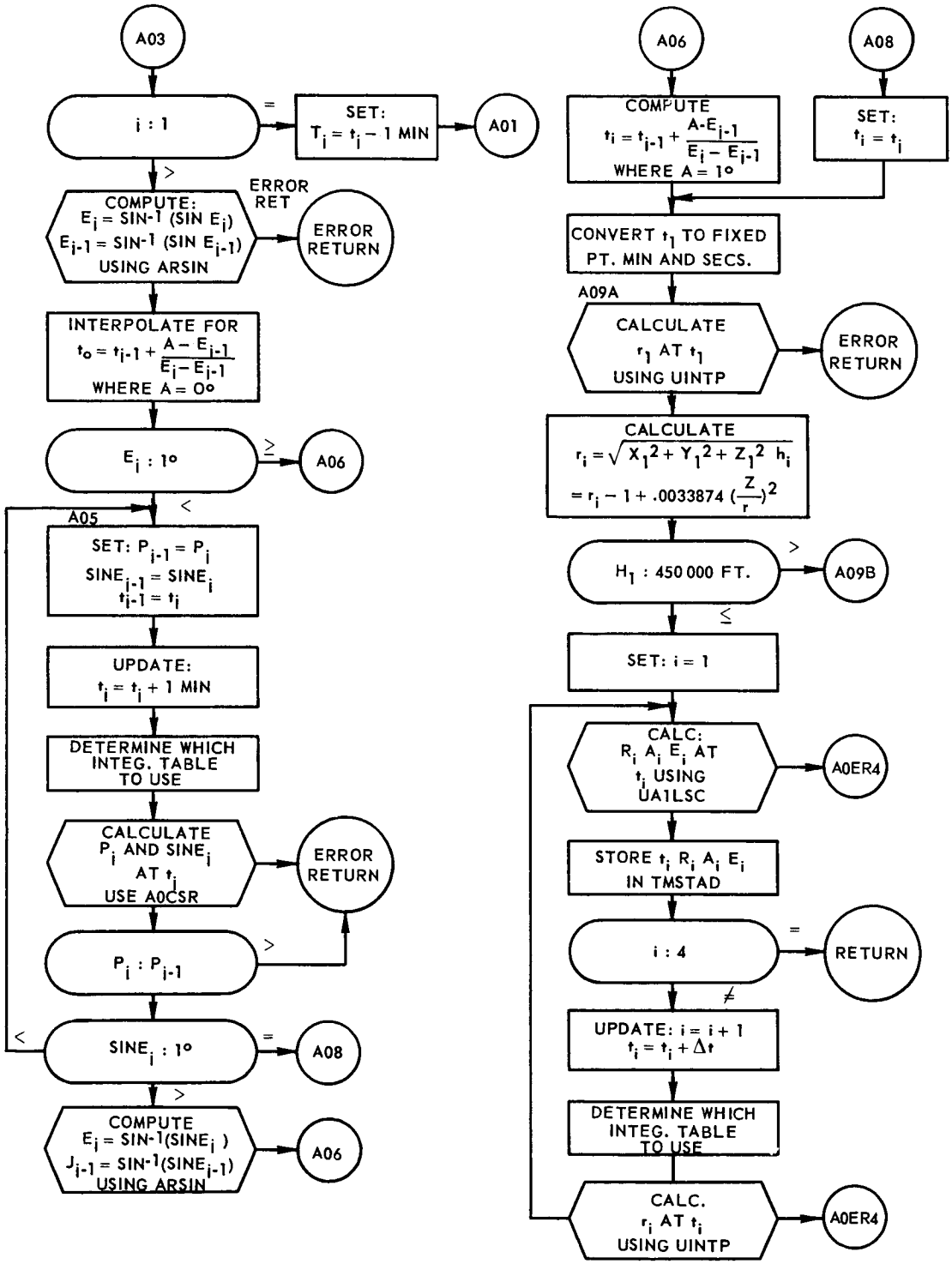


FIGURE 4-30. AOSTAD PROGRAM FLOW CHART *(Sheet 2 of 6)*

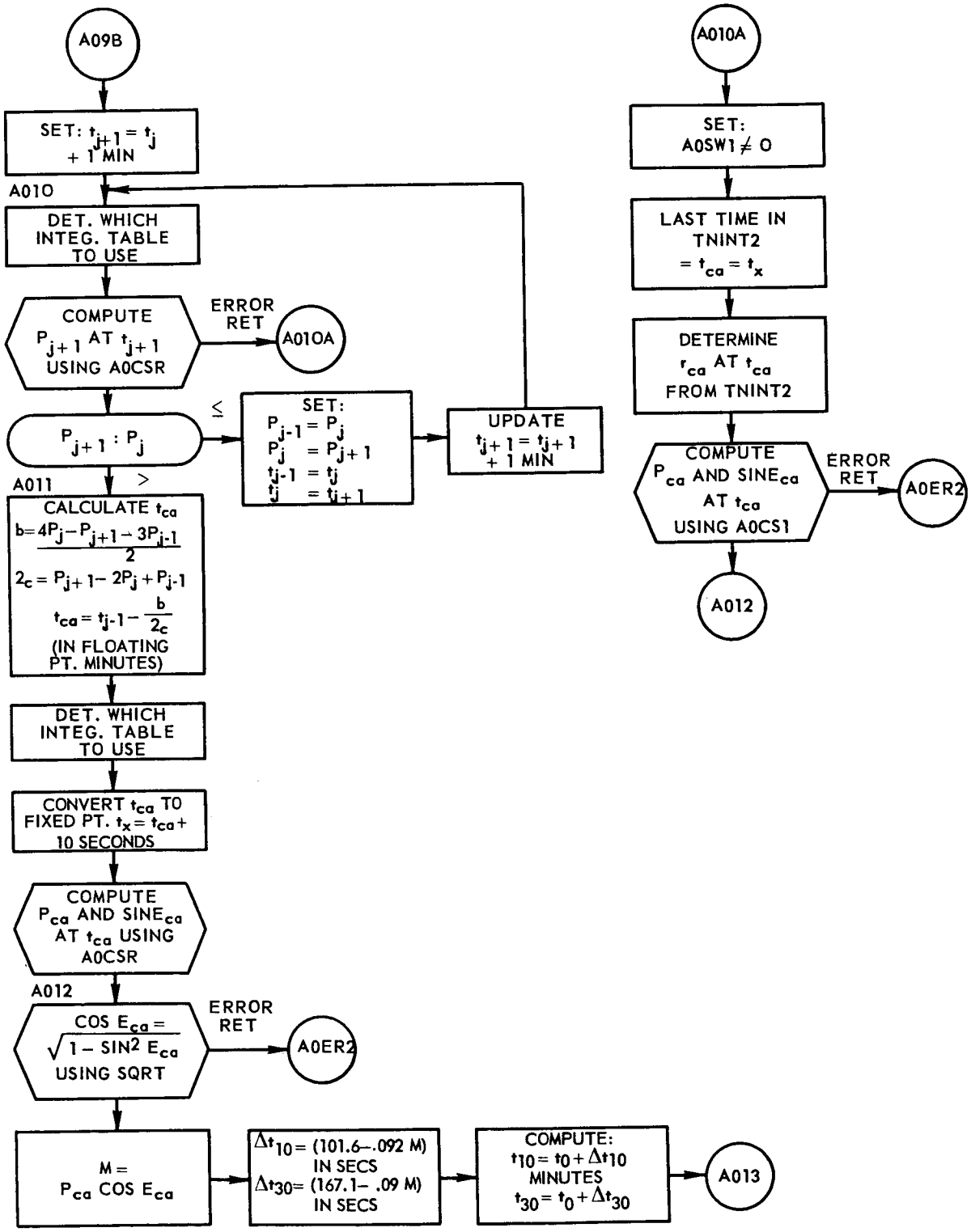


FIGURE 4-30. AOSTAD PROGRAM FLOW CHART *(Sheet 3 of 6)*

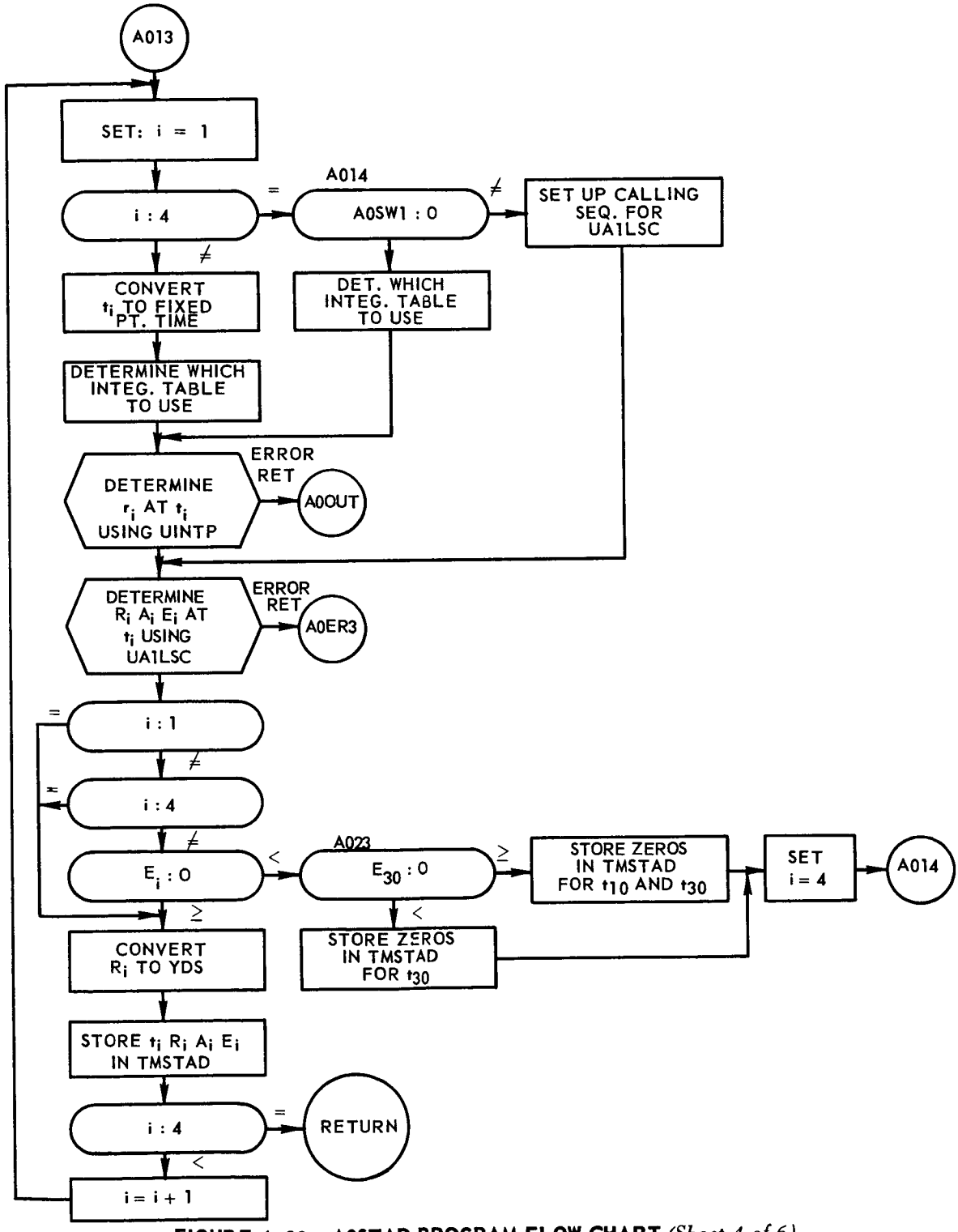


FIGURE 4-30. A0STAD PROGRAM FLOW CHART *(Sheet 4 of 6)*

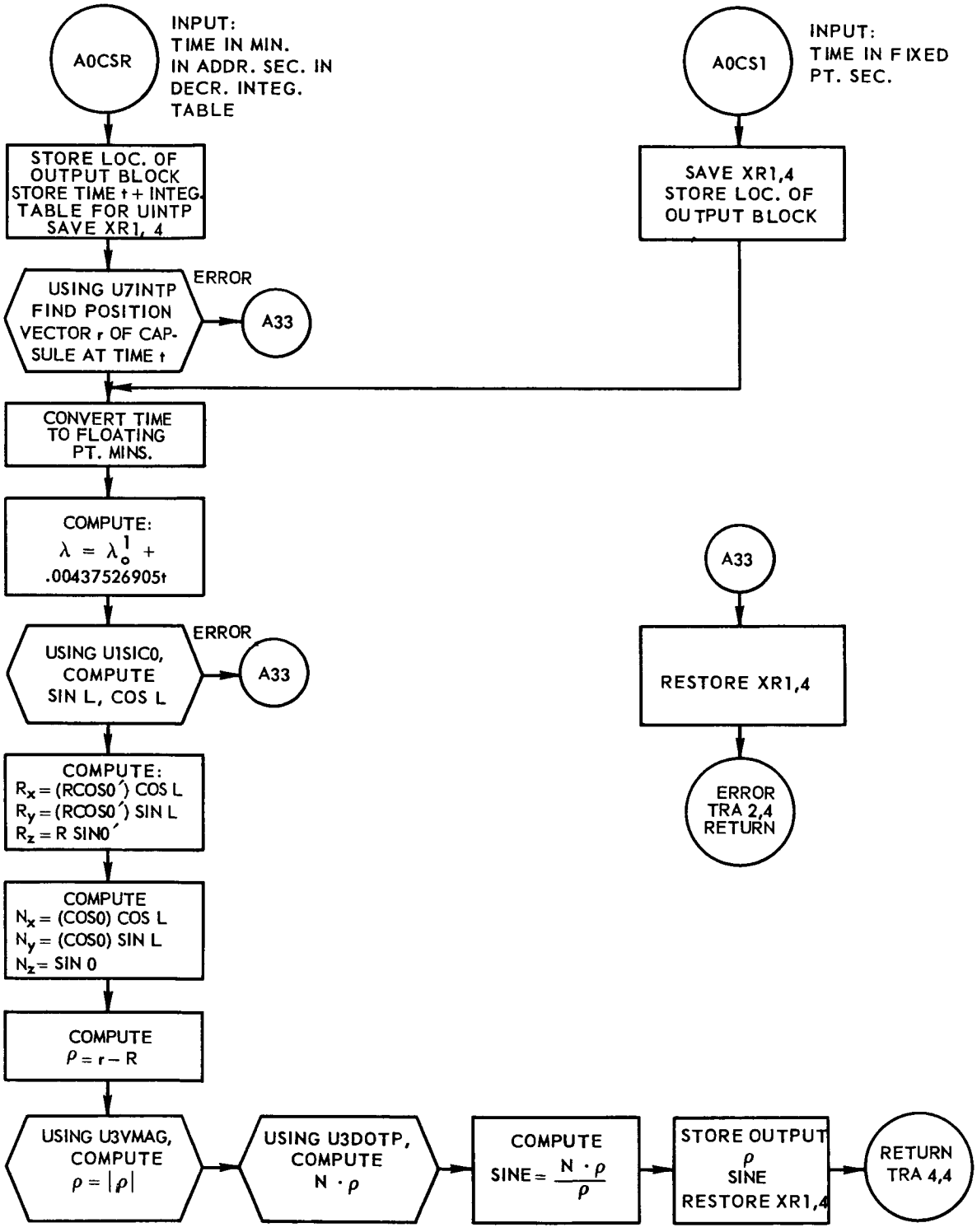


FIGURE 4-30. AOSTAD PROGRAM FLOW CHART *(Sheet 5 of 6)*

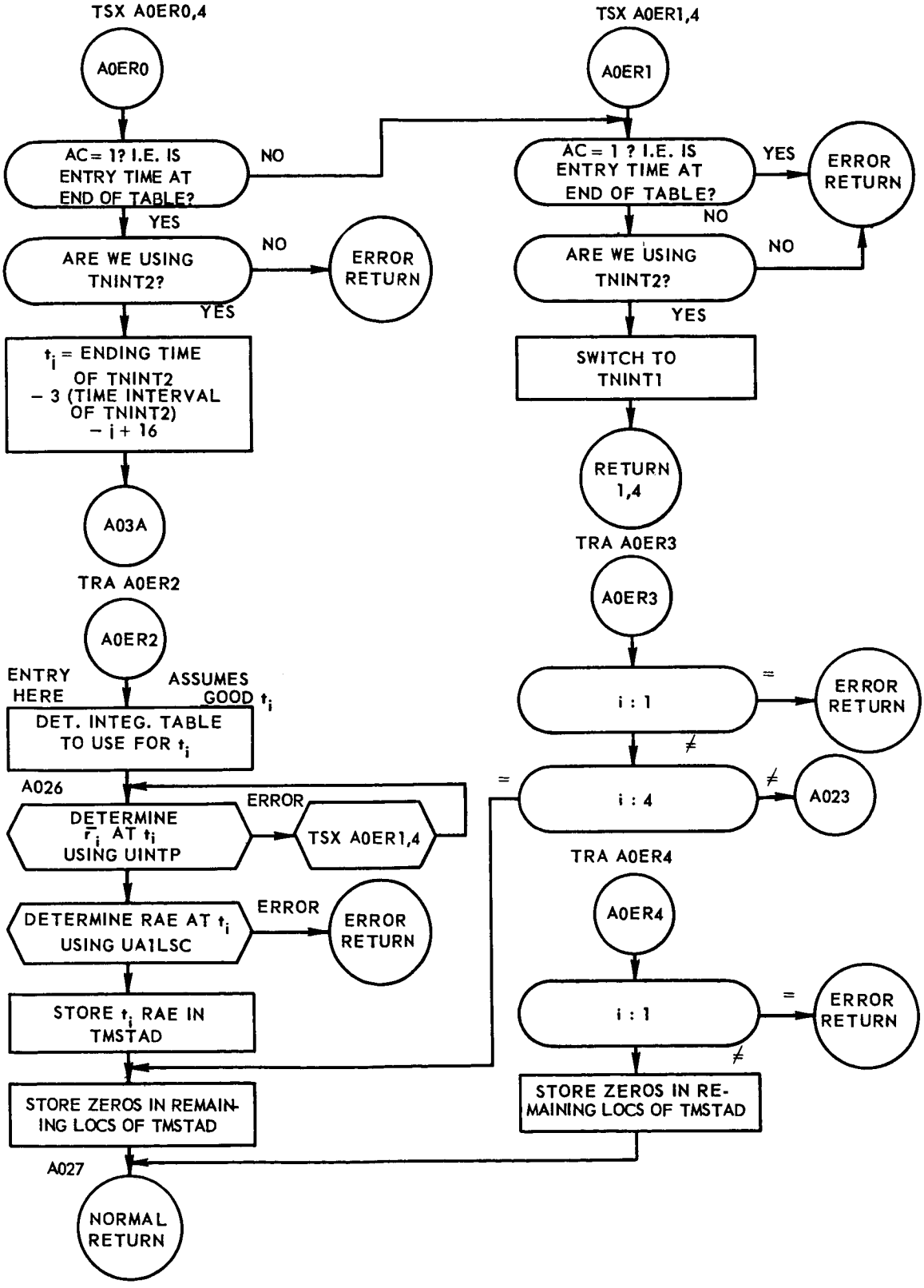


FIGURE 4-30. A0STAD PROGRAM FLOW CHART *(Sheet 6 of 6)*

4.30 CALCULATION OF HIGH SPEED OUTPUT DATA PROGRAM (O50RMC)

The purpose of O50RMC is to calculate part of the high speed output to be sent from Goddard to the Mercury Control Center displays for the orbit and reentry phases. The flow chart for O50RMC is shown in Figure 4-32.

4.30.1 Input Requirements

O50RMC uses the A3MSCP program and the following library routines:

U1SQRT,	U3DOTP
U1ATAB	U3XPRO
	U7INTP
	U3VMAG

The input requirements to the O50RMC program are:

a) The S(AC) will be plus if in the Orbit phase and minus if in the reentry phase.

b) Location MCRTMS will contain the real time (GMT) with minutes in the address and seconds in the decrement.

c) Location MCGTLO will contain the time of launch in fixed point seconds, B-35.

d) Location TMSSEC will contain the time-to-fire-retrorockets in fixed point seconds, B35.

e) Location TMORMC - 12 will contain the GMT of landing computed.

f) Location MCTOFS will contain time of retro setting in fixed point seconds, B35.

4.30.2 Output Requirements

The output of the O50RMC program is 28-word table (Table 4-9) TMORMC. In Table 4-9 time formats unless otherwise specified, are minutes in the address and seconds in the decrement. Latitude formats are $-\pi/2$ to $+\pi/2$, sign position is (+) for north and (-) for south. Longitudes are 0 to 2π. All latitudes and longitudes are in floating point radian quantities.

TABLE 4-9. TMORMC OUTPUT TABLE FORMAT (Sheet 1 of 2)

Location	Phase	Description	Format
TMORMC	O	Latitude of present position	
	R	Same	
+1	O	Longitude of present position	
	R	Same	
+2	O	Latitude of IP if retrofire in 30 seconds	
	R	Ignored	
+3	O	Longitude of IP if retrofire in 30 seconds	
	R	Ignored	
+4	O	Ground time remaining until retrofire will occur (GTRS)	
	R	Ground time remaining until landing	
+15	O	Height of capsule	Floating point, nautical miles
	R	Same	
+16	O	Apogee height	Floating point, nautical miles
	R	Flight path angle	Floating point, radians
+17	O	Present capsule setting (GMTRS)	
	R	Ground elapsed time since retrofire occurred	
+21	O	Inclination angle	Floating point, radians
	R	Same	
+23	O	Inertial velocity	Floating point, feet per second
	R	Same	
+24	O	Longitude of perigee	Floating point, radians
	R	Ignored	

TABLE 4-9. TMORMC OUTPUT TABLE FORMAT (Sheet 2 of 2)

Location	Phase	Description	Format
+25	O	Ground elapsed time since launch	Fixed point seconds B35
	R	Same	
+26	O	Eccentricity	Floating point, number
	R	Ignored	
+27	O	Predicted insertion height ($a - \overline{R}$)	Floating point, nautical miles
	R	Ignored	
+28	O	$r - \overline{R}$	Floating point, nautical miles
	R	Same	

4.30.3 Method

If the mission is in the orbit phase then the following equations are used:

R = mean earth radius in Mercury units

$$r = |r| \qquad r - \overline{R} \tag{1}$$

$$v = |v| \tag{2}$$

$$e \cos E = r\, v^2 - 1 \tag{3}$$

$$a = \frac{r}{1 - e \cos E} \qquad a - \overline{R} \tag{4}$$

$$e \sin E = \frac{\overline{r} \cdot \overline{v}}{a} \tag{5}$$

$$e = \sqrt{(e \sin E)^2 + (e \cos E)^2} \tag{6}$$

$$\text{Apogee height} = a(1 + e) - \overline{R} \tag{7}$$

$$|\overline{r} \times \overline{v}| = \sqrt{P} \tag{8}$$

$$\frac{r \times v}{|\overline{r} \times \overline{v}|} \equiv (R_x, R_y, R_z) \tag{9}$$

$$\sin i = \sqrt{R_x^{\,2} + R_y^{\,2}} \qquad \cos i = R_2 \tag{10}$$

$$\text{Inclination } i = \arctan \left(\frac{\sqrt{R_x^{\,2} + R_y^{\,2}}}{R_z} \right) \tag{11}$$

$$e > .003$$

$$E = \arctan \frac{\sin E}{\cos E} \tag{12}$$

$$M = E - e \sin E \tag{13}$$

Time from present position to next position of perigee

$$T_{pp} = \frac{2\pi - M}{a^{-\frac{3}{2}}} \tag{14}$$

T_{pp} gives r, v, which is used to compute longitude of perigee

If the mission is in re-entry phase then the following equations are used:

a) $r = |r| \quad v = |v|$

b) $r - \bar{R} \quad \bar{R}$ = means earth radius in Mercury Units

c) $\bar{r} \cdot \bar{v}$

d) γ = Flight path angle = $\arcsin \frac{\bar{r} \cdot \bar{v}}{r\,v}$

$$\left(-\frac{\pi}{2} \leq \gamma \leq \frac{\pi}{2}\right)$$

4.30.4 Usage

a) Calling Sequence

α	TSX	O50RMC, 4
$\alpha + 1$		Normal Return

A plus S(AC) at time of transfer to O50RMC indicates orbit phase.

A minus S(AC) at time of transfer to O50RMC indicates re-entry phase.

b) Space Required

294_{10} excluding subroutine and constants.

c) Error Codes

Error returns from library routines other than U7INTP and A3MSCP return to the suffix MFORMC.

An error return from U7INTP checks to see if the time given the program was prior to or later than the time of the table. If the time is before the first time used for interpolation, the program goes to MFORMC. If the time is later, the program uses the last good vector in the table to calculate output.

d) Constants Used

KECR.T	DEC	.003	Test for e critical
K001.0	DEC	1.0	
K00060	DEC	60	

e) Tables Used

TMSSEC	PZE	Present capsule setting fixed point seconds B35	
TNINT1	BSS	4000	Orbit table
TNINT2	EQU	TNINT1 + 2200	Re-entry table
TMORMC	BSS	36	Output table

f) Monitor Communication Cells Used

MCRTMS	PZE	Real time (A = min, D = sec)
MCTOFS	PZE	GMTRS, fixed point, B35
MCGTLO	PZE	Time of launch, fixed point, B35

g) Timing

Maximum excluding subroutine = 14 milliseconds

4.30.5 Checkout

Times were put into the O50RMC program as input and the desired times were computed. They were then checked against hand calculations for correctness.

Height, latitude, and longitude, obtained from A3MSCP, were tested against results from A2CSCP and compared favorably, but not exactly, due to different methods of calculations in the two programs.

Equations listed in 4.32.3, Method (orbit and re-entry) were checked using the Vanguard program EP-2 and results compared to at least 5 significant digits.

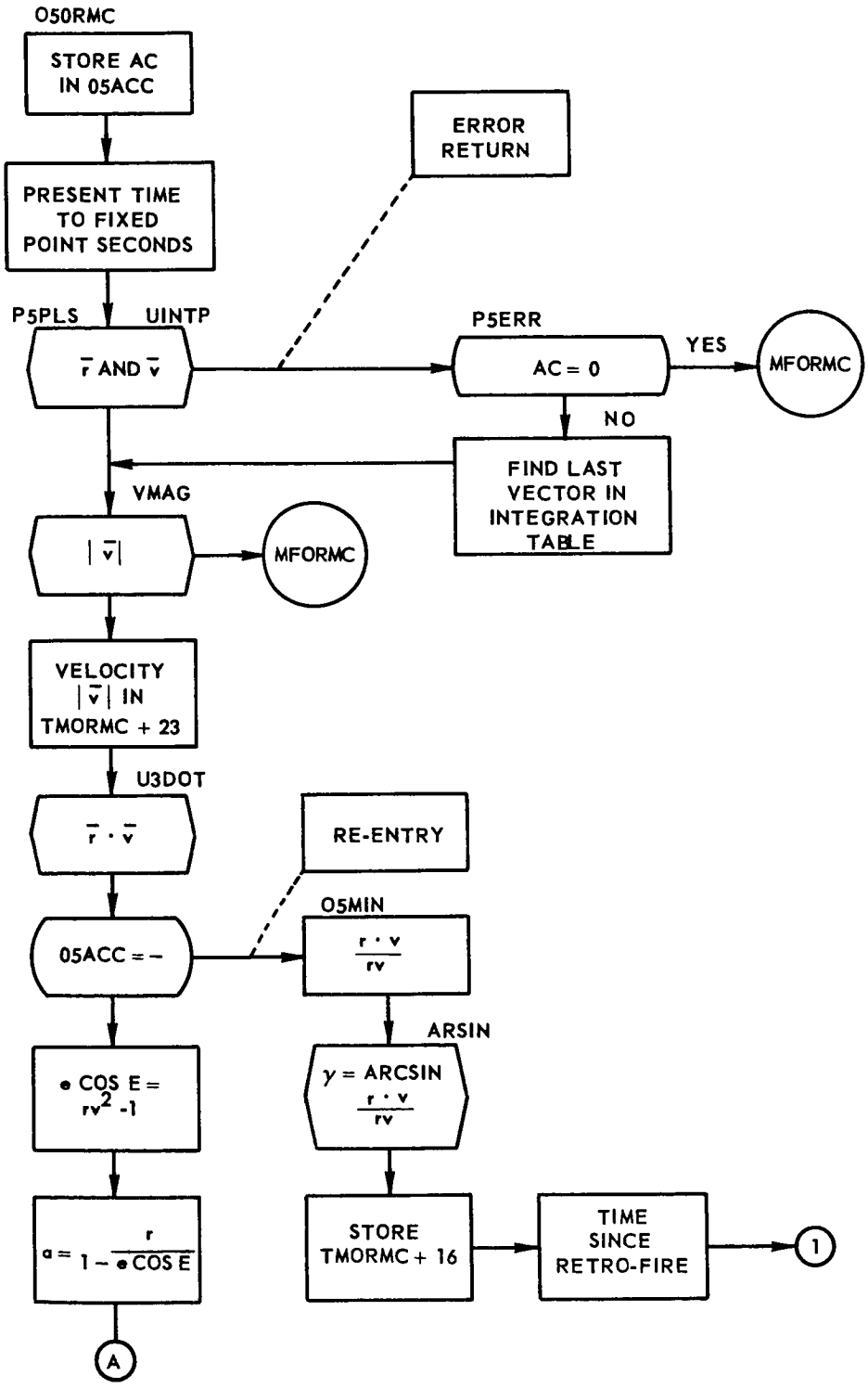


FIGURE 4-31. O50RMC PROGRAM FLOW CHART *(Sheet 1 of 3)*

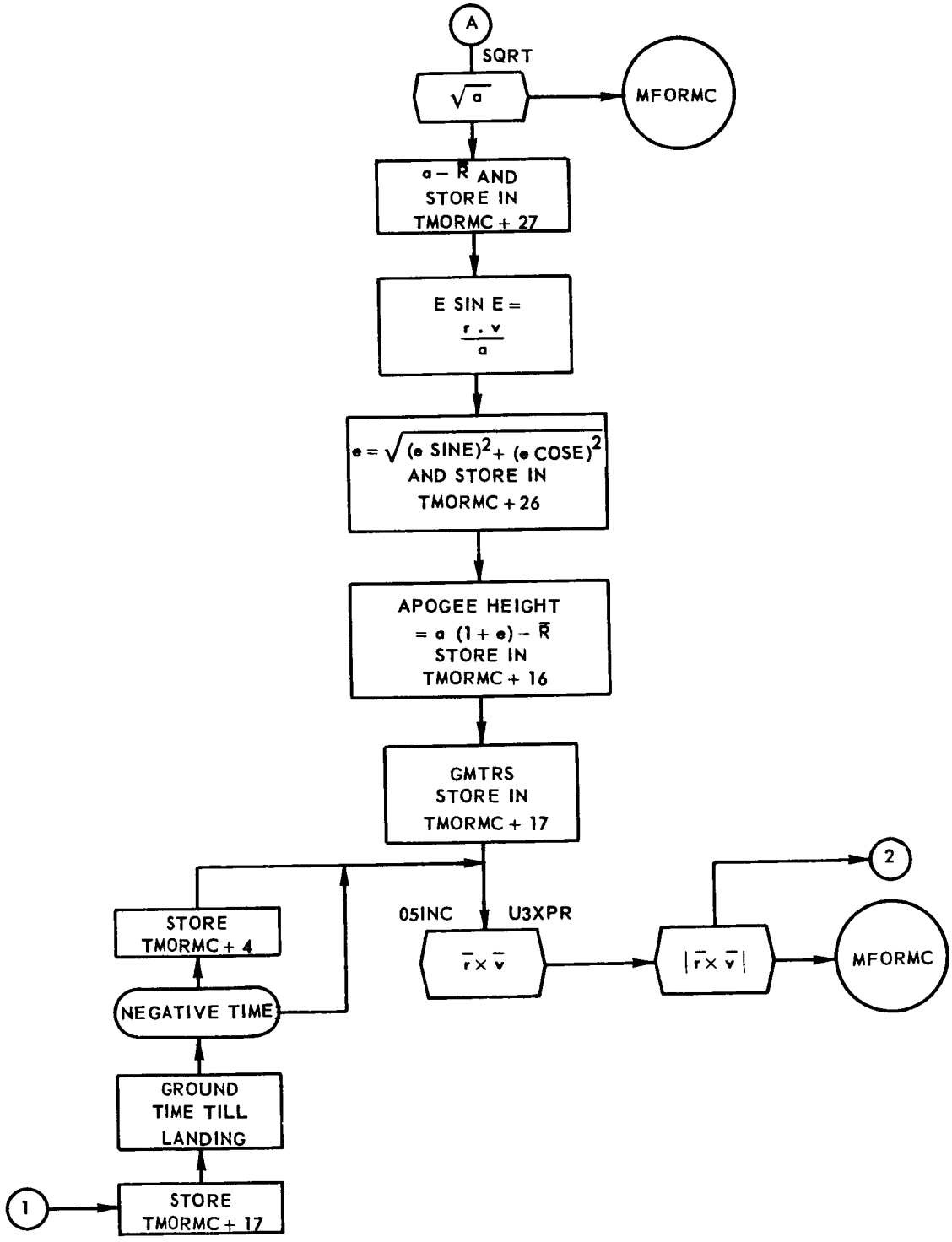


FIGURE 4-31. O50RMC PROGRAM FLOW CHART *(Sheet 2 of 3)*

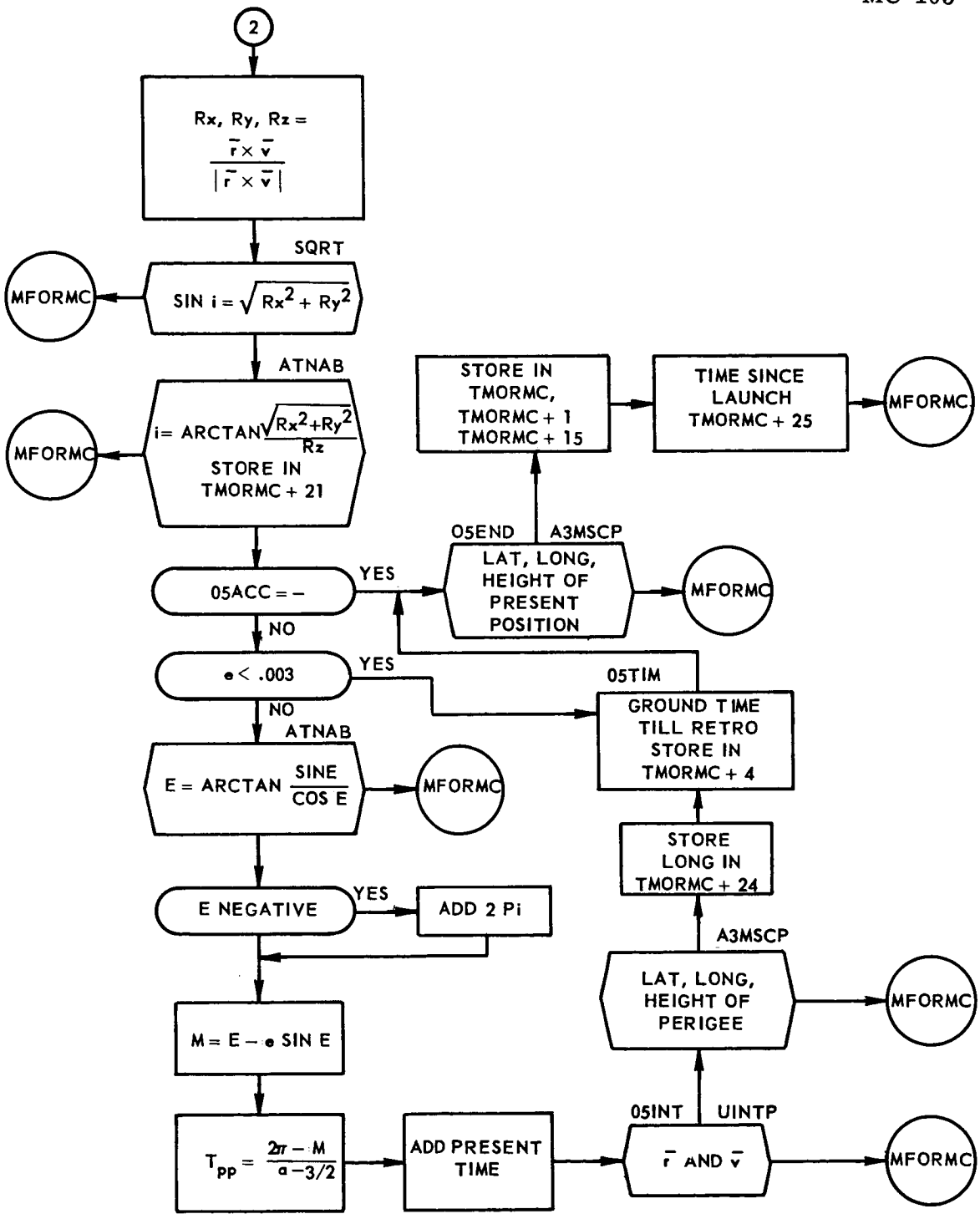


FIGURE 4-31. O50RMC PROGRAM FLOW CHART *(Sheet 3 of 3)*

Section 5
OUTPUT PROCESSORS

The Goddard output processors prepare data for transmission to the control centers, world-wide tracking sites, and to the Goddard plotboards during the launch/abort, orbit, and re-entry phases of a Project Mercury mission. These processors are divided into low and high speed outputs and are discussed in this section.

5.1 LOW-SPEED OUTPUT PROGRAM (O0LSTY)

This program takes acquisition data generated by the A0STAD program, converts it to teletype format and stores it in 7090 memory for transmission to the tracking sites as directed by Monitor. The flow chart for the O0LSTY program is shown in Figure 5-1.

5.1.1 Input Requirements

The following subroutines are used with O0LSTY.

O1PACK - Packs teletype data in Goddard memory

O1TME, O1RAD — DGHR subroutine / converts angular measure to BCD degrees, or time measure to hours, minutes and seconds.

O1BCX, O1BCF — BCD subroutine / converts fixed to floating data to BCD format.

Subroutines O1TME, O1RAD, O1BCX, and O1BCF are the same as subroutines O0TIME, O0AGLE, O0BCDX, O0BCDF, but adapted for and used only in O0LSTY.

The input to the O0LSTY program is derived from the following tables:

a) TMSTAD: A 16 location input block of the following format is provided:

Location	Contents	Description	Format
TMSTAD	Time (t_1)	Time of 1° above horizon	Minutes in address, seconds in the decrement
+1	Range	Of t_1	Floating point; yards
+2	Azimuth	Of t_1	Floating point; radians
+3	Elevation	Of t_1	Floating point; radians
+4	Time (t_{10})	Time of 10° above horizon	Minutes in address, seconds in the decrement
+5	Range	Of t_{10}	Floating point - yards
+6	Azimuth	Of t_{10}	Floating point - radians
+7	Elevation	Of t_{10}	Floating point - radians
+8	Time (t_{30})	Time of 30° above horizon	Minutes in address, seconds in the decrement
+9	Range	Of t_{30}	Floating point - yards
+10	Azimuth	Of t_{30}	Floating point - radians
+11	Elevation	Of t_{30}	Floating point - radians
+12	Time (t_x)	Time of closest approach + Δt(Δt is yet to be defined).	Minutes in address, seconds in the decrement
+13	Range	Of t_x	Floating point - yards
+14	Azimuth	Of t_x	Floating point - radians
+15	Elevation	Of t_x	Floating point - radians

b) MCHFSC: Contains GMT of data to be sent in half seconds (fixed point).

c) MNSCNO: Equal total number of entries in Tables O1TBA or O1TBB (fixed point).

d) MCACQ1: A(MCACQ1) = Location of output (fixed point)
D(MCACQ1) = Integer which is the internal station number (fixed point).

The two columns (O1TBA and O1TBB) shown in Table 5-1 specify the site to which the acquisition data is to be sent. Column O1TBA contains the teletype address switching codes for the station and Column O1TBB contains the three letter code that identifies the designated radar site. The entries in each column are stored in descending order according to the internal station number. When there are two radars at a site, there are two internal station numbers for that site.

The selection of the proper codes from these two columns is controlled by the contents of the decrement of MCACQ1 which contains the internal station number of the site, e.g., CLA O1TBA + MNSCNO, XR where, O1TBA is the first word in table, MNSCNO contains total number of stations, XR contains the decrement of MCACQ1.

5.1.2 Output Requirements

The output of O0LSTY is packed in the output block indicated in the control word MCACQ1. The output consists of 31 words. Words 1-5 consist of: (1) A three letter code indicating to which station the acquisition data should be sent; (2) the GMT that the message was processed; (3) the information needed to position the message on the page. Words 6-26 contain the acquisition data converted to teletype format. Time is in GMT hours, minutes, and seconds. Range is in thousandths of yards. Both azimuth and elevation are in degrees. Words 27-31 contain (1) the three letter station number code; (2) the GMT that the message was processed; (3) an end of message indication which is eight line feeds in succession. If the program tries to convert a floating point number greater than 234000000000, $(238)_{10}$ in the BCD subroutine, question marks are printed out in the output location of this number. This is also true if the program tries to convert a fixed point number larger than eleven characters long, using either the DGHR or the BCD subroutines. Neither case interrupts the normal flow of the program.

5.1.3 Usage

a) Calling Sequence

α TSX	O0LSTY, 4
$\alpha + 1$	Normal Return

b) Space Required

Instructions	196
Constants	46
Common	25
Tables	102
Subroutines	266
TOTAL	635

TABLE 5–1. STATION IDENTIFICATION TABLES

Internal Station Number	Station	01TBA	01TBB
36	Grand Bahama	YC↓YC↓	Z<≡GBI
35	Cape Canaveral	YC↓YC↓	Z<≡CNV
34	Grand Turk	YC↓YC↓	Z<≡GTI
33	Corpus Christi	YX↓YX↓	Z<≡TEX
32	Guaymas, Mexico	YF↓YF↓	Z<≡GYM
31	Point Arguello	YQ↓YQ↓	Z<≡CAL
30	Hawaii	YD↓YD↓	Z<≡HAW
29	Canton	YL↓YL↓	Z<≡CTN
28	Woomera	YW↓YW↓	Z<≡WOM
27	Muchea	YU↓YU↓	Z<≡MUC
26	Indian Ocean Ship	YI↓YI↓	Z<≡IOS
25	Zanzibar	YZ↓YZ↓	Z<≡ZZB
24	Keno	YK↓YK↓	Z<≡KNO
23	Grand Canary	YN↓YN↓	Z<≡CYI
22	Mid Atlantic Ship	YA↓YA↓	Z<≡ATS
21	Bermuda	YB↓YB↓	Z<≡BDA
20	San Salvador	YC↓YC↓	Z<≡SSI
19	Bermuda	YB↓YB↓	Z<≡BDA
18	Bermuda	YB↓YB↓	Z<≡BDA
17	Eglin	YR↓YR↓	Z<≡EGL
16	Eglin	YR↓YR↓	Z<≡EGL
15	Corpus Christi	YX↓YX↓	Z<≡TEX
14	White Sands	YE↓YE↓	Z<≡WHS
13	Guaymas, Mexico	YF↓YF↓	Z<≡GYM
12	Point Arguello	YQ↓YQ↓	Z<≡CAL
11	Point Arguello	YQ↓YQ↓	Z<≡CAL
10	Hawaii	YD↓YD↓	Z<≡HAW
9	Hawaii	YD↓YD↓	Z<≡HAW
8	Woomora	YW↓YW↓	Z<≡WOM
7	Muchea	YU↓YU↓	Z<≡MUC
6	Grand Canary	YN↓YN↓	Z<≡CYI
5	Bermuda	YB↓YB↓	Z<≡BDA
4	Bermuda	YB↓YB↓	Z<≡BDA
3	San Salvador	YC↓YC↓	Z<≡SSI
2	Grand Bahama	YC↓YC↓	Z<≡GBI
1	Cape Canaveral	YC↓YC↓	Z<≡CNV

↓ Refers to letters shift on teletype machine.
< Refers to carriage return.
≡ Refers to line feed.

c) Constants Used

K00000	DEC	0
K00001	DEC	1
K00003	DEC	3
K00006	DEC	6
K00010	DEC	10
K00013	DEC	13
K00015	DEC	15
K00017	DEC	17
K00018	DEC	18
K00028	DEC	28
K00032	DEC	32
K00034	DEC	34
K00037	DEC	37
K00040	DEC	40
K00046	DEC	46
K00053	DEC	53
K00060	DEC	60
K00163	DEC	163
K00165	DEC	165
K00167	DEC	167
K00168	DEC	168
K15360	DEC	15360
K01.E3	DEC	1000.0
K0.MPR	DEC	3437.7467707849
K.1RAD	DEC	57.2957791
KCH233	OCT	233000000000
KCH234	OCT	234000000000

d) Timing

Timing required for O0LSTY is 402.794 msecs

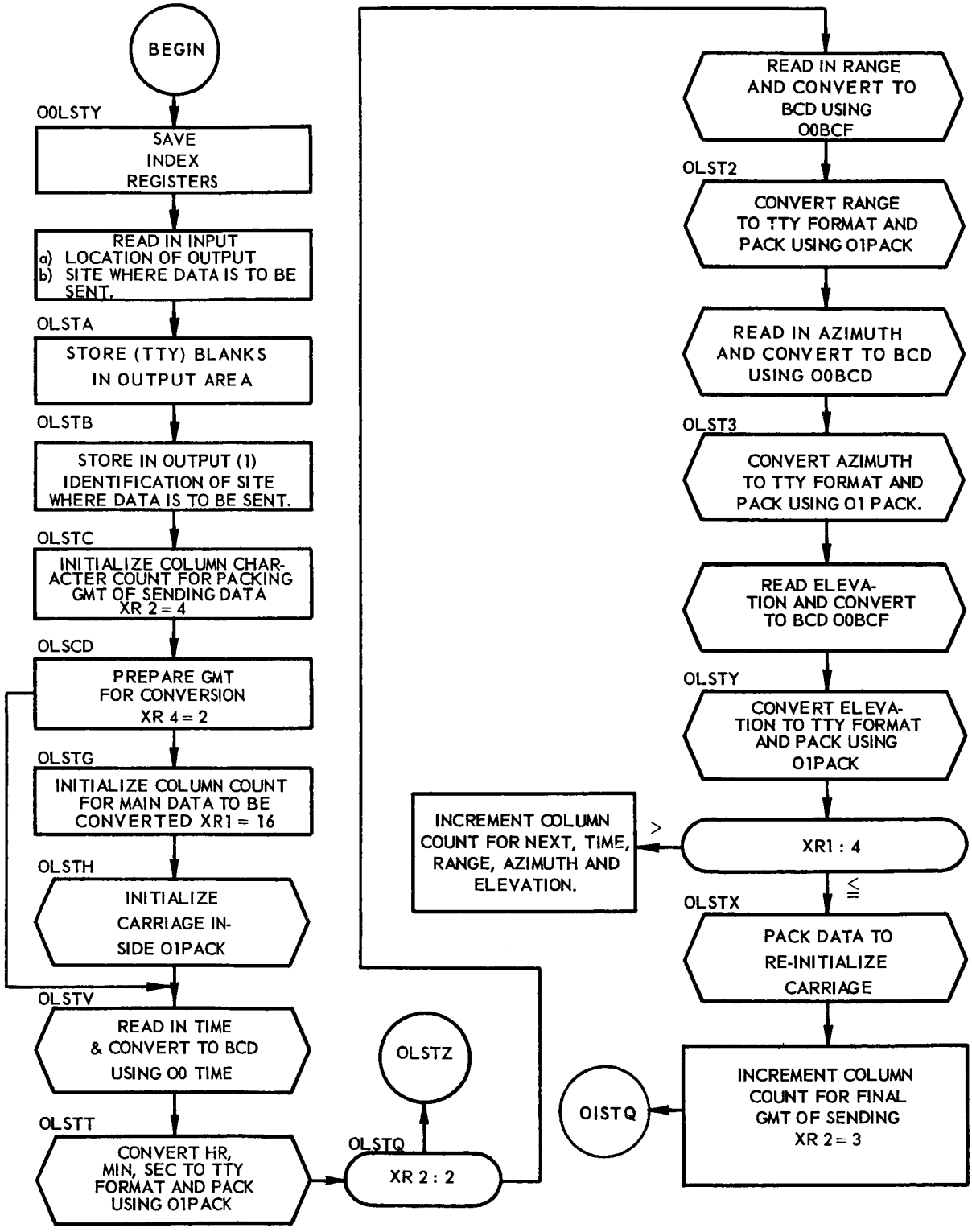


FIGURE 5-1. 00LSTY PROGRAM FLOW CHART *(Sheet 1 of 2)*

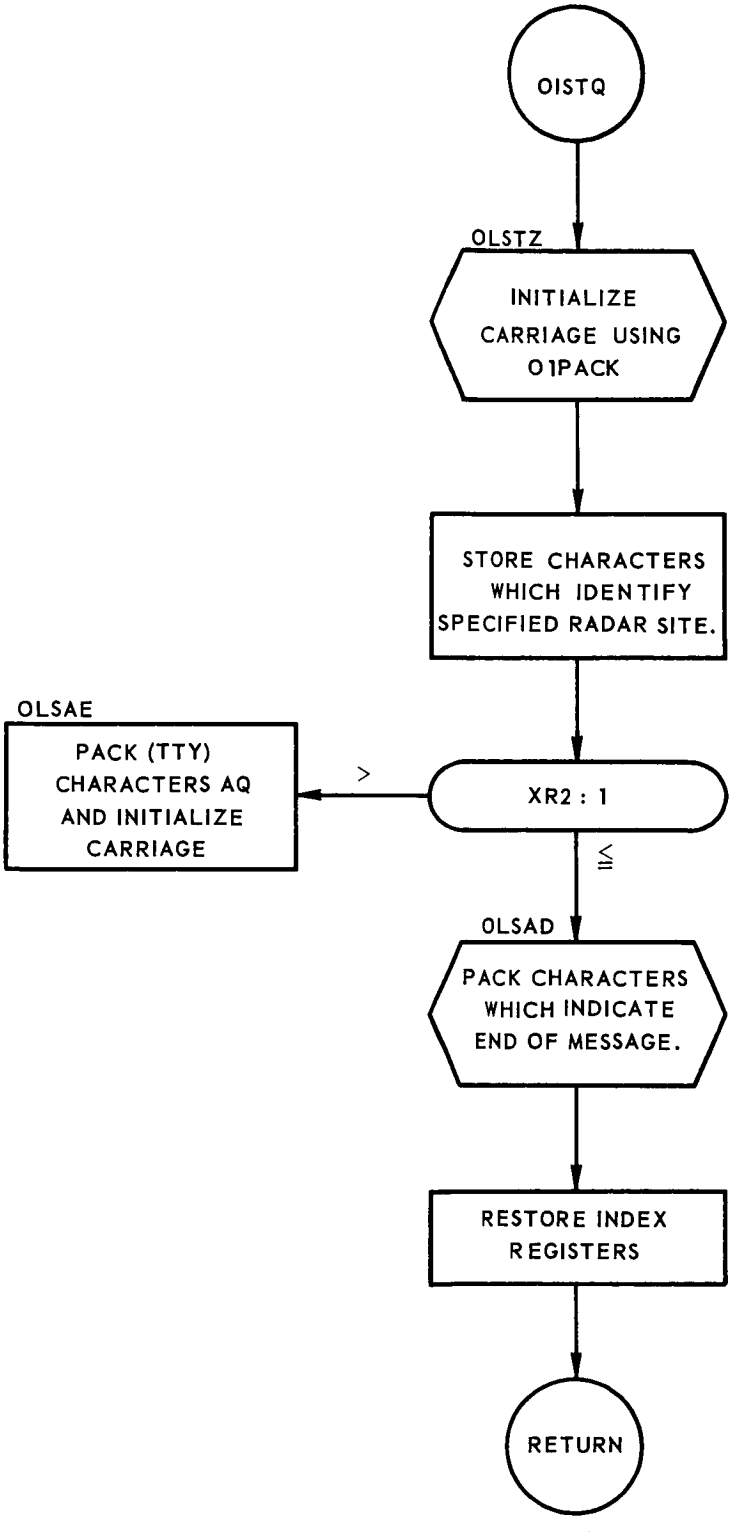


FIGURE 5-1. OOLSTY PROGRAM FLOW CHART *(Sheet 2 of 2)*

5.2 HIGH-SPEED OUTPUT PROGRAMS (O0LANA and O0ORRE)

These programs convert a block of data to the proper output format and ready it for transmission to the Mercury Control Center. O0LANA processes the output data during the launch/abort phase while O0ORRE processes it during the orbit and re-entry phases. The following subroutines are used with the high-speed output programs:

BCD - converts a fixed or floating point number to its BCD equivalent

PACK - packs N bits into a block of sequential words (32 bits/word)

DGHR - converts angular or time measure to BCD degrees (or hours), minutes, and seconds.

MBCD - converts up to 6 BCD characters to modified BCD characters.

PLOT - data preparation for Goddard plotboards

The detailed write ups for the above routines are included in Sections 5.3 to 5.7. The flow charts for O0LANA and O0ORRE are shown in Figure 5-2 and 5-3.

5.2.1 Input Requirements

The input to O0LANA consists of a 30 location table called TMLANA (see Table 5-2). Similarly, the input for O0ORRE is TMORRE (see Table 5-3). The word Time in the FORMAT column of the tables means fixed point minutes in the address and fixed point seconds in the decrement. Also, the range on all latitudes is $-\pi/2$ to $+\pi/2$ and the range on all longitudes is 0 to 2π.

In addition to the input blocks mentioned above, Monitor must provide a proper sign in the accumulator upon each entrance to these programs. This sign is used to indicate phase of operation. For O0LANA, a positive accumulator means normal launch while a negative sign means abort. For O0ORRE, a plus indicates orbit and a minus sign indicates re-entry.

5.2.2 Output Requirements

The output of both O0LANA and O0ORRE is a 13-location table TMOLAB, containing coded information which is transmitted to the following displays at the Mercury Control Center.

a) Wall maps

b) Strip chart

c) Plotboards

TABLE 5-2. TMLANA TABLE FORMAT

Location	Quantity	Phase Needed	Where Displayed	Format
TMLANA	$\gamma - \gamma$ nom (B–GE)	Launch	Strip Chart	Fl.Pt.–Radians
+1	$\gamma - \gamma$ nom (IP7090)	Launch	Strip Chart	Fl.Pt.–Radians
+2	V/Vr – V/Vr nom (B–GE)	Launch	Strip Chart	Fl.Pt.–Radians
+3	V/Vr – V/Vr nom (IP7090)	Launch	Strip Chart	Fl.Pt.–Radians
+4	Data Source Indicator (1 = GE–GOD, 2 = RAW, 3 = IP7090)	Launch/Abort	–	Fx.Pt.–B15
+5	ϕ_{PP}	Launch/Abort	Wall Map & Plotboard 4	Fl.Pt.–Radians
+6	λ_{PP}	Launch/Abort	Wall Map & Plotboard 4	Fl.Pt.–Radians
+7	ϕ Tower Impact Point – Before tower separation	Launch	Plotboard 4	Fl.Pt.–Radians
	ϕ 30 sec – After tower separation	Launch	Plotboard 4	Fl.Pt.–Radians
+8	λ Tower Impact Point – Before tower separation	Launch	Plotboard 4	Fl.Pt.–Radians
	λ 30 sec – After tower separation	Launch	Plotboard 4	Fl.Pt.–Radians
+9	GTRS – Time Remaining Until Retrofire	Abort	Wall digital display	Time
+10	GMTLC – GMT of Landing Computed	Abort	Recovery Status Monitor	Time
+11	Zero – Before tower separation	Launch	Plotboard 4	–
	ϕ Max – After tower separation	Launch	Plotboard 4	Fl.Pt.–Radians
	ϕ Refined Impact Point	Abort	Recovery Status Monitor, Wall Map & Plotboard 4	Fl.Pt.–Radians
+12	$(3\pi/2)$ – Before tower separation	Launch	Plotboard 4	–
	λ Max – After tower separation	Launch	Plotboard 4	Fl.Pt.–Radians
	λ Refined Impact Point	Abort	Recovery Status Monitor, Wall Map & Plotboard 4	Fl.Pt.–Radians
+13	Emergency Abort GMTRC	Launch/Abort	Retrofire Console	Time
+14	Zero – Before Separation	Launch	Retrofire Console	–
	Δ T – After separation	Launch	Retrofire Console	Time
+15	Emergency Abort ECTRC	Abort	Retrofire Console	Time

TABLE 5-2. TMLANA TABLE FORMAT (cont'd)

Location	Quantity	Phase Needed	Where Displayed	Format
+16	Zero – Before Seco + 5 sec.	Launch	Flight Dynamics Officer	–
	Negative of Orbit Capability – After Seco + 5 sec.	Launch	Flight Dynamics Officer	Fx.Pt.–B35
+17	Recovery Area	Launch/Abort	Retrofire Console	Special Code
+18	Abort Recommendation (1 = GO, 2 = NO GO)	Launch/Abort	Flight Dynamics Officer	Fx.Pt.–B35
+19	h – Height of Capsule	Launch/Abort	Plotboard 2	Fl.Pt.–Nautical Miles
+20	γ – Flight Path Angle	Launch/Abort	Flight Dynamics Officer, & Plotboard 1	Fl.Pt.–Radians
+21	i – Inclination Angle	Launch	Flight Dynamics Officer	Fl.Pt.–Radians
+22	V/V_r – Velocity Ratio	Launch	Flight Dynamics Officer & Plotboard 1	Fl.Pt.
	V – Inertial Velocity	Abort	Flight Dynamics Officer & Plotboard 1	Fl.Pt.–Ft/Sec
+23	$(Y - Y_{nom})$ – Cross Range Deviation	Launch	Plotboard 2	Fl.Pt.–Nautical Miles
+24	d – Down Range Distance	Launch/Abort	Plotboard 2	Fl.Pt.–Nautical Miles
+25	ICTRC ±Hrs	Abort	Retrofire Console	Fx.Pt.–B35
+26	ICTRC ±Mins	Abort	Retrofire Console	Fx.Pt.–B35
+27	ICTRC ±Secs	Abort	Retrofire Console	Fx.Pt.–B35
+28	$r - \bar{R}$	Launch/Abort	Flight Dynamics Officer & Plotboard 1	Fl.Pt.–Nautical Miles
+29	Data Source (1 = AN/FPS 16, 0 = AZUSA)	Launch	–	Fx.Pt.–B17

TABLE 5-3. TMORRE TABLE FORMAT

Location	Quantity	Phase Needed	Where Displayed	Format
TMORRE	ϕ_{PP}	Orbit/Re-entry	Wall Map & Plotboard 4	Fl. Pt. – Radians
+1	λ_{PP}	Orbit/Re-entry	Wall Map & Plotboard 4	Fl. Pt. – Radians
+2	ϕ 30 sec	Orbit	Wall Map & Plotboard 4	Fl. Pt. – Radians
	ϕ Refined Impact Point	Re-entry	Wall Map & Plotboard 4	Fl. Pt. – Radians
+3	λ 30 sec	Orbit	Wall Map & Plotboard 4	Fl. Pt. – Radians
	λ Refined Impact Point	Re-entry	Wall Map & Plotboard 4	Fl. Pt. – Radians
+4	GTRS – Ground Time Remaining Until Retrofire	Orbit	Wall digital display	Time
	Ground Time Remaining Until Landing	Re-entry	Wall digital display	Time
+5	Emergency Abort GMTRC	Orbit	Retrofire Console	Time
+6	Emergency Abort ECTRC	Orbit	Retrofire Console	Time
+7	End of Present Orbit GMTRC	Orbit	Retrofire Console	Time
+8	End of Present Orbit ECTRC	Orbit	Retrofire Console	Time
+9	End of Mission GMTRC	Orbit	Retrofire Console	Time
+10	End of Mission ECTRC	Orbit	Retrofire Console	Time
+11	Orbit Number	Orbit/Re-entry	Wall digital display	Fx. Pt. – B35
+12	GMTLC – GMT of Landing Computed	Orbit/Re-entry	Recovery Status Monitor	Time
+13	ϕ L.P. – Latitude of Mission End Landing Point	Orbit/Re-entry	Recovery Status Monitor	Fl. Pt. – Radians
+14	λ L.P. – Longitude of Mission End Landing Point	Orbit/Re-entry	Recovery Status Monitor	Fl. Pt. – Radians
+15	h – Height of Capsule	Orbit/Re-entry	Plotboard 2	Fl. Pt. – Nautical Miles
+16	h_a – Apogee height	Orbit	Flight Dynamics Officer	Fl. Pt. – Nautical Miles
	γ – Flight Path Angle	Re-entry	Flight Dynamics Officer	Fl. Pt. – Radians
+17	GMTRS – GMT of Retrofire	Orbit	Retrofire Console	Time
	Elapsed Ground Time since Retrofire	Re-entry	Retrofire Console	Time

TABLE 5-3. TMORRE TABLE FORMAT (cont'd)

Location	Quantity	Phase Needed	Where Displayed	Format
+18	ICTRC ±Hours	Orbit	Retrofire Console	Fx. Pt. – B35
+19	ICTRC ±Minutes	Orbit	Retrofire Console	Fx. Pt. – B35
+20	ICTRC ±Seconds	Orbit	Retrofire Console	Fx. Pt. – B35
+21	i – Inclination Angle	Orbit	Flight Dynamics Officer	Fl. Pt. – Radians
+22	Orbit Capability	Orbit	Flight Dynamics Officer	Fx. Pt. – B35
+23	V – Inertial Velocity	Orbit/Re-entry	Flight Dynamics Officer & Plotboard 1	Fl. Pt. – Ft/Sec
+24	λ_p – Longitude of Perigee	Orbit	Plotboard 3	Fl. Pt. – Radians
+25	te – Elapsed Time since Launch (seconds)	Orbit/Re-entry	Plotboard 2 & Plotboard 3	Fx. Pt. – B35
+26	ϵ – Eccentricity	Orbit	Plotboard 3	Fl. Pt.
+27	$a - \bar{R}$ – Predicted Insertion Height	Orbit	Plotboard 2	Fl. Pt. – Nautical Miles
+28	$r - \bar{R}$	Orbit/Re-entry	Flight Dynamics Officer & Plotboard 1	Fl. Pt. – Nautical Miles
+29	Recovery Area	Orbit/Re-entry	Retrofire Console	Special Code

d) Wall digital displays

e) Console digital displays

The coded information contained in TMOLAB consists of a total of 408 bits that indicate the values of various quantities to be displayed. Figure 5-4 (Sheets 1 through 5) illustrates each of these quantities in their proper order of transmission. The actual quantities to be displayed is determined by phase of operation and frame. There are two frames of information for each phase—odd and even—which are transmitted alternately.

During the launch phase a frame is sent to the displays every .5 seconds. During orbit a frame is sent every 6 seconds and during the abort and re-entry phases a frame is sent every 3 seconds.

Generally, the bits in TMOLAB are coded as follows:

a) For the strip chart, wall map, and plotboards—Each value is scaled to a 10 bit quantity and zero offset according to the particular ranges of each of these displays. If any of these ranges is exceeded, then $(1777)_8$ or 0 is packed into TMOLAB in place of the offending value accordingly as the maximum or minimum, respectively, is violated.

b) For the wall and console digital displays—Each value is converted to a modified BCD code. That is, a BCD code which reduces to a minimum the number of bits necessary to represent any one digit position for each particular quantity. No check for values which may exceed the maximum or minimum of the displays is made. If such a value does occur, it is incorrectly coded and packed. Furthermore, the first few bits immediately following this quantity in TMOLAB may also be disturbed.

Table 5-4 contains the ranges and scale factors for the strip chart, wall map, and plotboards. In addition, Table 5-5 contains the number of bits required for each of the wall and console digital displays.

On the first pass through O0LANA a time mark (bit #401) is sent out to the strip chart and plotboards. Thereafter, a time mark is put out every 10 seconds. If for some reason O0LANA is not entered at that time when a time mark is to be put out it will be omitted altogether. Similarly, O0ORRE puts out a time mark every minute during orbit and every 10 seconds during re-entry.

The hold phase is indicated to O0LANA by means of a negative sign in the TMLANA word containing orbit capability. At this time, O0LANA sends all zeros to the four plotboards located at Cape Canaveral. In addition, each of the digital displays on the Flight Dynamics Officer's Console are frozen with the exception of the Orbit Capability and GO, NO-GO displays.

	LAUNCH		ABORT		ORBIT		RE-ENTRY		
	ODD	EVEN	ODD	EVEN	ODD	EVEN	ODD	EVEN	
	$\gamma-\gamma_{NOM}$ (B-GE) 10		0	0	0	0	0	0	STRIP CHART
	$\gamma-\gamma_{NOM}$ (IP 7090) 10		0	0	0	0	0	0	
	V/V_r- V/V_{2nom} (B-GE) 10		0	0	0	0	0	0	
	V/V_r- V/V_{2nom} (IP 7090) 10		0	0	0	0	0	0	
42 BITS	D.S.	D.S.	D.S.	D.S.	0 1	0 1	0 1	0 1	
	ϕ_{PP} 10								WALL MAP
	λ_{PP} 10								
	ϕ_{PP} 10		ϕ REFINED I. P.		ϕ R. F. IN 30 SEC		ϕ REFINED I. P.		
82 BITS	λ_{PP} 10		λ REFINED I. P.		λ R.F. IN 30 SEC		λ REFINED I. P.		

FIGURE 5-4. DISPLAY FORMAT OF TMOLAB CONTENTS *(Sheet 1 of 5)*

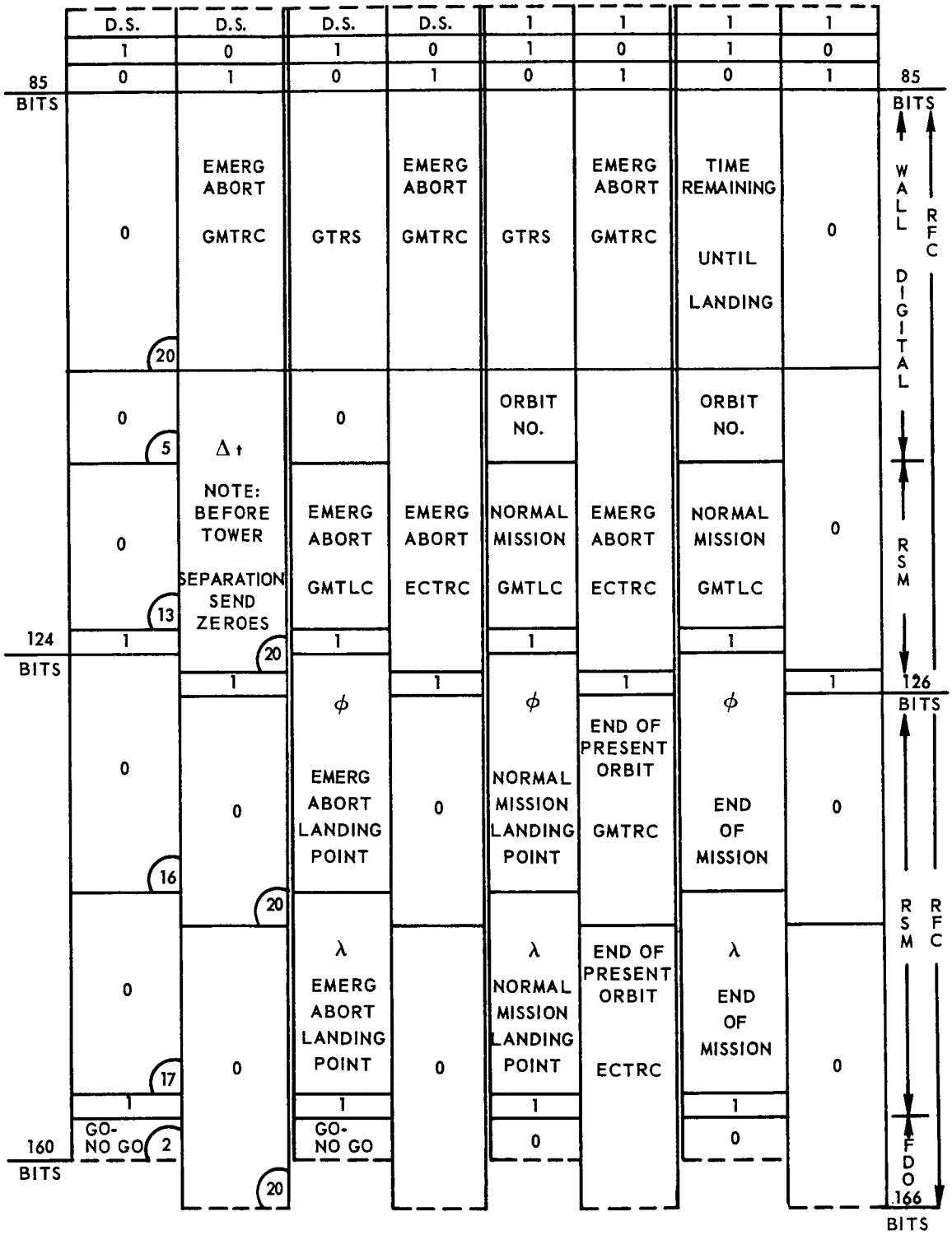


FIGURE 5-4 DISPLAY FORMAT OF TMOLAB CONTENTS *(Sheet 2 of 5)*

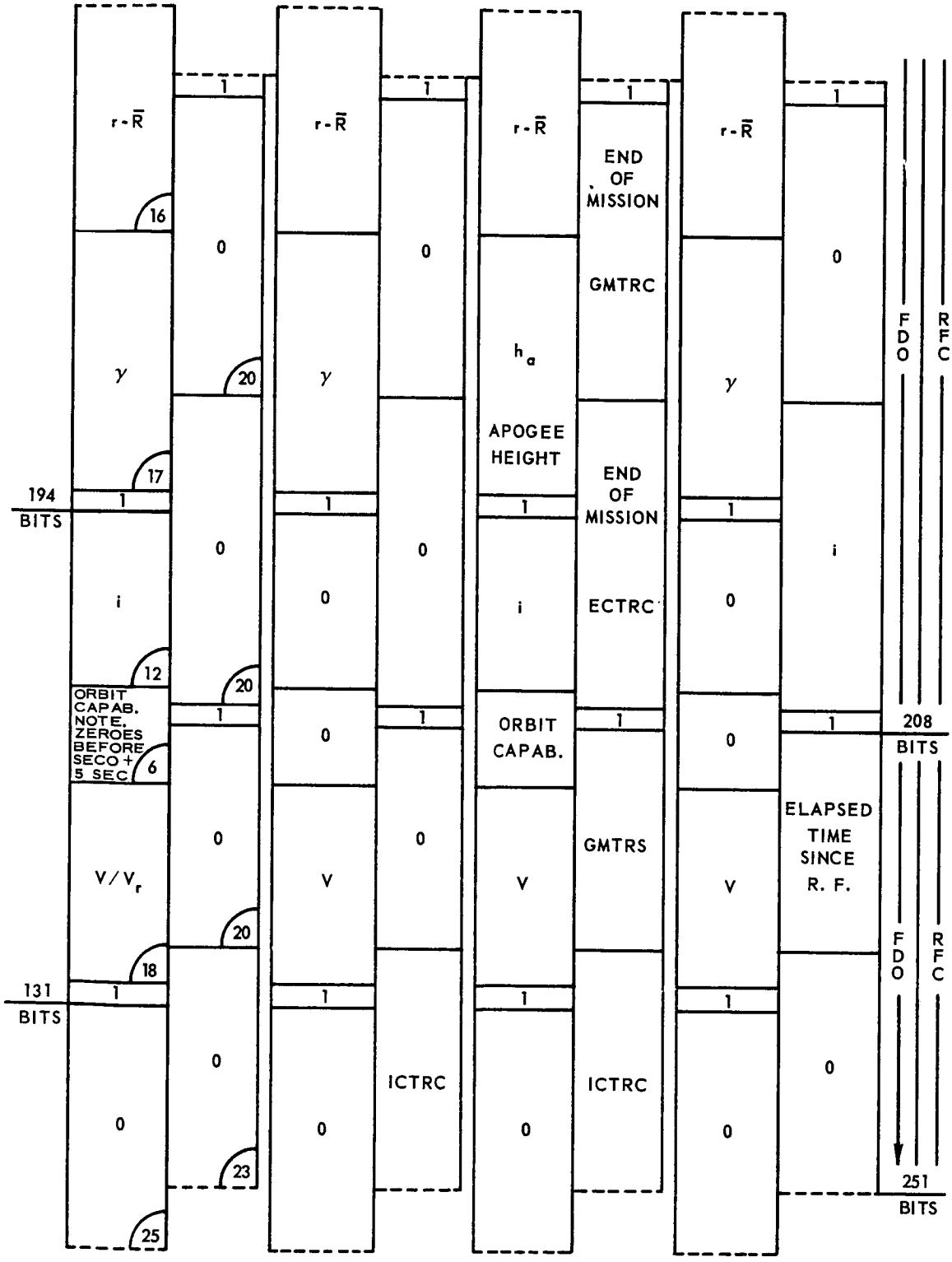


FIGURE 5-4. DISPLAY FORMAT OF TMOLAB CONTENTS *(Sheet 3 of 5)*

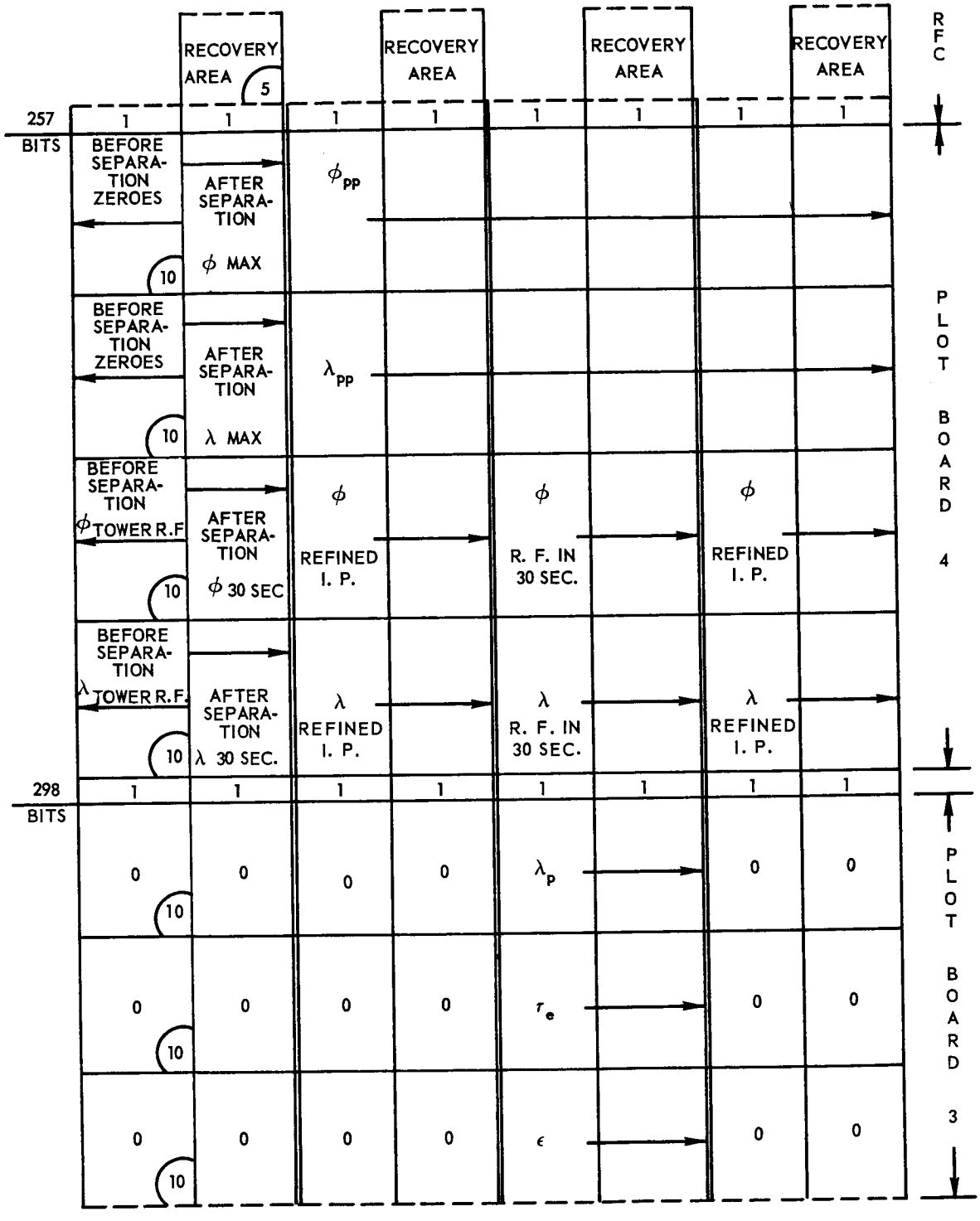


FIGURE 5-4. DISPLAY FORMAT OF TMOLAB CONTENTS *(Sheet 4 of 5)*

	$Y-Y_{NOM}$ (10)	→	0	0	$\alpha-\bar{R}$	→	0	0	PLOT BOARD 2
339 BITS	1	1	1	1	1	1	1	1	
	d (10)	→	→	→	T_e	→	→	→	
	h (10)	→	→	→	→	→	→	→	
	γ (10)	→	$r-\bar{R}$	→	→	→	→	→	PLOT BOARD 1
	V/V_r (10)	→	V	→	→	→	→	→	
380 BITS	1	1	1	1	1	1	1	1	
	0 (10)	0	0	0	0	0	0	0	
	0 (10)	0	0	0	0	0	0	0	
	T. M.	T. M.	T. M.	T. M.	T. M.	T. M.	T. M.	T. M.	
408 BITS	0 (10)	0	0	0	0	0	0	0	

FIGURE 5-4. DISPLAY FORMAT OF TMOLAB CONTENTS *(Sheet 5 of 5)*

TABLE 5-4. CHARACTERISTICS OF DISPLAY MEDIA

Display	Phase	Quantity	Range (Of Display)	Range in Degrees	Scale Factor
STRIP CHART	Launch	$\gamma - \gamma_{nom}$	$-.087266463^R$ to $+.087266463^R$	–5 to +5	5861.35824
	Launch	$V/V_r - V/V_{r\,nom}$	–.04 to + .04	–	12787.5
WALL MAP	All Phases	ϕ	$-.698131701^R$ to $+.698131701^R$	–40 to +40	732.669781
	All Phases	λ	-3.14159265^R to $+3.14159265^R$	–180 to +180	162.815507
PLOT BOARD 4	All Phases	ϕ	$+.209439510^R$ to $+.628318531^R$	+12 to +36	2442.23260
	Launch/Abort	λ	-1.43116999^R to $+.139626340^R$	–82 to +8	651.262027
	Orbit/Re-entry*		-2.35619449^R to $-.785398163^R$	–135 to –45	651.262027
PLOT BOARD 3	Orbit	λ_p	-3.14159265^R to $+9.42477796^R$	–180 to +540	81.4077534
	Orbit	ϵ	–.144 to + .144	–	3552.08333
	Orbit	t_e	0^s to $18{,}000^s$ (5^h)	–	.056833333
PLOT BOARD 2	Launch/Abort	h	0^{nm} to 60^{nm} ($d \leq 60$)	–	17.05
			0^{nm} to 150^{nm} ($d > 60$)	–	6.82
	Launch/Abort	d	0^{nm} to 60^{nm} ($d \leq 60$)	–	17.05
			0^{nm} to 600^{nm} ($d > 60$)	–	1.705
	Launch	$\gamma - \gamma_{nom}$	-130^{nm} to $+20^{nm}$	–	6.82
	Orbit/Re-entry	h	0^{nm} to 600^{nm}	–	1.705
	Orbit/Re-entry	t_e	0^{nm} to $18{,}000^s$ (5^h)	–	.056833333
	Orbit	$a - \bar{R}$	-400^{nm} to $+200^{nm}$	–	1.705

*See 5.2.3, Other Requirements.

TABLE 5-4. CHARACTERISTICS OF DISPLAY MEDIA (cont'd)

Display	Phase	Quantity	Range (Of Display)	Range in Degrees	Scale Factor
PLOT BOARD 1	Launch	γ	0^{nm} to $.654498469^{R}$ ($V/V_r < .19$)	0 to 37.5	1563.02886
			0^{nm} to 1.04719755^{R} ($.19 \leq V/V_r \leq .9$)	0 to 60	976.893041
			$-.017453293^{R}$ to $+.034906585^{R}$ ($V/V_r > .9$)	−1 to +2	19537.8608
	Launch	V/V_r	.04 to .19 ($V/V_r < .19$)	–	6820.0
			0 to 1.5 ($.19 \leq V/V_r \leq .9$)	–	682.0
			.9 to 1.05 ($V/V_r > .9$)	–	6820.0
	Abort Orbit Re-entry	$r - \bar{R}$	0^{nm} to 200^{nm}	–	5.115
	Abort Orbit Re-entry	V	$0^{F/S}$ to $30{,}000^{F/S}$	–	.0341

*See 5.2.3, Other Requirements.

TABLE 5-5. DISPLAYED QUANTITIES AND THEIR FORMAT

Quantity	Total Number of Bits	Pattern	Display Range	Example
Orbit Lifetime	6	24	0 to 39	19
Inclination Angle (i)	12	444	0 to 99.9	32.°1
Flight Path Angle (γ)	17	14444	–99.99 to +99.99	–02.°16
Apogee Ht (h_a)	17	14444	–999.9 to +999.9	+102.6 nautical miles
Velocity Ratio (V/V_r)	18	24444	0 to 3.9999	1.0013
Velocity (V)	18	24444	0 to 39999	24,362 ft/sec
$r - \bar{R}$	16	4444	0 to 999.9	102.4 nautical miles
Recovery Area	5	23		
Latitude of Landing Point	16	14434	–99.59 to +99.59	–89.59
Longitude of Landing Point	17	114434	–199.59 to +199.59	+179.59
Orbit Number	5	14	0 to 19	02
ICTRC	23	124134134	–39–59–59 to +39+59+59	–18 hrs +32 min –59 sec
GMT of Landing	13	2434	0000 to 3959	18 hrs 32 min
All Others	20	243434	000000 to 395959	18 hrs 32 min 10 sec

O0LANA also sets up 48 bits in locations TMMRLP and TMMRLP +1 which are used to operate the Goddard plotboards. These plotboards can duplicate any of the four plotboards located at Cape Canaveral (see O0PLOT).

5.2.3 Other Requirements

For O0LANA, Monitor cell MCCHSC is tested for zero. If it is, the longitudinal range for plotboard 4 during re-entry is set the same as during launch. If it is not zero, then the longitudinal range is set the same as during the orbit phase (-135° to -45°). In O0LANA it is assumed MCPGMT contains present GMT in fixed point seconds, scaled B35.

O0LANA and O0ORRE have been programmed to be compiled independently. Hence, many general symbolic locations have the same name and care should be taken to avoid multiple defined symbols if they are compiled together.

5.2.4 Usage

a) Transfer Information

1) O0LANA—when the input block of data (TMLANA) is ready for conversion to the output format, Monitor transfers to O0LANA. Return is made to MFLANA.

2) O0ORRE—Monitor transfers to O0ORRE when TMORRE is ready for conversion to the output format. Return is made to MFORRE.

b) Space Required

O0LANA

Instructions	437
Constants	55
Erasable	21
*Subroutines	
PACK	73
BCD	101
DGHR	26
MBCD	29
PLOT	73

**Tables	43
Ind. Add.	28
Other	15
Total	858

O0ORRE

Instructions	365
Constants	40
Erasable	21
*Subroutine	276
**Tables	
Ind. Add.	15
Other	6
Total	723

c) Timing—Running time for O0LANA or O0ORRE is approximately 50 to 100 milliseconds

d) Checkout—O0LANA and O0ORRE have been operational in the Mercury Program System since Dec. 1, 1960.

e) Constants Used for O0LANA (The asterisk indicates those constants used also by O0ORRE).

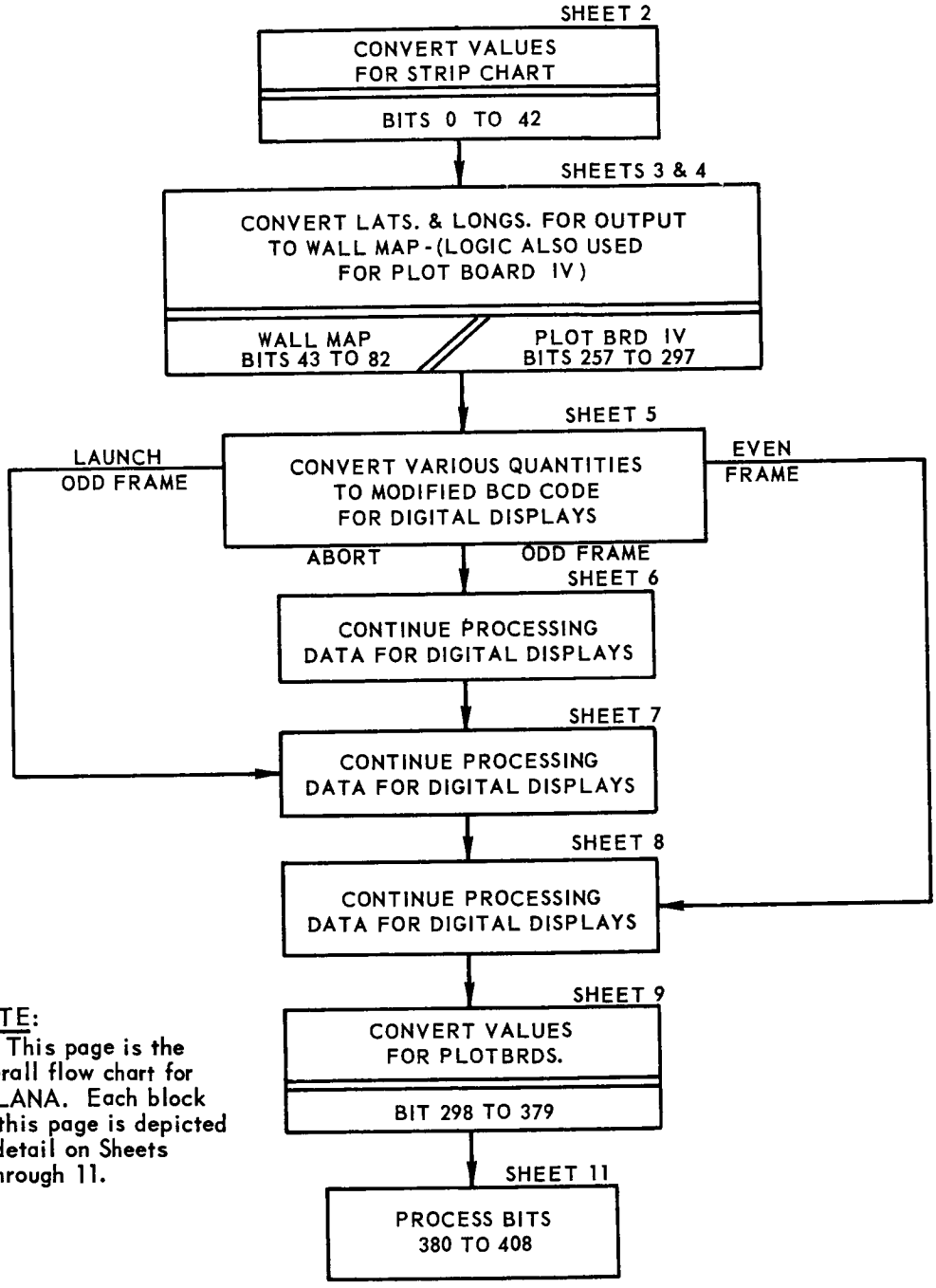
K0.SF1	DEC	5861.358244
K0.SF2	DEC	12787.5
*K0.SF3	DEC	732.6697805
*K0.SF4	DEC	162.8155068
*K0.SF5	DEC	2442.232602
*K0.SF6	DEC	651.2620273
K0.5DG	DEC	.0872664626
K00.04	DEC	.04
*K.40DG	DEC	.6981317008
*K00.PI	DEC	3.141592654
*K.12DG	DEC	-0.2094395102
*K.82DG	DEC	1.431169987
K0130.	DEC	130
K000.9	DEC	0.9
K0.1DG	DEC	0.9174532925

*Subroutine count includes only instructions and tables, the constants, erasable storage, etc., are accounted for elsewhere.
**Does not include TMOLAB or TMORRE

K00003	DEC	3B35
KD0012	DEC	3B15
K00005	DEC	5B35
*KD0020	DEC	5B15
*KU1777	OCT	233000001777
*K01023	DEC	1023B35
KBB33Z	OCT	200000000001
*KD0004	DEC	1B15
*KD2048	DEC	1B6
*KT0006	DEC	3B19
*KT0004	DEC	1B18
KBB23Z	OCT	000400000001
KD1024	DEC	1B7
*K00256	DEC	1B27
*K00512	DEC	1B26
*K00001	DEC	1
K01024	DEC	1B25
KD0008	DEC	1B14
*K00010	DEC	10
*K.30SC	DEC	0.1454441043E-3
K00.71	DEC	0.71
*KKK2PI	DEC	6.283185307
*K.1RAD	DEC	57.29577951
*K0.MPR	DEC	3437.746771
*K00000	PZE	0
*K00027	DEC	27
*K00060	DEC	60
K0060.	DEC	60
*K15360	DEC	15360
*KLLCW5	OCT	7400
*KLLCW6	OCT	4400
*KLLCW9	OCT	26400
*KMNMSK	OCT	777777777
KPDM31	OCT	7777777777
KAZ16M	OCT	-377776777777
*KCH233	OCT	233000000000
*KTGMSK	OCT	700000
*K10000	OCT	10000
*K13MK1	OCT	777777
*K00377	OCT	377

In addition to the above marked constants, O0ORRE uses the following four constants:

K.135D	DEC	2.3561944902
K0.144	DEC	0.144
K0400.	DEC	400.
K00600	DEC	600



NOTE:
This page is the overall flow chart for OOLANA. Each block on this page is depicted in detail on Sheets 2 through 11.

FIGURE 5-2. OOLANA PROGRAM FLOW CHART *(Sheet 1 of 13)*

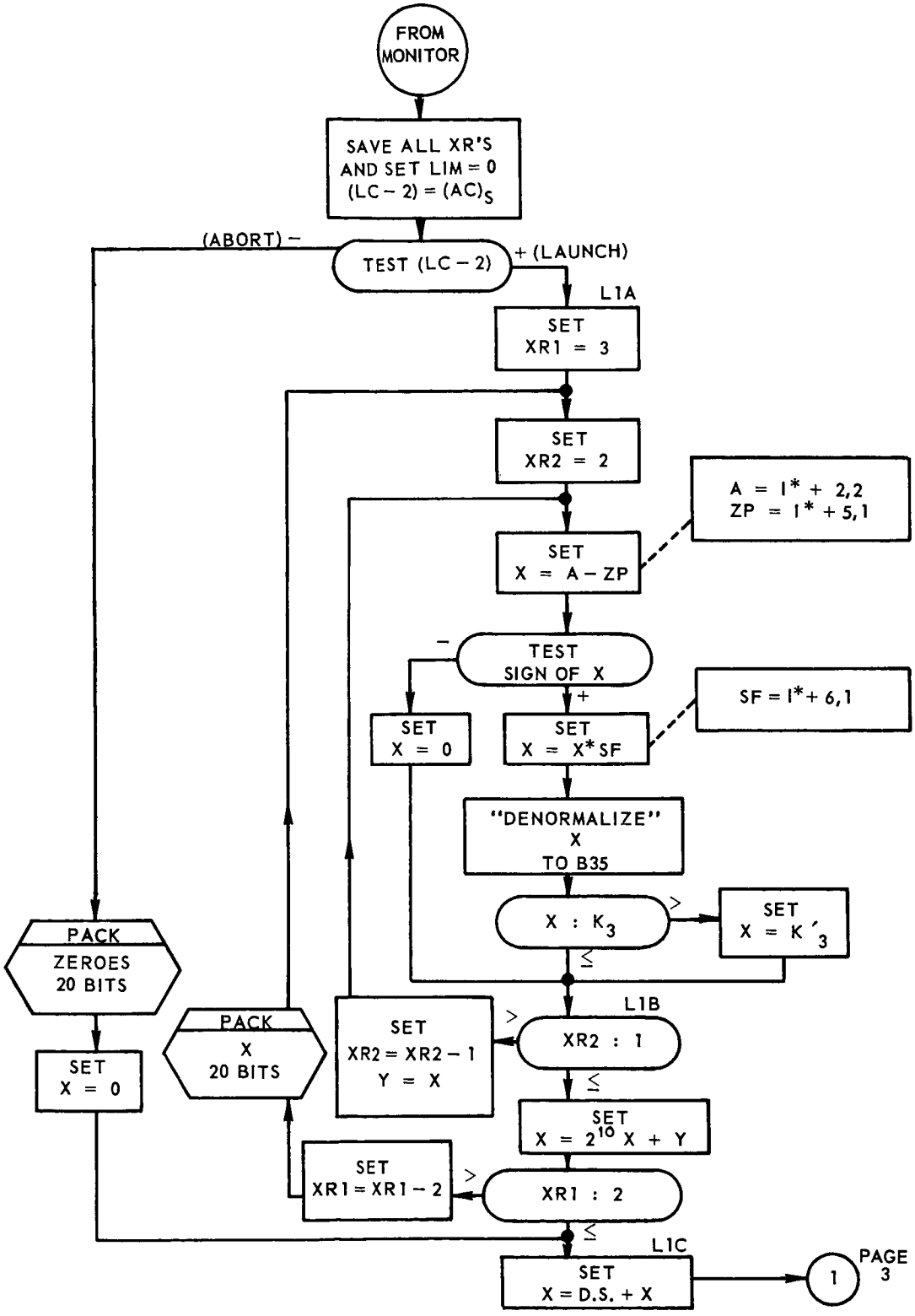


FIGURE 5-2. OOLANA PROGRAM FLOW CHART *(Sheet 2 of 13)*

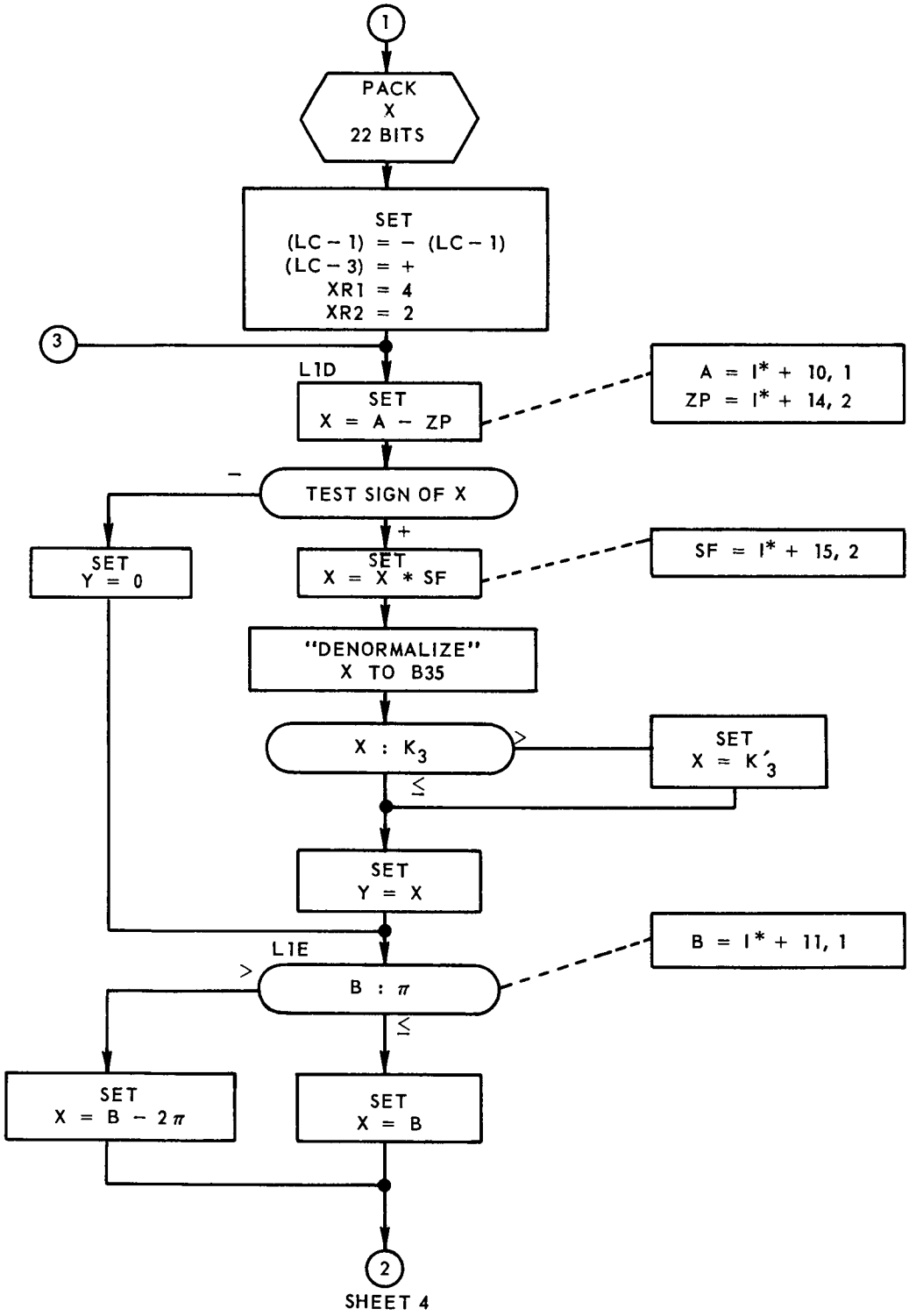


FIGURE 5-2. OOLANA PROGRAM FLOW CHART *(Sheet 3 of 13)*

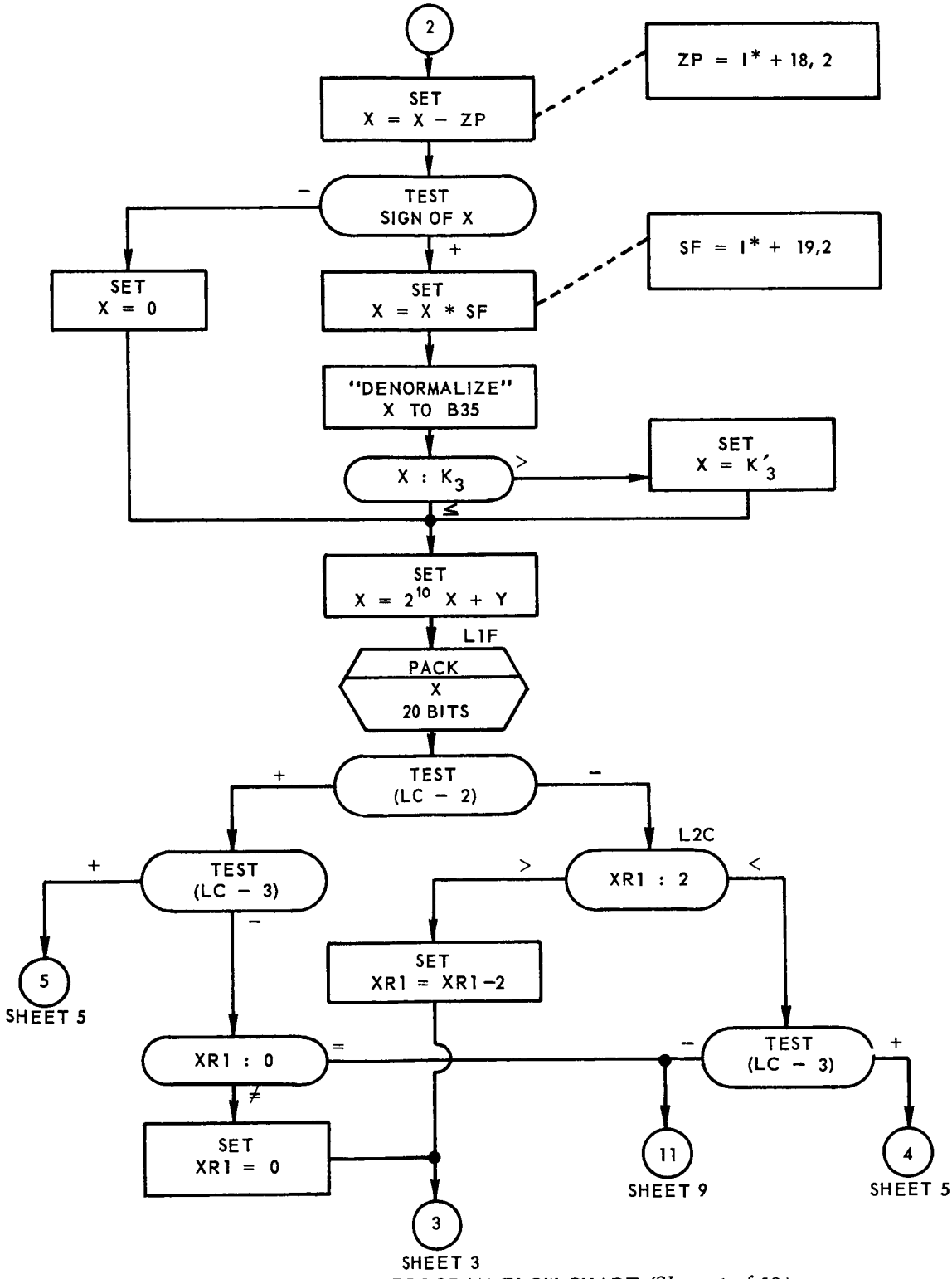


FIGURE 5-2. OOLANA PROGRAM FLOW CHART *(Sheet 4 of 13)*

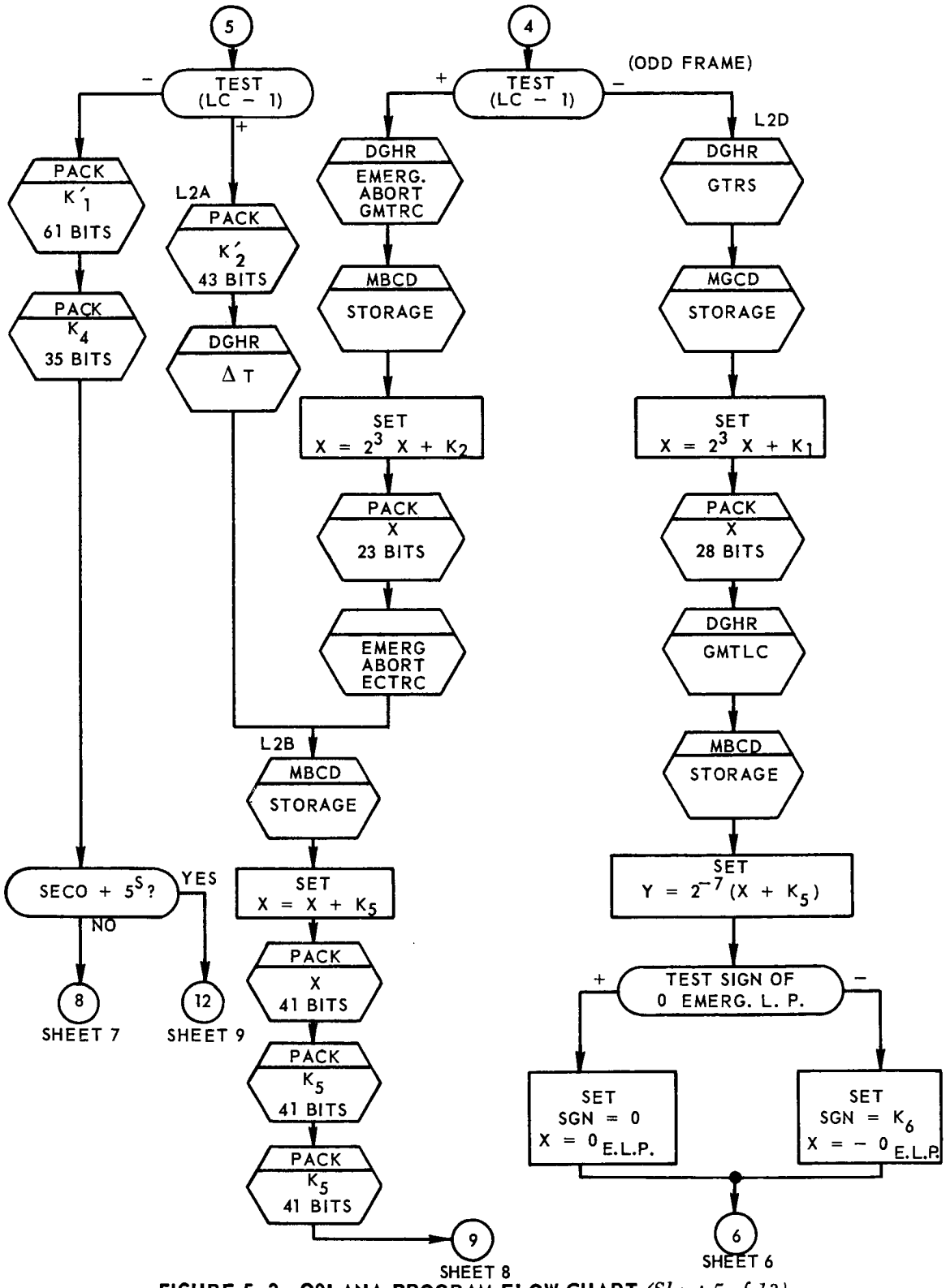


FIGURE 5-2. OOLANA PROGRAM FLOW CHART *(Sheet 5 of 13)*

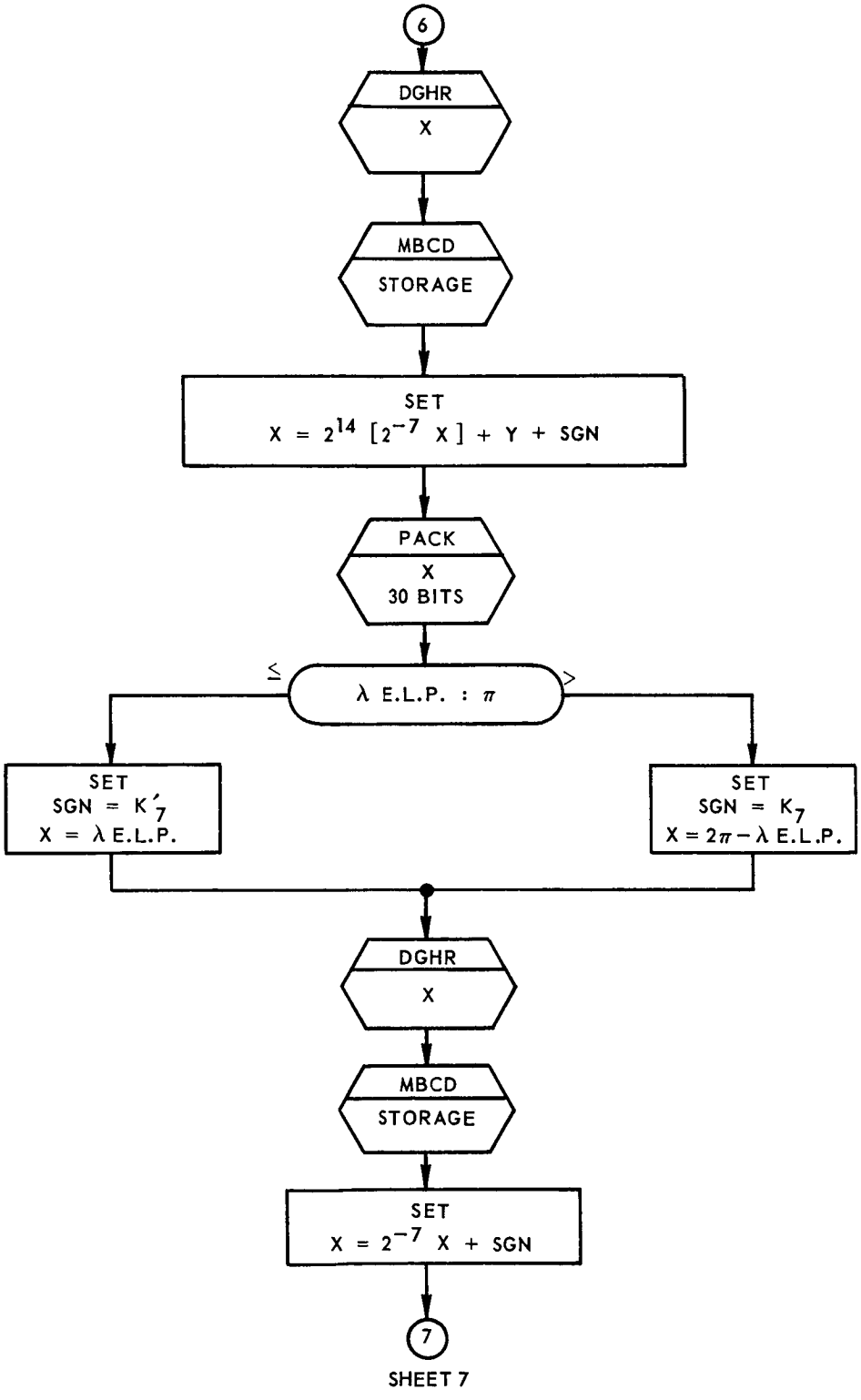


FIGURE 5-2. OOLANA PROGRAM FLOW CHART *(Sheet 6 fo 13)*

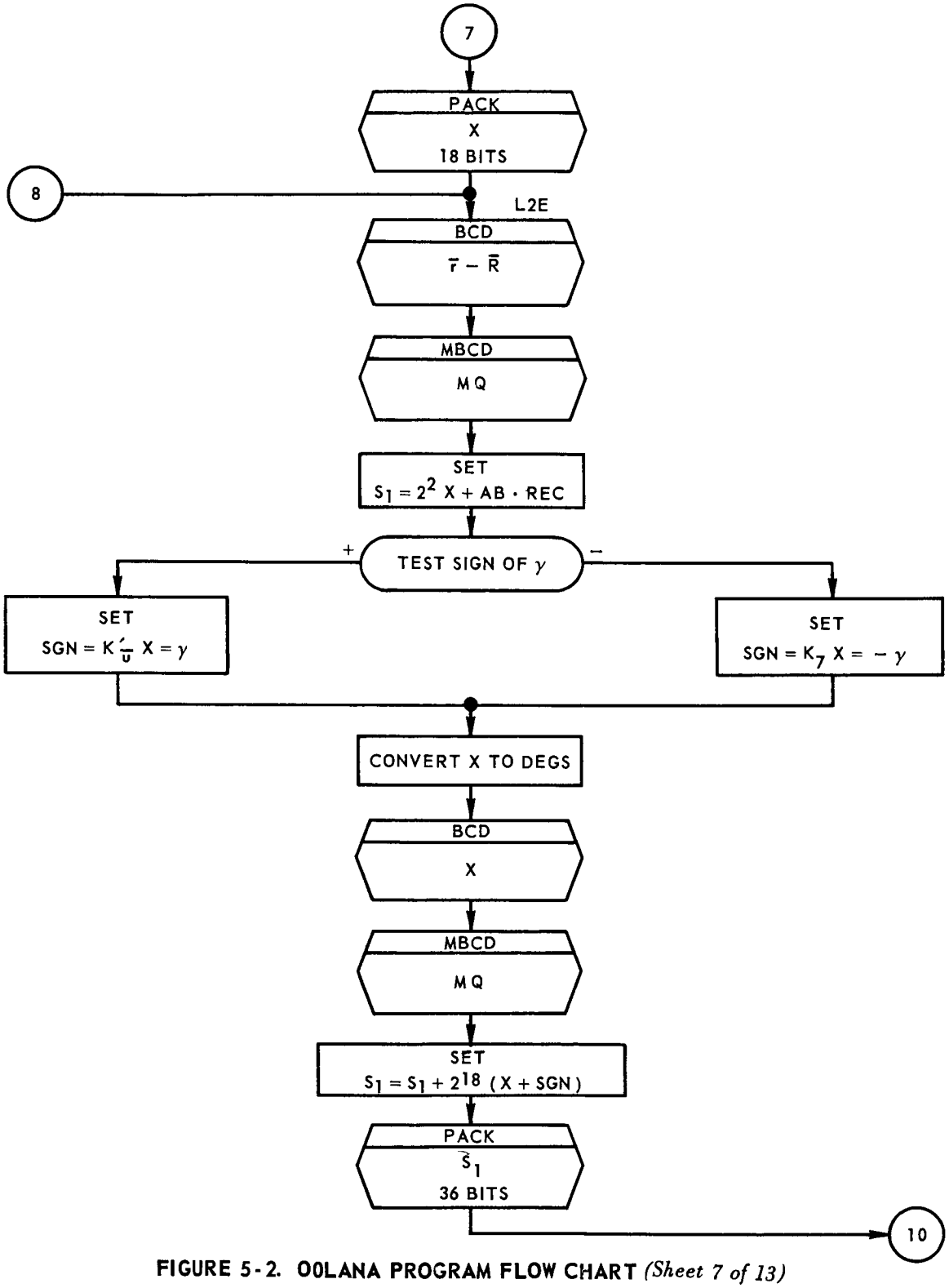


FIGURE 5-2. OOLANA PROGRAM FLOW CHART *(Sheet 7 of 13)*

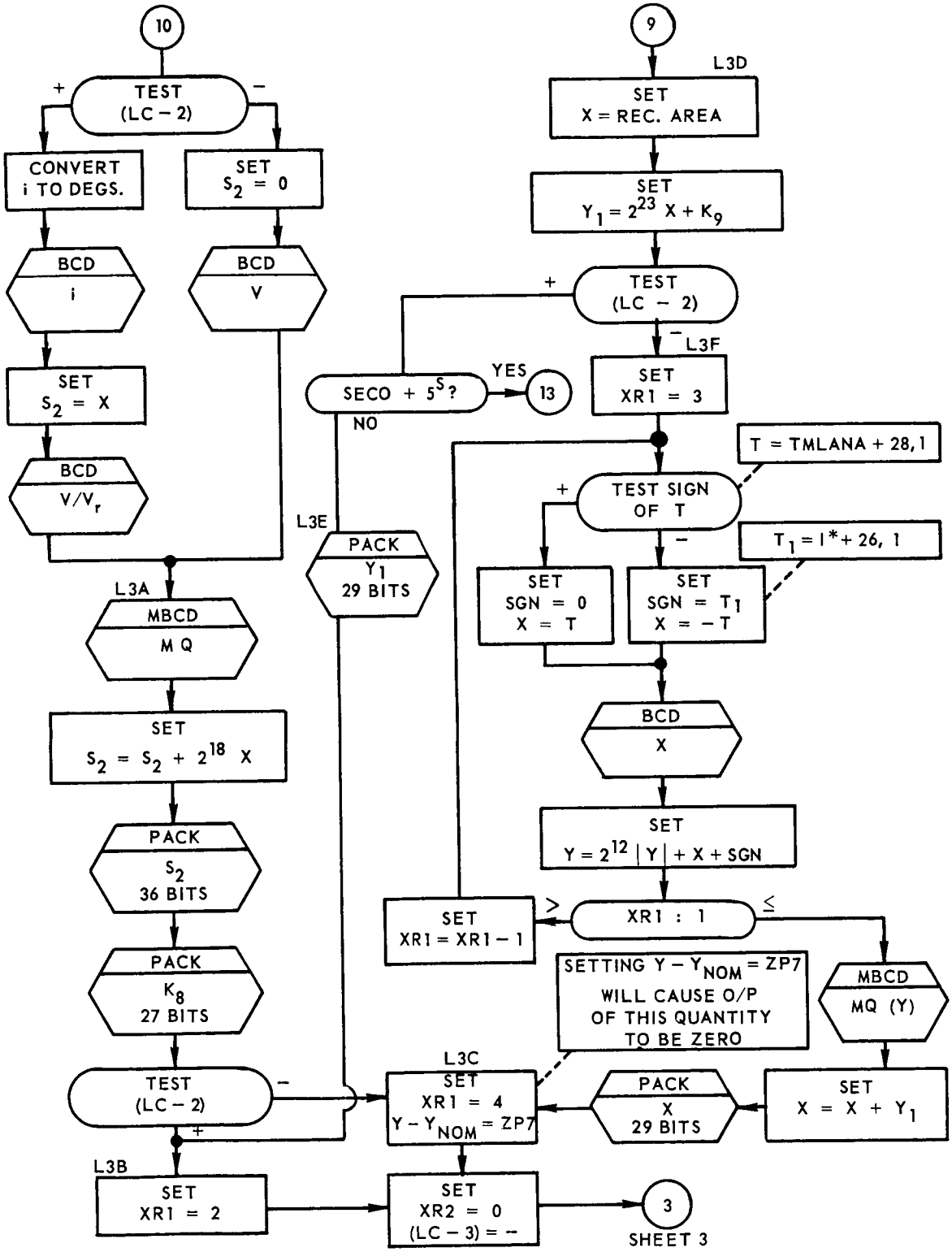


FIGURE 5-2. OOLANA PROGRAM FLOW CHART *(Sheet 8 of 13)*

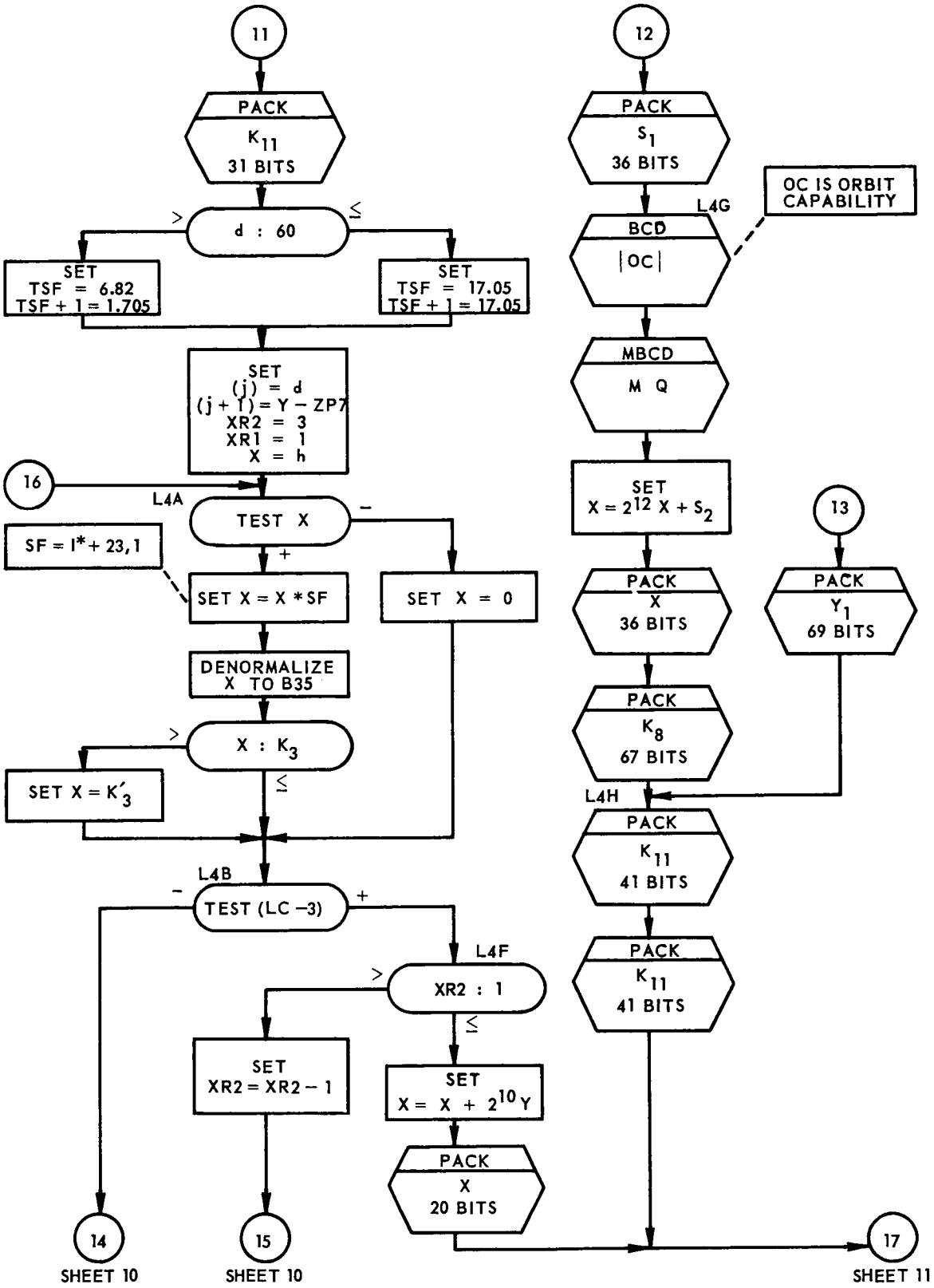
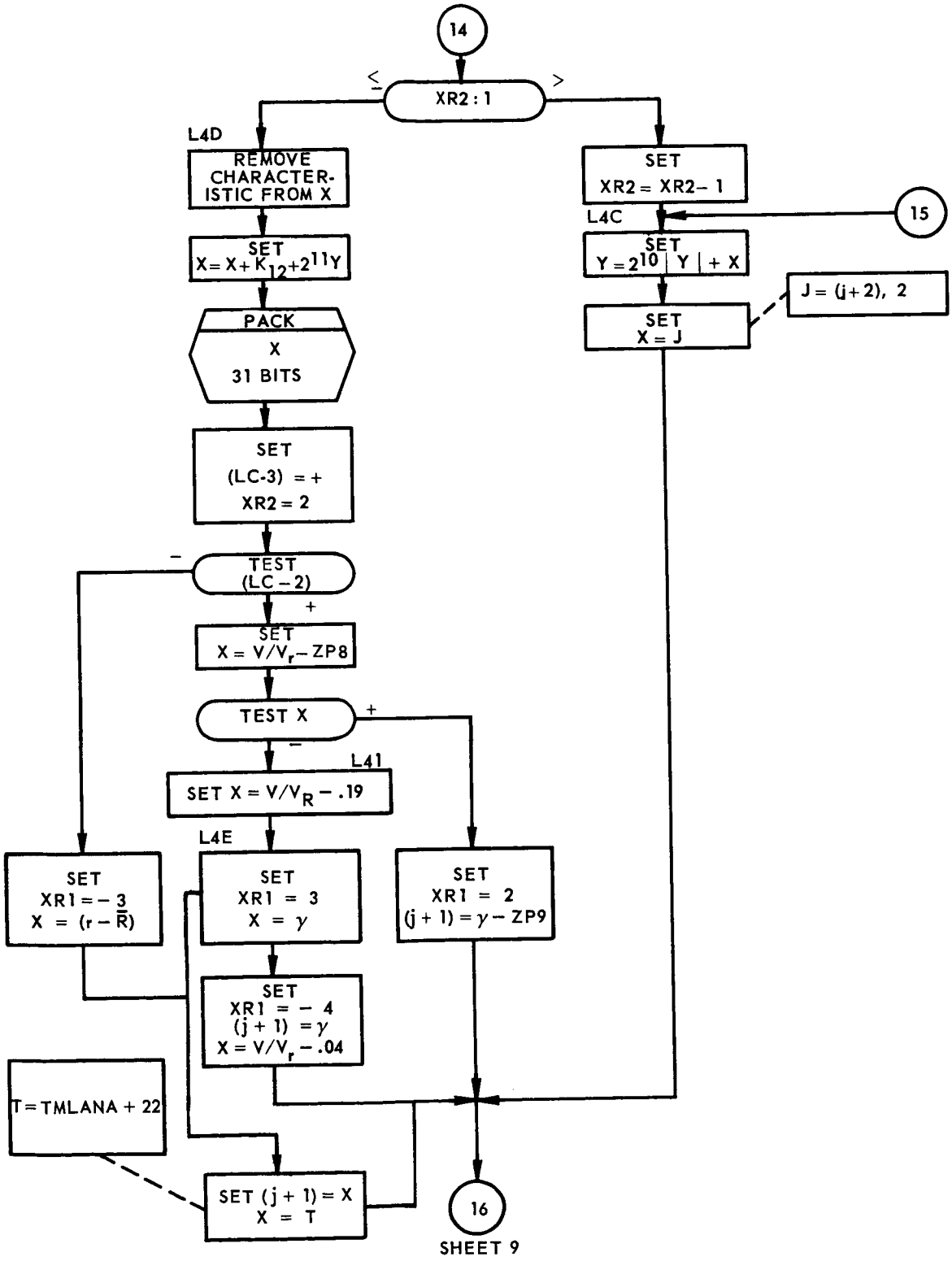


FIGURE 5-2. OOLANA PROGRAM FLOW CHART *(Sheet 9 of 13)*



SHEET 9

FIGURE 5-2. OOLANA PROGRAM FLOW CHART *(Sheet 10 of 13)*

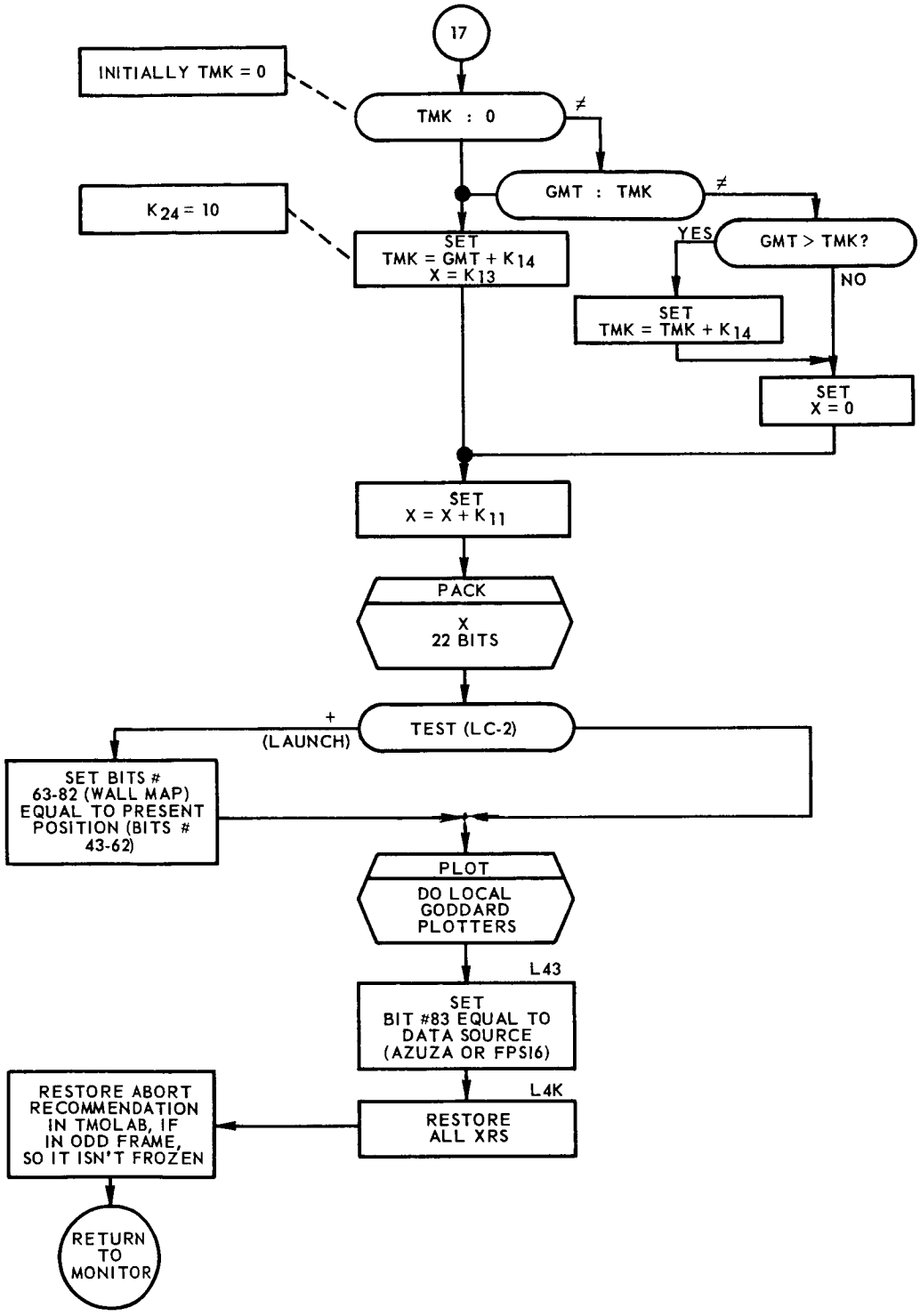


FIGURE 5-2. OOLANA PROGRAM FLOW CHART *(Sheet 11 of 13)*

GENERAL CONSTANTS FOR OOLANA

NAME	PROGRAM SYMBOL		VALUE
K_1	K00003	DEC	3B35
K'_1	KD0012	DEC	3B15
K_2	K00005	DEC	5B35
K'_2	KD0020	DEC	5B15
K_3	KU1777	OCT	233000001777
K'_3	K01023	DEC	1023B35
K_4	KBB33Z	OCT	200000000001
K_5	KD0004	DEC	1B15
K_6	KD2048	DEC	1B6
K_7	KT0006	DEC	3B19
K'_7	KT0004	DEC	1B18
K_8	KBB23Z	OCT	000400000001
K_9	KD1024	DEC	1B7
K_{10}	K00256	DEC	1B27
K'_{10}	K00512	DEC	1B26
K_{11}	K00001	DEC	1B35
K_{12}	K01024	DEC	1B25
K_{13}	KD0008	DEC	1B14
K_{14}	K00010	DEC	10B35
–	KD0060	DEC	60B17

LOGICAL CHOICES

LC-1 = +	EVEN FRAME
LC-1 = –	ODD FRAME
LC-2 = +	LAUNCH
LC-2 = –	ABORT
LC-3 = +	WALL MAP OR PLOT BOARD I
LC-3 = –	PLOT BOARD IV OR PLOT BOARD II

FIGURE 5-2. OOLANA PROGRAM FLOW CHART *(Sheet 12 of 13)*

INDIRECT ADDRESSES
(I*)

LOC.	ADD, T
I	TMLANA + 3, 1
+ 1	TMLANA + 4, 1
+ 2	ZP1
+ 3	SF1
+ 4	ZP2
+ 5	SF2
+ 6	TMLANA + 5
+ 7	TMLANA + 6
+ 8	TMLANA + 11
+ 9	TMLANA + 12
+10	TMLANA + 7
+11	TMLANA + 8
+12	ZP3
+13	SF3
+14	ZP5
+15	SF5
+16	ZP4
+17	SF4
+18	ZP6
+19	SF6
+20	TSF + 7, 2
+21	TSF + 5, 2
+22	TSF + 3, 2
+23	K_{10}
+24	K_{10}'
+25	K_{10}'
+26	TSF + 9, 2
+27	TSF + 11, 2

TABLE OF SCALE FACTORS
(TSF)

LOC.	ASSOCIATED QUANTITY	VALUE
TSF	h	**VALUE PUT IN TABLE BY PROG.
+ 1	d	**VALUE PUT IN TABLE BY PROG.
+ 2	$\gamma - \gamma_{nom}$	6.82
+ 3	V/V_r (>.9)	6820.0
+ 4	γ (V/V_r >.9)	19537.86801
+ 5	V/V_r ($\geq$ .19 $\leq$ .9)	682.0
+ 6	γ (V/V_r $\geq$ .19 $\leq$ .9)	976.8930407
+ 7	$\sqrt{}$	.0341
+ 8	r - R	5.1150
+ 9	V/V_r (<.19)	6820.0
+10	δ (V/V_r <.19)	1563.028865

ZPNS AND SFNS

DISPLAY	N	ASSOCIATED QUANTITY	SCALE FACTORS (SFN)	PROGRAM SYMBOL	ZERO POINTS (ZPN)	PROGRAM SYMBOL	ZPN IN DEGREES
S. C.	1	$\gamma - \gamma_{NOM}$	861.358244	K0.SF1	$-.0872664626^{RAD}$	K0.5DG	$-5°$
S. C.	2	$V/V_r - V/V_{r_{NOM}}$	12787.5	K0.SF2	$-.04$	K00.04	–
W. M.	3	$\phi_{I.P.}, \phi_{P.P.}$	732.6697805	K0.SF3	$-.6981317008^{RAD}$	K.40DG	$-40°$
W. M.	4	$\lambda_{I.P.}, \lambda_{P.P.}$	162.8155068	K0.SF4	-3.141592654^{RAD}	K00.PI	$-180°$
PB IV	5	$\phi_{I.P.}, \phi_{P.P.}$	2442.232602	K0.SF5	$+.2094395102^{RAD}$	K.12DG	$+12°$
PB IV	6	$\lambda_{I.P.}, \lambda_{P.P.}$	651.2620273	K0.SF6	-1.431169987^{RAD}	K.82DG	$-82°$
PB II	7	$Y - Y_{NOM}$	–	–	$-130.0^{N.M.}$	K0130.	–
PB I	8	V/V_r	–	–	$+.9$	K000.9	–
PB I	9	γ	–	–	$-.0174532925$	K0.1DG	$-1°$

FIGURE 5-2. OOLANA PROGRAM FLOW CHART *(Sheet 13 of 13)*

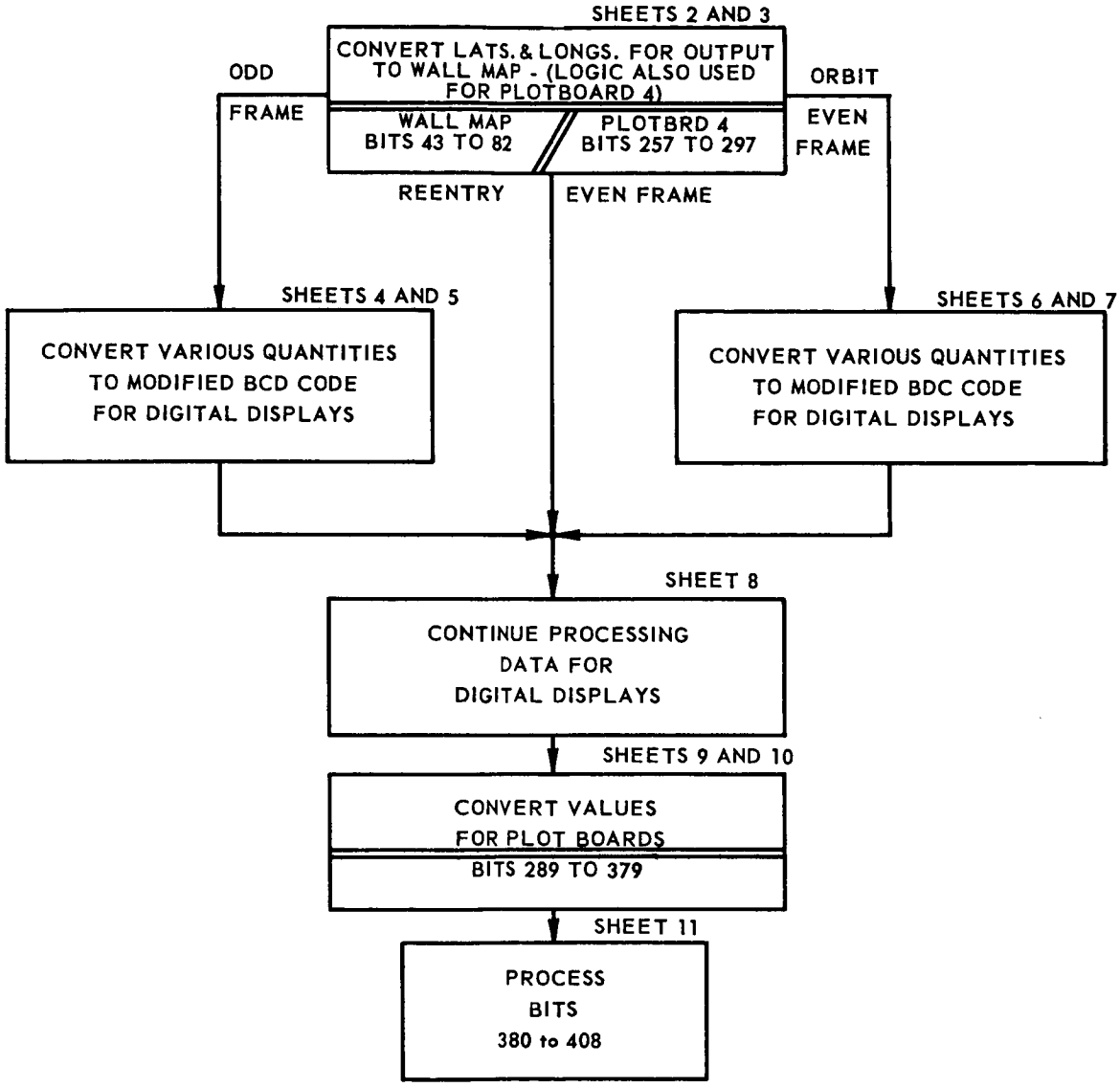


FIGURE 5-3. OOORRE PROGRAM FLOW CHART *(Sheet 1 of 13)*

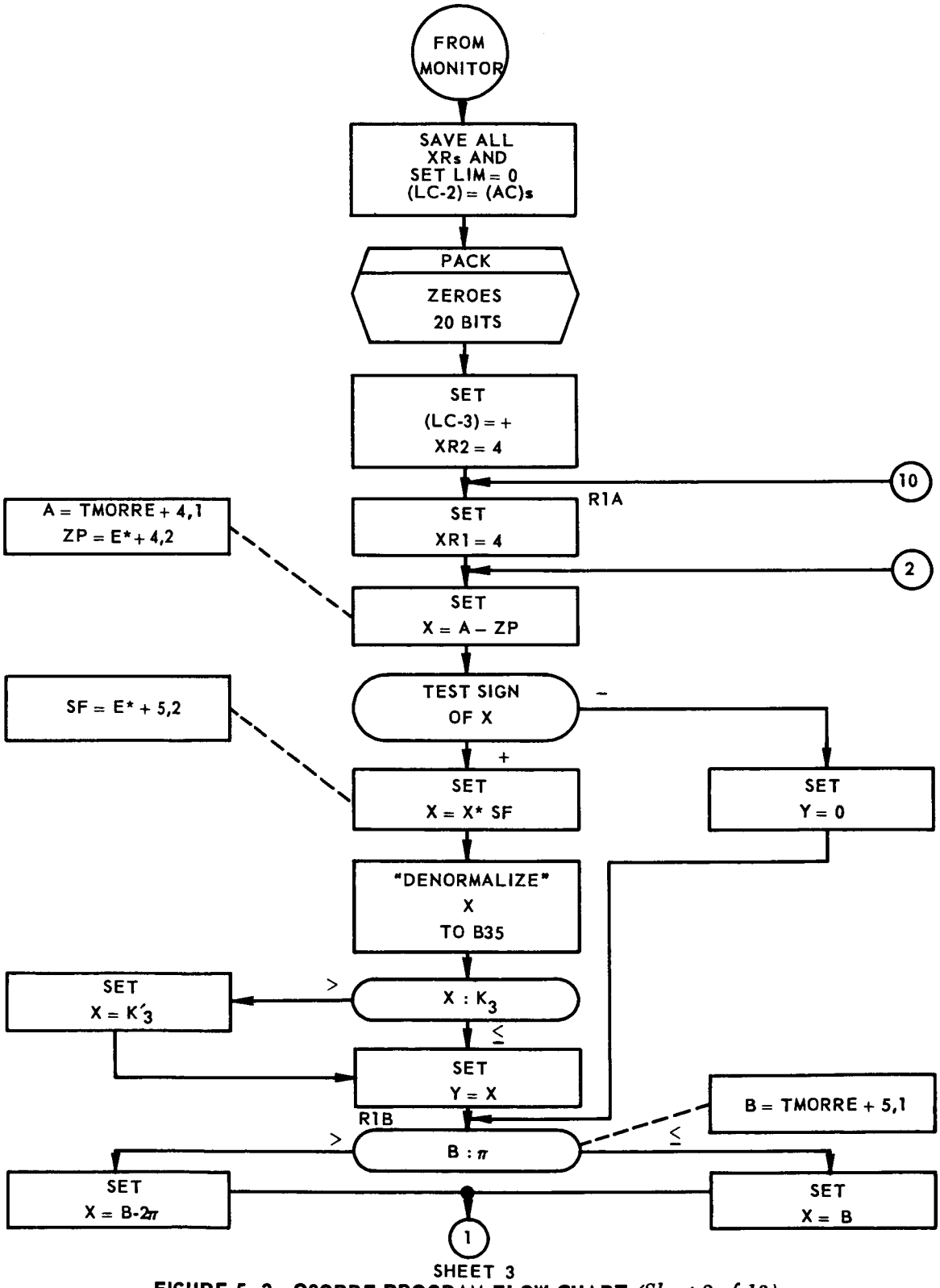


FIGURE 5-3. OOORRE PROGRAM FLOW CHART *(Sheet 2 of 13)*

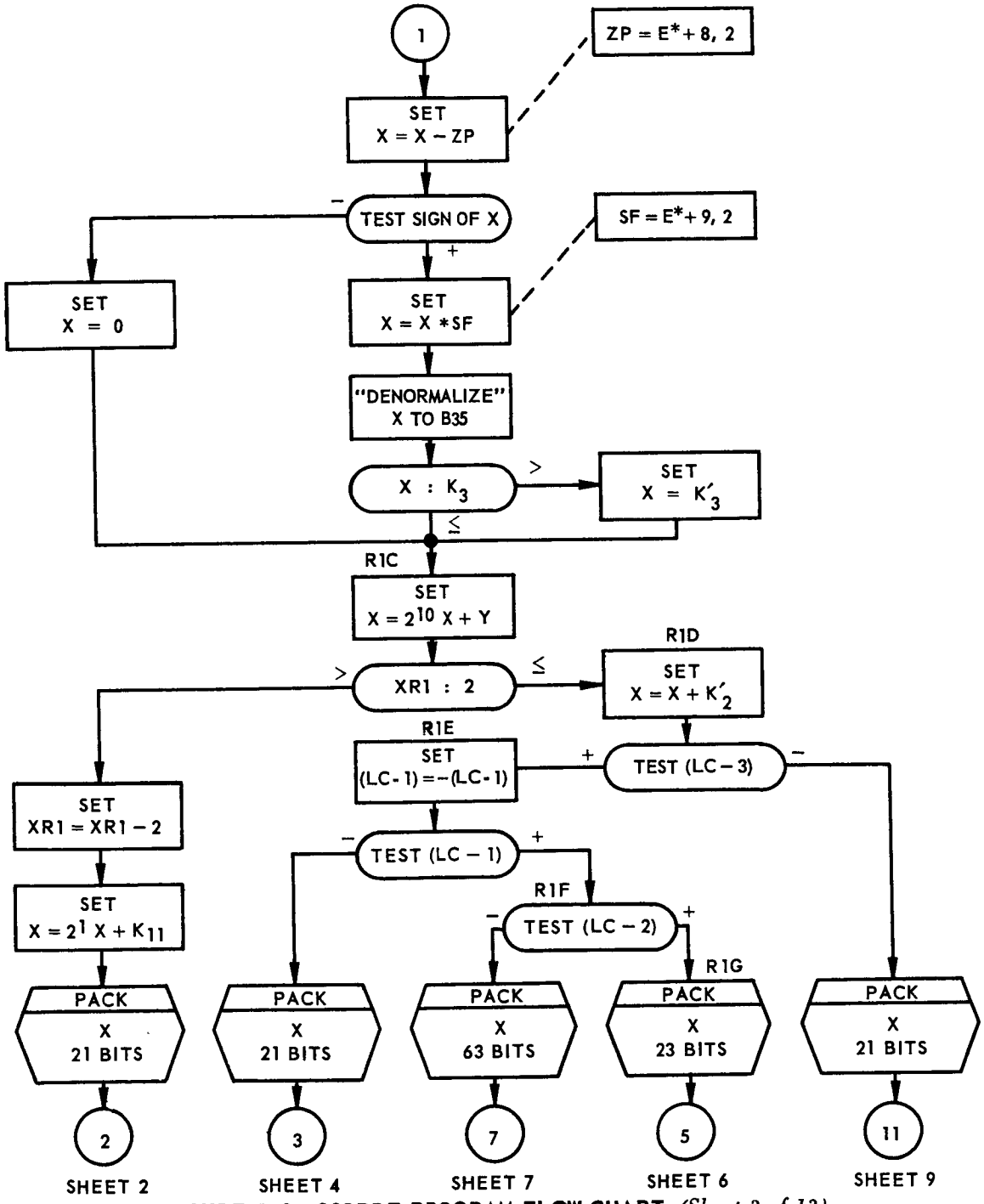
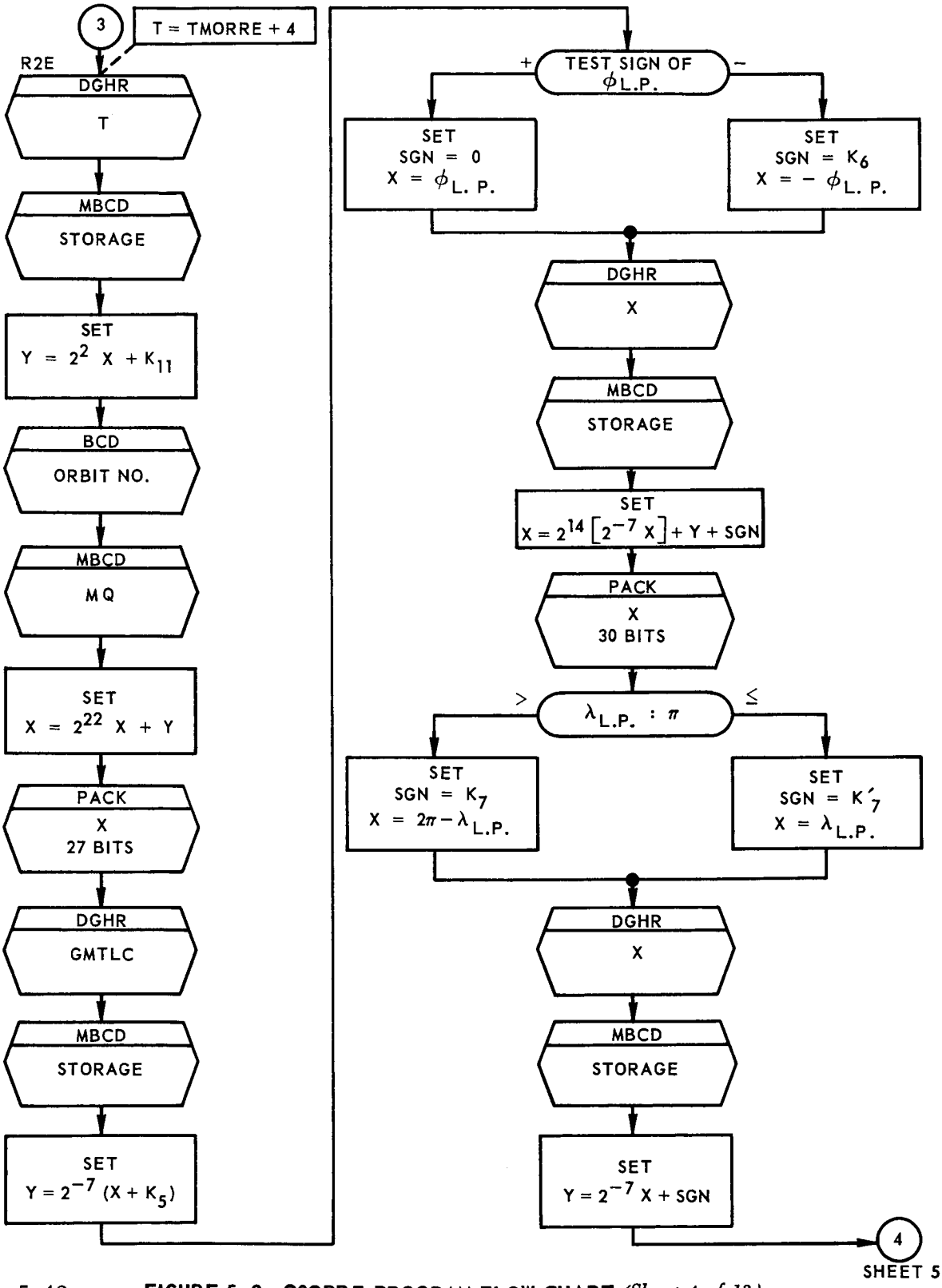


FIGURE 5-3. OOORRE PROGRAM FLOW CHART *(Sheet 3 of 13)*



5-42 **FIGURE 5-3. OOORRE PROGRAM FLOW CHART** *(Sheet 4 of 13)*

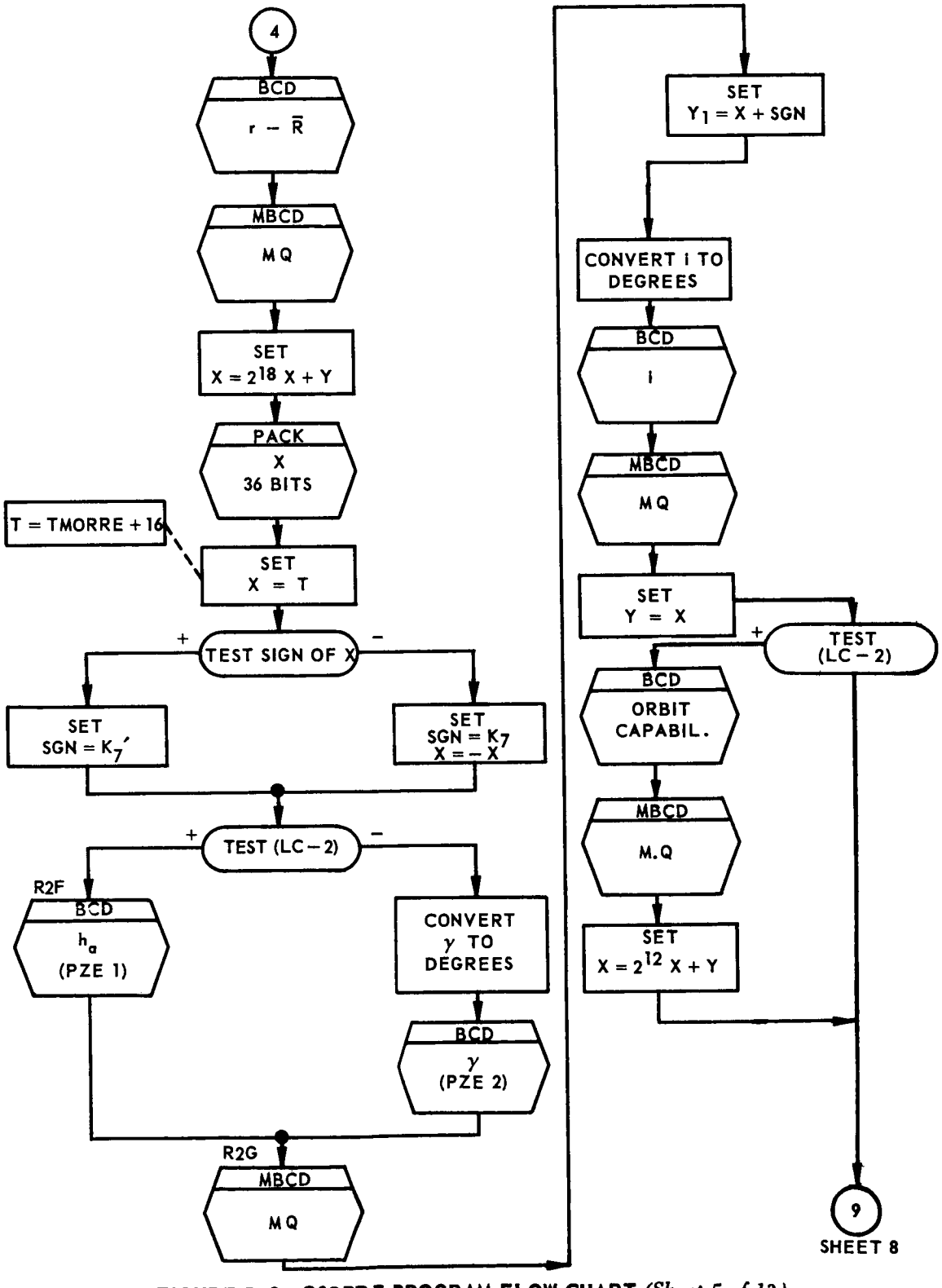


FIGURE 5-3. OOORRE PROGRAM FLOW CHART *(Sheet 5 of 13)*

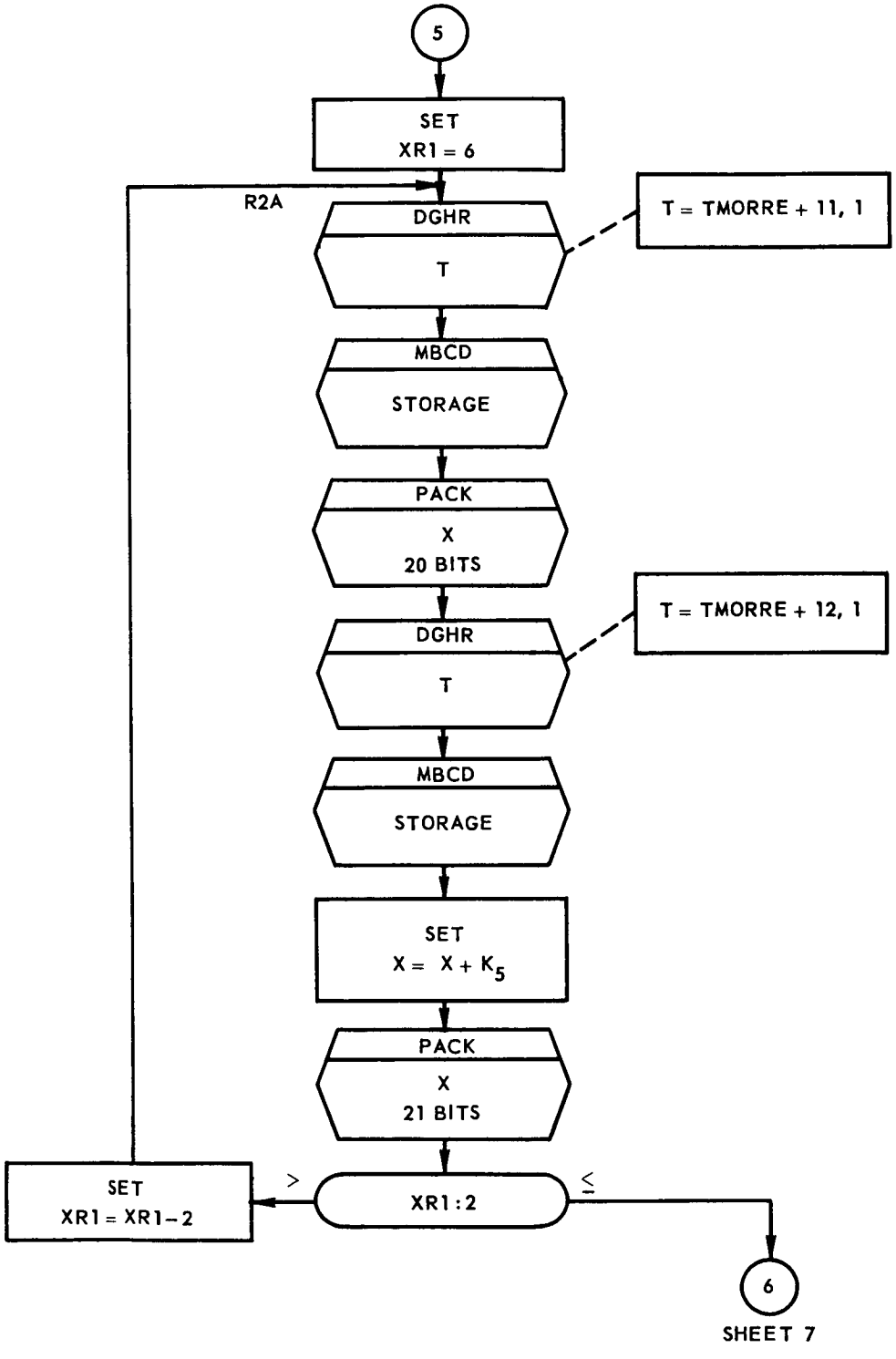


FIGURE 5-3. OOORRE PROGRAM FLOW CHART *(Sheet 6 of 13)*

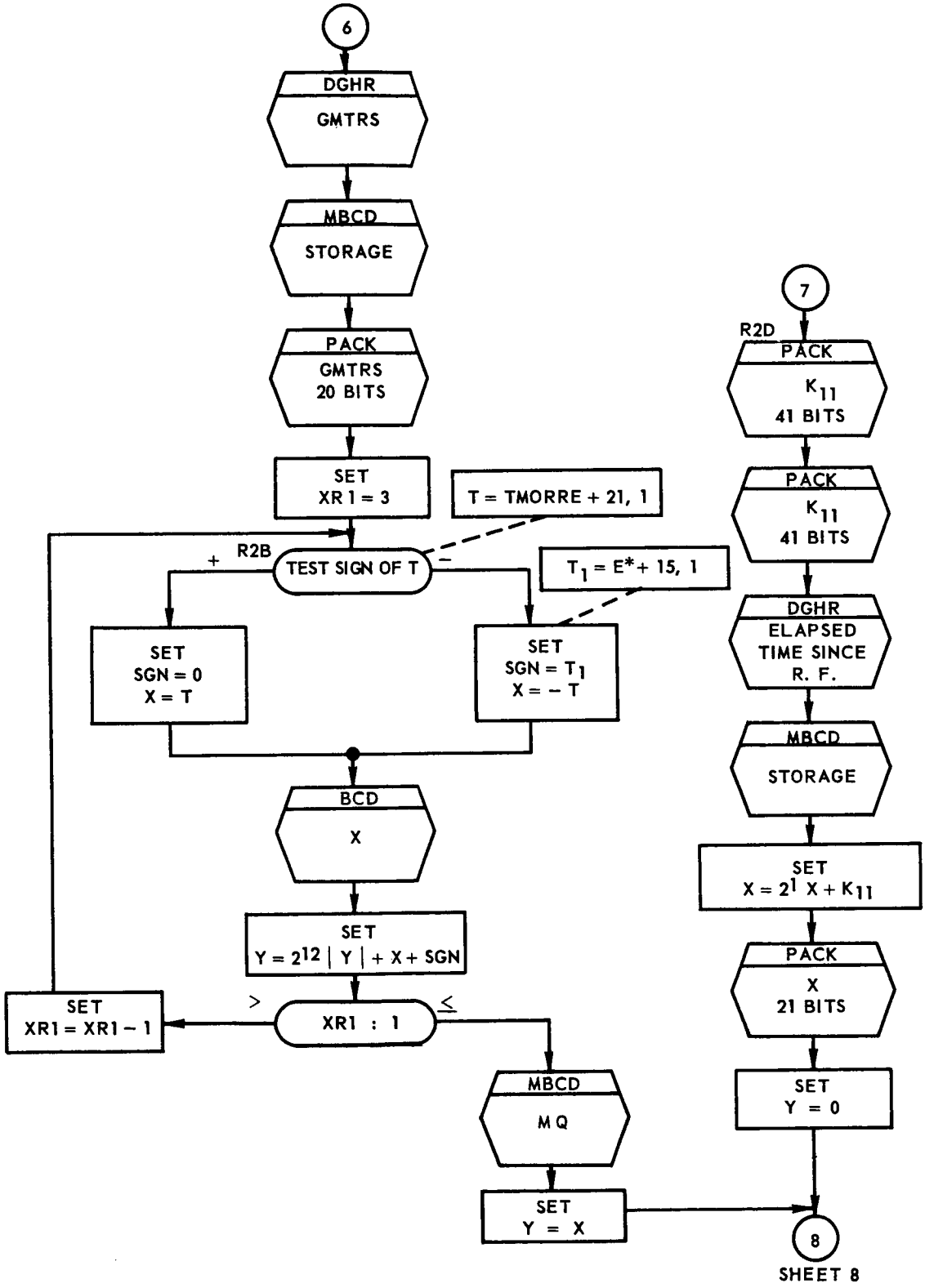


FIGURE 5-3. OOORRE PROGRAM FLOW CHART *(Sheet 7 of 13)*

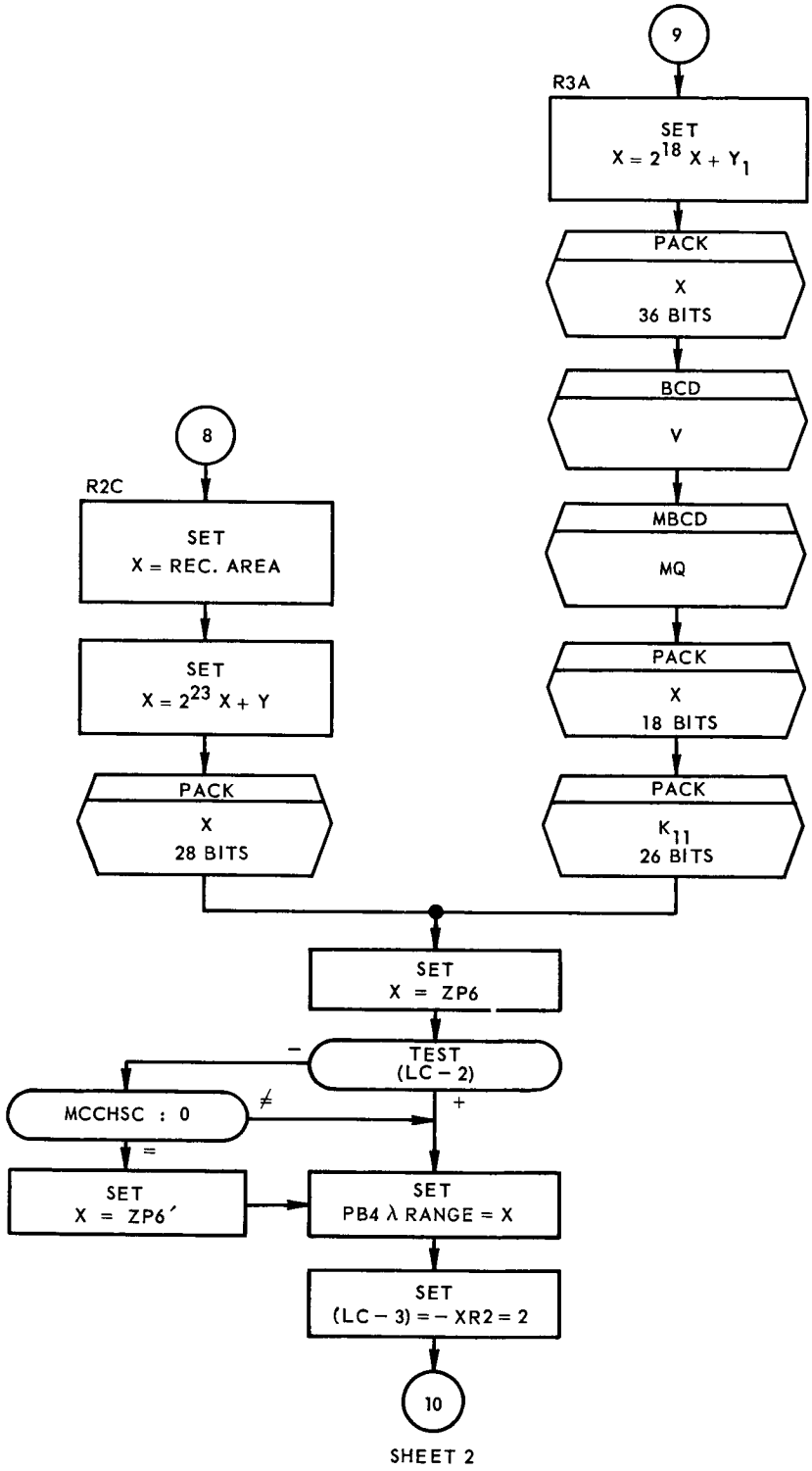


FIGURE 5-3. OOORRE PROGRAM FLOW CHART *(Sheet 8 of 13)*

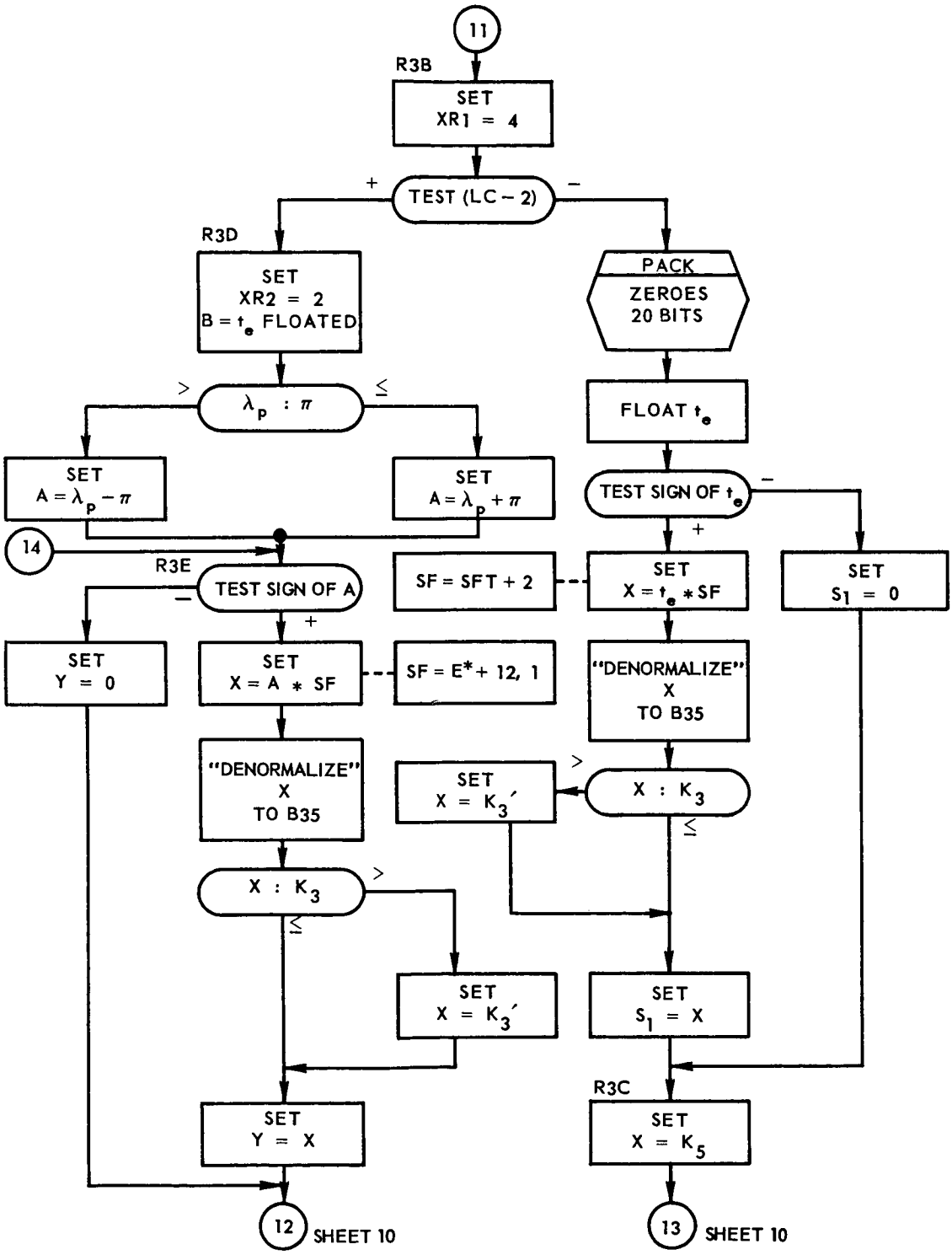


FIGURE 5-3. OOORRE PROGRAM FLOW CHART *(Sheet 9 of 13)*

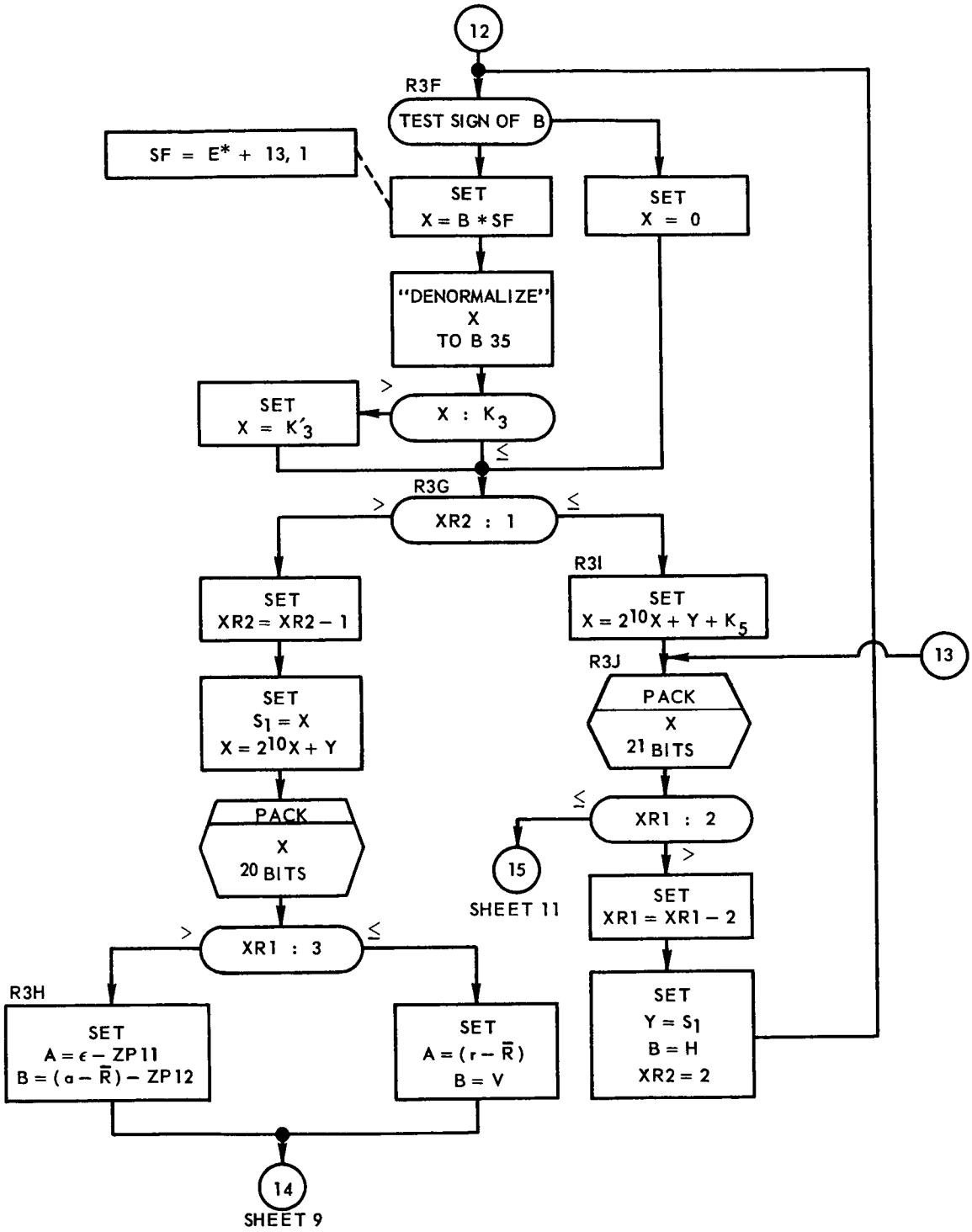


FIGURE 5-3. OOORRE PROGRAM FLOW CHART *(Sheet 10 of 13)*

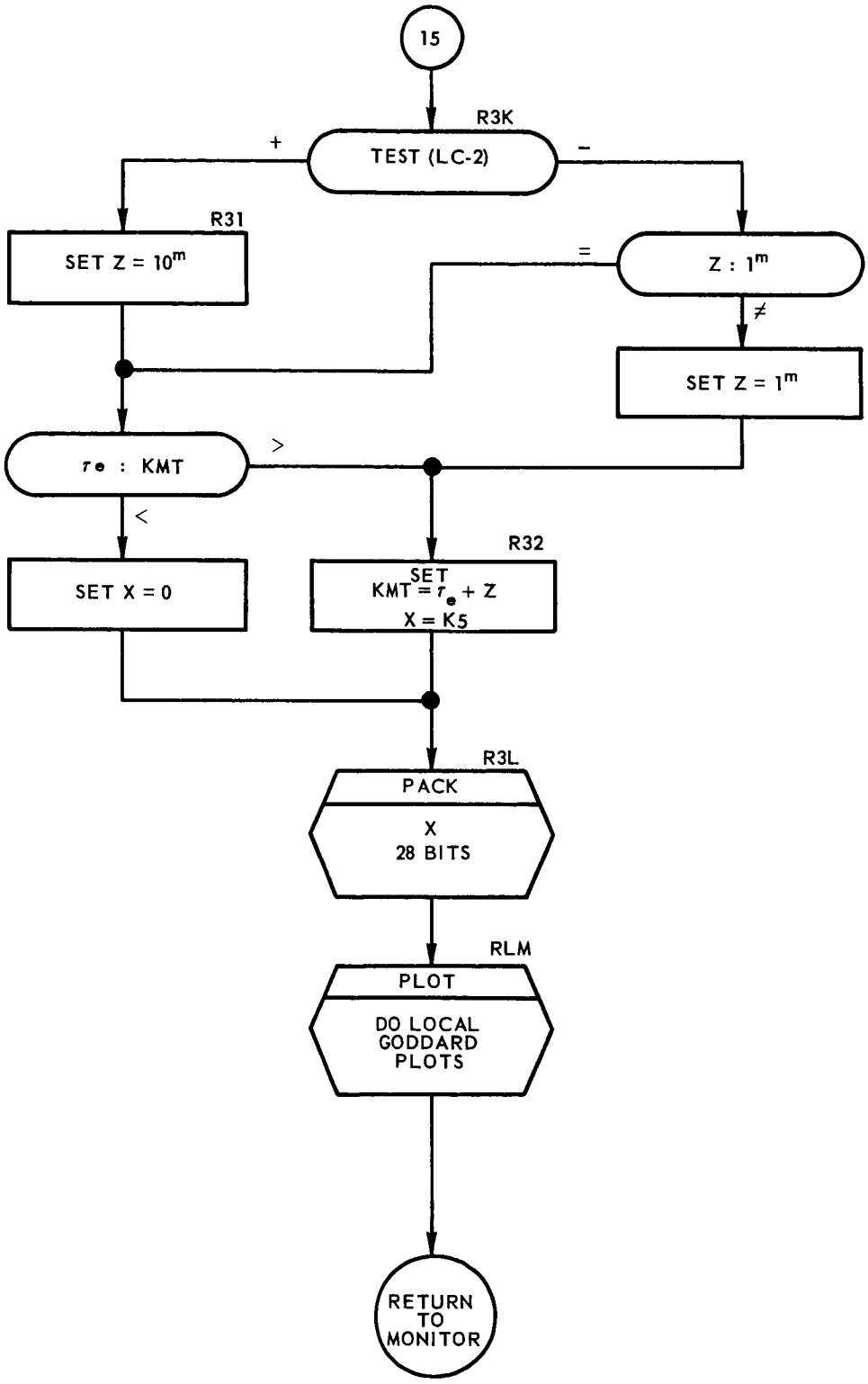


FIGURE 5-3. OOORRE PROGRAM FLOW CHART *(Sheet 11 of 13)*

INDIRECT ADDRESSES (E*)

LOC.	ADD, T
E	ZP3
+1	SF3
+2	ZP5
+3	SF5
+4	ZP4
+5	SF4
+6	ZP6 or ZP6′
+7	SF6
+8	SFT+2,2
+9	SFT+4,2
+10	SFT+6,2
+11	SFT+5,2
+12	K_{10}
+13	K_{10}'
+14	K_{10}'

SCALE-FACTOR TABLE (SFT)

LOC.	ASSOCIATED QUANTITY	VALUE
SFT	λ_p	81.40775337
+1	ϵ	3552.083333
+2	t_e	.0568333333
+3	$h, a - \bar{R}$	1.705
+4	V	.0341
+5	$r - \bar{R}$	5.1150

ZPNS AND SFNS

DISPLAY	N	ASSOCIATED QUANTITY	SCALE FACTORS (SFN)	PROGRAM SYMBOL	ZERO POINTS (ZPN)	PROGRAM SYMBOL	ZPN IN DEGS.
W. M.	3	ϕ	732.6697805	K0.SF3	$-.6981317008^{RAD}$	K.40DG	-40°
W. M.	4	λ	162.8155068	K0.SF4	-3.141592654^{RAD}	K00.PI	-180°
PB IV	5	ϕ	2442.232602	K0.SF5	$+.2094395102^{RAD}$	K.12DG	$+12^\circ$
PB IV	6	λ	651.2620273	K0.SF6	-2.356194490^{RAD}	K.135D	-135° *
PB III	11	ϵ	–	–	$-.144$	K0.144	–
PB II	12	$a - \bar{R}$	–	–	-400^{NM}	K0400.	–

*K.82DG (-82°) = ZP6′
K1135D (-135) = ZP6

FIGURE 5-3. OOORRE PROGRAM FLOW CHART *(Sheet 12 of 13)*

GENERAL CONSTANTS

NAME	PROGRAM SYMBOL	VALUE	
K'_2	KD0020	DEC	5B15
K_3	KU1777	OCT	233000001777
K'_3	K01023	DEC	1023B35
K_5	KD0004	DEC	1B15
K_6	KD2048	DEC	1B6
K_7	KT0006	DEC	3B19
K'_7	KT0004	DEC	1B18
K_{10}	K00256	DEC	1B27
K'_{10}	K00512	DEC	1B26
K_{11}	K00001	DEC	1B35
K_{15}	K00060	DEC	60B35

LOGICAL CHOICES

LC–1 = +	EVEN FRAME
LC–1 = –	ODD FRAME
LC–2 = +	ORBIT
LC–2 = –	REENTRY
LC–3 = +	WALL MAP
LC–3 = –	PLOT BOARD IV

FIGURE 5-3. OOORRE PROGRAM FLOW CHART *(Sheet 13 of 13)*

5.3 BINARY CODED DECIMAL CONVERSION PROGRAM (BCD)

This program converts a positive fixed or floating point number to its BCD equivalent, inserting a BCD decimal point if desired. The flow chart for the BCD program is shown in Figure 5-5.

5.3.1 Input Requirements

a) Place the number to be converted in the accumulator if it is in floating-point form. If a fixed point number, place it in the accumulator and MQ scaled B35.

b) No number can be converted into more than 11 BCD decimal digits. (If a decimal point is to be inserted, then no more than 10 decimal digits.)

c) No floating point number which has a characteristic greater than 233_8 can be converted with this subroutine.

d) All numbers to be converted must be positive.

5.3.2 Output Requirements

The converted number appears in the accumulator and MQ as follows:

Bits 6 through 35 of the accumulator and bits S, 1, through 35 of the MQ contain the BCD number with that digit which is D* places to the right of the decimal point occupying the low order 6 bits of the MQ. Bits P, 1, through 5 of the accumulator will contain a binary number equal to D + Z* + X where X is the number of digits in the integral portion of the answer. If D = X = 0, then this number will be 1 + Z.

5.3.3 Usage

a) Calling Sequence

Floating Point Input

```
α       TSX     00BCDF, 4
α + 1   PZE     Z,,D
α + 2   (NORMAL RETURN)
```

where D = The number of decimal digits to the right of the decimal point to be converted to BCD. The program will round the converted number to D decimal places.

---

*D and Z are defined in the calling sequence.

Z = The decimal point indicator

Z = 1 if a BCD decimal point is to be inserted in the answer

Z = 0 if no decimal point is desired.

Fixed point input

```
α       TSX     O0BCDX, 4
α + 1   PZE     Z,,D
α + 2   (NORMAL RETURN)
```

b) Space Required

Instructions and tables	101
Erasable storage	7
Constants	5
Total	113

c) Timing

105 to 468 machine cycles

(1.26 msec. to 5.62 msec.)

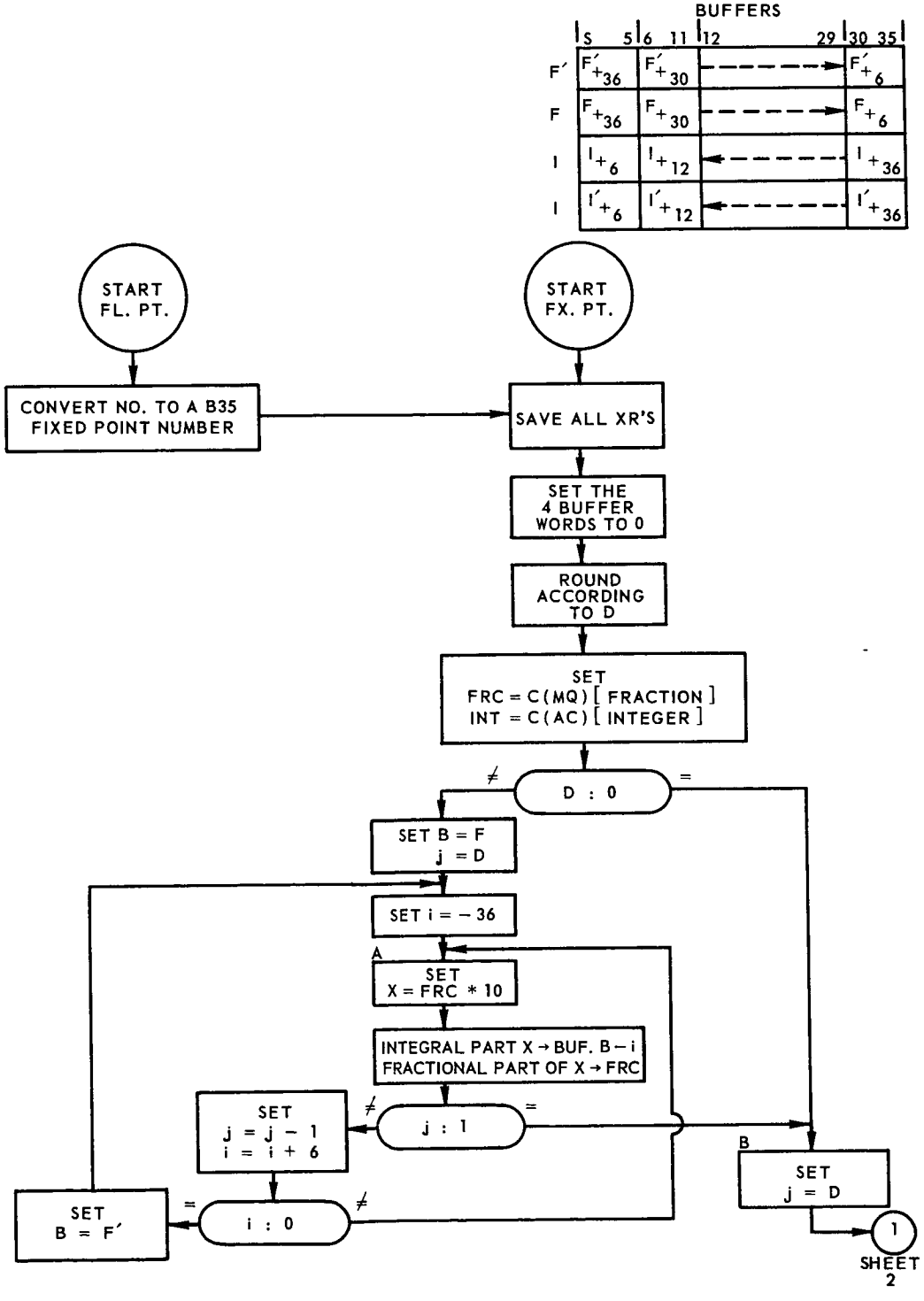


FIGURE 5-5. BCD ROUTINE FLOW CHART *(Sheet 1 of 2)*

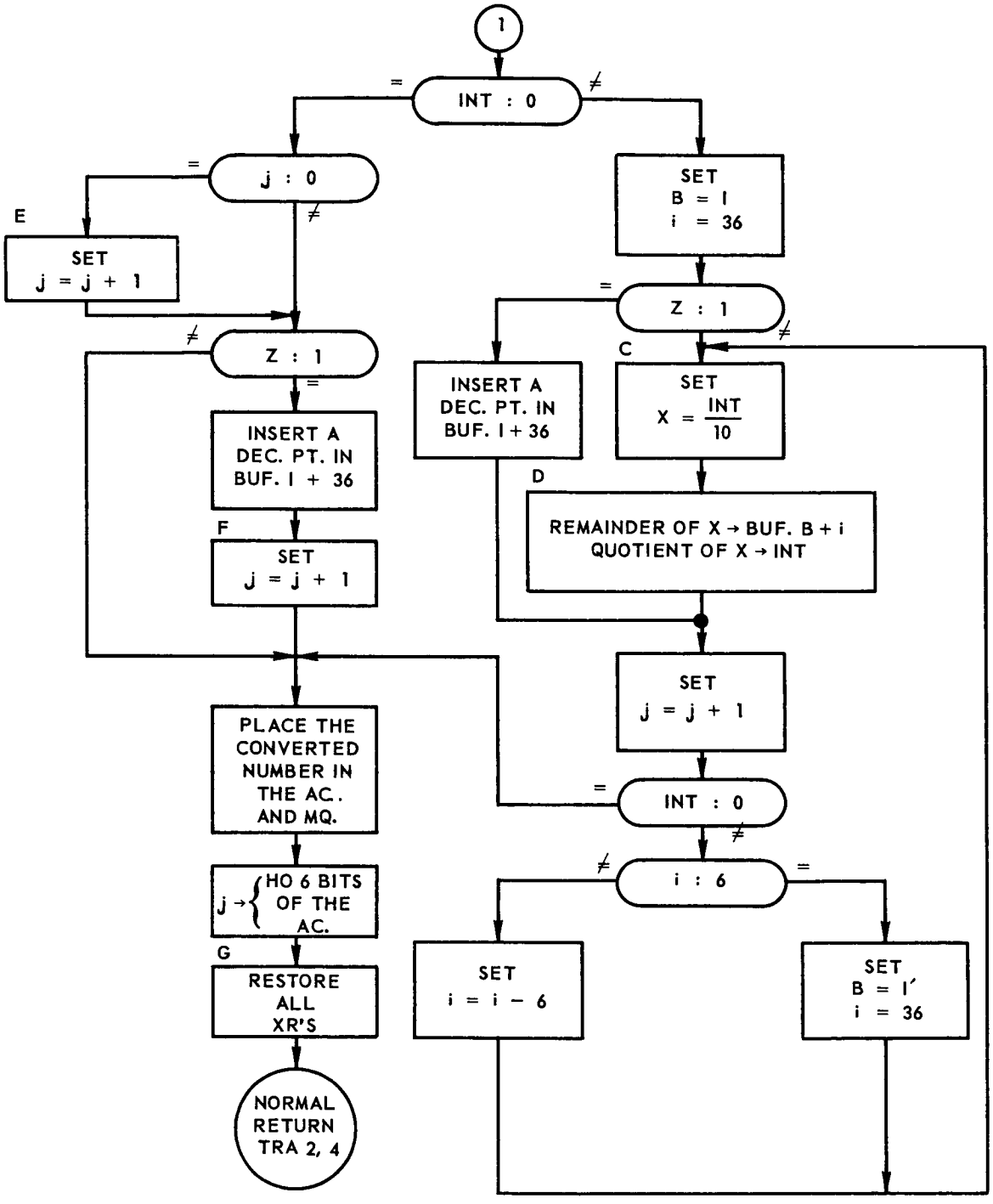


FIGURE 5-5. BCD ROUTINE FLOW CHART *(Sheet 2 of 2)*

5.4 PACKING ROUTINE (PACK)

This routine packs N number of bits (32 bits per word) located in the accumulator into a block of sequential words. The flow chart for O0PACK is shown in Figure 5-6.

5.4.1 Input Requirements

The bits to be packed must be placed in the low-order end of the accumulator. The number of bits, N, to be packed on any one pass is indicated in the calling sequence and must be of the following magnitude:

$\phi \leq N \leq X$ where X = 408 minus the number of bits already packed into the output block.

If N exceeds 35, the logical accumulator is packed and thereafter zeros are packed until N is satisfied.

5.4.2 Output Requirements

After 408 bits are packed, the packed bits will appear in the 32 low order positions of 13 sequential words starting at location TMOLAB. The order of packing is from the low order end to the high order end of each word and from TMOLAB to TMOLAB + 12.

To initiate the packing of a new output block, location PALIM must be set to zero. This must be controlled by the main program.

5.4.4 Usage

a) Calling Sequence

```
α       TSX  O0PACK,4
α + 1   PZE  N
α + 2   (RETURN)
```

where N = the number of bits to be packed.

b) Space Required

*Instructions and tables	60
Erasable storage	1
Total	61

*Includes 13 locations for TMOLAB.

c) Timing

50 to 76 machine cycles

(600 microseconds to 812 microseconds) (Based on $N \leq 32$)

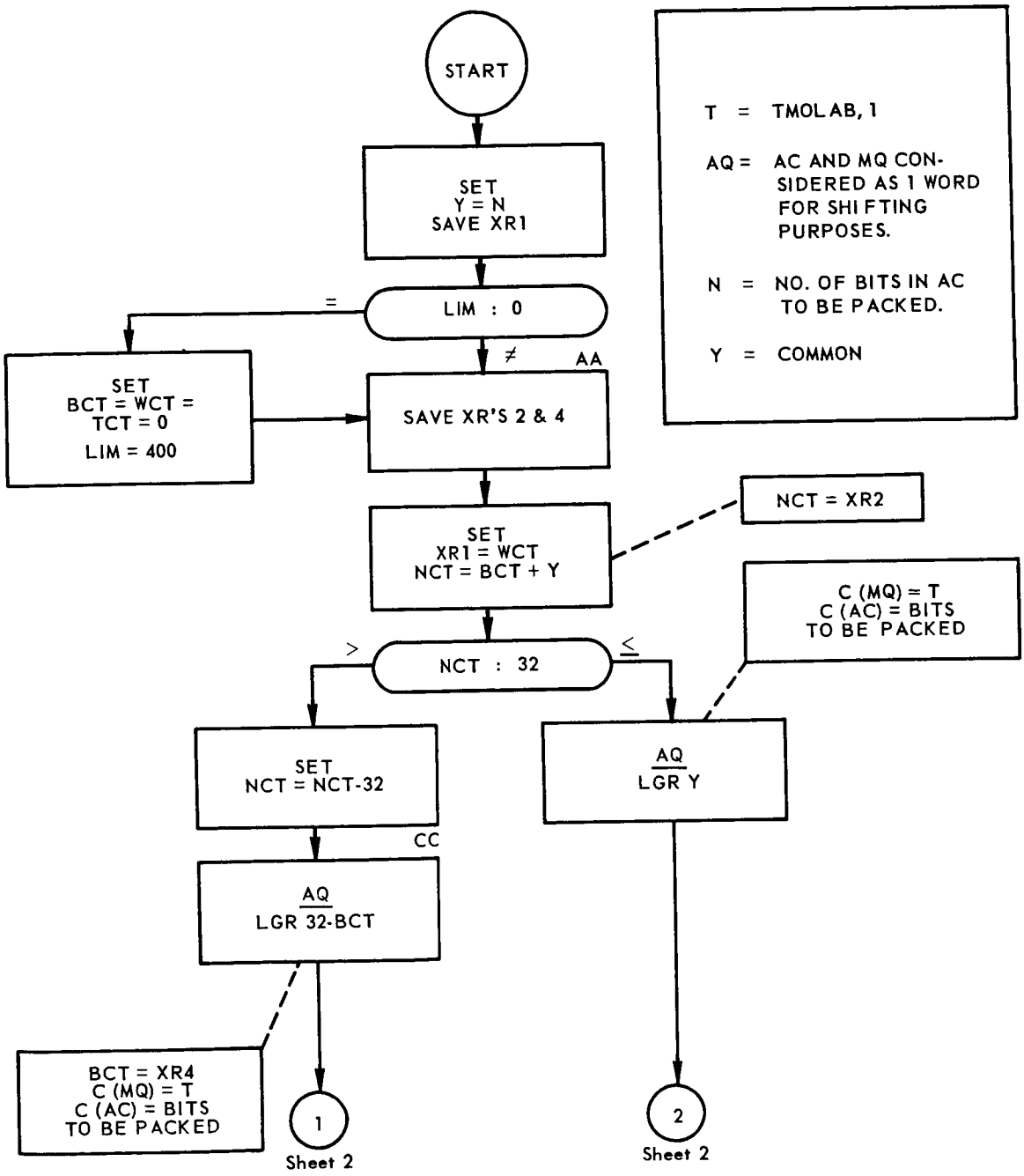


FIGURE 5-6. OOPACK ROUTINE FLOW CHART *(Sheet 1 of 2.)*

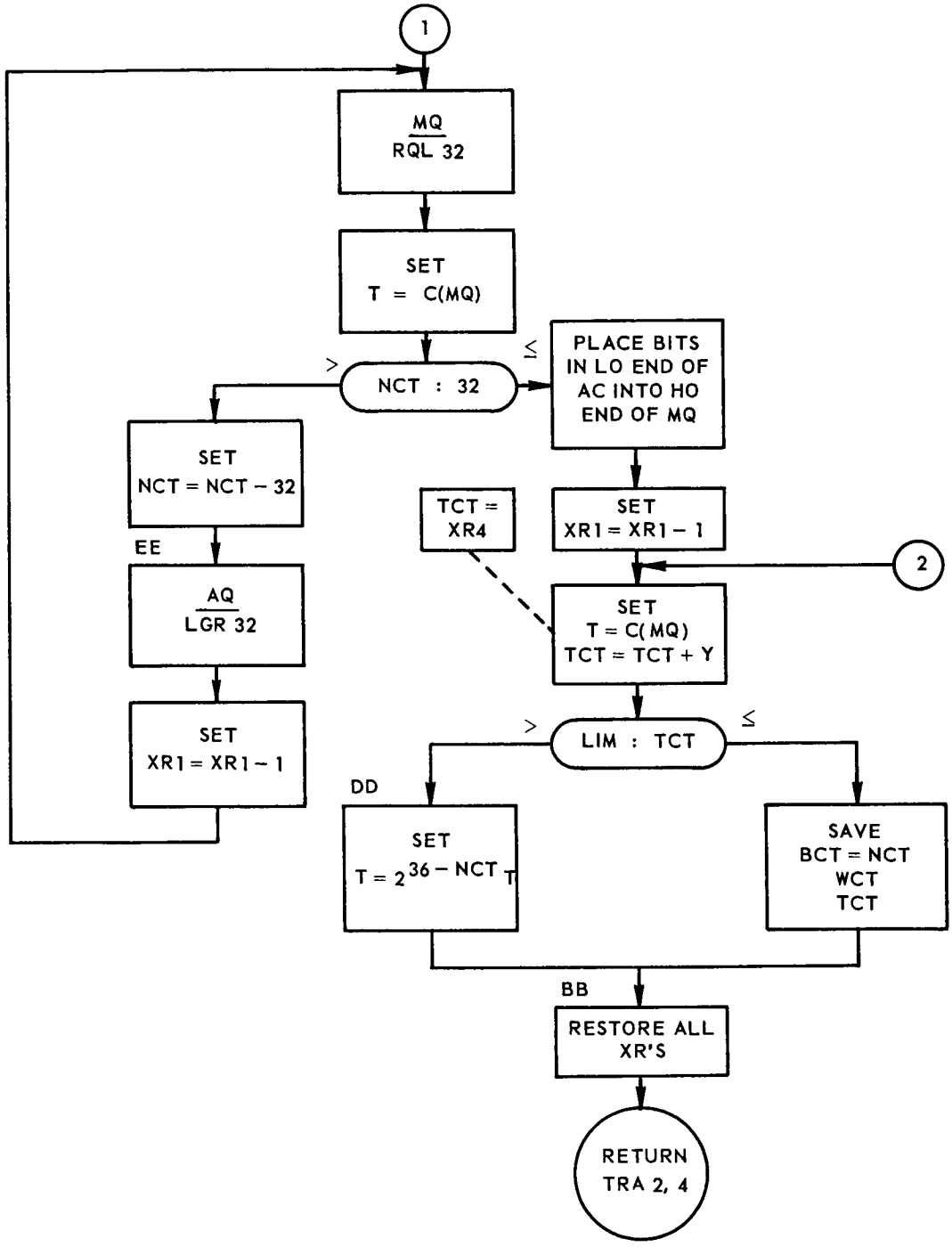


FIGURE 5-6. OOPACK ROUTINE FLOW CHART *(Sheet 2 of 2)*

5.5 TIME MEASURE TO BCD DEGREES CONVERSION PROGRAM (DGHR)

This program converts angular or time measure to BCD degrees (or hours), minutes, and seconds. The flow chart for DGHR is shown in Figure 5-7.

5.5.1 Input Requirements

This program uses the BCD routine. The following information serves as input to the DGHR program:

Angle—The number to be converted, A, must be floating point radians ($0 \leq A \leq \pi$.) and be located in the MQ upon entrance.

Time—The word to be converted must be placed in the accumulator upon entrance and be of the following format:

Seconds in the decrement.

Minutes in the address.

5.5.2 Output Requirements

The output of DGHR is as follows:

BCD seconds in the low-order end of O0SEC.

BCD minutes in the low-order end of O0MIN.

BCD degrees (or hours) in the low-order end of O0HRD.

5.5.3 Usage

a) Calling Sequence

Angular Input

```
α          TSX     O0AGLE, 4
α + 1      (NORMAL RETURN)
```

Time Input

```
α          TSX     O0TIME, 4
α + 1      (NORMAL RETURN)
```

b) Space Required (Not including BCD)

Instructions	26
Erasable storage	1
Constants	4
Total	31

c) Timing

40 to 88 machine cycles, not including BCD.

355 to 709 machine cycles, including BCD. (4.26 ms to 8.51 ms)

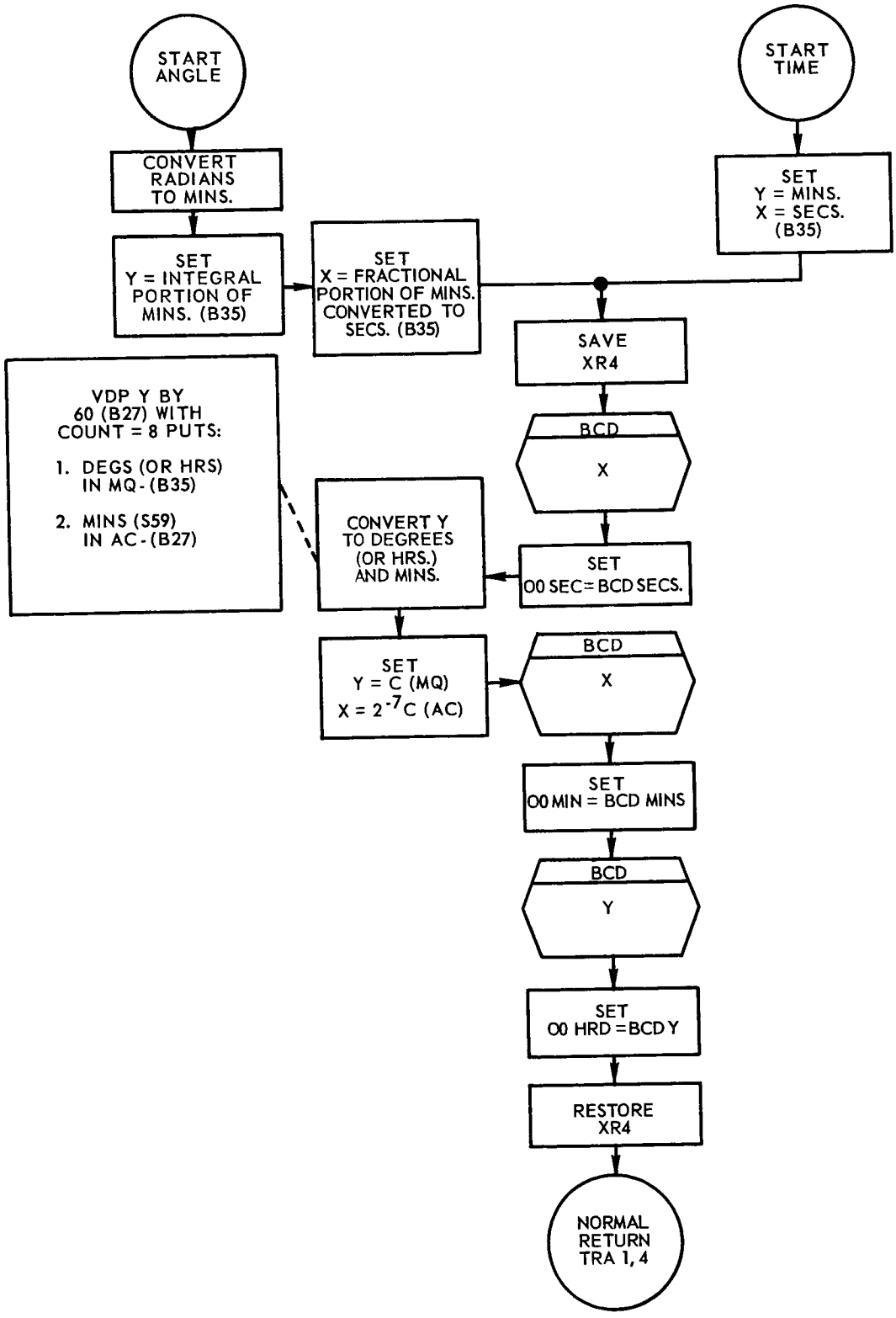


FIGURE 5-7. DGHR ROUTINE FLOW CHART

5.6 BCD TO MODIFIED BCD CONVERSION PROGRAM (MBCD)

This program converts up to 6 BCD characters to modified BCD characters.

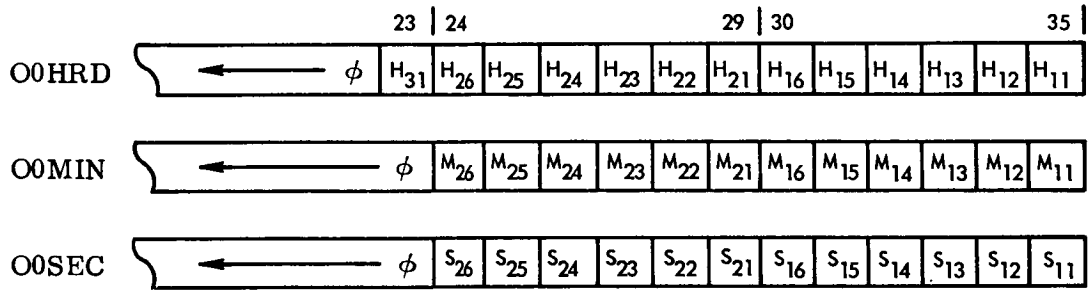
5.6.1 Input and Output Requirements

Locations "O0HRD", "O0MIN", and "O0SEC" must be defined by the main program.

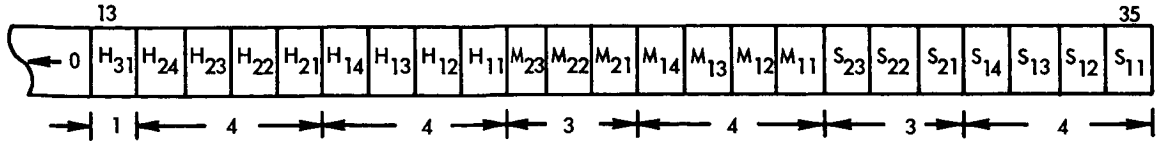
If the BCD word to be modified is in the MQ, then the following modification takes place. The first two bits of each character are eliminated. Hence, the modified BCD word consists of at most 24 bits. Upon return from the subroutine, these 24 bits occupy the low-order end of the accumulator. All other bit positions of the accumulator are zero.

When the BCD characters to be modified are in locations "O0HRD", "O0MIN", and "O0SEC" (the output of DGHR), the subroutine will modify the low-order 13 bits of "O0HRD" and the low-order 12 bits of the other two. All other bit positions in these locations must be zero.

Hence, the input should be as follows:



The following 23 bit result will appear in the low order end of the accumulator:



5.6.2 Usage

a) Calling Sequence

MQ Modification

```
α          TSX     O0MODQ, 4
α + 1      (RETURN)
```

Storage Modification

```
α          TSX     O0MODS, 4
α + 1      (RETURN)
```

b) Space Required: 29 locations

c) Timing

1) MODQ—14 Machine cycles (336 us)

2) MODS—15 Machine cycles (360 us)

5.7 DATA PREPARATION FOR GODDARD PLOTBOARDS (PLOT)

This subroutine packs coded bits for transmission to the Goddard plotboards. By proper setting of the console panel keys, any of the four Mercury Control Center plotboards can be duplicated on the Goddard plotboard. The flow chart for PLOT is shown in Figure 5-8.

5.7.1 Input Requirements

The PLOT routine examines the tag of the console panel keys for an indication as to what should be plotted. The following numbers in the tag have the indicated effect.

0 - A ϕ causes the arms to remain fixed on the plotboards in their present position.

1. - causes the duplication of plotboard 1 at Cape Canaveral

2. - causes the duplication of plotboard 2 at Cape Canaveral

3. - during Orbit, causes the duplication of plotboard 3 at Cape Canaveral During the other phases, zeros are sent to both arms.

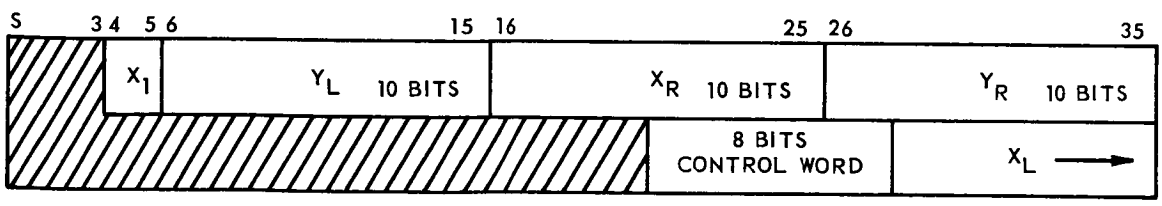
4. - causes the duplication of plotboard 4 at Cape Canaveral

5, 6, or 7. - puts both arms on standby.

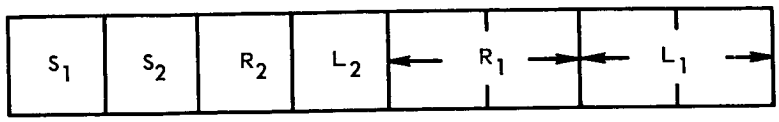
5.7.2 Output Requirements

According to the tag, PLOT taps off the proper bits in TMOLAB and stores them for output in the two location block, TMMRLP. The format of TMMRLP is as follows:

The low-order 10 bits consist of the "Y" component for the right arm. These bits are the same as those sent to the Mercury Control Center. The next 10 low-order bits of TMMRLP is the "X" component for the right arm. Following are 20 bits for the components of the left arm. The final 8 bits are the control word.



Above is the complete format for the 48 bit output. The format of the 8 bit control word is as follows:



L_1: 01 = Lift Top pen (color 1) of left arm
10 = Lift bottom pen (color 2) of left arm
11 = Lift both pens of left arm

R_1: Same as L_1 except for the right arm.

L_2: 1 = Left arm on standby

R_2: 1 = Right arm on standby

S_1, S_2: Spares

It should be noted that a standby command overrides all other bit controls.

5.7.3 Usage

a) Calling Sequence:

α	TSX	PLOT, 4
$\alpha + 1$	Return	

b) Space Required:

Instructions	69
Erasable storage	1
Constants	8
Tables	4
Total	82

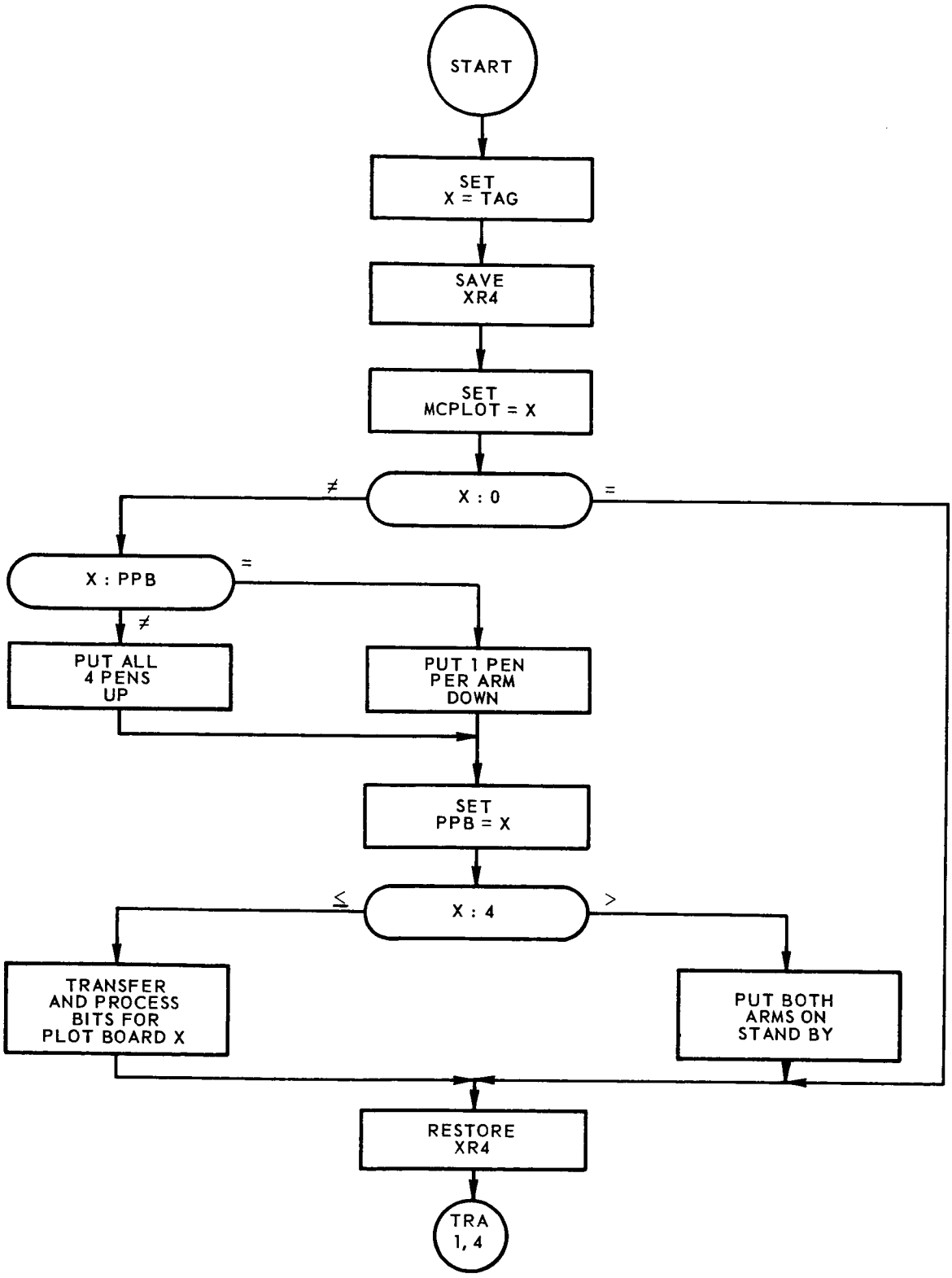


FIGURE 5-8. PLOT ROUTINE FLOW CHART